

AN14259

对MCXN947烧写CMPA和带单调计数器的CFPA的SDK示例

第1.0版—2024年7月18日

应用笔记

文档信息

信息	内容
关键词	AN14259、单调计数器、CMPA、CFPA
摘要	本应用笔记介绍了如何通过ROM API更改CFPA/CMPA位域的示例。



1 介绍

本文档提供了使用ROM API更改客户制造可编程区（CMPA）和客户现场可编程区（CFPA）位域的示例。此外，也可使用MCUXpresso安全预处理工具来完成此操作，请参阅[MCUXpresso安全预处理工具](#)。MCUXpresso安全预处理工具基于GUI界面，可提供简化的开发流程，使准备（更改CMPA/CFPA位域、生成密钥等）、编程和烧写镜像以及在恩智浦MCU器件上备制可启动可执行文件变得更简单。无论是使用ROM API还是MCUXpresso安全预处理工具，都可以烧录一次性可编程（OTP）熔丝，从而通过特定设置锁定CMPA和CFPA区域。对于使用MCXNx4x器件发布的成品来说，这种锁定是非常必要的。

CMPA和CFPA是MCX器件的受保护闪存区域（PFR）的一部分。PFR区域包含启动配置、安全策略、PRINCE（内部和外部）、SoC特定参数等的设定值。PFR在闪存中有一个固定地址，从地址0x10000000开始。[图1](#)所示为PFR布局的不同区域。PFR包含CFPA暂存页、CFPA ping和pong页、CMPA页、客户密钥存储页（CMPA的一部分）以及恩智浦制造可编程区（NMPA）页。

注：当使用OTP时，本文档提供的信息不会锁定具有特定设置的CMPA和CFPA区域。

对MCXN947烧写CMPA和带单调计数器的CFPA的SDK示例

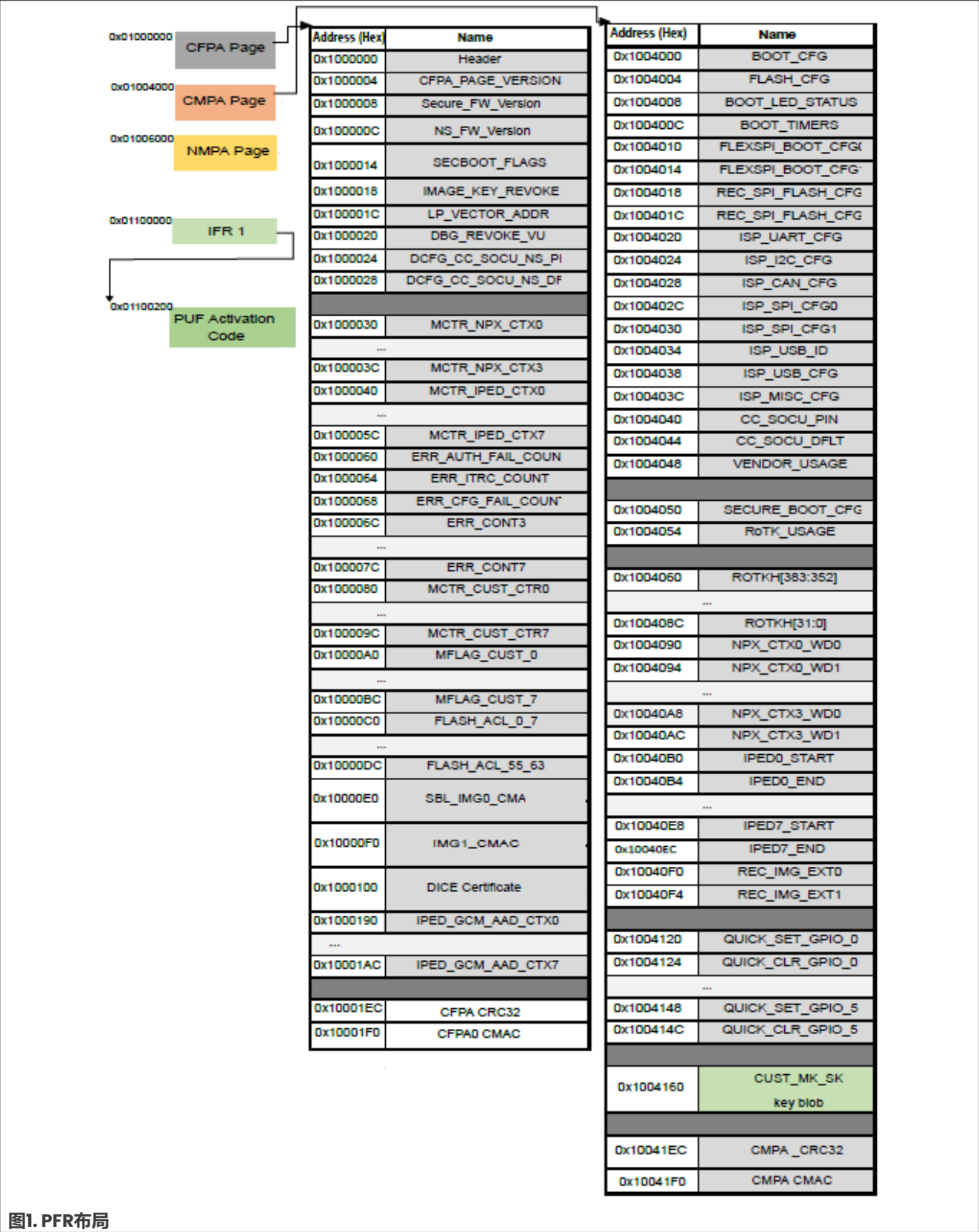


图1. PFR布局

1.1 CMPA

CMPA区域提供以下配置的设定值：

1. **启动配置**：包含与启动模式、速度、超时、默认启动源、USB产品/供应商ID等相关的配置。
2. **RoT密钥表哈希**：包含48个字节的根密钥表哈希（ROTKH），它是根据四个OEM公钥计算得出的SHA-384或SHA-256摘要。存储了四个OEM公私钥对，以便在其中一个私钥被泄露时有另外的密钥对可用。ROTKH也被称为“哈希的哈希”，因为此哈希值是通过四个OEM公钥哈希值计算得出的。
3. **调试配置**：包括启用/禁用调试访问等配置。这些配置用于确定生命周期和调试身份认证状态。
4. **PRINCE配置**：允许运行时解密的加密闪存代码和数据。包含外部闪存内联PRINCE加密/解密（IPED）和内部闪存PRINCE相关的设置。
5. **FlexSPI配置**：包含FlexSPI接口的配置设定值，该接口可连接到外部SPI闪存，支持1、2、4和8位。FlexSPI支持就地执行（XIP）和使用IPED模块即时解密。

CMPA还包含对CMPA区域计算得出的CMAC/CRC32字段，以便对CMPA区域中的数据进行完整性检查。当CFPA区域中的CMAC_UPD[CMPA_UPD]字段设置为0b010或0b011时，ROM会将计算值写入CMPA。如果CMAC_UPD[CMPA_UPD]位设置为0b011，ROM也会将该值写入CMPA_CMAL OTP，并且在CMPA_CMAL熔丝烧写后，用户就无法再更改CMPA位域。如果需要在MCU的开发阶段进行这样的转换，或者芯片已准备好在现场部署，则通常可以烧写CMPA_CMAL熔丝。

CMPA还提供了一个客户密钥存储页，可用于存储客户应用程序所需的不同密钥以及必须从外部提供给加密子系统（CSS）的一些预共享密钥，例如，以RFC3394 blob形式存储在地址0x1004160的CUST_MK_SK密钥。有关更多详细信息，请参阅《MCXNx4x参考手册》（文件[MCXNX4XRM](#)）及其附件MCXNx4x_IFR.xlxs。

1.2 CFPA

CFPA暂存页、ping页和pong页具有相同的结构。然而，CFPA的ping页和pong页用于选择实际的CFPA页（版本号较高的那个页面），而暂存页用于更新CFPA页。这意味着每当应用程序代码使用FLASH API更新CFPA页时，数据都会首先写入暂存页，然后在内核复位后，根据CFPA页版本号将暂存页的内容复制到ping页或pong页。版本号较低的那个页面将更新为最新版本。

CFPA提供了以下配置的设定值：

1. **ROTKH 撤销**：最多可以有四个RoT密钥，每个密钥都可以被撤销。当尝试使用已撤销的RoT密钥签名的镜像时，如果SEC_BOOT_EN设置为仅启动已签名的镜像，它们将在身份认证过程中被拒绝并且无法启动。
2. **镜像撤销**：提供一个镜像密钥撤销ID，可用于撤销镜像。有关更多详细信息，请参阅《MCXNx4x安全参考手册》（文档MCXNx4xSRM）中“安全启动ROM”章节的“CFPA页”部分。
3. **PRINCE区域IV码存储**：包括对于内部和外部PRINCE区域的单调擦除计数器。然而，这些字段完全由ROM处理，用户不得在这些字段中写入任何内容。有关更多详细信息，请参阅《MCXNx4x安全参考手册》（文档MCXNx4xSRM）中“安全启动ROM”章节的“CFPA页”部分。

4. **调试配置：**包括启用/禁用调试访问等配置。这些配置用于确定生命周期和调试身份认证状态。有关更多详细信息请参阅《MCXNx4x 参考手册》（文档 [MCXNX4XRM](#)）中的“调试子系统”章节。

除了上述配置外，CFPA页还包含各种版本的字段，如CFPA页版本和安全与非安全固件版本，以及可存储在闪存中的双启动镜像的CMAC值，也可以作为其字段存储。有关更多详细信息，请参阅《MCXNx4x参考手册》（文档 [MCXNX4XRM](#)）及其附件MCXNx4x_IFR.xlslx。

1.2.1 单调计数器

特定字段（如各种版本、供应商使用情况、内部和外部PRINCE区域的擦除计数以及客户定义的单调计数器字段）是使用单调（递增）计数器实现的。单调计数器用于生成递增或递减值。对于递增值的实现，一旦计数值变为更高的数字，则此后不得出现任何小于此更高数字的值。无论采用哪种实现方式，都必须在此器件的整个生命周期内保持计数值的单调性，这一点在经历无数次的电源开关循环时尤其重要。如果单调计数器未能保持其计数值，甚至回退至之前较早的数值，可能会对器件的安全性构成严重威胁。

1.3 NMPA

NMPA是PFR的最后一个区域，包含芯片和晶圆的独特数据，如PUF激活码、恩智浦颁发的器件正品证明证书、以及UUID等信息。有关更多详细信息，请参阅《MCXNx4x参考手册》（文档[MCXNX4XRM](#)）及其附件MCXNx4x_IFR.xlslx。

1.4 ROM API的结构

ROM API的结构如[图2](#)所示。它包含多个API驱动程序，这些驱动程序包含绝对ROM API函数地址，可使用函数指针进行调用。ROM API提供的部分功能包括支持两个可编程的FLASH区域（存储应用代码和数据）和FLASH防火墙区域（存储设备配置、启动配置、现场可编程数据）。它还能进行外部串行NOR FLASH、片上OTP-eFUSE的读取和编程、支持加密功能、应用内编程等等。通过FLASH API驱动程序，可使用FLASH API来更新可编程闪存区域并指定CMPA和CFPA区域的参数。

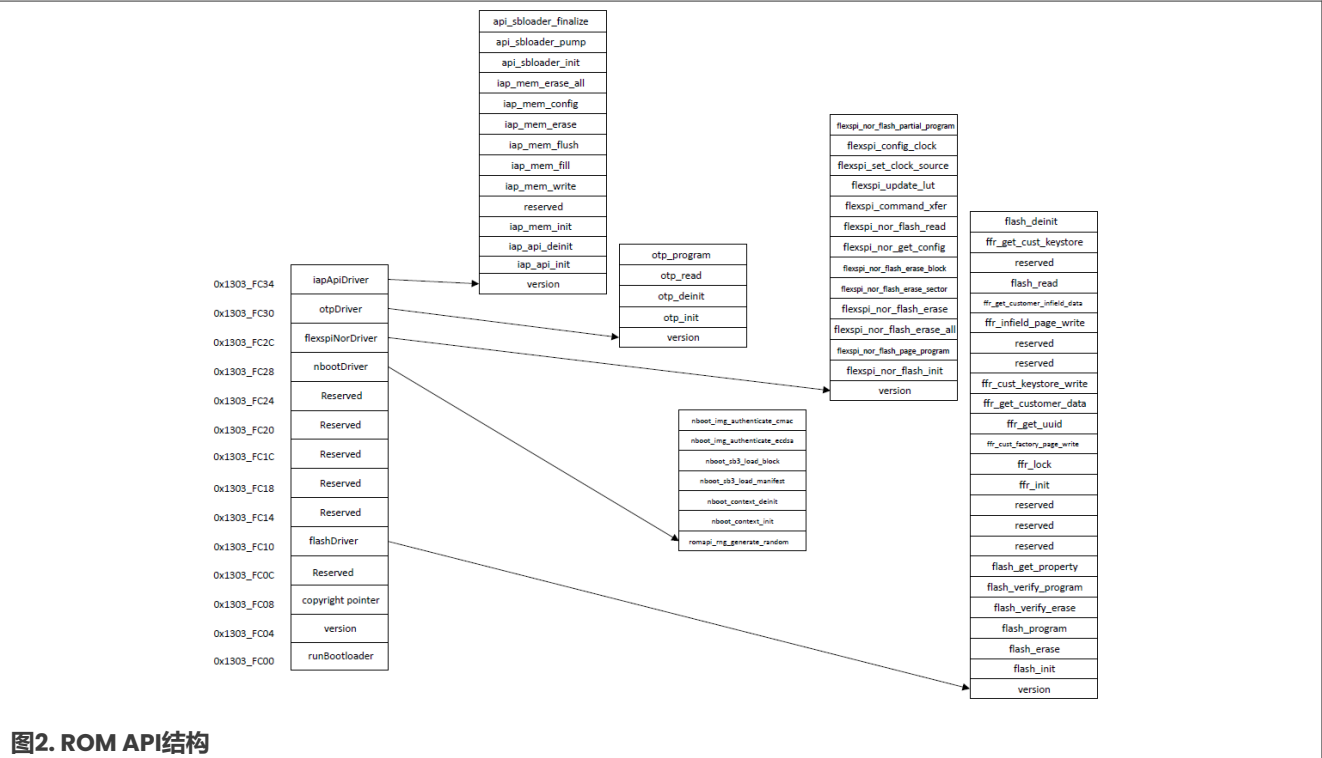


图2. ROM API结构

1.5 FLASH API驱动程序

FLASH API集可实现以下功能：

1. 初始化FLASH控制器。
2. 擦除并验证指定的FLASH区域。
3. 编程并验证指定的FLASH页面。
4. 检索FLASH属性。
5. 初始化或锁定FFR。
6. 编程并读取CMPA。
7. 编程并读取CFPA。
8. 非阻塞FLASH擦除/状态检查API，用于时序关键型用例。

FLASH API的设计严格遵循了FLASH驱动程序API接口的结构。

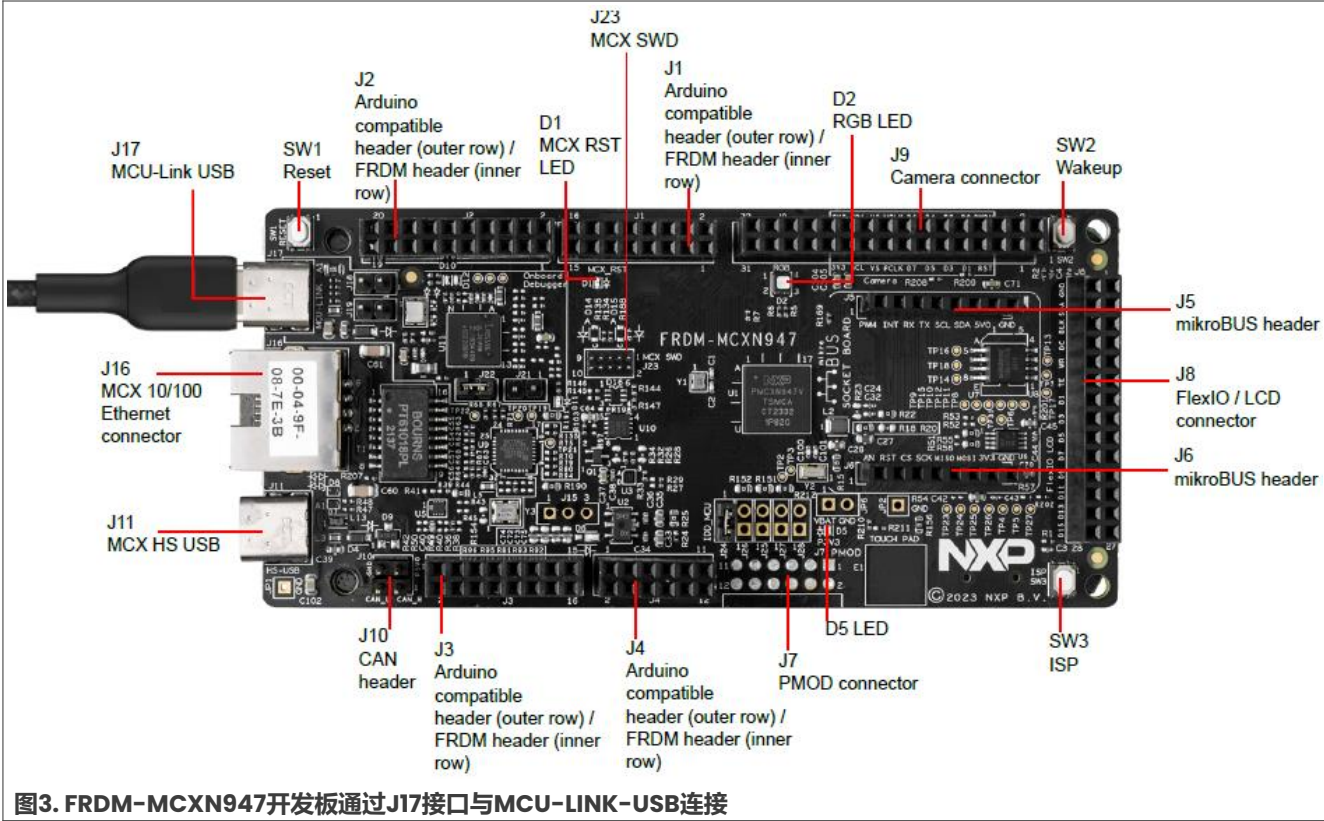
有关不同API的更多详细信息，请参阅《MCXNx4x参考手册》（文档[MCXNX4XRM](#)）中“ROM API”章节的“FLASH API”部分。SDK发布包中还提供了整个FLASH API的封包。

2 设置和SDK示例

本节提供了有关硬件和软件设置的、SDK示例以及输出结果的信息。

2.1 硬件设置

图3所示为主机计算机如何通过MCU-Link USB J17接口与FRDM-MCXN947开发板建立连接。J17利用MCU-Link VCOM输出，充当了主机的串行转USB桥接器，并提供了CMSIS-DAP调试接口。



2.2 软件设置

以下应用示例使用MCUXpresso IDE v11.8.0进行下载，详见[MCUXpresso集成开发环境 \(IDE\)](#)。此外，为了构建和调试代码，还需要安装MCX-N9XX-EVK的SDK。以下示例使用了SDK_2.13.1_MCX-N9XX-EVK。

注：输出结果可显示在Tera Term应用程序上。

使用Tera Term或任何其他终端应用程序时，必须使用以下设置：

- 115200 波特率
- 无奇偶校验
- 8 个数据位
- 1 个停止位

2.3 SDK示例

本文档包含了SDK示例工程。要导入SDK示例工程（位于归档文件夹中），请按照以下步骤操作：

1. 转入**File -> Import -> General -> Existing Projects**。
2. 浏览至工作区，然后单击**Next**。
3. 选择.zip归档文件。
4. 单击**Finish**。
5. 此工程应显示在**Project Explorer**窗口中。
6. **要构建工程：**请点击位于IDE左下角的**MCUXpresso IDE - Quickstart Panel**中的**Build**按钮，或点击顶部工具栏中的锤子图标。
7. **要将工程烧录到MCU：**右击**Project**选项卡，选择**Run as -> 1 Local C/C++ Application**。

8. **要调试工程：**从**MCUXpresso IDE – Quickstart Panel**中选择**Debug**按钮，或点击顶部工具栏中的**Bug**按钮（以蓝色突出显示）。

在以下SDK示例中，提供了三个选项：

- 1. 显示UUID和CFPA页的内容。
- 2. 升级CFPA页的版本。
- 3. 增大供应商使用次数。

CFPA页的版本和供应商使用次数字段均采用单调计数器来实现。因此，一旦它们的值发生变化，新设置的值必须总是大于之前的值。每次对CFPA页面进行更改时，用户都必须确保CFPA页面的版本号是递增的（单调计数器）。由于CFPA的更改是在CFPA暂存页上进行的，因此必须进行内核复位，以便将新的数值从CFPA暂存页复制到主CFPA页面上（根据页面版本选择ping页或pong页）。

该示例工程采用了FLASH API的封包（在SDK发行包中作为fsl_flash.h头文件和fsl_flash.c源文件提供）中提供的函数，这些函数使用了底层的FLASH API。以下章节将介绍一些有关所使用的封包API的详细信息。

注：这些函数调用最多使用2kB的堆栈来执行读取-修改-写入PFR页。请根据需要适当调整堆栈大小。

2.3.1 FFR_GetUUID

- 1. 底层FLASH API函数：ffr_get_uuid。
- 2. 函数原型：status_t FFR_GetUUID(flash_config_t *config, uint8_t *uuid)。
- 3. 参数说明：
 - config – 这是一个指向内存中flash_config_t数据结构的指针，可用于存储驱动程序的运行时状态。
 - uuid – 这是指向一个数值地址的指针，该值是从nmpa的配置中回读的uuid值。
- 4. 此示例工程的代码片段如下所示：

```
volatile uint32_t g_UUID[4];
/* Initialize flash driver */
FLASH_Init(&flashInstance);
if (FFR_Init(&flashInstance) == kStatus_Success)
    PRINTF("Flash init successfull!! Halting...\r\n");
else
    error_trap();

status = FFR_GetUUID(&flashInstance, (uint8_t *)g_UUID);
PRINTF("\nUUID 0x%8x 0x%8x 0x%8x 0x%8x\r\n", g_UUID[0], g_UUID[1], g_UUID[2],
g_UUID[3]);
```

2.3.2 FFR_GetCustomerInfieldData

- 1. 底层FLASH API函数：ffr_get_customer_infield_data。
- 2. 函数原型：status_t FFR_GetCustomerInfieldData(flash_config_t *config, uint8_t *pData, uint32_t offset, uint32_t len)。
- 3. 参数说明：
 - config – 指向内存中flash_config_t数据结构的指针，用于存储驱动程序的运行时状态。
 - pData – 指向数据目标缓冲区的指针，用于存储从客户现场页读取的数据。
 - offset – 指向基于设备的CFPA地址0x3dc00的偏移值的指针。
 - len – 指向要读取的数据长度的指针，且偏移值 + 长度 <= 512B。

4. 此示例工程的代码片段如下所示:

```
flash_config_t flashInstance;
uint32_t g_CFPADData[FLASH_FFR_MAX_PAGE_SIZE / sizeof(uint32_t)];

/* Read data stored on the Customer Factory page */
status = FFR_GetCustomerInfieldData(&flashInstance, (uint8_t *)g_CFPADData,
0x0,
FLASH_FFR_MAX_PAGE_SIZE);

PRINTF("\r\nCFPA page data:\r\n");
PRINTF("Header 0x%08x ,          Version 0x%08x , SecureFW Version 0x%08x ,
NonSecureFW Version 0x%08x\r\n", g_CFPADData[0], g_CFPADData[1],
g_CFPADData[2],
g_CFPADData[3]);
PRINTF("ImageKey Revoke 0x%08x , RothRevoke 0x%08x , Vendor Usage 0x%08x\r\n",
g_CFPADData[4], g_CFPADData[20], g_CFPADData[21]);
PRINTF("NS PIN 0x%08x ,          NS DFLT 0x%08x ,      Enable FA Mode 0x%08x\r\n",
g_CFPADData[22], g_CFPADData[23], g_CFPADData[24]);
```

2.3.3 FFR_InfieldPageWrite

1. 底层FLASH API函数: ffr_infield_page_write.
2. 函数原型: status_t FFR_InfieldPageWrite(flash_config_t *config, uint8_t *page_data, uint32_t valid_len).
3. 参数说明:
 - config - 指向内存中flash_config_t数据结构的指针, 用于存储驱动程序的运行时状态。
 - page_data - 指向要烧录到现场页的数据源缓冲区的指针。
 - valid_len - 要烧录的字节长度, 长度必须等于页大小。

4. 此示例工程的代码片段如下所示:

```
flash_config_t flashInstance;
uint32_t g_CFPADData[FLASH_FFR_MAX_PAGE_SIZE / sizeof(uint32_t)];

status_t markStatus = FFR_InfieldPageWrite(&flashInstance, (uint8_t *)g_CFPADData,
FLASH_FFR_MAX_PAGE_SIZE);
if (kStatus_FLASH_Success == markStatus)
{
    status = kStatus_Success;
    PRINTF("CFPA Write Done!\r\n");
}
else
{
    status = kStatus_Fail;
    PRINTF("CFPA Write Failed!\r\n");
}

/* Core reset to allow updating the main CFPA page */
NVIC_SystemReset();
break;
```

2.4 输出

输出结果使用UART COM5端口显示在Tera Term应用程序上。在成功编译并烧录代码到MCU后，用户会收到一个包含三个选项的提示，具体内容详见第2.3节。

- 1. 如果用户通过输入“0”选择第一个选项，将出现如图4所示的输出界面。输出内容包括设备特定的UUID和CFPA页的内容。



图4. 在Tera Term上显示的选项0的生成输出结果

- 2. 如果用户通过输入“1”选择第二个选项，则写入版本字段的CFPA页面版本将被更新，如图5所示。在确认信息（“CFPA Write Done!”）出现后，用户可以再次输入“0”来查看更新后的CFPA页面内容。

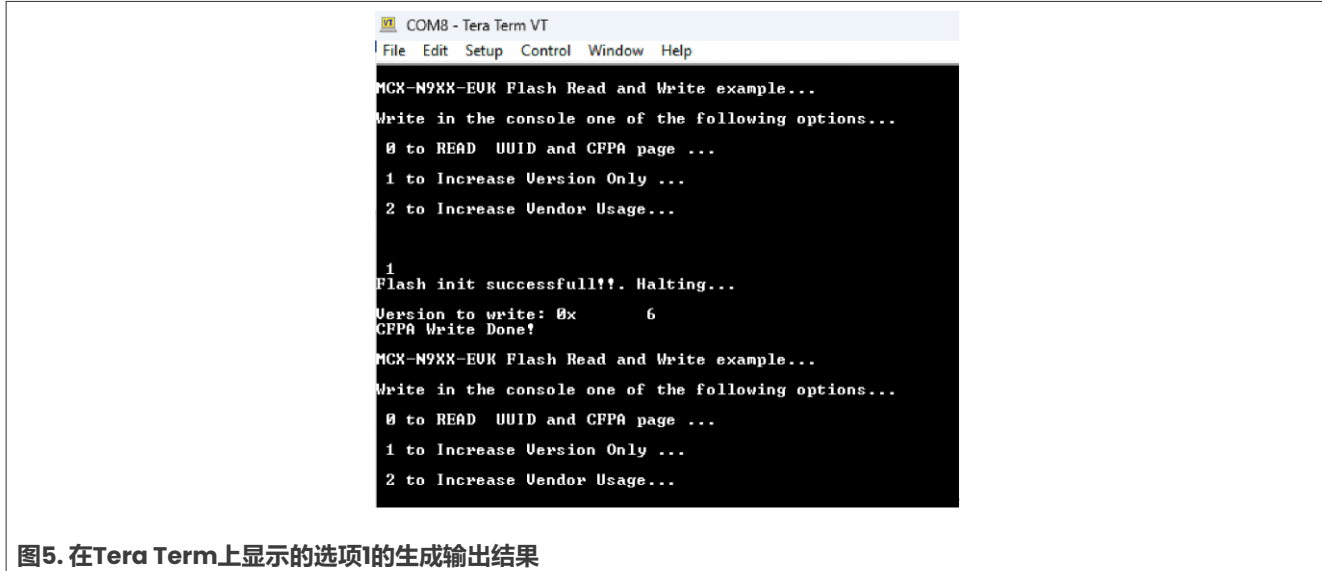


图5. 在Tera Term上显示的选项1的生成输出结果

- 3. 如果用户通过输入2选择第三个选项，则相应字段中将写入供应商使用次数页的新版本，以及更新的CFPA页版本，如图6所示。在确认信息（CFPA Write Done!）出现后，用户可以再次输入“0”，查看CFPA页的更新内容。

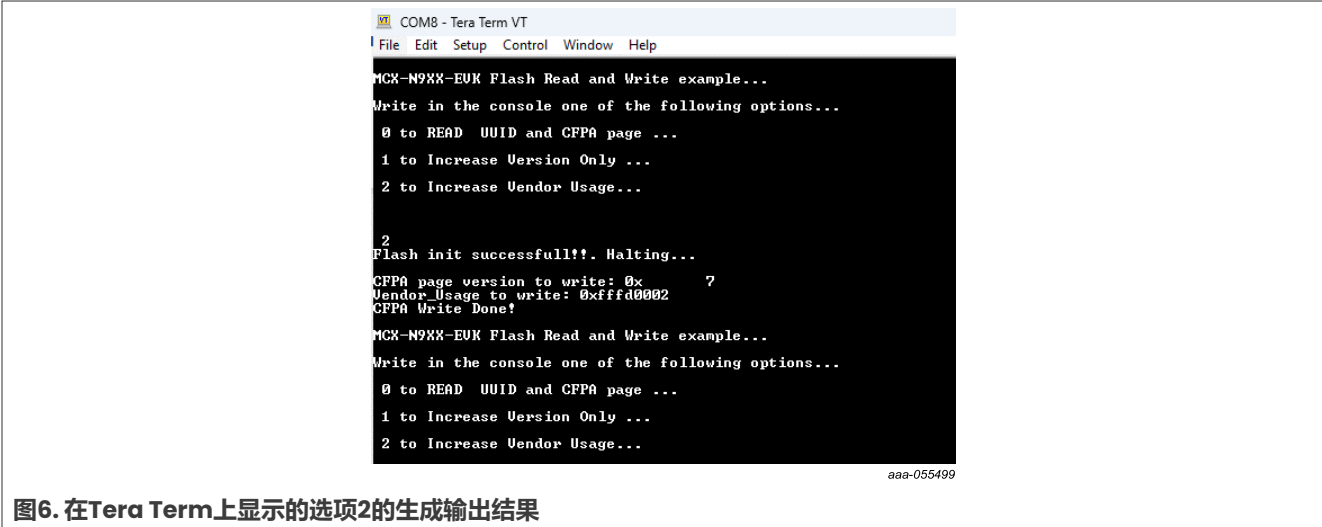


图6. 在Tera Term上显示的选项2的生成输出结果

CFPA页根据用户的输入进行更新，从而显示了软件的成功运行。

3 参考资料

表1列出了其他可参考的文档和资源，以获取更多信息。以下列出的一些文档可能仅在签署保密协议（NDA）后才能访问。如需获取这些文档的访问权限，请联系您当地的现场应用工程师（FAE）或销售代表。

表1. 相关文档/资源

文档	链接/访问方式
MCUXpresso安全预处理工具	MCUXPRESSO-SECURE-PROVISIONINGI
MCUXpresso集成开发环境（IDE）	MCUXpresso-IDE
MCXNx4x参考手册（文档MCXNX4XRM）	MCXNX4XRM
MCXNx4x安全参考手册（文档MCXNx4x SRM）	联系当地FAE或销售代表

4 缩略语和缩写

本章节列出了本文所用的缩略语和缩写词汇。

表2. 缩略语和缩写

缩略语	定义
API	应用编程接口
CFPA	客户现场可编程区
CMPA	客户制造/工厂可编程区
COM	通信端口
FAE	现场应用工程师
GUI	图形用户界面
IPED	内联Prince加密/解密
MCU	微控制器单元

表2. 缩略语和缩写 (续)

缩略语	定义
NDA	保密协议
NMPA	恩智浦制造可编程区域
NOR	或非
OEM	原始设备制造商
OTP	一次性可编程
PFR	受保护的闪存区域
ROTKH	根密钥表哈希
ROM	只读存储器
SDK	软件开发套件
SoC	片上系统
SPI	串行外设接口
UART	通用异步接收器发送器
USB	通用串行总线
VCOM	虚拟COM
XIP	就地执行

5 关于本文中源代码的说明

本文中所示的示例代码具有以下版权和BSD-3-Clause许可：

2024年恩智浦版权所有；在满足以下条件的情况下，可以源代码和二进制文件的形式重新分发和使用本源代码（无论是否经过修改）：

- 1. 重新分发源代码必须保留上述版权声明、这些条件和以下免责声明。
- 2. 以二进制文件形式重新分发时，必须在文档和/或随分发提供的其他材料中复制上述版权声明、这些条件和以下免责声明。
- 3. 未经事先书面许可，不得使用版权所有者的姓名或参与者的姓名为本软件的衍生产品进行背书或推广。

本软件由版权所有者和参与者“按原样”提供，不承担任何明示或暗示的担保责任，包括但不限于对适销性和特定用途适用性的暗示保证。在任何情况下，无论因何种原因或根据何种法律条例，版权所有或参与者均不对因使用本软件而导致的任何直接、间接、偶然、特殊、惩戒性或后果性损害（包括但不限于采购替代商品或服务；使用损失、数据损失或利润损失或业务中断）承担责任，无论是因合同、严格责任还是侵权行为（包括疏忽或其他原因）造成的，即使事先被告知有此类损害的可能性也不例外。

6 修订历史

表3汇总了本文的修订情况。

表3. 修订历史

文档ID	发布日期	说明
AN14259 v.1.0	2024年7月18日	首次公开发布

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at <https://www.nxp.com.cn/profile/terms>, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this document expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. 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 B.V. — NXP B.V. is not an operating company and it does not distribute or sell products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

MCX — is a trademark of NXP B.V.

目录

1 介绍..... 2

1.1 CMPA..... 4

1.2 CFPA..... 4

1.2.1 单调计数器 5

1.3 NMPA..... 5

1.4 ROM API的结构 5

1.5 FLASH API驱动程序..... 6

2 设置和SDK示例..... 6

2.1 硬件设置..... 6

2.2 软件设置..... 7

2.3 SDK示例 7

2.3.1 FFR_GetUUID..... 8

2.3.2 FFR_GetCustomerInfieldData..... 8

2.3.3 FFR_InfieldPageWrite..... 9

2.4 输出 10

3 参考资料..... 11

4 缩略语和缩写..... 11

5 关于本文中源代码的说明..... 12

6 修订历史..... 12

法律声明..... 14

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.