零死角玩转STM32

设置FLASH的读写保护及解除

淘宝：firestm32.taobao.com

论坛： www.chuxue123.com

扫描进入淘宝店铺

主讲内容

01 选项字节与读写保护

02 修改选项字节的过程

03 操作选项字节的库函数

04 实验：设置读写保护及解除

参考资料:《零死角玩转STM32》

“设置FLASH的读写保护及解除”章节

设置FLASH的读写保护及解除

选项字节与读写保护

在实际发布的产品中，在STM32芯片的内部FLASH存储了控制程序，如果不作任何保护措施的话，可以使用下载器直接把内部FLASH的内容读取回来，得到bin或hex文件格式的代码拷贝，别有用心的厂商即可利用该代码文件山寨产品。为此，STM32芯片提供了多种方式保护内部FLASH的程序不被非法读取，但在默认情况下该保护功能是不开启的，若要开启该功能，需要改写内部FLASH选项字节(Option Bytes)中的配置。

设置FLASH的读写保护及解除

选项字节的内容

选项字节是一段特殊的FLASH空间，STM32芯片会根据它的内容进行读写保护、复位电压等配置，选项字节的构成如下：

地址	[63:16]	[15:0]
0x1FFF C000	保留	ROP 和用户选项字节 (RDP & USER)
0x1FFF C008	保留	扇区 0 到 11 的写保护 nWRP 位
0x1FFE C000	保留	保留
0x1FFE C008	保留	扇区 12 到 23 的写保护 nWRP 位

设置FLASH的读写保护及解除

选项字节的内容

选项字节具体的数据位配置：

选项字节（字，地址 0x1FFF C000）	
RDP： 读保护选项字节。 读保护用于保护 Flash 中存储的软件代码。	
位 15:8	0xAA：级别 0，无保护 其它值：级别 1，存储器读保护（调试功能受限） 0xCC：级别 2，芯片保护（禁止调试和从 RAM 启动）
USER： 用户选项字节 此字节用于配置以下功能： 选择看门狗事件：硬件或软件 进入停止模式时产生复位事件 进入待机模式时产生复位事件	
位 7	nRST_STDBY 0：进入待机模式时产生复位 1：不产生复位
位 6	nRST_STOP 0：进入停止模式时产生复位 1：不产生复位

设置FLASH的读写保护及解除

选项字节的内容

选项字节具体的数据位配置：

选项字节（字，地址 0x1FFF C000）	
位 5	WDG_SW 0：硬件看门狗 1：软件看门狗
位 4	0x1：未使用
位 3:2	BOR_LEV： BOR 复位级别 这些位包含释放复位信号所需达到的供电电压阈值。 通过对这些位执行写操作，可将新的 BOR 级别值编程到 Flash。 00： BOR 级别 3 (VBOR3)，复位阈值电压为 2.70 V 到 3.60 V 01： BOR 级别 2 (VBOR2)，复位阈值电压为 2.40 V 到 2.70 V 10： BOR 级别 1 (VBOR1)，复位阈值电压为 2.10 V 到 2.40 V 11： BOR 关闭 (VBOR0)，复位阈值电压为 1.8 V 到 2.10 V
位 1:0	0x1：未使用
选项字节（字，地址 0x1FFF C008）	
位 15	SPMOD：选择 nWPRi 位的模式 0：nWPRi 位用于写保护(默认) 1：nWPRi 位用于代码读出保护(Proprietary code readout protection ， PCROP)
位 14	DB1M：设置内部 FLASH 为 1MB 的产品的双 Bank 模式 0：单个 Bank 的模式 1：使用两个 Bank 的模式
位 13:12	0xF：未使用
nWRP： Flash 写保护选项字节。 扇区 0 到 11 可采用写保护。	
位 11:0	nWRPi (i 值为 0-11，对应 0-11 扇区的保护设置)： 0：开启所选扇区的写保护 1：关闭所选扇区的写保护
选项字节（字，地址 0x1FFE C000）	
位 15:0	0xFF：未使用
选项字节（字，地址 0x1FFE C008）	
位 15:12	0xF：未使用
nWRP： Flash 写保护选项字节。 扇区 12 到 23 可采用写保护。	
位 11:0	nWRPi： 0：开启扇区 i 的写保护。 1：关闭扇区 i 的写保护。

主要讲解选项字节配置中的RDP位和PCROP位，它们分别用于配置读保护级别及代码读出保护。

设置FLASH的读写保护及解除

RDP读保护级别

修改选项字节的RDP位的值可设置内部FLASH为以下保护级别：

- **0xAA：级别0，无保护**

这是STM32的默认保护级别，它没有任何读保护，读取内部FLASH及“备份SRAM”的内容都没有任何限制。(注意这里说的“备份SRAM”是指STM32备份域的SRAM空间，不是指主SRAM，下同)

- **其它值：级别1，使能读保护**

把RDP配置成除0xAA或0xCC外的任意数值， 都会使能级别1的读保护。在这种保护下，若使用调试功能(使用下载器、仿真器)或者从内部SRAM自举时都不能对内部FLASH及备份SRAM作任何访问(读写、擦除都被禁止)；而如果STM32是从内部FLASH自举时，它允许对内部FLASH及备份SRAM的任意访问。

设置FLASH的读写保护及解除

RDP读保护级别

也就是说，在级别1模式下，任何尝试从外部访问内部FLASH内容的操作都被禁止，例如无法通过下载器读取它的内容，或编写一个从内部SRAM启动的程序，若该程序读取内部FLASH，会被禁止。而如果是芯片自己访问内部FLASH，是完全没有问题的，例如前面的“读写内部FLASH”实验中的代码自己擦写内部FLASH空间的内容，即使处于级别1的读保护，也能正常擦写。

当芯片处于级别1的时候，可以把选项字节的RDP位重新设置为0xAA，恢复级别0。在恢复到级别0前，芯片会自动擦除内部FLASH及备份SRAM的内容，即降级后原内部FLASH的代码会丢失。在级别1时使用SRAM自举的程序也可以访问选项字节进行修改，所以如果原内部FLASH的代码没有解除读保护的操作时，可以给它加载一个SRAM自举的程序进行保护降级，后面我们将会进行这样的实验。

设置FLASH的读写保护及解除

RDP读保护级别

- **0xCC：级别2，禁止调试**

把RDP配置成0xCC值时，会进入最高级别的读保护，且设置后无法再降级，它会永久禁止用于调试的JTAG接口(相当于熔断)。在该级别中，除了具有级别1的所有保护功能外，进一步禁止了从SRAM或系统存储器的自举(即平时使用的串口ISP下载功能也失效)，JTAG调试相关的功能被禁止，选项字节也不能被修改。它仅支持从内部FLASH自举时对内部FLASH及SRAM的访问(读写、擦除)。

由于设置了级别2后无法降级，也无法通过JTAG、串口ISP等方式更新程序，所以使用这个级别的保护时一般会在程序中预留“后门”以更新应用程序，若程序中没有预留后门，芯片就无法再更新应用程序了。所谓的“后门”是一种IAP程序(In Application Program)，它通过某个通讯接口获取将要更新的程序内容，然后利用内部FLASH擦写操作把这些内容烧录到自己的内部FLASH中，实现应用程序的更新。

设置FLASH的读写保护及解除

RDP读保护级别

不同级别下的访问限制如下表：

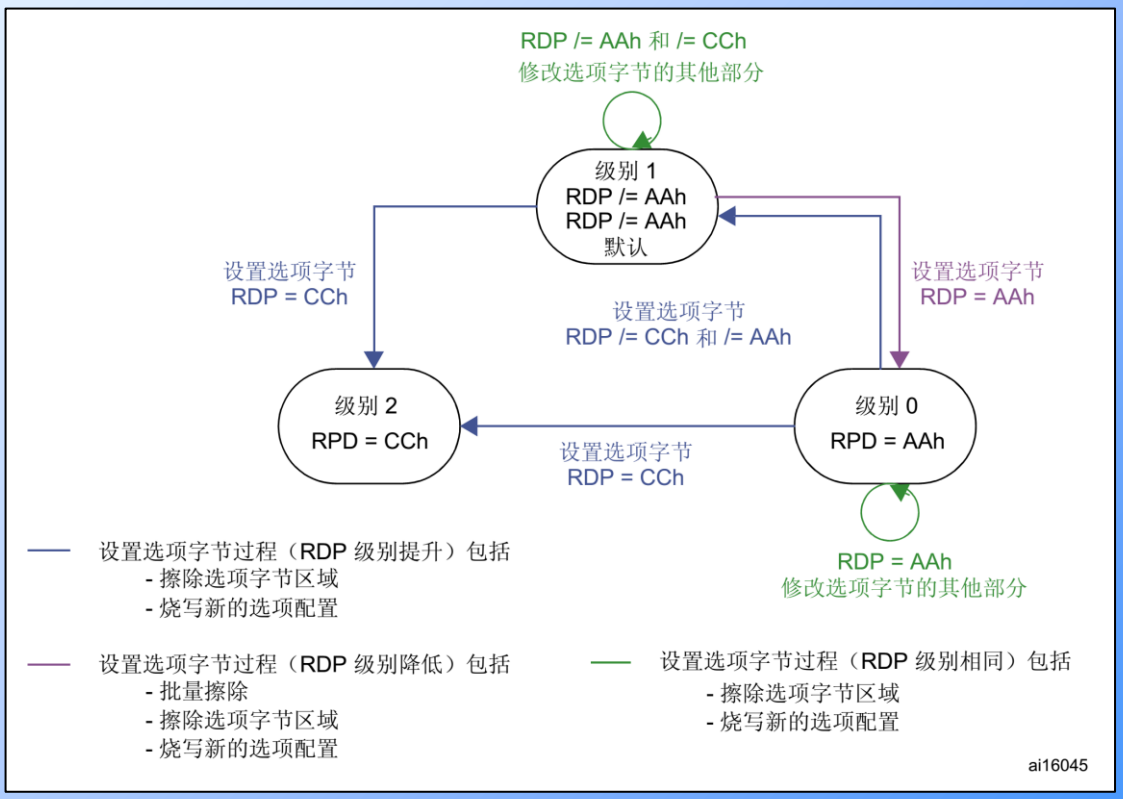
存储区	保护级别	调试功能，从 RAM 或系统存储器自举			从 Flash 自举		
		读	写	擦除	读	写	擦除
主 Flash 和备份 SRAM	级别 1	否		否(1)	是		
	级别 2	否			是		
选项字节	级别 1	是			是		
	级别 2	否			否		
OTP	级别 1	否		NA	是		NA
	级别 2	否		NA	是		NA

1. 只有在 RDP 从级别 1 更改为级别 0 时，才会擦除主 Flash 和备份 SRAM。OTP 区域保持不变。

设置FLASH的读写保护及解除

RDP读保护级别

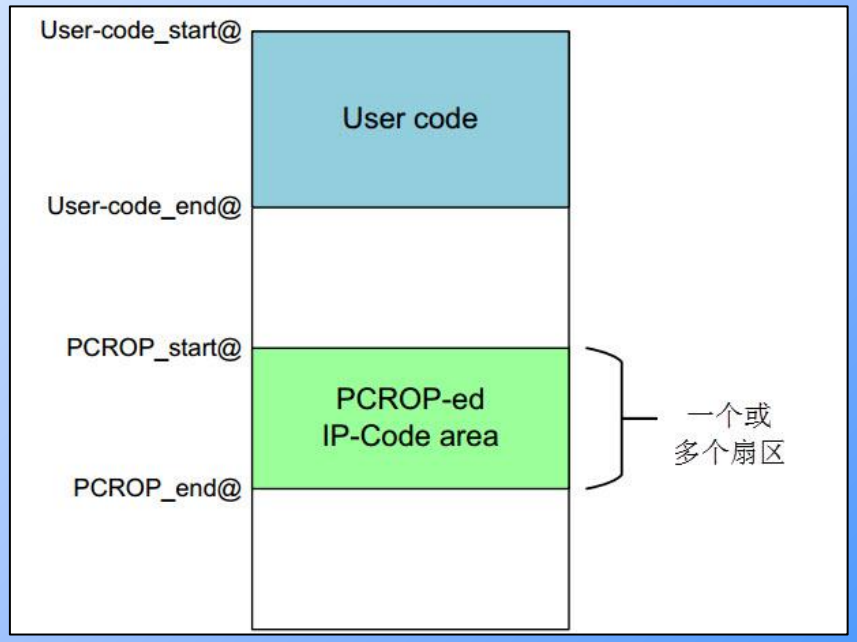
不同保护级别之间的状态转换图：



设置FLASH的读写保护及解除

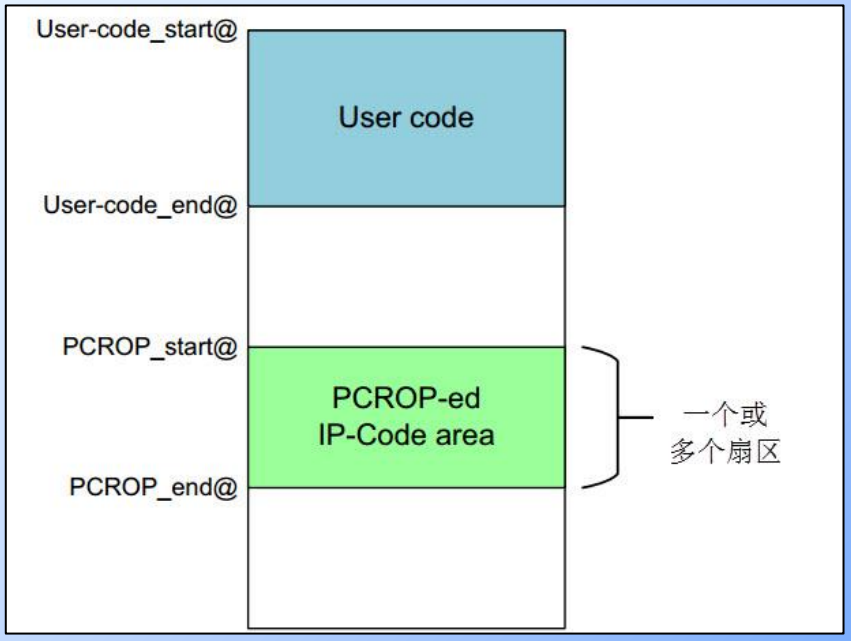
PCROP代码读出保护

在STM32F42xx及STM32F43xx系列的芯片中，除了可使用RDP对整片FLASH进行读保护外，还有一个专用的代码读出保护功能（Proprietary code readout protection，下面简称PCROP），它可以为内部FLASH的某几个指定扇区提供保护功能，所以它可以用于保护一些IP代码，方便提供给另一方进行二次开发：



设置FLASH的读写保护及解除

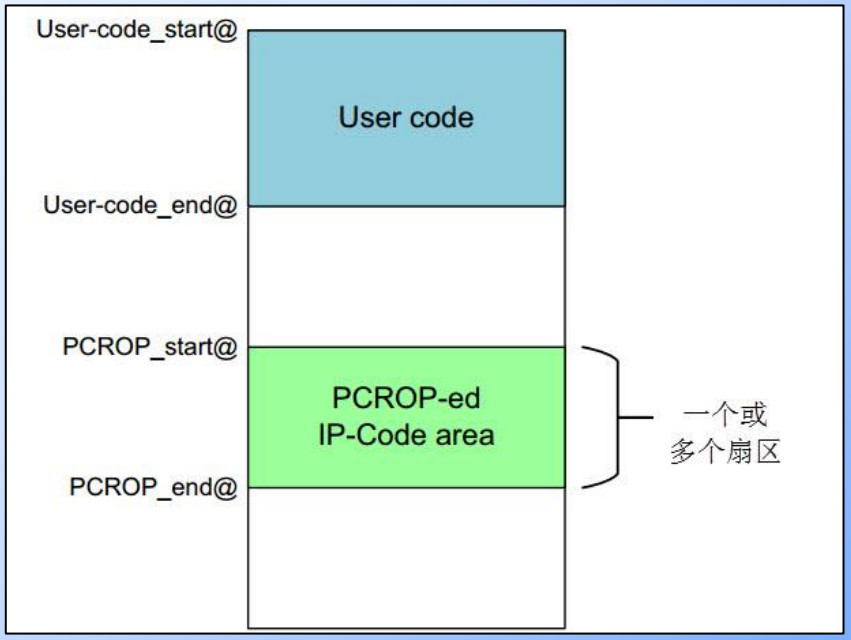
PCROP代码读出保护



当SPMOD位设置为0时(默认值)，nWRPi位用于指定要进行写保护的扇区，这可以防止错误的指针操作导致FLASH 内容的改变，若扇区被写保护，通过调试器也无法擦除该扇区的内容；当SPMOD位设置为1时，nWRPi位用于指定要进行PCROP保护的扇区。其中PCROP功能可以防止指定扇区的FLASH内容被读出，而写保护仅可以防止误写操作，不能被防止读出。

设置FLASH的读写保护及解除

PCROP代码读出保护



当要关闭PCROP功能时，必须要使芯片从读保护级别1降为级别0，同时对SPMOD位置0，才能正常关闭；若芯片原来的读保护为级别0，且使能了PCROP保护，要关闭PCROP时也要先把读保护级别设置为级别1，再在降级的同时设置SPMOD为0。

THANKS

论坛： www.chuxue123.com

淘宝：firestm32.taobao.com

扫描进入淘宝店铺