



STM32F10x闪存编程

简介

本编程手册介绍了如何烧写STM32F10x微控制器的闪存存储器。

STM32F10x内嵌的闪存存储器可以用在线编程(ICP)或在程序中编程(IAP)烧写。

在线编程(In-Circuit Programming – ICP)方式用于更新闪存存储器的全部内容，它通过JTAG或SWD协议下载用户应用程序到微控制器中。ICP是一种快速有效的流程，消除了封装和管座的困扰。

与ICP方式对应，在程序中编程(In-Application Programming – IAP)可以使用微控制器支持的任何一种通信接口(如I/O端口、USB、CAN、UART等)下载程序或数据到存储器中。IAP允许你在程序运行时重新烧写闪存存储器中的内容。然而，IAP要求至少有一部分程序已经用ICP烧到某个闪存块中。

本文的内容适用于STM32F101和STM32F103系列。

闪存接口是在AHB协议上实现了对指令和数据的访问，通过对存储器的分区和预取缓存的实现，加快了存储器的访问；闪存接口还实现了闪存编程和擦除所需的逻辑电路，这里还包括访问和写入保护以及选择字节的控制。

目录

1	概述	4
1.1	特性	4
1.2	闪存模块组织	4
2	读/编写STM32F10x内置闪存	6
2.1	简介	6
2.2	读操作	6
2.2.1	取指令	6
2.2.2	D-Code接口	6
2.2.3	闪存访问控制器	6
2.2.4	信息模块访问	7
2.3	闪存编程和擦除控制器(FPEC)	7
2.3.1	键值	7
2.3.2	解除闪存锁	7
2.3.3	闪存编程	7
2.3.4	信息块的编程	8
2.3.5	闪存擦除	9
2.4	保护	11
2.4.1	读保护	11
2.4.2	写保护	11
2.4.3	信息块保护	12
2.5	选择字节加载	12
2.6	低功耗管理	13
3	寄存器说明	14
3.1	闪存访问控制寄存器(FALSH_CR)	14
3.2	FPEC键寄存器(FLASH_KEYR)	15
3.3	闪存OPTKEY寄存器(FLASH_OPTKEYR)	15
3.4	闪存状态寄存器(FLASH_SR)	16
3.5	闪存控制寄存器(FLASH_CR)	17
3.6	闪存地址寄存器(FLASH_AR)	18
3.7	选择字节寄存器(FLASH_OBR)	19
3.8	写保护寄存器(FLASH_WRPR)	20
3.9	闪存寄存器映像	21
4	版本历史	22

术语

下面列出了本文档中所用到的术语和缩写的简要说明：

- Cortex-M3内核集成了两个调试端口：
 - JTAG调试接口(JTAG-DP)提供基于JTAG(Joint Test Action Group 联合测试行动小组)协议的5线标准接口。
 - SWD调试接口(SWD-DP)提供基于SWD(Serial Wire Debug 串行线调试)协议的2线标准接口。

有关JTAG和SWD协议，请参考*Cortex M3 Technical Reference Manual*。

- 选择字节：存放在闪存中的产品配置位。
- 字(Word)：32位长的数据或指令
- 半字(Half Word)：16位长的数据或指令
- 字节(Byte)：8位长的数据或指令
- FPEC(FLASH Program/Erase controller 闪存编程/擦除控制器)：对内置闪存的写操作是由一个内嵌的FPEC管理的。
- IAP(In-Application Programming)：IAP是在用户程序运行时对闪存微控制器中存储器重新编程的一种能力。
- ICP(In-Circuit Programming)：ICP是在芯片安装在用户应用板上后通过JTAG协议对闪存微控制器中存储器编程的一种能力。
- I-Code：这是连接Cortex-M3核心的指令总线与闪存程序存储器接口的总线，指令预取是在这个总线上实现的。
- D-Code：这是连接Cortex-M3核心的D-Code总线(常数和调试访问)与闪存数据存储器接口的总线。
- SIF：用户选择字节的小信息块(详见图4)。
- AHB：先进高性能总线。

1 概述

1.1 特性

- 128K字节闪存
- 擦写周期：1000次
- 存储器配置：
 - 主存储块：16K x 64位
 - 信息块：320 x 64位

闪存接口的特性：

- 带预取缓冲器的读接口(2x64位)
- 选择字节加载
- 闪存编程/擦除操作
- 读出/写入保护

1.2 闪存模块组织

存储器组织成主存储器块和信息块；主存储器块包含128页，每页1K字节；信息块包括2页(2K字节和0.5K字节)；如表一：

表一 闪存模块组织

块	名称	地址范围	长度(字节)
主存储器	页0	0x0800 0000 – 0x0800 03FF	4 x 1K
	页1	0x0800 0400 – 0x0800 07FF	
	页2	0x0800 0800 – 0x0800 0BFF	
	页3	0x0800 0C00 – 0x0800 0FFF	
	页4~7	0x0800 1000 – 0x0800 1FFF	4 x 1K
	页8~11	0x0800 2000 – 0x0800 2FFF	4 x 1K
	· · ·	· · ·	· · ·
	页124~127	0x0801 F000 – 0x0801 FFFF	4 x 1K
信息块	启动程序代码	0x1FFF F000 – 0x1FFF F7FF	2K
	用户选择字节	0x1FFF F800 – 0x1FFF F9FF	512
闪存寄存器	FLASH_ACR	0x4002 2000 – 0x4002 2003	4
	FLASH_KEYR	0x4002 2004 – 0x4002 2007	4
	FLASH_OPTKEYR	0x4002 2008 – 0x4002 200B	4
	FLASH_SR	0x4002 200C – 0x4002 200F	4
	FLASH_CR	0x4002 2010 – 0x4002 2013	4
	FLASH_AR	0x4002 2014 – 0x4002 2017	4
	保留	0x4002 2018 – 0x4002 201B	4
	FLASH_OBR	0x4002 201C – 0x4002 201F	4
	FLASH_WRPR	0x4002 2020 – 0x4002 2023	4
	保留	0x4002 2024 – 0x4002 2087	100

闪存存储器被组织成32位宽的存储器单元，可以存放代码和数据常数。每一个STM32F10x微控制器的闪存模块都有一个特定的起始地址，有关的起始地址请参考STM32F10x参考手册。

系统存储器是用于存放在系统存储器自举模式下的启动程序，这个区域只保留给ST使用；ST在生产线上对这个区域编程并锁定以防止用户擦写。

本译文仅供参考，如有翻译错误，请以英文原稿为准

对主存储器和信息块的写入由内嵌的闪存编程/擦除控制器(FPEC)管理；编程与擦除的高电压由内部产生。

闪存存储器有两种保护方式防止非法的访问(读、写、擦除)：

- 页写入保护
- 读出保护

详情请参考2.4节。

在执行闪存写操作时，任何对闪存的读操作都会锁住总线，在写操作完成后读操作才能正确地进行；既在进行写或擦除操作时，不能进行代码或数据的读取操作。

进行闪存编程操作时(写或擦除)，必须打开内部的RC振荡器(HSI)。

闪存存储器可以用ICP或IAP方式编程。

2 读/编写 STM32F10x 内置闪存

2.1 简介

本节介绍如何读和对STM32F10x的内置闪存编程

2.2 读操作

内置闪存模块可以在通用地址空间直接寻址，任何32位数据的读操作都能访问闪存模块的内容并得到相应的数据。

读接口在闪存端包含一个读控制器，还包含一个AHB接口与CPU衔接。这个接口的主要工作就是产生读闪存的控制信号并预取CPU要求的指令块，预取指令块仅用于在I-Code总线上的取指操作，数据常量是通过D-Code总线访问的。这两条总线的访问目标是相同的闪存模块，访问D-Code将比预取指令优先级高。

2.2.1 取指令

Cortex-M3在I-Code总线上取指令，在D-Code总线上取数据。预取指令块可以有效地提高对I-Code总线访问的效率。

预取缓冲器

预取缓冲器包含两个数据块，每个数据块有8个字节；预取指令(数据)块直接映像到闪存中，因为数据块的大小与闪存的宽度相同，所以读取预取指令块可以在一个读周期完成。

设置预取缓冲器可以使CPU更快地执行，CPU读取一个字的同时下一个字已经在预取缓冲器中等待，即当代码跳转的边界为8字节的倍数时，闪存的加速比例为2。

预取控制器

预取控制器根据预取缓冲器中可用的空间决定是否访问闪存，预取缓冲器中有至少一块的空余空间时，预取控制器则启动一次读操作。

清除闪存访问控制寄存器中的一个控制位能够关闭预取缓冲器，这样预取缓冲器将处于关闭状态。

注意： 当AHB时钟的预分频系数不为“1”时，必须打开预取缓冲器(FLASH_ACR[4]=1)。

如果在系统中没有高频率的时钟，即HCLK频率较低时，闪存的访问只需半个HCLK周期(半周期的闪存访问只能在时钟频率低于8MHz时进行，使用HIS或HSE并且关闭PLL时可得到这样的频率)；在闪存访问控制寄存器中有一个控制位可以选择这种工作方式。

注意： 当使用了预取缓冲器和AHB时钟的预分频系数不为“1”时，不能使用半周期访问方式。

访问时间调节

为了维持读闪存的控制信号，预取控制器的时钟周期与闪存访问时间的比例由闪存访问控制器控制；这个值给出了能够正确地读取数据时，闪存控制信号所需的时钟周期数目；复位后，该值为1，闪存访问为两个时钟周期(FLASH_ACR的复位值为01，长度=1)。

2.2.2 D-Code接口

D-Code接口包含CPU端简单的AHB接口和对闪存访问控制器的仲裁器提出访问请求的逻辑电路。D-Code的访问优先于预取指令的访问。这个接口使用预取缓冲器的访问时间调节模块。

2.2.3 闪存访问控制器

这个模块就是在I-Code上的指令预取请求和D-Code接口上读请求的仲裁器。

D-Code接口的请求优先于I-Code的请求。

2.2.4 信息模块访问

信息模块的内容可以通过I-Code总线执行并且可以在D-Code总线上读出，小信息模块可以在任一模式下读写。

选择字节模块包含配置选择字节和其他用户定义的信息。

2.3 闪存编程和擦除控制器(FPEC)

FPEC模块处理闪存的编程和擦除操作，它包括7个32位的寄存器：

- FPEC键寄存器(FLASH_KEYR)
- 选择字节键寄存器(FLASH_OPTKEYR)
- 闪存控制寄存器(FLASH_CR)
- 闪存状态寄存器(FLASH_SR)
- 闪存地址寄存器(FLASH_AR)
- 选择字节寄存器(FLASH_OBR)
- 写保护寄存器(FLASH_WRPR)

只要CPU不访问闪存，闪存操作不会延缓CPU的执行。

2.3.1 键值

共有三个键值：

- RDPRT键 = 0x000000A5
- KEY1 = 0x45670123
- KEY2 = 0xCDEF89AB

2.3.2 解除闪存锁

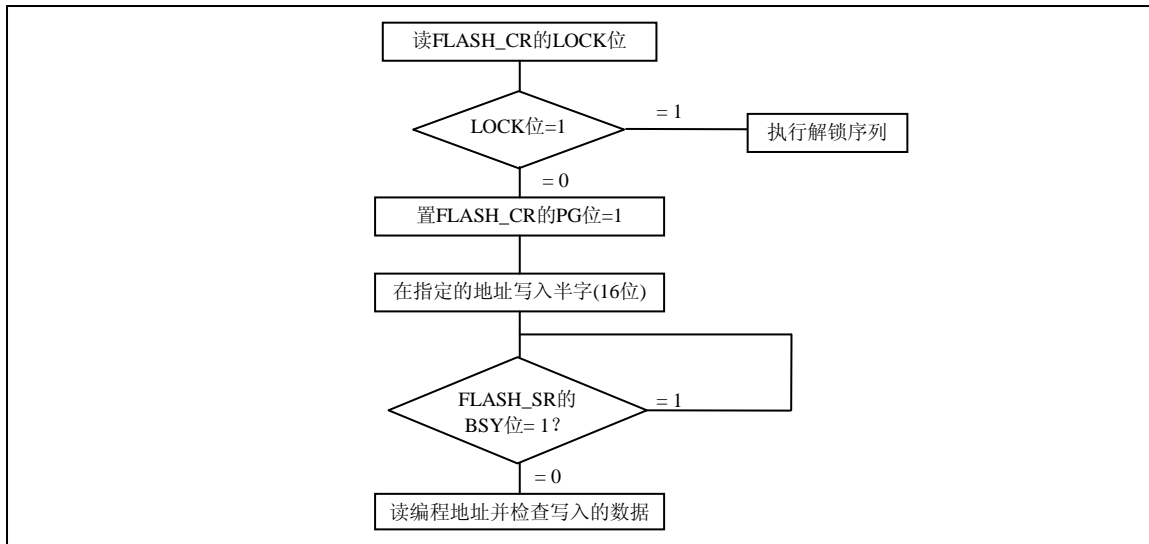
复位后，FPEC模块是被保护的，不能写入FLASH_CR寄存器；通过写入特定的序列到FLASH_KEYR寄存器可以打开FPEC模块，这个特定的序列是两个键值(KEY1和KEY2，见2.3.1节)；写入任何其他序列都会在下次复位前锁死FPEC模块和FLASH_CR寄存器。

写入错误的键序列还会产生总线错误；总线错误发生在第一次写入的不是KEY1，或第一次写入的是KEY1但第二次写入的不是KEY2时；FPEC模块和FLASH_CR寄存器可以由程序设置FLASH_CR寄存器中的LOCK位锁住，这时可以通过在FLASH_KEYR中写入正确的键值对FPEC解锁。

2.3.3 闪存编程

每次闪存编程可以写入16位。当FLASH_CR寄存器的PG位为1时，写入一个半字到一个闪存地址将启动一次编程；写入任何非半字的数据，FPEC都会产生总线错误。在编程过程中(BSY位为1)，任何读写闪存的操作都会使CPU暂停，直到此次闪存编程结束。

图一 编程过程



标准编程

这种模式下CPU以标准的半字写的方式烧写闪存，FLASH_CR寄存器的PG位必须置1。FPEC先读出指定地址的内容并检查它是否被擦除，如未被擦除则不执行编程并在FLASH_SR寄存器的PGERR位提出警告(唯一的例外是当要烧写的数值是0x0000时，0x0000可被正确烧入且PGERR位不置位)；如果指定的地址在FLASH_WRPR中指定为写保护，则不执行编程并在FLASH_SR寄存器的WRPRERR位置1提出警告。FLASH_SR寄存器的EOP为1时表示编程结束。

标准的闪存编程顺序如下：

- 检查FLASH_SR寄存器的BSY位，以确认没有其他正在进行的编程操作；
- 设置FLASH_CR寄存器的PG位为1；
- 写入要编程的半字到指定的地址；
- 等待BSY位变为0；
- 读出写入的地址并验证数据。

注意： 当FLASH_SR寄存器的BSY位为1时，不能对任何寄存器执行写操作。

2.3.4 信息块的编程

选择字节编程

选择字节是通过特殊的地址进行编程。选择字节只有6个字节(4个用于写保护，1个由于读保护，另一个用于器件配置)。对FPEC解锁后，分别写入KEY1和KEY2(见2.3.1节)到FLASH_OPTKEYR，再设置FLASH_CR寄存器的OPTWRE位为1，此时可以对小信息块进行编程：设置FLASH_CR寄存器的OPTPG位为1后写入半字到指定的地址。

FPEC先读出指定地址的选择字节内容并检查它是否被擦除，如未被擦除则不执行编程并在FLASH_SR寄存器的WRPRERR位提出警告。FLASH_SR寄存器的EOP为1时表示编程结束。

FPEC使用半字中的低字节并自动地计算出高字节(高字节为低字节的反码)，并开始编程操作，这将保证选择字节和它的反码始终是正确的。

烧写编程的顺序如下：

- 检查FLASH_SR寄存器的BSY位，以确认没有其他正在进行的编程操作；
- 设置FLASH_CR寄存器的OPTWRE位为1；
- 设置FLASH_CR寄存器的OPTPG位为1；
- 写入要编程的半字到指定的地址；

本译文仅供参考，如有翻译错误，请以英文原稿为准

- 等待BSY位变为0;
- 读出写入的地址并验证数据。

当读闪存保护选项从“保护”变为“未保护”时，在重新设置读保护选项前会自动执行一个全部擦除用户闪存的操作。如果用户要改变读保护之外的选项，则不会出现全部擦除操作。读保护选项上的这一擦除操作保护了闪存中的内容不被非法读出。

数据编程

选择字节块之后剩余的字节(在0x1FFF F804未用的OPT字节)可以用于存储数据。对这部分地址的编程可以通过标准编程操作完成。

擦除过程

小信息块的擦除顺序(OPTERASE)如下:

- 检查FLASH_SR寄存器的BSY位，以确认没有其他正在进行的闪存操作;
- 设置FLASH_CR寄存器的OPTWRE位为1;
- 设置FLASH_CR寄存器的OPTER位为1;
- 设置FLASH_CR寄存器的STRT位为1;
- 等待BSY位变为0;
- 读出小信息块并做验证。

2.3.5 闪存擦除

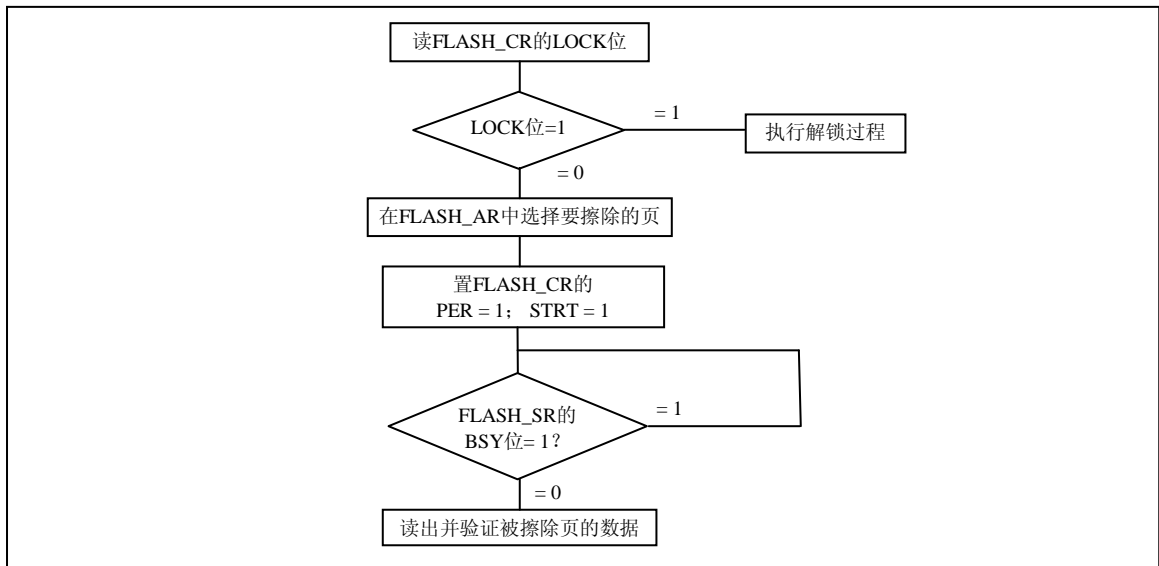
闪存可以按页擦除，也可以全部擦除。

页擦除

闪存的任何一页都可以通过FPEC的页擦除功能擦除；擦除一页应遵守下述过程:

- 检查FLASH_SR寄存器的BSY位，以确认没有其他正在进行的闪存操作;
- 用FLASH_AR寄存器选择要擦除的页;
- 设置FLASH_CR寄存器的PER位为1;
- 设置FLASH_CR寄存器的STRT位为1;
- 等待BSY位变为0;
- 读出被擦除的页并做验证。

图二 闪存页擦除过程

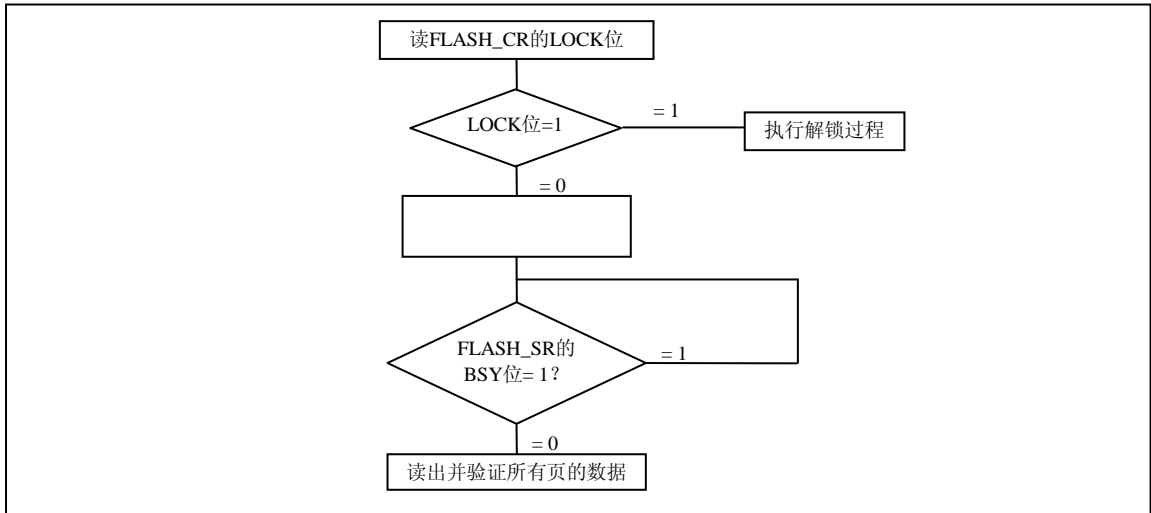


全部擦除

可以用全部擦除功能擦除所有用户区的闪存，信息块不受此操作影响。建议使用下述过程：

- 检查FLASH_SR寄存器的BSY位，以确认没有其他正在进行的闪存操作；
- 设置FLASH_CR寄存器的MER位为1；
- 设置FLASH_CR寄存器的STRT位为1；
- 等待BSY位变为0；
- 读出所有页并做验证。

图三 闪存全擦除过程



2.4 保护

闪存中的用户代码区可以防止非法的读出；闪存区可以以每4页为单位加以保护，以防止在程序跑飞的情况下不被意外地改变。

2.4.1 读保护

这项保护是通过设置信息块中的一个选择字节启动的。当保护字节被写入相应的值以后，在调试模式中不允许读出闪存存储器，所有在RAM中加载和执行的功能(如JTAG/SWD，从RAM启动等)仍然有效，这样可以用于解除读保护(访问闪存仍然被禁止)。当启动读保护后，第0~3页被自动加上了写保护。当信息块中的访问保护选择字节被修改到未保护状态时，全部擦除操作将自动运行。

当RDP选择字节和它的反码包含下列数值对时，闪存被置于保护状态：

表二 闪存存储器保护状态

RDP字节的值	RDP反码的值	读保护状态
0xFF	0xFF	保护
RDP	RDP字节的反码	未保护
任意值	非RDP字节的反码	保护

注意：擦除选择字节块将不会导致自动的全部擦除操作，因为擦除选择字节的结果0xFF相当于保护状态。

未保护

解除读保护的过程是：

- 擦除整个小信息块(用户部分)，读保护码(RDP)将变为0xFF，此时读保护仍然有效；
- 写入正确的RDP代码0xA5以解除存储器的保护，该操作将导致对所有用户闪存的全部擦除操作；
- 进行复位(上电复位)以重新加载选择字节(和新的RDP代码)，此时读保护被解除。

2.4.2 写保护

写保护是以每4页为单位实现的，这样使用32个选择位可以控制到128K字节。以4K位单位实施保护也是合理的，因为通常启动代码都会大于1K。下表列出了对用户页面保护措施：

表三 用户页面的保护

RDP	WRP	作用
有效	有效	CPU只能读；禁止调试和非法访问
有效	无效	CPU可以读写；禁止调试和非法访问；页0为写保护
无效	有效	CPU可读；允许调试和非法访问
无效	无效	CPU可以读写；允许调试和非法访问

如果试图在一个受保护的页面进行编程或擦除操作，在闪存状态寄存器(FLASH_SR)中会返回一个保护错误标志。

解除保护

下属步骤用于解除写保护：

本译文仅供参考，如有翻译错误，请以英文原稿为准

- 使用闪存控制寄存器(FLASH_CR)的OPTER位擦除整个小信息块(用户部分);
- 烧写RDP代码(用于解除读保护);
- 烧写正确的RDP代码0xA5，允许读访问;
- 进行系统复位，重装载选择字节(包含新的WRP[3:0]字节); 写保护被解除。

2.4.3 信息块保护

选择字节块写保护

默认状态下，选择字节块始终是可以读且被写保护。要想对选择字节块进行写操作(编程/擦除)首先要在OPTKEYR中写入正确的键序列(与上锁时一样)，随后选择字节块的写操作被允许，FLASH_CR寄存器的OPTWRE位标示允许写，清除这位将禁止写操作。

2.5 选择字节加载

- 在闪存的信息块中，存放了一组选择字节；这些字节包含产品的配置信息(如封装等)。用户部分的选择字节可以由用户根据应用程序自己选择，选择软件的看门狗或硬件的看门狗就是一个很好的例子。

选择字节中一个32位的字被分为下述几部分：

表四 选择字节格式

位31~24	位23~16	位15~8	位7~0
选择字节1的反码	选择字节1	选择字节0的反码	选择字节0

信息块中选择字节的组织结构如下：

表五 信息块的组织结构

块	地址	[31:24]	[23:16]	[15:8]	[7:0]
保留	0x1FFF F7F8	保留			
	0x1FFF F7FC	保留			
小信息块 (SIF)	0x1FFF F800	nUSER	USER	nRDP	RDP
	0x1FFF F804	未用			
	0x1FFF F808	nWRP1	WRP1	nWRP0	WRP0
	0x1FFF F80C	nWRP3	WRP3	nWRP2	WRP2

选择字节有6个字节，他们主要用于保护内部的闪存接口(读出和写入操作)，只有一个字节用于用户的程序功能。

表六 用户选择字节说明

<p>RDP: 读出保护选择字节</p> <p>读出保护功能帮助用户保护存在闪存中的软件。该功能由设置信息块中的一个选择字节启用。写入正确的数值(RDPRT键=0x00A5)到这个选择字节后，闪存被开放允许读出访问。</p>	
<p>USER: 用户选择字节</p> <p>这个字节用于配置下列功能：</p> <ul style="list-style-type: none"> - 选择看门狗事件：硬件或软件 - 进入停机(STOP)模式时的复位事件 - 进入待机模式时的复位事件 	
位19:23	0xFF: 不用
位 18	<p>nRST_STDBY</p> <p>0: 当进入待机模式时产生复位</p> <p>1: 进入待机模式时不产生复位</p>
位 17	<p>nRST_STOP</p> <p>0: 当进入停机(STOP)模式时产生复位</p> <p>1: 进入停机(STOP)模式时不产生复位</p>
位 16	<p>WDG_SW</p> <p>1: 硬件看门狗</p> <p>0: 软件看门狗</p>
<p>WRP_x: 闪存写保护选择字节</p> <p>用户选择字节WRP_x中的每个位用于保护主存储器中4页的内容，每页为1K字节。总共有4个用户选择字节用于保护所有128K主闪存。</p> <p>WRP0: 页0至页31的写保护</p> <p>WRP1: 页32至页63的写保护</p> <p>WRP2: 页64至页95的写保护</p> <p>WRP3: 页96至页127的写保护</p>	

每次系统复位后，选择字节装载机读出信息块的数据并保存在寄存器中；每个选择位都在信息块中有它的反码位，在装载选择位时反码位用于验证选择位是正确的，如果有任何的差别，将产生一个选择字节错误标志(OPTERR)。当发生选择字节错误时，对应的选择字节被强置为0xFF。当选择字节和它的反码均为0xFF时(擦除后的状态)，上述验证功能被关闭。

所有的选择位(不包括它们的反码位)用于配置该微控制器，CPU可以读选择寄存器，详见3寄存器说明。

CPU可以读选择字节寄存器。

2.6 低功耗管理

在低功耗模式下所有的闪存访问被终止。详情参见STM32F10x参考手册。

3 寄存器说明

本节将使用下述缩写：

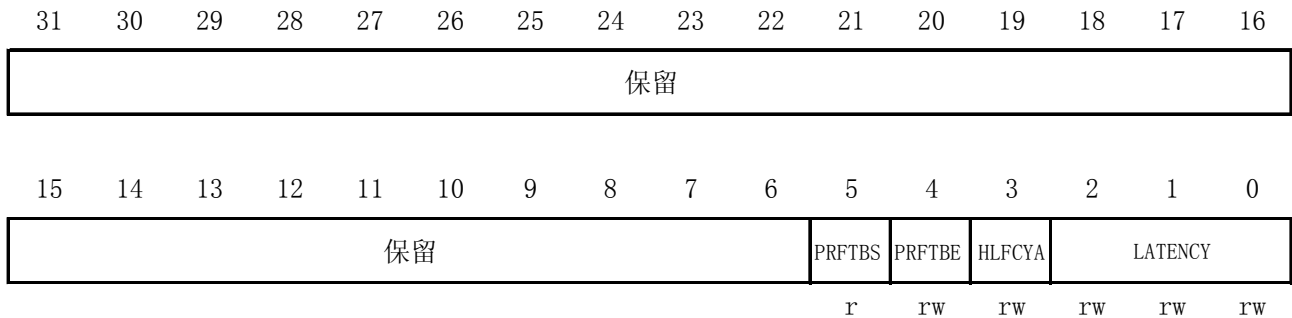
表七 缩写

读/写(rw)	软件可以读或写这些位。
只读(r)	软件只能读这些位。
读/清除(rc_w0)	软件可以读并写'0'清除这些位，写'1'将不对操作的位产生影响。
读/设置(rs)	软件可以读并写'1'设置这些位，写'0'将不对操作的位产生影响。
保留(res)	保留位，必须保持在复位状态。

3.1 闪存访问控制寄存器(FLASH_ACR)

地址偏移：0x00h

复位值：0x0000 0001h

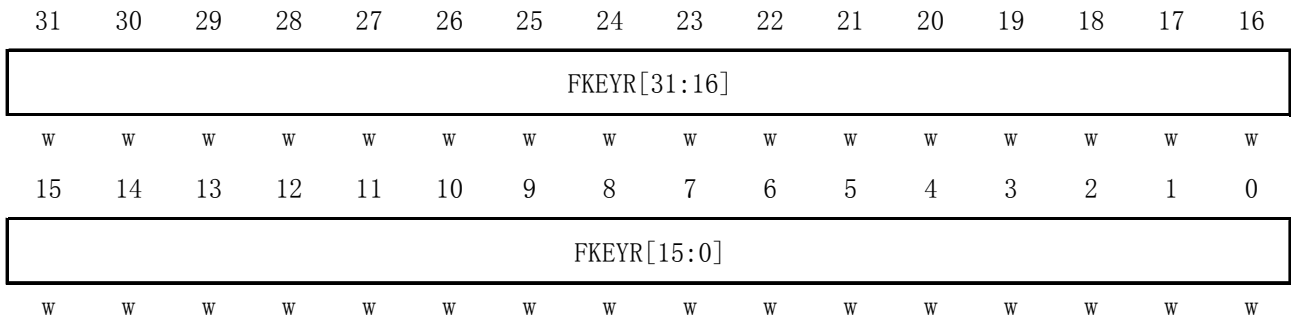


位31~6	保留。必须保持为清除状态'0'
位5	PRFTBS ：预取缓冲区状态 该位显示预取缓冲区的状态 0：预取缓冲区被关闭 1：预取缓冲区被启用
位4	PRFTBE ：预取缓冲区使能 0：关闭预取缓冲区 1：启用预取缓冲区
位3	HLFCYA ：闪存半周期访问使能 0：禁止半周期访问 1：启用半周期访问
位2~0	LATENCY ：时延 这些位表示SYSCLK(系统时钟)周期与闪存访问时间的比例 000：零等待状态，当 $0 < \text{SYSCLK} \leq 24\text{MHz}$ 001：一个等待状态，当 $24\text{MHz} < \text{SYSCLK} \leq 48\text{MHz}$ 010：两个等待状态，当 $48\text{MHz} < \text{SYSCLK} \leq 72\text{MHz}$

3.2 FPEC键寄存器(FLASH_KEYR)

地址偏移: 0x04h

复位值: xxxx xxxxh



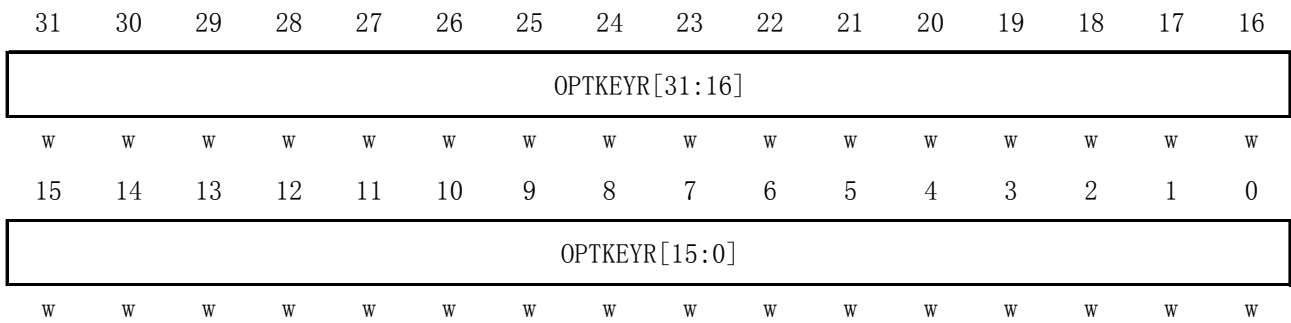
注: 所有这些位是只写的，读出时返回0。

位31~0	FKEYR: FPEC键 这些位用于输入FPEC的解锁键。
-------	---

3.3 闪存OPTKEY寄存器(FLASH_OPTKEYR)

地址偏移: 0x08h

复位值: xxxx xxxxh



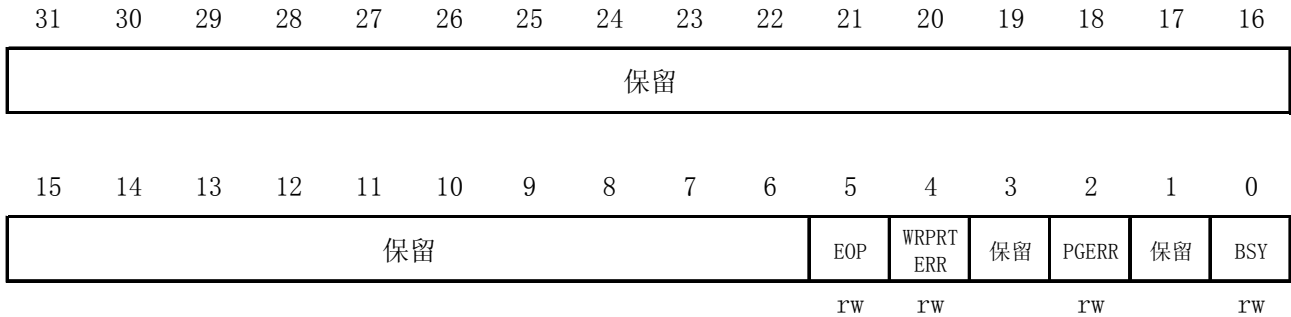
注: 所有这些位是只写的，读出时返回0。

位31~0	OPTKEYR: 选择字节键 这些位用于输入选择字节的键以解除OPTWRE。
-------	--

3.4 闪存状态寄存器(FLASH_SR)

地址偏移: 0x0Ch

复位值: 0x0000 0000h

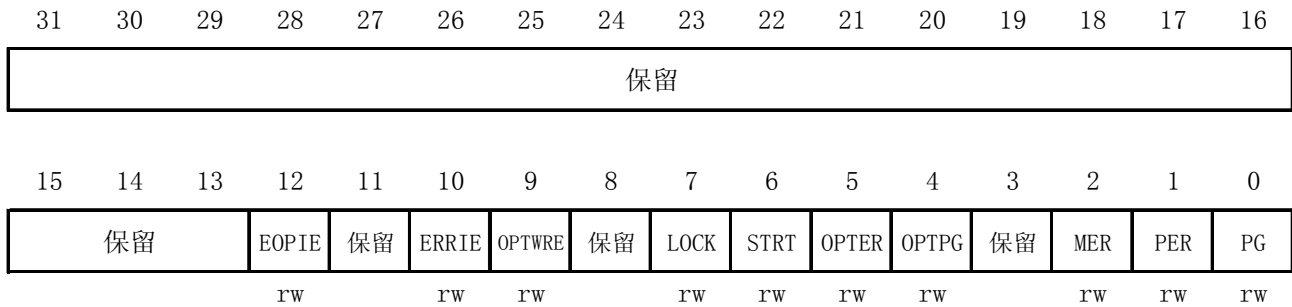


位31~6	保留。必须保持为清除状态'0'
位5	EOP: 操作结束 当闪存操作(编程/擦除)完成时，硬件设置这位为'1'，写入'1'可以清除这位状态。 注：每次成功的编程或擦除都会设置EOP状态
位4	WRPRTERR: 写保护错误 试图对写保护的闪存地址编程时，硬件设置这位为'1'，写入'1'可以清除这位状态。
位3	保留。必须保持为清除状态'0'
位2	PGERR: 编程错误 试图对内容包含'0'的地址编程时，硬件设置这位为'1'，写入'1'可以清除这位状态。 注：试图写'0'前必须清除STRT位
位1	保留。必须保持为清除状态'0'
位0	BSY: 忙 该位指示闪存操作正在进行。在闪存操作开始时，该位被设置为'1'，在操作结束或发生错误时该位被清除为'0'。

3.5 闪存控制寄存器(FLASH_CR)

地址偏移：0x10h

复位值：0x0000 0080h

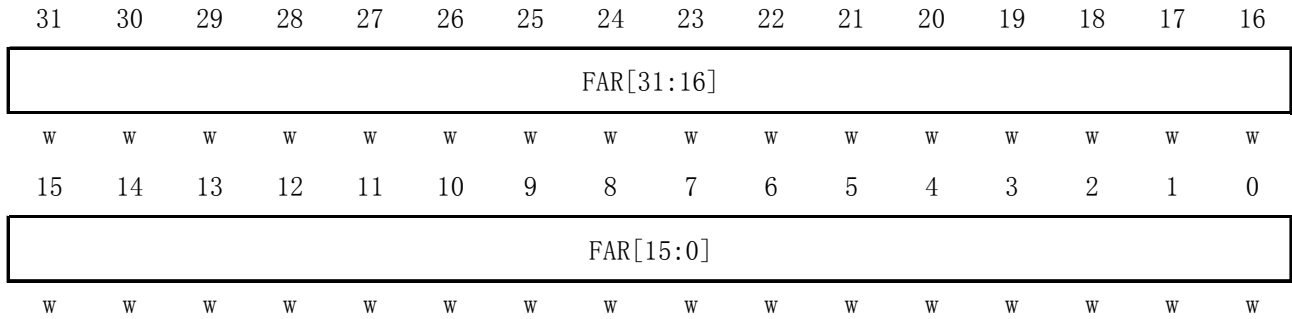


位31~13	保留。必须保持为清除状态'0'
位12	<p>EOPIE: 允许操作完成中断 该位允许在FLASH_SR寄存器中的EOP位变为'1'时产生中断。</p> <p>0: 禁止产生中断 1: 允许产生中断</p>
位11, 8, 3	保留。必须保持为清除状态'0'
位10	<p>ERRIE: 允许错误状态中断 该位允许在发生FPEC错误时产生中断(当FLASH_SR寄存器中的PGERR/WRPRTERR置为'1'时)。</p> <p>0: 禁止产生中断 1: 允许产生中断</p>
位9	<p>OPTWRE: 允许写选择字节 当该位为'1'时, 允许对选择字节/小信息块进行编程操作。当在FLASH_OPTKEYR寄存器写入正确的键序列后, 该位被置为'1'。 软件可清除此位。</p>
位7	<p>LOCK: 锁 只能写'1'。当该位为'1'时表示FPEC和FLASH_CR被锁住, 在检测到正确的解锁序列后, 硬件清除此位为'0'。在一次不成功的解锁操作后, 下次复位前, 该位不能再被改变。</p>
位6	<p>STRT: 开始 当该位为'1'时将触发一次擦除操作。该位只可由软件置为'1'并在BSY变为'1'时清为'0'。</p>
位5	<p>OPTER: 擦除选择字节 擦除选择字节/小信息块。</p>
位4	<p>OPTPG: 烧写选择字节 对选择字节编程。</p>
位2	<p>MER: 全擦除 选择擦除所有用户页</p>
位1	<p>PER: 页擦除 选择擦除页</p>
位0	<p>PG: 编程 选择编程操作</p>

3.6 闪存地址寄存器(FLASH_AR)

地址偏移：0x14h

复位值：0x0000 0000h



这些位由硬件修改为当前/最后使用的地址。在页擦除操作中，软件必须修改这个寄存器以指定要擦除的页。

位31~0	FAR： 闪存地址 当进行编程时选择要编程的地址，当进行页擦除时选择要擦除的页。 注意：当FLASH_SR中的BSY位为'1'时，不能写这个寄存器。
-------	---

3.7 选择字节寄存器(FLASH_OBR)

地址偏移：0x1Ch

复位值：0xFFFF FFFFh

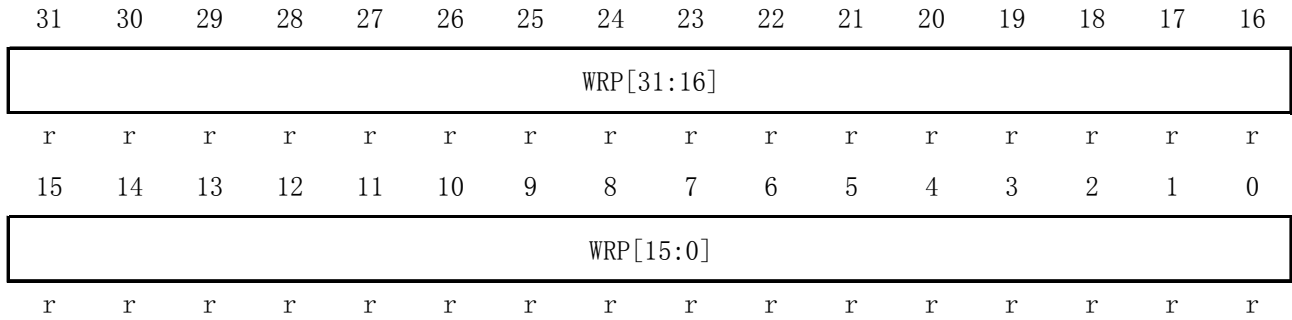


位31~10	保留。必须保持为清除状态'0'。
位9~2	USER: 用户选择字节 这里包含OBL加载的用户选择字节 位[9:5]: 未用 位4: nRST_STDBY 位3: nRST_STOP 位2: WDG_SW
位1	RDPRT: 读保护 当设置为1，表示闪存存储器被写保护。 注：该位为只读。
位0	OPTERR: 选择字节错误 当该位为'1'时表示选择字节和它的反码不匹配。 注意：这位为只读。

3.8 写保护寄存器(FLASH_WRPR)

地址偏移: 0x20h

复位值: 0xFFFF FFFFh



位31~0	<p>WRP: 写保护 该寄存器包含由OBL加载的写保护选择字节。</p> <p>0: 写保护生效 1: 写保护失效</p> <p>注意: 这些位为只读。</p>
-------	--

3.9 闪存寄存器映像

表八 闪存接口—寄存器映像和复位值

偏移	寄存器	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
000h	FLASH_ACR	保留																								PRFTBS	PRFTBE	HLFCYA	LATENCY [2:0]					
	复位值																									0	0	0	0	0	1			
004h	FLASH_KEYR	FKEYR[31:0]																																
	复位值	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x		
008h	FLASH_OPTKEYR	OPTKEYR[31:0]																																
	复位值	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x		
00Ch	FLASH_SR	保留																								EOP	WRPRTER	保留	PGERR	ERLYBSY	BSY			
	复位值																									0	0	0	0	0				
010h	FLASH_CR	保留																			EOPIE	保留	ERRIE	OPTWRE	保留	LOCK	STRT	OPTER	OPTPG	保留	MER	PER	PG	
	复位值																				0	0	0	0	1	0	0	0	0	0	0	0	0	
014h	FLASH_AR	FAR[31:0]																																
	复位值	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
018h	保留																																	
01Ch	FLASH_OBR	保留																				不使用					OPTERR	nRST_STDB	nRST_STOP	WDG_SW	RDPT	OPTERR		
	复位值																										1	1	1	1	1	1	1	1
020h	FLASH_WRPR	[31:0]																																
	复位值	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		

4 版本历史

表九 文档版本历史

日期	版本	改变
2007年3月22日	0.1	最初版本
2007年6月1日	1	少量内容修改。 修改STRT位的定义，3.5节：闪存控制寄存器(FLASH_CR)。 修改了2.3.4节中关于数据编程的描述文字。 修改了3.1节中有关Latency位的描述。 修改了有关ICP的定义。 在调试接口定义中，JTAG之外增加了SWD。 3.7节中修改了OBR的复位值。

本译文仅供参考，如有翻译错误，请以英文原稿为准

请仔细阅读下述内容

Please Read Carefully:

Information in this document is provided solely in connection with ST products. STMicroelectronics NV and its subsidiaries ("ST") reserve the right to make changes, corrections, modifications or improvements, to this document, and the products and services described herein at any time, without notice.

All ST products are sold pursuant to ST's terms and conditions of sale.

Purchasers are solely responsible for the choice, selection and use of the ST products and services described herein, and ST assumes no liability whatsoever relating to the choice, selection or use of the ST products and services described herein.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any part of this document refers to any third party products or services it shall not be deemed a license grant by ST for the use of such third party products or services, or any intellectual property contained therein or considered as a warranty covering the use in any manner whatsoever of such third party products or services or any intellectual property contained therein.

UNLESS OTHERWISE SET FORTH IN ST'S TERMS AND CONDITIONS OF SALE ST DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY WITH RESPECT TO THE USE AND/OR SALE OF ST PRODUCTS INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

UNLESS EXPRESSLY APPROVED IN WRITING BY AN AUTHORIZED ST REPRESENTATIVE, ST PRODUCTS ARE NOT RECOMMENDED, AUTHORIZED OR WARRANTED FOR USE IN MILITARY, AIR CRAFT, SPACE, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS, NOR IN PRODUCTS OR SYSTEMS WHERE FAILURE OR MALFUNCTION MAY RESULT IN PERSONAL INJURY, DEATH, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE. ST PRODUCTS WHICH ARE NOT SPECIFIED AS "AUTOMOTIVE GRADE" MAY ONLY BE USED IN AUTOMOTIVE APPLICATIONS AT USER'S OWN RISK.

Resale of ST products with provisions different from the statements and/or technical features set forth in this document shall immediately void any warranty granted by ST for the ST product or service described herein and shall not create or extend in any manner whatsoever, any liability of ST.

ST and the ST logo are trademarks or registered trademarks of ST in various countries.

Information in this document supersedes and replaces all information previously supplied.

The ST logo is a registered trademark of STMicroelectronics. All other names are the property of their respective owners.

© 2007 STMicroelectronics - All rights reserved

STMicroelectronics group of companies

Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia - Malta - Morocco - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

www.st.com

