

1 简介

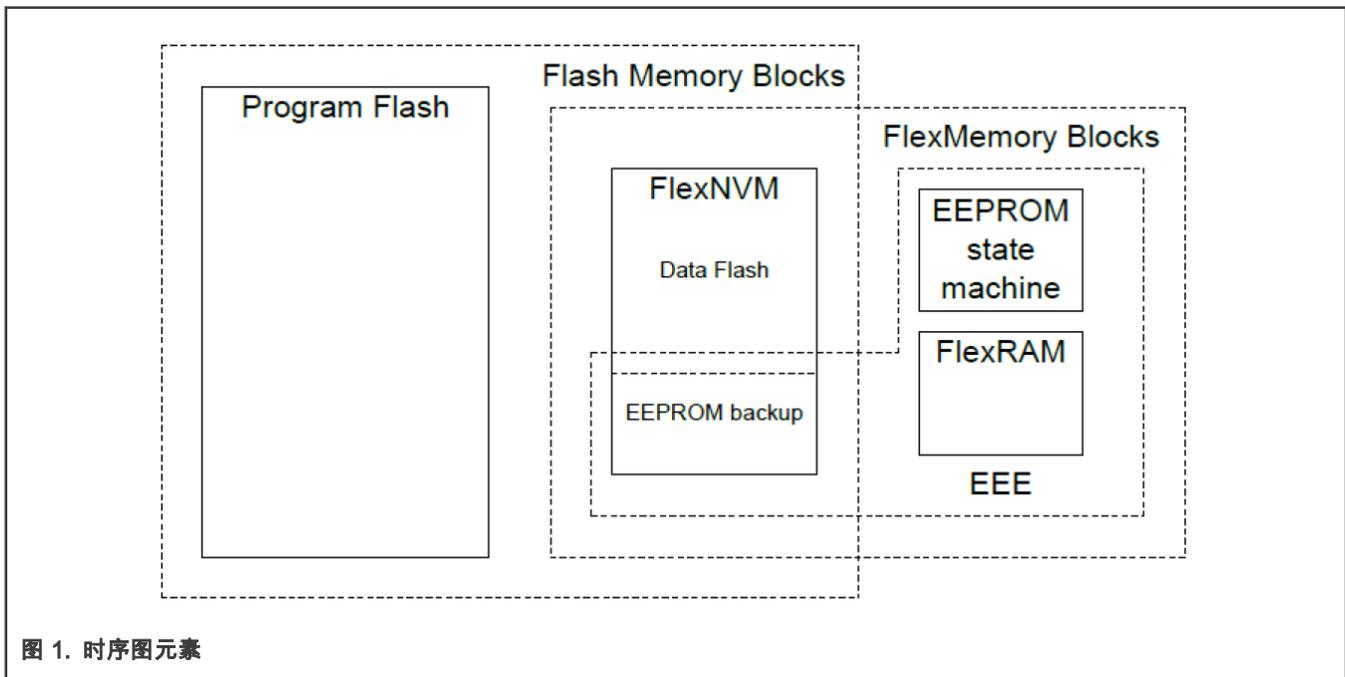
- FlexMemory (FlexNVM 和 FlexRAM) 可以在恩智浦的 Kinetis KE1xF 系列上使用。FlexMemory 允许用户将 FlexNVM 块配置为基本 D-Flash (数据 Flash)、增强 EEPROM (EEE) 或两者的组合。
- D-Flash memory 对于需要快速存储大量数据或存储静态数据的应用程序非常有用。EEPROM 特性被广泛应用于在系统断电时存储少量变化的数据。
- 本应用笔记描述了 FlexMemory 的特性，以及如何实现它作为 D-Flash 和增强的 EEPROM 的功能。这两种用法的示例包含在 SDK2.0 KE1xF 发布包中。

2 FlexMemory 功能

2.1 FlexMemory 组件

图 1 显示了 KE1xF 系列设备上的闪存存储模块和 FlexMemory 组件。

闪存模块包括程序闪存和 FlexNVM；FlexMemory 模块由 FlexNVM 和 FlexRAM 组成，增强 EEPROM (EEE) 由 FlexNVM、FlexRAM 和 EEE 状态机中的 EEPROM 备份组成。FlexNVM 和 FlexRAM 是唯一用于增强型 EEPROM 的内存。



目录

1	简介.....	1
2	FlexMemory 功能.....	1
2.1	FlexMemory 组件.....	1
2.2	FlexNVM 功能.....	2
2.3	FlexRAM 功能.....	2
2.4	FlexMemory 分区.....	2
3	D-Flash 实现.....	5
4	EEPROM 实现.....	6
4.1	EEE 启动.....	6
4.2	EEE 读写.....	6
5	演示示例.....	7
5.1	Flexnvm_dflash 演示.....	8
5.2	Flexnvm_eeprom 演示.....	10
6	结论.....	11
7	修订记录.....	12

2.2 FlexNVM 功能

当对 FlexNVM 进行数据闪存分区时：

- 扇区大小为 2 KB
- 保护方案防止程序意外或存储数据的擦除
- 具有验证功能的自动内置程序和擦除算法
- 分区编程，加快批量编程时间
- 编程或擦除程序闪存块中的数据时，可以读取数据闪存模块。

2.3 FlexRAM 功能

- 可作为传统 RAM 或高耐久性 EEPROM 存储器使用的内存
- 为 EEPROM 或传统 RAM 操作配置的 4 KB FlexRAM
- 当为 EEPROM 配置时：
 - 保护方案可防止程序意外或为 EEPROM 写入数据的擦除
 - 嵌入式硬件仿真方案，实现 EEPROM 记录维护功能的自动化
 - 可编程的 EEPROM 数据集大小和 FlexNVM 分区代码有助于 EEPROM 存储器耐久性的权衡
 - 支持一次 1、2 或 4 字节的 FlexRAM 对齐写入
 - 在编程或擦除程序或数据闪存中的数据时，可以读取 FlexRAM
- 为传统 RAM 配置时：
 - 在编程或擦除程序或数据闪存中的数据时，可以对 FlexRAM 进行读写访问。

2.4 FlexMemory 分区

用户可以将 FlexNVM 块配置为：

- 普通 Flash
- EEPROM
- 两者的结合

用户的 FlexNVM 配置选项是使用程序分区命令指定的。分区过程告诉 EEE 状态机将使用多少 EEPROM 内存以及将使用多少 FlexNVM 来备份 EEPROM。

为了满足不同的客户需求，可以将 FlexRAM 和 FlexNVM 块拆分为分区，如 [图 2](#) 所示。

1. EEPROM 分区 (EEESIZE) — 用于 EEPROM 的 FlexRAM 的数量可以从 0 字节 (无 EEPROM) 到最大 FlexRAM 大小 (KE1xF 中为 4 KB)。当 FlexRAM 被配置为 EEPROM 时，FlexRAM 未被 EEPROM 分配的剩下的部分不能被访问。EEPROM 分区从 FlexRAM 地址 (KE1xF 中的 0x1400 0000) 空间的底部向上增长。
2. 数据闪存分区 (DEPART) — 用于数据闪存的 FlexNVM 内存量可以从 0 字节 (所有 FlexNVM 块都可用于 EEPROM 备份) 编程到 FlexNVM 块的最大大小 (KE1xF 中为 64 KB)。
3. FlexNVM EEPROM 备份分区 — 用于 EEPROM 备份的 FlexNVM 内存量，等于 FlexNVM 块总大小减去数据闪存分区大小。EEPROM 备份大小必须至少是 FlexRAM 中 EEPROM 分区大小的 16 倍。

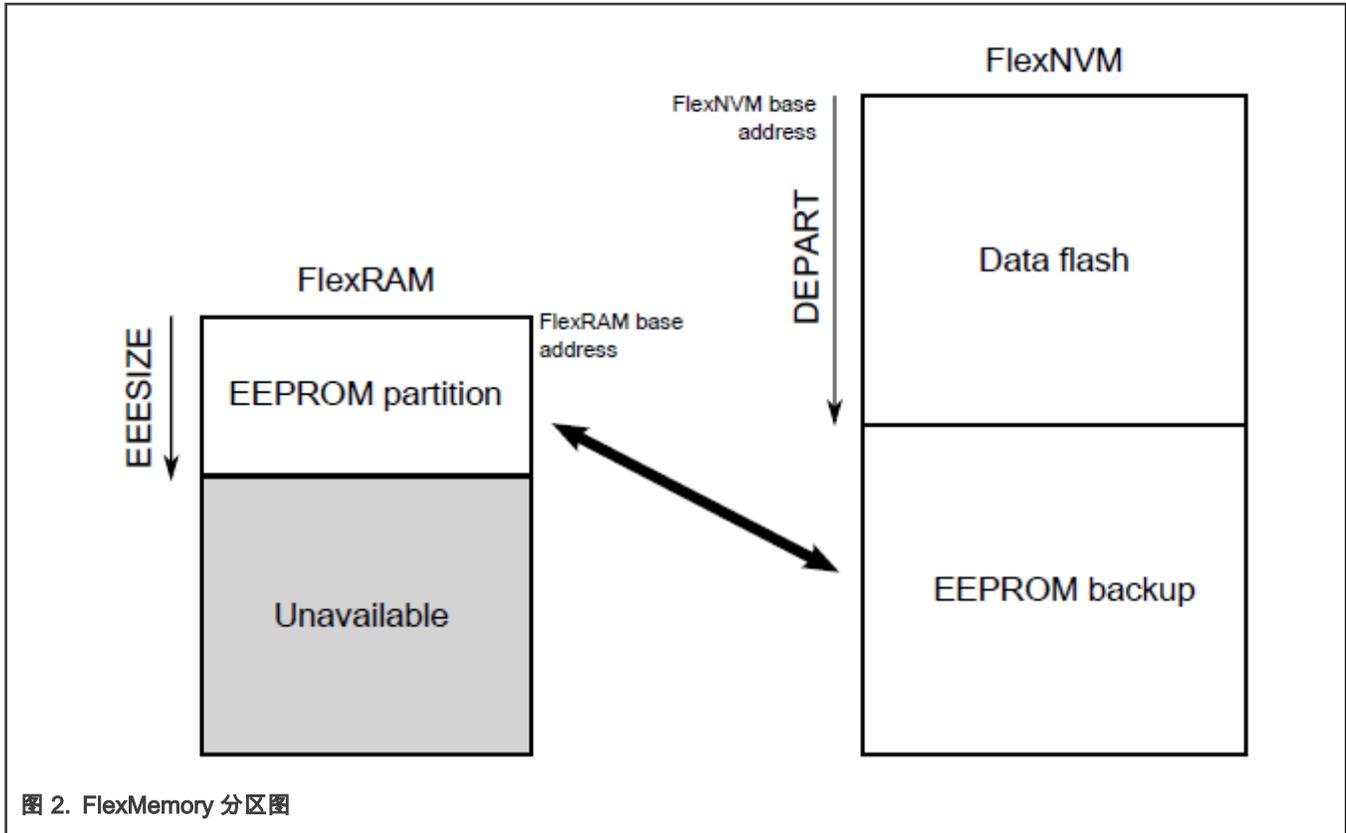


图 2. FlexMemory 分区图

2.4.1 程序分区命令

程序分区命令，如 [表 1](#) 所示，FlexNVM 模块可用作数据 Flash、EEPROM 备份或两者的组合，并初始化 FlexRAM 大小。程序分区命令不能从闪存启动，因为在程序分区命令执行期间，闪存资源是不可访问的。

表 1. 程序分区命令 FCCOB 要求

FCCOB 编号	FCCOB 含量 [7:0]
0	0x80 (PGMPART)
1	未使用
2	未使用
3	复位期间 FlexRAM 加载选项 (仅使用位 0) : <ul style="list-style-type: none"> • 0 — FlexRAM 在复位过程中加载了有效的 EEPROM 数据 • 1 — 复位过程中未加载 FlexRAM
4	EEPROM Data Set Size Code 选项 : 0 , 32 , 64 , 128 , 256 , 512 , 1024 , 2048 , 4096 字节
5	FlexNVM Partition Code 选项 : 0 , 32 , 48 , 64 KB

2.4.2 Set FlexRAM Function 命令

如表 2 所示，Set FlexRAM Function 命令可用于更改 FlexRAM 的功能：

- 当不为 EEPROM 分区时，FlexRAM 通常用作传统 RAM。
- 为 EEPROM 分区时，FlexRAM 通常用于存储 EEPROM 数据。

表 2. 设置 FlexRAM 功能命令 FCCOB 要求

FCCOB 编号	FCCOB 含量 [7:0]
0	0x81 (SETRAM)
1	FlexRAM 功能控制代码 选项： <ul style="list-style-type: none"> • 0xFF：使 FlexRAM 可用作 RAM • 0x00：使 FlexRAM 可用于 EEPROM

2.4.3 读取资源命令

如表 3 所示，Read Resource 命令用于用户从位于闪存模块内的专用存储器资源读取数据。可用的专用内存资源包括程序 flash IFR、数据 flash IFR 空间和版本 ID 字段。

对于 FlexMemory 的使用，需要数据闪存 IFR 空间，如表 4 所示。

表 3. 设置 FlexRAM 功能命令 FCCOB 要求

FCCOB 编号	FCCOB 含量 [7:0]
0	0x03 (RDRSRC)
1	闪存地址 [23:16]
2	闪存地址 [15:8]
3	闪存地址 [7:0]
4	资源选择代码 选项： <ul style="list-style-type: none"> • 0x00：Program Flash 0 IFR • 0x00：Data Flash 0 IFR • 0x01：Version ID
返回的值	
4	读取数据 [64:56]
5	读取数据 [55:48]
6	读取数据 [47:40]
7	读取数据 [39:32]

下页继续...

表 3. 设置 FlexRAM 功能命令 FCCOB 要求 (续上页)

FCCOB 编号	FCCOB 含量 [7:0]
8	读取数据 [31:24]
9	读取数据 [23:16]
A	读取数据 [15:8]
B	读取数据[7:0]

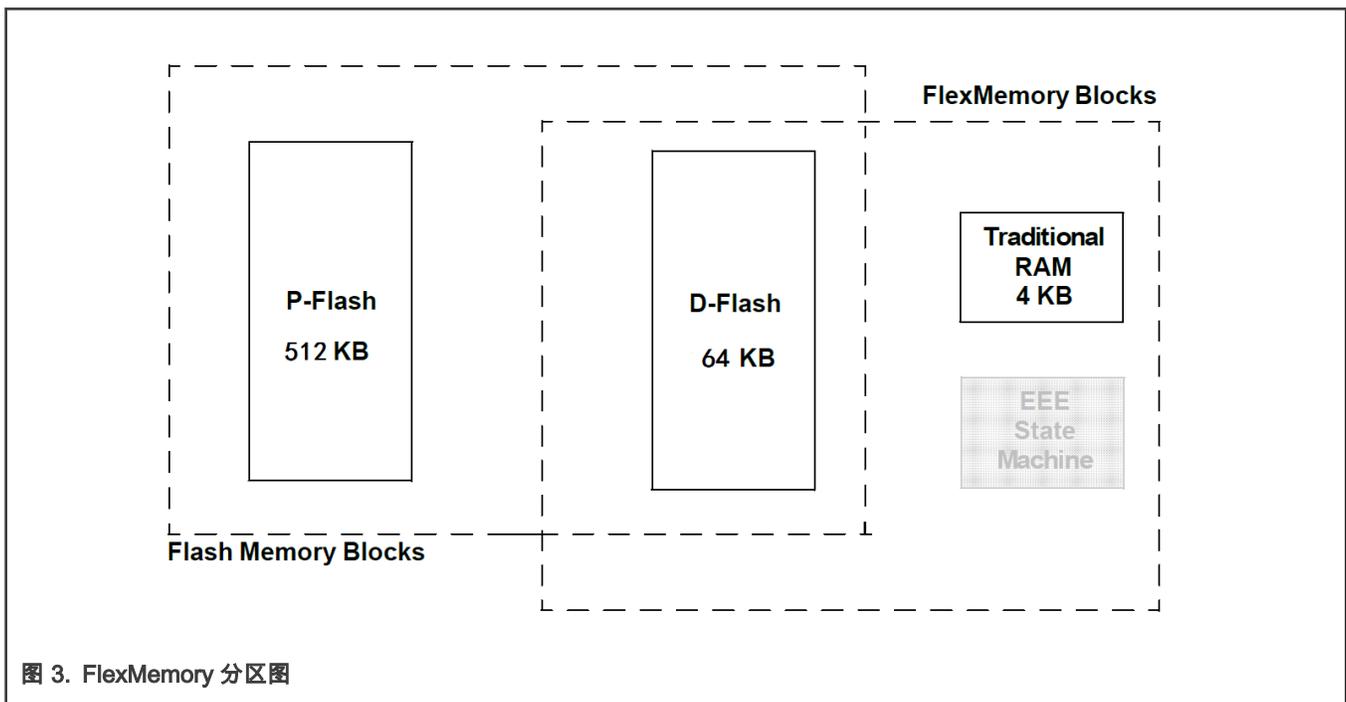
数据闪存 0IFR 是 1 KB 的非易失性信息存储器，可以读取和擦除，但是用户在数据闪存 0IFR (程序分区命令、擦除所有块命令和读取资源命令) 中的程序能力有限。表 4 总结了数据闪存 0IFR 的内容。数据闪存 0IFR 位于数据闪存 0 内存块内。

表 4. 数据闪存 IFR

地址范围 (偏移地址)	大小 (字节)	字段描述
0x00–0x3FB、0x3FE–0x3FF	1022	保留
0x3FD 型	1	EEPROM 数据集大小代码
0x3FC 型	1	FlexNVM 分区代码

3 D-Flash 实现

图 3 显示了当增强型 EEPROM (EEE) 功能被禁用时，内存模块如何工作，这意味着所有 FlexNVM 都用作 D-Flash，FlexRAM 用作传统 RAM。



P-Flash 模块始终是 P-Flash。任何 FlexMemory 配置都不会改变其功能。因为在这种情况下不使用 EEE 功能，所以整个 FlexNVM 被分配为 D-Flash 空间，不需要 E-Flash (EEPROM 备份)。FlexRAM 变成了 4 KB 的传统 RAM。这意味着它可以被用作额外的内存空间，但是需要注意的是，它以闪存时钟速度而不是内核速度运行。设备中存在 EEE 状态机，但未激活。当用作 D-Flash

时，读、写和程序操作与 P-Flash 块相同，FCCOB 命令序列也相同。用户可以寻址位 23，在程序闪存 (=0) 和数据闪存 (=1) 之间进行选择。

对于需要快速存储大量数据或存储静态数据的应用程序，D-Flash 内存非常有用。

4 EEPROM 实现

图 4 显示了当增强 EEPROM (EEE) 功能被启用并且整个 FlexNVM 被用来备份 EEE 数据时，内存模块是如何工作的。

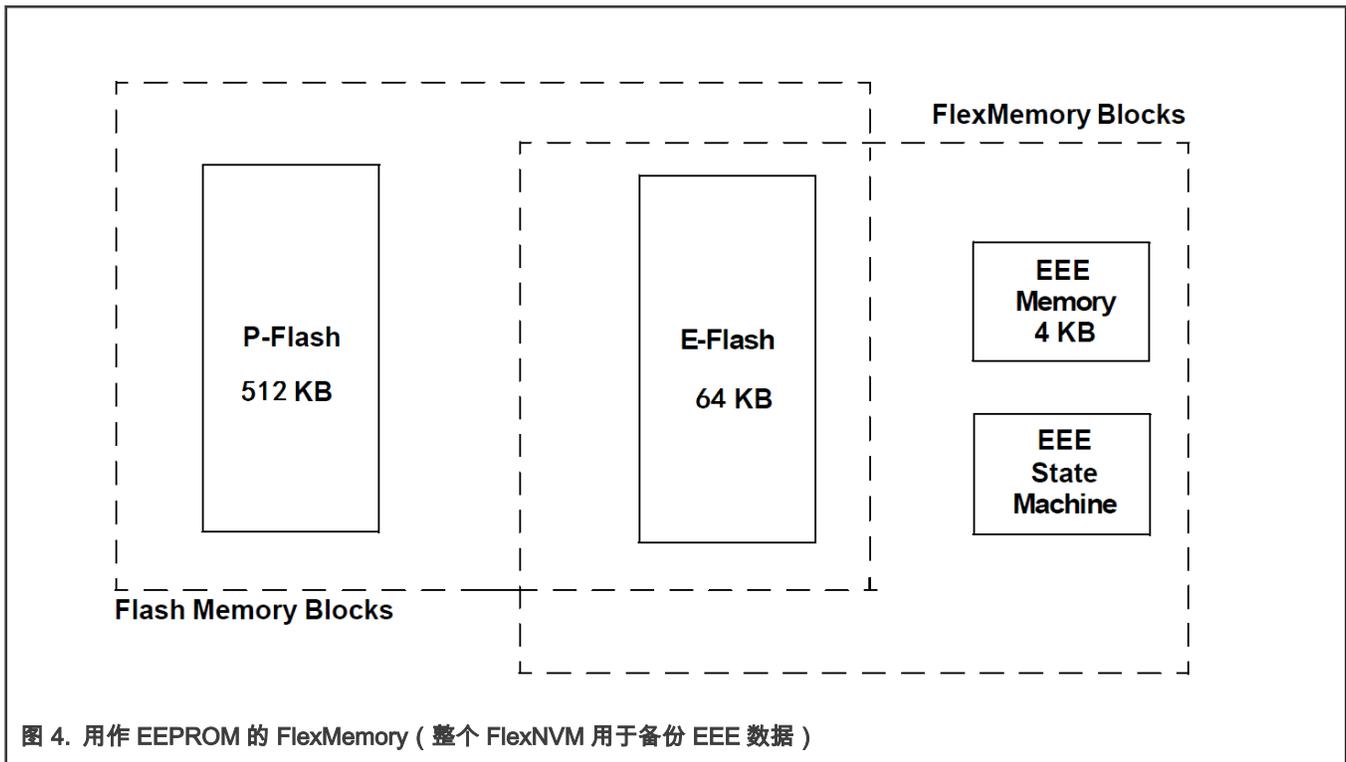


图 4. 用作 EEPROM 的 FlexMemory (整个 FlexNVM 用于备份 EEE 数据)

启用增强型 EEPROM (EEE) 功能时，可以使用许多配置选项。FlexNVM 还可以混合使用 D-Flash 和 E-Flash (EEPROM 备份)。图 4 显示了将整个 FlexNVM 用作 E-Flash (EEPROM 备份) 存储器的示例。FlexRAM 作为 EEE 中 4 KB 的内存空间。因为 E-Flash 不能直接访问，EEE 数据的任何读写都会使用这个 4 KB 的内存空间。EEE 状态机自动管理对 EEE 内存空间的所有写入，并根据需要生成 Flash 程序和擦除操作到 E-Flash 中。

4.1 EEE 启动

重置后，分区过程中写入的 EEE 配置选项将自动加载。如果启用了 EEE，则状态机将来自的 EEE 数据加载到 FlexRAM。

在系统引导期间使用 E-Flash。将数据从 E-Flash 复制到 FlexRAM 所需的时间量可能因 EEE 的配置大小和需要解析的备份 E-Flash 的数量而异。在 EEE 数据加载完成之前，清除 FTFE_FCNFG[EEERDY] 标志，因此软件必须等待 EEERDY 标志设置，然后才能尝试访问 FlexRAM 中的 EEE 数据。如果需要中断驱动选项而不是软件轮询，那么可以使用 CCIF 中断而不是轮询 EEERDY。

4.2 EEE 读写

通过访问 FlexRAM 地址空间来读取和写入 EEE 数据。EEE 空间从 FlexRAM 开始分配。可寻址空间是 FlexRAM 基址 (0x1400_KE1xF 中的 0000)，最大为编程的 EEE 大小。启用 EEE 功能时，不得访问 FlexRAM 中未用作 EEE 的任何空间。例如，如果 EEE 大小配置为总共 32 个字节，则允许对 0x1400_0000 和 0x1400_001F 之间的任何地址进行读写访问，但对 0x1400_0020 到 0x17FF_FFFF 的访问会产生总线错误。

4.2.1 EEE 写入

向 EEE 空间写入一个 EEE 操作，将数据存储在 E-Flash 存储器中。因为这是一个 Flash 编程操作，所以在写入 EEE 空间之前，软件必须轮询 CCIF 位以确定是否有任何其他 Flash 操作正在进行。由于不允许在同一闪存块内执行多个并发写入和 read-while-write 操作，因此在 EEE 写入完成之前，不允许访问 EEE 或 D-Flash 空间。

4.2.2 EEE 读取

读取 EEE 时，由于数据由 FlexRAM 提供，因此不会触发闪存编程操作。

但是，在 EEE 写入过程中不允许 EEE 的读取。软件必须在读操作之前轮询 EEERDY 位，或者在写访问之后等待 EEERDY 位，然后才允许软件继续。在许多情况下，软件在写之前和之后轮询 EEERDY (或 CCIF) 并阻止其他 EEE 操作直到 EEERDY 在写之后设置为止是最有效的。这样，EEE 写入需要一个特殊的函数，但是 EEE 读取不需要任何特殊的操作。这种方法的另一个优点是，如果您有多个 EEE 读访问，并且其间没有 EEE 写周期，则不需要额外的延迟或标志检查。

必须考虑的 EEE 读取的一个特殊情况是复位后第一次访问 EEE。对于复位后 EEE 的第一次读取，可能需要轮询 EEERDY 位，以确保状态机已完成从 E-Flash 到 FlexRAM 的初始数据加载。如果系统启动时间长，这保证了初始数据加载在第一次 EEE 读取之前有时间完成，那么在第一次读取之前可能不需要对 EEERDY 标志进行测试。但是，在对 EEE 进行第一次读取访问之前显式测试 EEERDY 位更安全。

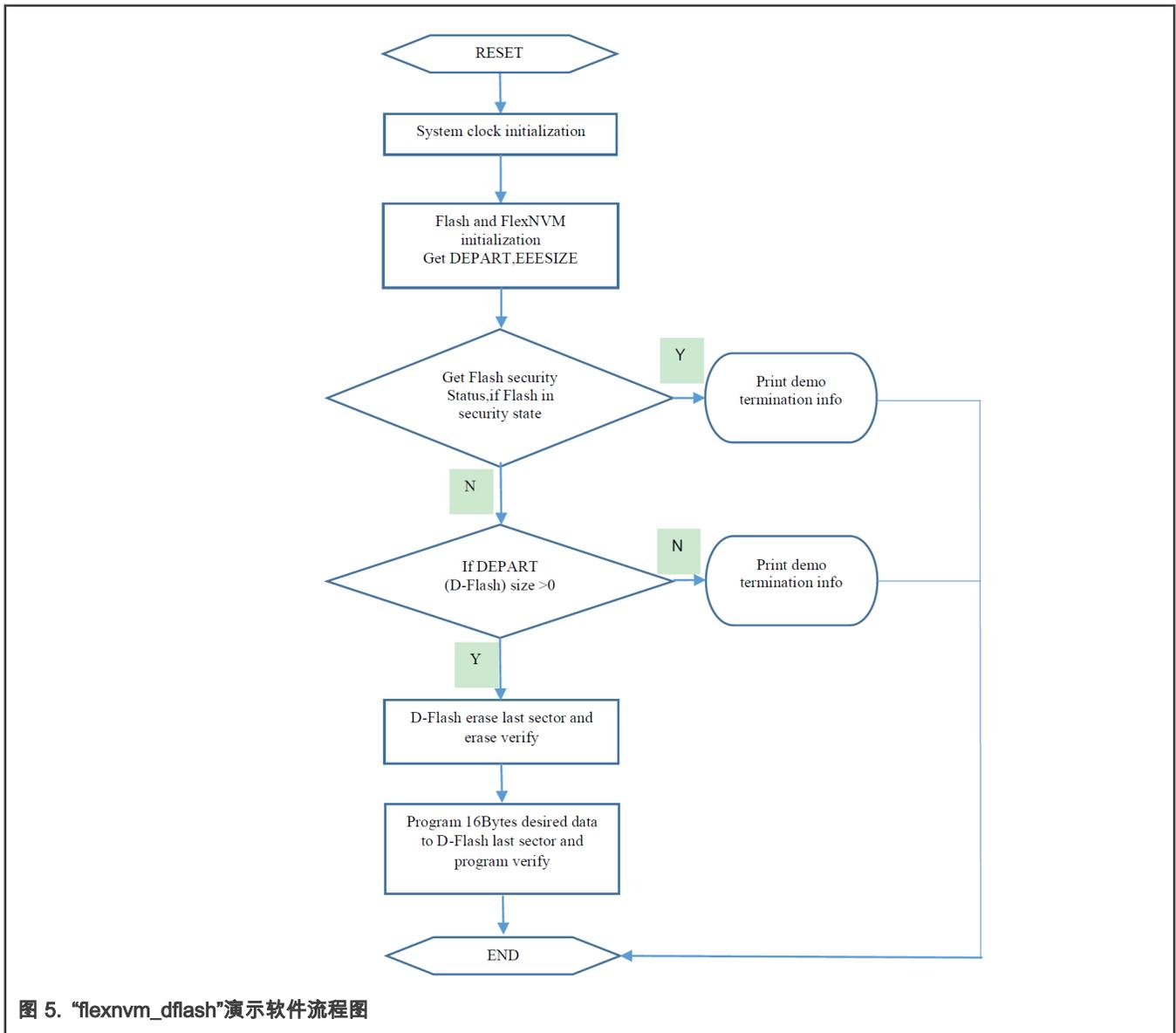
5 演示示例

在 SDK2.0 KE1xF 发行包中的 **flash driver example** 文件夹下有两个演示项目。

- “flexnvm_dflash” 演示，显示了如何使用 flexmemory 和 SDK 闪存驱动程序作为 D-Flash。
- “flexnvm_eeprom” 演示，显示了 EEPROM 的用法。

图 5 显示了“flexnvm_dflash”的软件流程图项目。图 8 显示了“flexnvm_eeprom”项目的软件流程图。在 Flash 中下载并执行代码后，我们可以使用 J-link commander 工具来检查内存的读写和编程状态。

5.1 Flexnvm_dflash 演示



默认情况下，KE1xF 64KB FlexNVM 没有分区，完全用作 D-Flash。用户可以通过分区来配置所需的 EEPROM 大小。下载并执行“flexnvm_dflash”演示后，日志消息将打印在 OpenSDA 串行终端上，如 图 6 所示。当使用 J-link commander 工具检查内存状态时，预定义的 16 字节数据被编程到 D-Flash 的最后一个扇区，如 图 7 所示。

5.2 Flexnvm_eeprom 演示

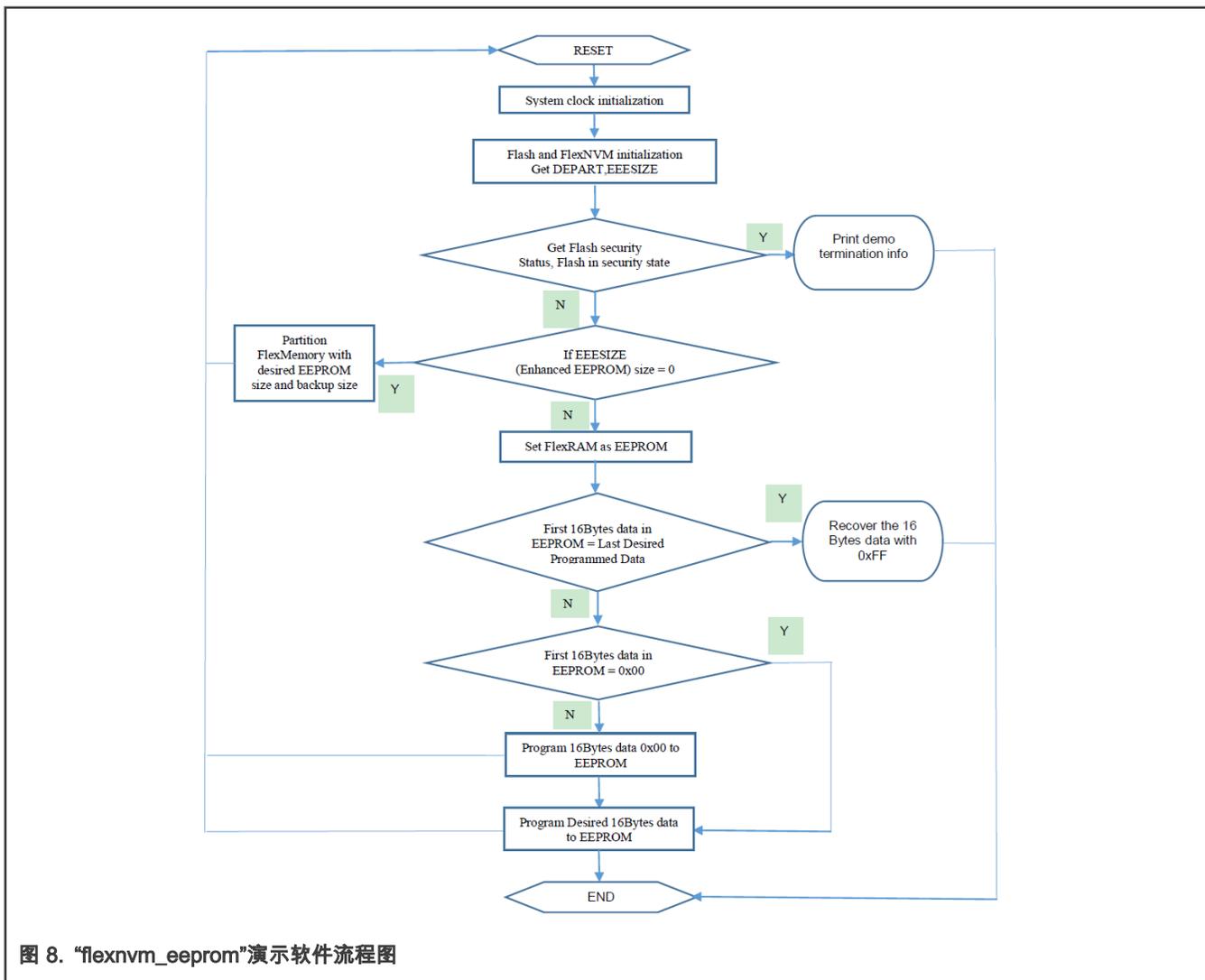


图 8. “flexnvm_eeprom”演示软件流程图

默认情况下，KE1xF 64 KB FlexNVM 没有分区，完全用作 D-Flash。用户可以通过分区来配置所需的 EEPROM 大小。下载并执行“flexnvm_eeprom”演示后，日志消息将打印在 OpenSDA 串行终端上，如 图 9 所示。FlexNVM 被配置为 D-Flash 和 E-Flash (EEPROM 备份) 的混合使用。两者的大小都是 32 KB。FlexRAM 中的 EEPROM 数据大小配置为 32 字节。如果我们在源代码中屏蔽恢复数据代码并使用 J-link commander 工具检查内存状态，可以看到预定义的 16 字节数据被编程到 FlexRAM 的起始地址，如 图 10 所示。

注意

虽然 FlexNVM 有不同的分区可用，但其目的是在给定应用程序的整个生命周期中使用单个分区选择。FlexNVM 分区代码的选择会影响 EEPROM 的大小和最大写入次数。所以分区只能执行一次。如果闪存被重新划分为不同的配置，那么记录的数据将丢失，并且不能保证获得预期的擦写寿命。如果用户想修改“flexnvm_eeprom”演示中的参数，对 EEPROM 大小进行另一种配置，必须首先发出一个大批量擦除 (mass erase) 命令。

```

COM24 - PuTTY
FlexNVM EEprom Example Start

Flash is UNSECURE!

EEprom Information:
EEprom Base Address: (0x10008000)
EEprom Total Size: 32 B
Make FlexRAM available for EEPROM
Now EEPROM data is read and written by accessing the FlexRAM address space
FlexRAM Base Address: (0x14000000)
Read 16 bytes data from start of EEPROM space
The first 16 bytes data in EEPROM are not all 0x00s
Program the first 16 bytes memory space of EEprom as 0x00s
Successfully Programmed Location 0x14000000 -> 0x14000020

Perform a system reset
FlexNVM EEprom Example Start

Flash is UNSECURE!

EEprom Information:
EEprom Base Address: (0x10008000)
EEprom Total Size: 32 B
Make FlexRAM available for EEPROM
Now EEPROM data is read and written by accessing the FlexRAM address space
FlexRAM Base Address: (0x14000000)
Read 16 bytes data from start of EEPROM space
The first 16 bytes data in EEPROM are all 0x00s
Program a buffer(16 bytes) into the first 16 bytes memory space of EEprom
Successfully Programmed Location 0x14000000 -> 0x14000020

Perform a system reset
FlexNVM EEprom Example Start

Flash is UNSECURE!

EEprom Information:
EEprom Base Address: (0x10008000)
EEprom Total Size: 32 B
Make FlexRAM available for EEPROM
Now EEPROM data is read and written by accessing the FlexRAM address space
FlexRAM Base Address: (0x14000000)
Read 16 bytes data from start of EEPROM space
The first 16 bytes data in EEPROM are what we have programmed before
Recover the first 16 bytes memory space of EEprom as 0xFFs
End of FlexNVM EEprom Example
    
```

图 9. “flexnvm_eeeprom”演示终端上的日志打印

```

J-Link>mem 14000000,10
14000000 = 01 00 00 00 02 00 00 00 03 00 00 00 04 00 00 00
J-Link>mem 14000000,20
14000000 = 01 00 00 00 02 00 00 00 03 00 00 00 04 00 00 00
14000010 = FF FF
    
```

图 10. 使用 “flexnvm_eeeprom”演示的 J-link 命令检查内存

6 结论

本笔记概述了 Kinetis KE1xF 设备的功能、用户视角以及如何将 FlexMemory 作为 D-Flash 和 EEPROM 实现。介绍了两个演示项目（包含在与本应用说明相关的 SDK2.0 KE1xF 发行包中），以帮助更好地理解 FlexMemory 功能实现，然后帮助用户使用其应用程序中的功能。

7 修订记录

表 5. 修订记录

版本号	日期	说明
0	2016 年 9 月	初始版本

How To Reach Us

Home Page:

nxp.com

Web Support:

nxp.com/support

Limited warranty and liability — 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.

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.

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 marks and the Power and Power.org logos and related marks are trademarks and service marks licensed by Power.org. M, M Mobileye and other Mobileye trademarks or logos appearing herein are trademarks of Mobileye Vision Technologies Ltd. in the United States, the EU and/or other jurisdictions.

© NXP B.V. 2016-2021.

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: 2016 年 9 月

Document identifier: AN5338

