Hello and welcome to this presentation of the STM32C0 System memories protection. It will cover the different means for protecting code and data.

Overview

- Purposes:
  1. Provide read and write protection of embedded firmware and data in:
     - Flash memory
     - Backup registers
  2. Provide secure execution of sensitive firmware

Application benefits

- Protection of STM32 embedded software intellectual property
- Prevents hacking or dumping code through a JTAG interface or other possible means of external attack
- Protects code/data from unwanted/accidental erasure (i.e. loader, calibration data)
- Allows development of secure applications (secure boot or secure firmware update…)

life.augmented

2

Memory protections have been designed for different purposes.
A read protection, for example, will prevent the dumping of embedded software code through an external access and will protect the developer's intellectual property.
A write protection will prevent certain Flash sectors from being accidentally erased by a load overflow in a software or data update procedure.
STM32C0 microcontrollers provide several features for protecting code and data located in Flash memory and backup registers.
In addition to these typical memory protections, the STM32C0 also implements a mechanism to ensure the safe execution of sensitive firmware.
The following slides will describe all these protection features.

Key features

- **Readout protection (RDP)**
  - Global protection of flash memory and backup registers against external access
  - Memories and registers are protected from SWD access when boot is different from user Flash memory
  - Three RDP levels defined from no protection to full and permanent protection

Protection depends on the RDP level

- **Proprietary Code Read Out Protection (PCROP)**
  - Flash memory area protection against software IP read and write access
  - Flash memory Code with PCROP attribute can only be executed

Request	Permission
Read	NO
Write	NO
Execute	**YES**

life.augmented

3

The following means are provided for code protection purposes:

- RDP: ReaDout Protection
- PCROP: Proprietary code readout protection
- WRP: Write protection

Secure User Memory protection ensures the safe execution of sensitive applications in addition to code and data protection.

Readout Protection, or RDP is a global mechanism that prevents external read access to Flash memory, option bytes and backup registers.

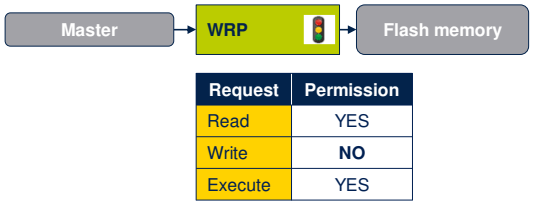
An external access can be gained by using a JTAG connector, a Serial Wire port or the boot software embedded in SRAM.

Three levels of RDP protection are defined from Level 0, which offers no protection at all, to Level 2 which has full

and permanent protection.
Protection levels will be described in the following slides.
PCROP is a memory access protection against code dumping. It is used to protect the intellectual property of the code.
The protected firmware remains executable but read and write access performed by the CPU executing malicious 3rd-party code (like Trojan horse) are prohibited.

Key features

- **Write protection (WRP)**
  - Flash memory sectors protection against Write/Erase/Program access
  - Flash memory code with write protection attribute is protected against unwanted write or erase operations



Request	Permission
Read	YES
Write	**NO**
Execute	YES

- **Secure User Memory protection**
  - Flash memory area protection with specific access mechanisms for sensitive firmware execution
  - Code and data in this area are only accessible after reset
  - Code is executed prior to any other process

Request	Permission
Secure Read	YES
Non-Secure read	NO
Secure Write	YES
Non-Secure Write	NO
Secure Execute	YES
Non-Secure Execute	NO

life.augmented

4

The write protection mechanism prevents accidental or malicious write/erase operations.
Secure User memory is a Flash memory area with a specific protection mechanism to ensure the safe execution of sensitive firmware in addition to code and data protection.
After system reset, the code in the securable memory area can only be executed until the securable area becomes secured and never again until the next system reset.
This allows implementing software security services such as secure key storage or safe boot.
All protection mechanisms are configurable via the STM32C0 option bytes.

Key features

Protection mechanism	Protected memory	Granularity	Number of regions	Region size
Readout Protection	Main flash Option bytes SRAM	Entire main flash	Global	Global
Write Protection	Main flash	2-KB page	2	Defined by first and last pages
Proprietary Code Read-Out Protection	Main flash	512-byte sub-page	2	Defined by first and last sub-pages
Securable memory	Main flash	2-KB page	One	Defined by a number of pages, from 0 to 15, starting at page 0

life.augmented

This table summarizes the features of the various protection mechanisms.
It provides the following information:

- Type of memory which is protected
- Granularity of the protection
- Number of protection areas
- Definition of the size of the protected area.

Readout protection

Protection levels 0 and 1

- RDP Level 0 (Default)
  - No protection is set, all operations (R/W/Erase) are permitted on Flash memory, SRAM, and backup registers
  - Option bytes can be modified
- RDP Level 1
  - No access (read, erase, program) to Flash memory and backup registers can be performed while the debug port is connected or while booting from RAM or system Flash memory bootloader
    - A bus error and a hard fault interrupt are generated in case of a read or write request
  - Access to protected memories from user code are allowed when booting from user Flash memory
  - Option bytes can be modified and protection level regression to Level 0 is possible, but this causes the Flash memory and backup registers to be mass-erased

life.augmented

6

The read protection is activated by setting the RDP option byte and then, by applying a system reset to reload the new RDP option byte.
There are three levels of read protection from no protection (Level 0) to maximum protection or no debug (Level 2).
When the lowest RDP level, Level 0, is set, the device has no protection. All read or write operations (if no write protection is set) on the Flash memory and the backup registers are possible in all boot configurations (Flash user boot, debug or boot from RAM).
Option bytes are also changeable in this level.
Level 0 is the factory default level.
In Level 1, read protection is set for the flash memory and the backup registers.
In this level, protected memories are only accessible when

booting from User Flash memory.
Whenever a debugger access is detected or boot is not set to a user flash memory area, any access to the protected memories generates a system hard fault and a bus error, which block all code execution until the next power-on reset.
Note that option bytes can still be modified in this level, making it possible to remove the protection. This mechanism is explained in the next slides.

Readout protection

Level regression and Protection level 2

- Protection level regression from Level 1 to Level 0
  - Mass erase of Flash memory and backup registers
    - Protected areas (PCROP and Secure User memory) may be kept unchanged depending on their erase policy
  - Option bytes and OTP bytes are not erased
- RDP Level 2
  - All protections provided by Level 1 are active and permanent
  - Option bytes can no longer be changed, internally or externally
  - Serial Wire Debug is disabled
  - Boot from RAM or System memory (boot loader) are no longer allowed
  - Only boot in user Flash memory is allowed and enables all operations (R/W/Erase) on the Flash memory and backup registers

life.augmented

7

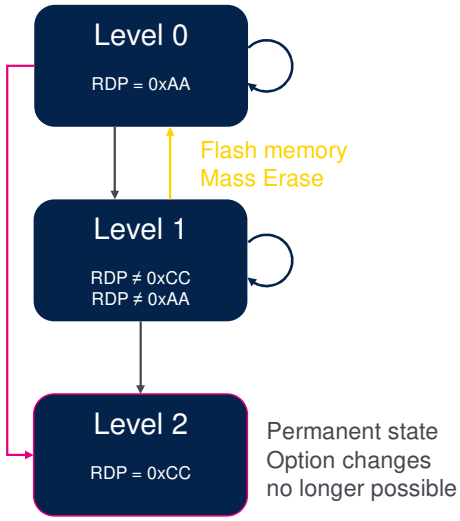
We have seen in the previous slide that it is possible to modify option bytes in Level 1. It is then possible to remove the protection by changing the protection level to Level 0.
This protection level regression will cause the Flash memory and the backup registers to be mass-erased. Flash areas protected by PCROP or configured as Secure User Memory can be erased or left unchanged depending on their erase policy configuration.
Readout protection Level 2 provides the same protection as in Level 1 but the protection becomes permanent. Option bytes cannot be modified, so once the RDP protection is set to this level, there is no way to modify it and level regression with mass-erase mechanism is no longer possible. This level must only be considered in the final product when the development stage is completed.

Note that to ensure that there are no backdoors, this protection cannot be bypassed even at ST factory.

Readout protection

Transition scheme

- Level 0 / RDP= 0xAA
  - Option byte change is allowed
  - Transition to Level 1 or Level 2 possible
- Level 1 / RDP != (0xAA | 0xCC)
  - Option byte change is allowed
  - Transition to Level 0 with mass erase of user Flash memory, backup registers and SRAM
  - Transition to permanent protection (Level 2) possible
- Level 2 / RDP = 0xCC
  - Option bytes are frozen
  - No transition possible



life.augmented

This slide shows the possible transitions between each readout protection level.
It is always possible to raise the protection level, but regression is only possible between Level 1 and Level 0 with the consequence of a full main Flash erase operation.
The RDP option byte is protected by a complementary byte.
Note that the RDP level is coded in one option byte; Level 0 is coded by a 0xAA value, Level 2 is coded by a 0xCC value and Level 1 is coded by any value other than 0xAA or 0xCC.

Readout protection

Summary

Area	Protection level (RDP)	Access rights when Boot in User Flash memory	Access rights when not booting from main flash or Debug Access detected
Main Flash memory	1	R/W/E	R
	2	R/W/E	-(1)
System Flash memory (Boot loader)	1	R	R
	2	R	-(1)
Option bytes	1	R/W/E	R/W/E
	2	R	-(1)
Backup registers	1	R/W	No Access
	2	R/W	-(1)
OTP	1	R/W	No Access
	2	R/W	-(1)

(1): In RDP2, only Boot in User Flash memory is allowed R: Read W: Write E: Erase

life.augmented

This table summarizes the different types of access authorized for the Flash memory and backup registers according to the readout protection (RDP) level, configured boot mode and with debug access, as seen in previous slides.

PCROP

Protect confidentiality of software IP code

- Software intellectual properties protection
  - ST or third-parties can develop and sell specific software IPs for STM32 MCUs
    - These IPs are used for further applications development and need to be protected against unauthorized copy
  - The PCROP feature ensures software IP protection against dumping from internal (malicious firmware) or external Flash memory access (debug port)

- PCROP attributes
  - The PCROP area is execute-only
    - Read/Write/Erase operations are not permitted
    - PCROP code needs to be compiled with the appropriate options (armcc) " –execute_only" to be compliant with this memory attribute
  - Protection is enabled regardless of the RDP level

life.augmented

10

PCROP means Proprietary code readout protection
Third-parties may develop and sell specific software IPs for STM32 microcontrollers and original equipment manufacturers may use them when developing their own application code.
In order to protect the software intellectual property (IP), the code must not be copied or read.
The PCROP's purpose is to protect the confidentiality of 3rd-party software intellectual property code against malicious users independent of the RDP level setting.
The protected firmware can only be executed by the Cortex®-M0+ core. Any other accesses (DMA, debug and data read, write and erase) are strictly prohibited.
To be compliant with this constraint, the firmware must be compiled with the appropriate compilation option. For example: "–execute_only" (for Keil tools). Without this

option, constants are interleaved with functions in the read-only section, called the literal pool.
The Cortex-M0+ MPU does not support execute-only access permissions.

PCROP

Setting/Unsetting

- Setting
  - Two PCROP areas can be defined
  - The PCROP area is defined with a granularity of 512 bytes and can be set from 512 bytes up to the full bank
  - PCROP areas are defined through option byte registers
- Resetting
  - The only way to deactivate PCROP is by RDP level regression from Level 1 to Level 0
    - This regression level will trigger a Flash memory mass erase operation
  - An additional option bit (PCROP_RDP) allows the selection of the PCROP areas to erase when the RDP protection is changed from Level 1 to Level 0

life.augmented

11

The proprietary code readout protected areas in Flash memory are defined through the option bytes.
Two PCROP areas can be defined. Each area is configured with a granularity of 512 bytes and can be set from 512 bytes up to the full bank.
The areas are protected against data accesses.
Note that pages protected with the PCROP feature are also protected against Write access, offering protection against unwanted page write or erase operations.
The PCROP protection can only be removed by RDP level regression from Level 1 to Level 0. When executed, this mechanism triggers a full mass erase of the Flash memory.
Depending on the PCROP_RDP option bit, the PCROP areas are erased when the RDP protection is changed from Level 1 to Level 0.

Flash memory write protection

Protects code and data from unwanted or accidental erasure

- Write protection attributes
  - Protected pages cannot be erased or programmed
- Setting/Resetting
  - Protection is set independently for each page (2 Kbytes) of the Flash memory
  - Protection is set in option byte registers
  - Write protection can be reset in RDP Level 0 and Level 1
    - It cannot be modified in RDP Level 2
  - If any sector is write-protected, the level regression mechanism does not work
    - Write protection must be removed prior to a level regression and a Flash memory Mass Erase

life.augmented

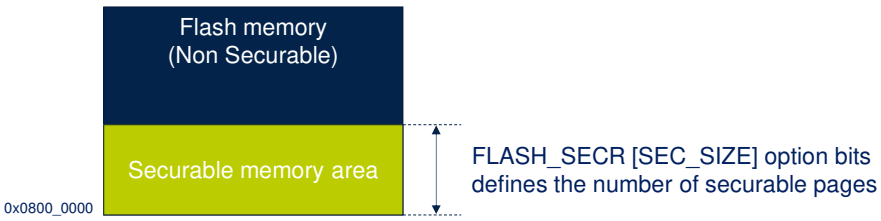
12

The write protection protects code and non-volatile data from unwanted or accidental erasure.
This protection is only available on the main Flash memory. The write protection can be set on a selection of Flash memory sectors only.
There are 16 pages of 2 Kbytes in STM32C0 microcontrollers.
When a sector is protected, it cannot be erased or programmed. Any attempt to write-access the sector will cause a Flash memory error.
If at least one sector is write-protected, a mass-erase of the Flash memory cannot be performed. The protection needs to be removed first.

Securable memory area

Introduction

- The main purpose of the securable memory area is to protect a specific part of Flash memory against undesired access
  - This allows implementing software security services such as secure key storage or safe boot



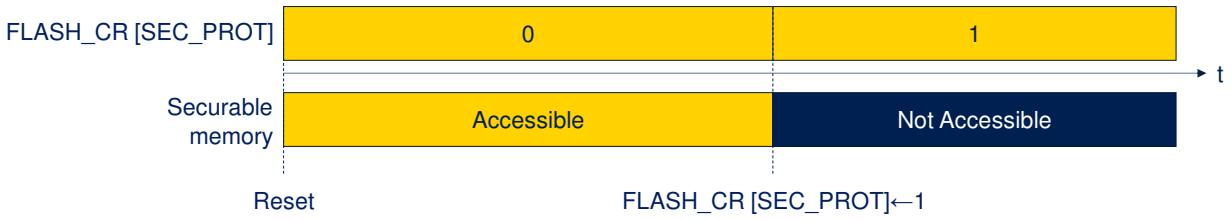
  - When FLASH_SECR[SEC_SIZE] option bits are equal to zero, securable memory is not implemented
    - This field can only be modified in RDP Level 0

life.augmented

13

The purpose of the securable memory is to store code and data, available during the boot time, that becomes inaccessible once the boot program sets a control bit.
The typical use case consists in performing an authentication and possibly decryption of the software image present in the flash memory by using cryptographic keys contained in the securable memory.
The authentication and decryption programs are also stored in the securable memory.
Option bits are used to set the size of the securable memory in page units. Base address is always 0x0800_0000, which corresponds to Cortex-M0+ reset vectors.
When the SEC_SIZE field in the option bytes is equal to zero, securable memory is disabled.
This field can only be modified in RDP Level 0.

Securable memory area



- By default, after a reset, the securable memory is accessible
  - Once the SEC_PROT bit is set in FLASH_CR register, the securable memory becomes inaccessible until the next reset
  - Only a reset can clear the SEC_PROT bit

life.augmented

When software sets the SEC_PROT bit in the FLASH_CR register, the securable memory is no longer accessible.
In case of secure boot, used to perform image authentication and decryption, the SEC_PROT bit is set to one when the authentication is successful, just before branching to the first instruction of the image.
Once the SEC_PROT bit is set, it cannot be cleared by software.
The only way to clear this bit is to apply a reset.

Securable memory area

- The content of the securable memory is erased upon changing from RDP Level 1 to Level 0, even if it overlaps with PCROP pages

Securable memory size (SEC_SIZE[6:0])	Securable memory ?	PCROP_RDP	Erased pages
0	NO	1	All (mass erase)
0		0	All but PCROP
>0	YES	1	All (mass erase)
>0		0	All but PCROP outside the securable memory area

  - PCROP_RDP bit controls whether PCROP is preserved when RDP level decreases from Level 1 to Level 0:
    - =0: PCROP is not erased
    - =1: PCROP is erased

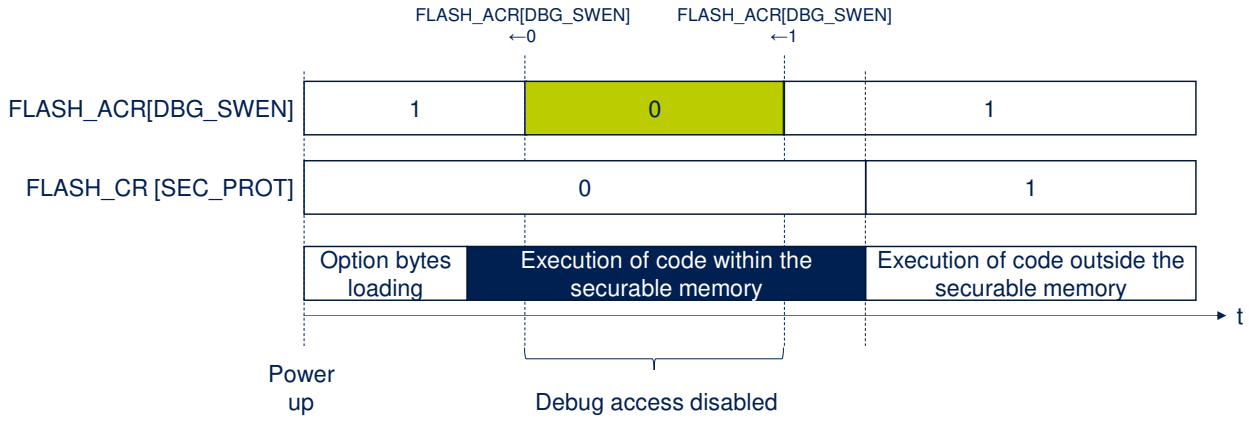
life.augmented

15

Of course, code present in the securable memory may decide to erase a part or the securable memory.
Furthermore, changing the Flash Read Protection (RDP) level from Level 1 to Level 0 triggers the erasure of the securable memory.
Note that the code present in the securable area can also be protected against read and write accesses, by mapping it into Proprietary Code Read Out Protection (PCROP) areas.
Changing the RDP level from Level 1 to Level 0 will erase these PCROP areas, whatever the value of the PCROP_RDP bit.
Only the contents of PCROP areas outside the securable memory address range will be preserved.

Disabling core debug access

- For executing sensitive code or manipulating sensitive data in securable memory area, the debug access to the core can temporarily be disabled



life.augmented

Taking control of the Cortex-M0+ by using invasive debug can be temporarily disabled through the DBG_SWEN control bit.
For instance, the secure boot can decide to clear this bit before performing authentication/decryption and then to set this bit to one to re-enable invasive debug once the authentication is successful.

Forcing boot from Flash memory

- STM32C0 boot memories:
  - Embedded SRAM
  - System memory (bootloader)
  - Main Flash memory
- To increase the security and establish a chain of trust, the BOOT_LOCK option bit of the FLASH_SECR register allows forcing the system to boot from the Main Flash memory regardless the other boot options
  - It is always possible to set the BOOT_LOCK bit
  - Conditions to reset this bit:
    - RDP is set to Level 0, or
    - RDP is set to Level 1, while Level 0 is requested, and a full mass-erase is performed

life.augmented

17

In the STM32C0, three different boot modes can be selected: boot from embedded SRAM, boot from system memory and boot from main Flash memory.
Executing a secure boot from securable memory implies that the boot area is the Flash memory.
To disable the other boot areas, the BOOT_LOCK option bit has to be set in the FLASH_SECR register.
It is always possible to set the BOOT_LOCK bit
However, resetting is possible only when RDP level is zero or RDP is changed from Level 1 to Level 0, which causes a mass-erase.

Option bytes loading fail-safe

- All memory protections presented in this module are stored in Option Bytes (OB)
- In case of mismatch on loading the OB:
  - For any WRP option mismatch, “No protection” is set for both WRP
  - For RDP option, the value of mismatch is the default value “Level 1”
  - For any PCROP mismatch all PCROP are set to “all memory protected”
  - For BOOT_LOCK, the value of mismatch is “boot forced from Main Flash memory”
- Those mismatch values force a secure configuration that might permanently lock the device
  - To prevent this, only program option bytes in a safe environment – safe supply, no pending watchdog, and clean reset line

life.augmented

18

During option byte loading, the options are read by double word. If the word and its complement are matching, the option word is copied into the option register.
If the comparison between the word and its complement fails, a status bit OPTVERR is set.
Mismatch values are forced into the option registers as indicated in the second bullet.
Upon an option byte programming failure (for any reason, such as loss of power or a reset during the option byte change sequence), the mismatch values of the option bytes are loaded after reset .
Those mismatch values force a secure configuration that might permanently lock the device.
The STM32C0 implements a new feature: debug capabilities remain enabled in case of option byte mismatch.

Related peripherals

- Refer to this training related to this peripheral:
  - STM32C0- Flash memory

life.augmented

Please refer to the Flash memory presentation to learn more about the memory architecture, option bytes and Flash memory operations.

Thank you

life.augmented

Thank you for attending this presentation!