

1 引言

LPC546xx 具有外部 SPI FLASH 接口(SPIFI), 可以采用单线/双线/四线模式访问外部 FLASH。它还支持 XIP 模式, 这意味着代码可以直接在 SPI 闪存上执行。SPIFI 大大扩展了应用程序的代码大小, 并使存储大数据(图像甚至视频)成为可能)。然而, 在开发阶段将数据下载到 spi flash 中总是一个麻烦的问题, 因为不同的 spi 闪存供应商有不同的命令集。

客户经常遇到 MDK 中原有的 flash 编程算法不能适应其硬件的情况, 导致下载数据不正确。

这个简短的应用笔记给出了这个问题的解决方案。它提供了如何为 MDK 创建闪存编程文件的一步指南, 并提供了一种可直接使用的预编译闪存编程算法。

此应用笔记适用于:

1. 使用带有 LPC546xx/LPC540xx 的 MDK, 并希望为自己的 SPI FLASH 部分创建一个闪存编程算法的读者。
2. 想了解 LPC546xx SPI 外部设备的读者。

1.1 术语表

表 1. 术语

缩写	说明
SPIFI	在 LPC546xx 系列上的 SPIFI 模块, 用于访问外部 SPI FLASH。
MDK (Keil)	Keil 集成开发环境
FLM	MDK 的闪存编程算法文件 (ELF 格式文件)。

2 执行情况

2.1 概述

Flash 编程算法 (对于 MDK, 简称 FLM) 是一种将应用程序擦除或下载到 Flash 设备的软件。它通常包含由 DFP 支持的预定义 FLM 文件。CMSIS 包中有一个创建算法的模板。在 MDK 中, Flash 编程算法是一个 FLM 文件。事实上, FLMS 文件格式就是 FLF 格式。

LPC546xx 中的 SPIFI 是用于访问外部 SPI FLASH 的外设。目前, 如果您想通过 MDK 将数据下载到外部 SPI Flash 中, 只有 LPC540xx_MT25QL128, FLM 可在 DFP 软件包上使用, 但这只支持 MT25QL128。虽然该算法可以支持大多数 MICRON SPI Flash 部件, 但通常不能支持另一个供应商, 如 Winbond 或 ISSI。原因很明显: LPC540xx_MT25QL128FLM 算法使用四线制烧写和四线制读取指令, 这些读写指令在不同厂家之间不兼容。

为了解决这个问题, 我们必须创建一个通用的 FLM 文件, 它与市场上大多数 SPI Flash 兼容。虽然, 可能会有一些折衷, 为了实现这一目标, 我们必须首先在 LPC546xx 上实现 SPIFI 驱动程序, 该驱动程序用于创建 flash 编程算法。

目录

1	引言	1
1.1	术语表.....	1
2	执行情况	1
2.1	概述.....	1
2.2	SPIFI.....	2
2.3	如何修改 SPIFI 驱动程序以支持新的 SPI Flash 设备.....	2
2.4	创建 Flash 编程算法文件.....	2
3	如何使用 FLM 文件	3
3.1	将 FLM 添加到 Flash 编程算法列表中.....	3
3.2	确定结果.....	4
4	结论	5



2.2 SPIFI

MCUXpresso SDK 提供了用于访问外部 SPI FLASH 的 SPIFI 驱动程序，以及 SPIFI 的示例代码。SPIFI 示例如下：

```
\boards\lpcxpresso54608\driver_examples\spifi\polling_transfer
```

在创建 FLM 文件之前，您需要 SPIFI 驱动程序代码来让您的读写外部 SPI FLASH。此驱动程序代码和示例在 polling_transter 演示中。确保您可以在硬件平台上成功地运行 polling_transfer 演示，此演示使用的 SPIFI 驱动程序代码在闪存编程算法中重用。如果 polling_transter 演示无法在硬件上成功运行，请参阅[如何修改 SPIFI 驱动程序以支持新的 SPI Flash 设备](#)。

2.3 如何修改 SPIFI 驱动程序以支持新的 SPI Flash 设备

有一些原因导致这个演示无法运行：

- 不兼容的引脚配置
- spi flash 命令设置不兼容

2.3.1 不兼容的引脚配置

这个问题很容易找到，也很容易解决。确保 SPI 闪存的硬件连接与 LPC546xxpresso 板相同，否则，必须修改 pin_mux 中的函数 BOARD_InitPins 以与硬件对齐。

2.3.2 配置 SPI 闪存命令列表

不同的 spi 闪存产品有不同的命令操作数，特别对于四线擦除和四线编程功能。几乎所有 spi flash 产品都使用相同的命令操作数进行单擦除、单读和单编程。处理不兼容命令操作数的一个简单方法是使用单读和单程序函数。这种方法可能会导致编程速度较慢。图 1 显示使用单线读取和编程命令的 SPIFI 命令表，此命令列表可兼容市场上的大多数 SPI Flash。

```
#define COMMAND_NUM (6)
#define READ (0)
#define PROGRAM_PAGE (1)
#define GET_STATUS (2)
#define ERASE_SECTOR (3)
#define WRITE_ENABLE (4)
#define WRITE_REGISTER (5)

spifi_command_t command[COMMAND_NUM] = {
    {PAGE_SIZE, false, kSPIFI_DataInput, 0, kSPIFI_CommandAllSerial, kSPIFI_CommandOpcodeAddrThreeBytes, 0x03}, /* single wire read */
    {PAGE_SIZE, false, kSPIFI_DataOutput, 0, kSPIFI_CommandAllSerial, kSPIFI_CommandOpcodeAddrThreeBytes, 0x02}, /* program page */
    {1, false, kSPIFI_DataInput, 0, kSPIFI_CommandAllSerial, kSPIFI_CommandOpcodeOnly, 0x05}, /* get status */
    {0, false, kSPIFI_DataOutput, 0, kSPIFI_CommandAllSerial, kSPIFI_CommandOpcodeAddrThreeBytes, 0x20}, /* erase 4K sector */
    {0, false, kSPIFI_DataOutput, 0, kSPIFI_CommandAllSerial, kSPIFI_CommandOpcodeOnly, 0x06}, /* write enable */
    {1, false, kSPIFI_DataOutput, 0, kSPIFI_CommandAllSerial, kSPIFI_CommandOpcodeOnly, 0x01}}; /* write register */
```

图 1. 用于 SPI Flash 操作的通用 LUT

2.4 创建 Flash 编程算法文件

2.4.1 实现 Flash 编程代码

闪存编程算法的定义与功能擦除和编程闪存设备。需要特殊的编译器和链接器设置。关于如何创建闪存编程算法的一步指南是可用的，请参阅 [CMSIS 文档](#)

以下是一些重要的重点和提示：

1. 不要创建一个新的 uVision 项目，从 ARM : CMSISPack 文件夹 (通常是 C : \Keil\ARM\Pack)复制内容) 将 \ARM\CMSIS\version\Device_Template_Flash)转换为新文件夹。
2. 闪存编程算法没有 main 函数入口，它是一个位置独立的代码。
3. 文件 FlashPrg.c 包含 Flash 编程功能 `init`, `UnInit`, `EraseSector` 还有 `ProgramPage`。可选地，根据设备特性 (或加快执行速度)，这些功能 `EraseChip`, `BlankCheck`，还有 `Verify` 可以实现，如表 2 显示：

表 2. Flash 编程算法 API 需要实现

函数名称	指示	说明
BlankCheck	可选的	检查和比较模式
EraseChip	可选的	删除整个 Flash 内存内容
EraseSector	强制性的	删除特定 secto 的 Flash 内存内容
Init	强制性的	为 Flash 编程初始化和准备设备
ProgramPage	强制性的	将应用程序写入 Flash 内存
UnInit	强制性的	在 Flash 编程步骤之一之后，取消对微控制器的初始化。
Verify	可选的	将 Flash 内存内容与程序代码进行比较。

与不同供应商保持最大兼容性的功能已经在附加软件中实现。Read 和 ProgramPage API 使用 SPI 一线模式来保持与不同闪存供应商的兼容性。

1. 文件 **Flash Dev.c** 包含由 MDK 识别的 Flash 设备结构的参数定义。此结构包含所有闪存描述，包括名称、闪存大小、扇区大小、开始地址等如 图 2 演示。

```

struct FlashDevice const FlashDevice = {
    FLASH_DRV_VERS,           // Driver Version, do not modify!
    "LPC5460x SPIFI ALL", // Device Name
    EXTSPI,                   // Device Type
    0x10000000,              // Device Start Address
    0x01000000,              // Device Size (16MB)
    4096,                    // Programming Page Size (16 pages a 256)
    0,                       // Reserved, must be 0
    0xFF,                    // Initial Content of Erased Memory
    300,                     // Program Page Timeout 300 mSec
    3000,                    // Erase Sector Timeout 3000 mSec

    // Specify Size and Address of Sectors
    0x001000, 0x000000,     // Sector Size 4kB (4096 Sectors)
    SECTOR_END
};

```

图 2. Flash 编程算法 API 需要实现

2.4.2 编译并创建 FLM 文件

在完成实现 **Flash Dev.c** 和 **FlashPrg.c** 之后，编译整个工程。将生成输出文件。在附件软件中，输出名称为 **LPC5460x_SPIFI_ALL.FLM**。

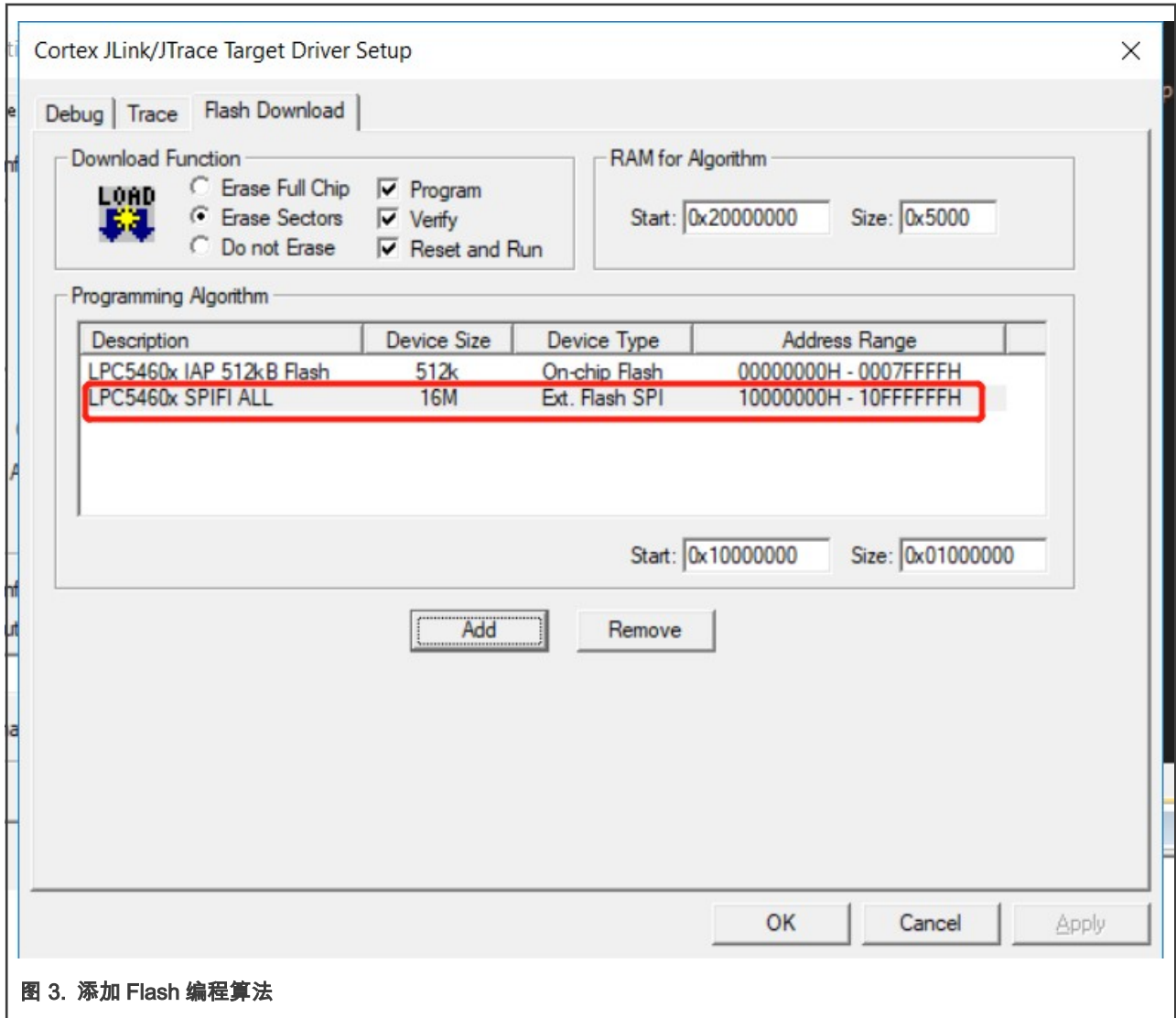
3 如何使用 FLM 文件

3.1 将 FLM 添加到 Flash 编程算法列表中

执行以下步骤将 FLM 添加到 Flash 编程算法列表中：

1. 将 FLM 文件复制到 **ARM\Keil\ARM\Flash** 文件夹中。

2. 打开您的应用程序 MDK 项目。打开 Flash 下载对话框，将 LPC5460xSP 全部添加到编程算法中。
3. 如果存在，请删除以前的 SPI Flash FLM 文件。这将使 MDK 在将数据下载到 SPI Flash 时使用我们的 FLM，如 图 3 演示。



3.2 确定结果

单击“加载”按钮开始下载应用程序，如 图 4 演示。



MDK 在 Build Output 窗口中显示输出日志。日志“验证 OK”指示您的应用程序成功下载到 MCU，如 图 5 演示。

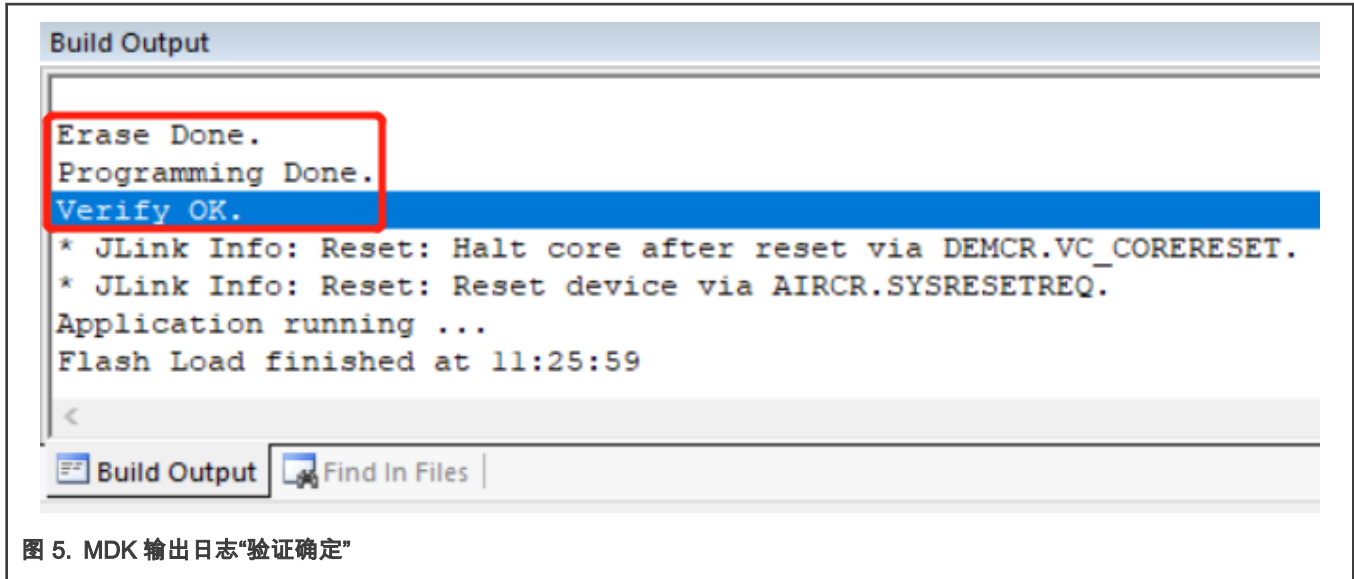


图 5. MDK 输出日志“验证确定”

4 结论

本应用笔记介绍了如何为 LPC546xx 的外部 SPI FLASH 设计 MDK 专用的编程算法。它还提供了一个预先编译的，准备使用 FLM 文件，可以支持市场上的大多数 SPI Flash。

为了保持兼容性，附加软件中的 FLM 项目使用默认的核心时钟、单线读取和单线编程命令操作数，下载速度比使用四模式的慢。

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 word 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. 2019-2021.

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: 2019 年 8 月 26 日

Document identifier: AN12563

