AN12130

S32K1xx MCU 的闪存编程最佳实践

Rev. 1 - 2019年9月 应用笔记

URL：https://www.nxp.com/docs/en/application-note/AN12130.pdf

目录

1 介绍.....1
2 编程接口.....1
3 编程步骤.....1
4 其他编程方面的考虑事项。.....5
5 常见问题.....6
6 进一步阅读.....7
7 修订历史.....7

1 介绍

本应用笔记介绍了在 S32K 器件上进行生产用闪存编程的推荐步骤。与微控制器通信以对闪存进行编程的物理接口有不同的选项，并有多个命令可用于修改闪存内容。

本应用笔记将讨论不同的物理接口、闪存命令，并提供推荐的序列，以最大限度地减少闪存编程时出现的问题并减少闪存编程时间。 目标受众是开发生产闪存编程硬件、生产用闪存编程软件或支持闪存编程的调试器接口的任何人。

尽管本文档中的某些信息可能对软件控制的现场固件更新有用，但本文并不涵盖此主题。

2 编程接口

有一个物理接口可以用于S32K1xx设备上的生产闪存编程：SWD/JTAG。

2.1 SWD/JTAG（串行线调试/联合测试行动组）

S32K14x 器件支持与 ARM 调试访问端口 (DAP) 相连的 SWD 和 JTAG 接口。 S32K11x 器件仅支持 SWD 接口（JTAG 仅支持边界扫描寄存器或 BSR）。 SWD 和 JTAG 是不同的物理接口，但用于闪存编程的命令和寄存器的访问是相同的。为简单起见，本文档的其余部分将使用术语 SWD，但所描述的操作可用于 SWD 或 JTAG。 SWD 是处理器的主要调试接口。除调试外，SWD 还可用于闪存编程。通常，带有支持所需闪存命令的软件的闪存编程算法将通过 SWD 加载到微控制器的片上 RAM 中，然后 SWD 将用于运行控制闪存擦除和从 RAM 执行编程代码。 SWD 还可用于直接写入闪存寄存器以针对所需命令配置它们。发送 SWD 命令比在内核上执行代码要慢，因此通常不推荐这种方法，因为如果 SWD 直接写入闪存 FCCOB 寄存器，闪存编程时间将显着延长。

3 编程步骤

以下部分描述了闪存编程的主要过程。

3.1 连接

对闪存flash进行编程的第一步是在所需的接口SWD上建立连接。

311 SWD连接步骤

下面列出了通过SWD建立与处理器连接的推荐步骤。

1. 给处理器上电，或者如果已经上电，则置位 RESET 引脚以复位处理器。 对于没有 RESET 引脚的设备，在建立通信后写入 MDM-AP 控制寄存器中的系统复位请求位。

2.保持复位低电平并与ARM DAP建立通信。 可以读取 MDM-AP ID 寄存器以验证连接是否正常工作。

3. 读取 MDM-AP 状态寄存器，直到 Flash Memory Ready 位置位。

4. 读取系统加密位以确定是否启用了加密特性。 如果系统加密特性 = 0，则继续。 如果系统加密特性= 1，如果不发出整体擦除命令或通过其他方式（例如，后门钥匙解锁）解除部件的加密特性，则无法与处理器内部（包括闪存）进行通信。

5. 写入 MDM-AP 控制寄存器以设置调试请求位。 这将防止内核在复位引脚被释放时尝试启动。

注

如果处理器使能加密特性，则无法写入调试请求位。

6.释放RESET信号或清除MDM-AP控制寄存器中的系统复位请求位。 完成上述步骤后，即可开始调试或闪存编程。

3.2 擦除闪存

在对闪存扇区进行编程之前，必须先将其擦除。 可以使用多种不同的闪存擦除机制：

•整体擦除（SWD）

•擦除所有块

•擦除闪存块

•擦除闪存扇区

以下部分描述了每个命令以及生产闪存编程器何时可以使用它们。

321 整体擦除

与上面列出的其他擦除选项不同，整体擦除不是使用 FCCOB 寄存器执行的闪存命令。相反，整体擦除是由编程接口直接请求的。

即使启用了加密特性或区域受到保护，调用整体擦除也会导致整个闪存被擦除。如果启用加密特性，还必须使能整体擦除功能（FTFC_FSEC[MEEN] 不等于 0b10），否则将不允许整体擦除。如果允许（使能了整体擦除），则执行以下步骤：

1.无论保护设置如何，所有闪存块都会被擦除。这包括程序闪存、数据闪存、程序闪存 IFR 交换指示器地址、数据闪存 IFR 空间（包括 EEE 分区信息）、EEPROM 备份存储器（E-闪存）和 FlexRAM。

2. 读取闪存以验证擦除是否成功完成。

3.如果擦除验证失败，则设置FTFC_FSTAT[MGSTAT0]位并且过程停止。如果擦除验证通过，则过程继续。

4. FTFC_FSEC[SEC] 寄存器字段设置为 0b10（不加密）。这会立即释放加密特性。

5. Flash 配置字段中的 Flash 加密字节也被编程为 0xFE（不加密）。在随后的复位中，处理器将在不加密模式下启动，除非闪存配置字段被修改。

3211 SWD整体擦除

下面列出了通过 SWD 接口执行整体擦除的推荐程序。

1. 按照 SWD 连接步骤中描述的步骤进行操作。

2. 读取 MDM-AP 状态寄存器 Mass Erase Enable 位以确定是否启用了整体擦除命令。如果整体擦除使能 = 0，则整体擦除被禁用且处理器无法被擦除或不加密。如果整体擦除使能= 1，则可以使用整体擦除命令。

3. 写入 MDM-AP 控制寄存器以设置 Flash Mass Erase in Progress 位。这将启动整体擦除过程。

4. 读取 MDM-AP 控制寄存器，直到 Flash Memory Mass Erase in Progress 位清零。

5. 当 Flash Memory Mass Erase 位清零时，整体擦除已完成。只要整体擦除的整体擦除验证部分成功完成，加密特性就会被释放。

重要的是：在整体擦除正在进行时，要确保处理器不接收新的复位或执行任何可能干扰整体擦除命令操作的代码。为避免这种情况，可通过在整体擦除期间保持 RESET 有效或保持系统复位请求置位来保持处理器处于复位状态。

322 擦除所有块

擦除所有块是一个闪存命令，它执行与 SWD 整体擦除类似的操作，但操作不完全相同。 与整体擦除一样，擦除所有块命令将擦除所有用户闪存内容，即程序闪存、数据闪存、数据闪存 IFR（包括 EEPROM 分区配置）、EEPROM 备份存储器（E-闪存）和 FlexRAM。 整体擦除和擦除所有块命令之间有两个重要区别：

•与整体擦除不同，擦除所有块会考虑保护设置。 如果任何闪存或 FlexRAM 区域受到保护，则擦除所有块命令将中止。

• 如果验证完成，擦除所有块将通过将 FTFC_FSEC[SEC] 字段设置为不加密来释放加密特性，但闪存配置字段中的加密字节未编程为 0xFE。 在随后的复位中，处理器将处于加密状态，除非闪存配置字段中的 FSEC[SEC] 被重新编程。

323 擦除闪存块

擦除闪存块命令可用于擦除整个闪存块。 擦除所有块命令可以擦除多个块（如果设备有多个闪存块），但擦除闪存块命令只能擦除单个块。 如果块内的任何区域受到保护，则擦除将中止。 此外，如果 EEPROM 被分区并且被擦除的块是数据闪存，则擦除将被中止。 当前的加密设置不受执行擦除闪存块命令的影响，但如果闪存配置字段被擦除，则在随后的复位中处理器将处于加密状态，除非闪存配置字段中的 FTFC_FSEC[SEC] 设置被重新编程。

注

仅包含单个闪存块的设备不支持擦除闪存块命令。如果设备上只有一个闪存块，那么擦除所有块命令提供了与不支持的擦除闪存块命令相同的功能。

324 擦除闪存扇区

顾名思义，擦除闪存扇区命令用于擦除闪存的整个扇区。 擦除闪存扇区命令会擦除最小内存单元。 扇区的大小取决于闪存配置（交错或非交错）。 请参阅 RM 中的程序闪存功能以确定扇区大小。

如果所选扇区受到保护，擦除闪存扇区命令将中止。

325 擦除建议

虽然在 S32K1xx 上擦除闪存有多种选择，但大多数生产闪存编程器应该只需要实现整体擦除和扇区擦除。

3.2.5.1 整体擦除建议

整体擦除是一项重要功能，可用于取消部件的加密保护。所有闪存编程器都应支持通过硬件接口 (SWD) 执行整体擦除请求。如果连接时检测到设备已加密，则应提示用户询问他们是否要擦除整个闪存以释放加密特性。如果检测到设备是加密的，可以包含一个选项来自动发出整体擦除请求，但恩智浦建议默认情况下不要自动整体擦除。通过这种方式，用户会收到可能存在问题的警报，如果闪存中已有信息需要保留，他们可以选择中止操作。

整体擦除会导致闪存中的所有用户信息丢失，因此不建议闪存编程器默认使用。如果整体擦除专门用于擦除程序，当用户在不同时间对多个文件进行编程，则它不能支持任何类型的多级编程。 NXP 建议默认使用扇区擦除，但有一个用户可选择的选项来使用整体擦除。

3.2.5.2 整体擦除与CSEc建议

Flash 模块添加了符合 SHE9 规范的功能。通过使用嵌入式处理器、固件和硬件辅助 AES-128 子块，Flash 模块为加密消息传递应用程序启用加密、解密和消息生成及身份验证算法。该模块称为加密服务引擎或 CSEc。启用 CSEc 后，在启动整体擦除命令之前必须考虑一些事项。如果未配置 CSEc 密钥或部件未分区，则将允许整体擦除操作运行。如果程序分区命令中的 CSEc 密钥大小字段设置为零密钥 (0x00)，那么您可以随时进行整体擦除，只要在闪存配置字段中启用整体擦除，它就应该可以运行。如果已配置任何 CSEc 密钥（因此该部分已被分区），用户必须完成以下完整流程，否则 整体擦除命令将不起作用：

- 启动程序分区命令(PGMPART) 时，务必将CSEc 密钥大小分配到非零值。
- 编程 CSEc 密钥，至少是 MASTER_ECU_KEY。
- 通过发出 CMD_DBG_CHAL 和 CMD_DBG_AUTH 命令擦除密钥。 CMD_DBG_AUTH 命令将擦除数据 IFR，从而擦除分区代码并允许整体擦除再次工作。

注

如果任何 CSEc 密钥被“写保护”，则上述过程将不起作用，因此无法启动整体擦除。

3.2.5.3 扇区擦除建议

为避免丢失预编程信息和配置，生产闪存编程器应默认使用扇区擦除命令。 首选方法是对扇区执行擦除以确保其真正处于擦除状态，对该扇区进行编程，然后根据需要移动到下一个扇区。 这样，扇区仅在需要时才被擦除，并且如果在擦除上层扇区或对任何扇区进行编程时出现错误导致整个操作中止，则已修改的闪存数量是最小的。

3.3 编程闪存

擦除闪存扇区（或整个闪存）后，可以开始编程。 设备可以支持两种不同的闪存编程命令：

- 写扇区
- 写短语（64 位）

331 写扇区

写扇区命令允许对比常规写短语命令更大的存储器区域进行编程。 一个扇区是 S32K1xx 设备的 FlexRAM 大小的四分之一。 由于写扇区命令可以对更多内存进行编程，因此它是对内存进行编程的最有效方法。 如果可用，这是用于生产编程的推荐程序命令。 要通过 SWD 使用写扇区命令，需要将要编程的数据写入 FlexRAM。 FlexRAM 必须配置为传统 RAM 模式而不是 EEPROM 模式。 数据加载到 FlexRAM 后，即可执行写扇区命令。

3.3.2 写短语

写短语命令可将 64 位（8 字节）内容编程到闪存中。 与写扇区命令不同，要编程的数据是直接写入闪存的 FCCOBn 寄存器中，因此这个命令不使用FlexRAM 进行编程。

3.3.3 编程建议

为了提高编程速度，写扇区命令是推荐用于闪存生产编程的编程方法。

执行任何闪存命令时，始终轮询相应的状态寄存器以等待命令完成。 新设备的命令执行时间几乎一致，但依赖固定延迟循环而不是状态查询可能会导致问题。 相反，需要轮询闪存 FSTAT[CCIF] 位以确定该命令是否已完成，然后再移至下一个命令。

4 其他编程考虑

对闪存进行编程时，闪存内的一些特殊位置和一些闪存功能需要注意。 以下部分提供了有关闪存编程器应给予特殊处理的闪存地址和功能的信息。

4.1 闪存配置字段

闪存配置字段是位于 0x400-0x40F 的闪存的 16 字节部分。闪存配置字段中的值在复位时被复制到闪存寄存器，因此它们为多个闪存配置设置默认值，最重要的是闪存保护和系统加密设置。

为防止意外加密设备，闪存配置字段需要特殊处理。恩智浦建议默认情况下闪存编程器将以下值写入闪存配置字段：

- 0xFFFF_FFFF
- 0xFFFF_FFFF
- 0xFFFF_FFFF
- 0xFFFF_FFFE

这些值保存在非保护的闪存区域，使设备处于不加密状态。有关更多信息，请参阅参考手册中的 Flash 配置字段描述部分。

一些用户可能希望对闪存配置字段进行编程，因此应包括覆盖选项，以使用户能够将其他值编程到闪存配置字段中。请注意，对闪存配置字段的更改将在下次复位后生效。为防止更新的闪存配置字段值对处理器加密并阻止进一步访问处理器，请确保编程器在重置处理器之前完成所有编程步骤和写入值的验证。

通过擦除所有 (ERSALL)、擦除扇区 (ERSSCR) 和擦除块 (ERSBLK) 等命令擦除配置字段后而不对配置字段进行编程，可能会导致设备锁定，因此必须在再次复位之前将闪存配置字段编程为不加密状态。

注

当用户不打算加密设备时，应该使用删除所有块且不加密的命令(ERSALLU)而不是ERSALL。

4.2 模拟EEPROM支持

S32K1xx 器件包括 FlexMemory 功能。 FlexMemory 使用户能够将一些片上闪存配置为模拟 EEPROM、附加闪存或两者的组合。

默认情况下，空白设备将所有 FlexNVM 配置为闪存。 为了使客户能够配置 EEPROM 功能，生产闪存编程器需要支持配置 EEPROM。 仅当闪存处于非加密状态时，才能通过 SWD/JTAG 接口进行 EEPROM 配置。

4.2.1 FlexNVM分区

要配置将使用的 EEPROM 内存量和将用于备份 EEPROM 的闪存量，必须对闪存进行分区。闪存程序分区命令 (PGPART) 用于设置内存分区和配置 EEPROM 以供使用。

当 PGPART 命令执行时，它将导致所有 FlexNVM 被擦除（甚至不会用于 EEPROM 备份的扇区也会被删除）。 FlexNVM 被擦除后，用于 EEPROM 备份的扇区将被格式化以供使用。由于 FlexNVM 被擦除并且分区后 FlexNVM 的部分内容不再可用，因此如果要使用 EEPROM 功能，则 EEPROM 配置应该是对器件进行编程的第一步。配置 EEPROM 并选择性加载后，可以将代码和数据编程到闪存的其余部分中。由于分区会影响可用于常规编程的闪存数量，因此最好从设备读取并保存分区代码，以便由于设备分区、使得用户在尝试对不再像传统闪存那样的地址进行编程时，收到警告。

注

分区只应该进行一次。如果为不同的配置重新分区了闪存，那么记录数据将丢失，并且这些记录的数据不能保证能够获得预期的持久性。

注

如果设备被整体擦除，分区信息、EEE数据和EEE位置使用信息将会丢失。如果EEE在整体擦除之前已经使用，那么这可能会影响EEE的最大持久性。

4.2.2 编程模拟EEPROM

闪存分区后，可以通过写入 FlexRAM 存储空间（从 0x1400_0000 开始）来对初始 的EEPROM 值进行编程。 使用 SWD可以通过直接内存写入来完成。

注

在尝试编程EEPROM数据时，必须把FlexRAM配置为EEPROM模式。在尝试使用写扇区命令之前，请确保将 FlexRAM 模式切换回传统的RAM模式。

4.3 加密器件

当部件处于加密状态时，JTAG/SWD 接口将被禁用。这意味着当部件处于此状态时，调试控制器无法读取或写入 器件的存储器映射地址。 当闪存配置字段中的 FTFC_FSEC 字节处于加密状态时，该部件是加密的。一旦发生这种情况，您将无法通过 JTAG/SWD 运行任何 CMD_DBG_CHAL 和 CMD_DBG_AUTH 命令。 因此，客户应用程序代码必须将整体擦除中显示的流程和 CSEc 考虑因素嵌入到他们的应用程序中，并从不同的接口（例如 CAN 或 UART/串行接口）触发例程。

5 常见问题

下面的列表描述了已知会导致编程问题的状况。

- 不保护闪存配置字段以防止意外编程和/或故意使用不正确的值（考虑字节序）对闪存配置字段进行编程。如果闪存配置字段被编程为启用加密特性并禁用整体擦除和后门密钥访问，则无法对闪存进行编程。
- 擦除或编程命令的打断会导致闪存内容损坏。中断可能包括复位、断电或与处理器上运行的代码发生冲突。
- 不使用调试请求位来防止在对闪存进行编程时执行代码。如果内核在建立调试连接或修改闪存时尝试运行代码，则代码可能会干扰。如果没有编程的代码（空白器件），则处理器将由于内核锁定而定期复位。如果在器件中编写了修改电源模式、时钟或配置SWD/JTAG 引脚的代码，那么当正在执行闪存命令时，这些会阻止连接和/或导致闪存损坏。
- 在处理器未通电时向 I/O 引脚施加电压。如果违反了引脚的电压电平规范，处理器将会尝试部分加电和/或将闪存置于未定义状态。这可能导致闪存内容损坏、闪存控制逻辑损坏或设备配置和调较值损坏，进而导致处理器报告为加密状态（锁定设备）或处理器无法响应和完成闪存命令。
- 必须特别注意不要终断那些与加密相关的配置进程，因为器件可能会无意中被加密/锁定。在处理器复位之前，请验证写入的值。
- 必须尽量缩短在生产线上处于加密状态的零件测试的持续时间。
- 在闪存编程期间确保电源无噪声且符合规范

6 进一步阅读

可能有用的其他文档包括：

- S32K Microcontroller documentation, software, and tools available at www.nxp.com/s32k
- Flash Memory Controller, Flash Memory Module and Debug chapters of the applicable device's reference manual.
- Using S32K1xx EEPROM functionality (document AN11983)
- Using S32K148 FlexNVM memory (document AN12003)
- Getting Started with CSEc Security Module (document AN5401)
- Production Flash Programming Best Practices for Kinetis K- and L-series MCUs (document AN4835)
- Using the Kinetis Security and Flash Protection Features (document AN4507).
- Using the Kinetis Family Enhanced EEPROM Functionality (document AN4282).
- Robust Over-the-Air Firmware Updates Using Program Flash Memory Swap on Kinetis Microcontrollers (document AN4533).

7 修订历史记录

表1.修订历史记录

修订版	日期	描述
0	2018年2月	初始发布
1	2019年9月	● 更新的加密器件。 ● 更新的常见问题。

How To Reach Us

Home Page:

nxp.com

Web Support:

nxp.com/support

For more information, please visit: http://www.nxp.com

For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: September 2019

Document identifier: AN12130

arm