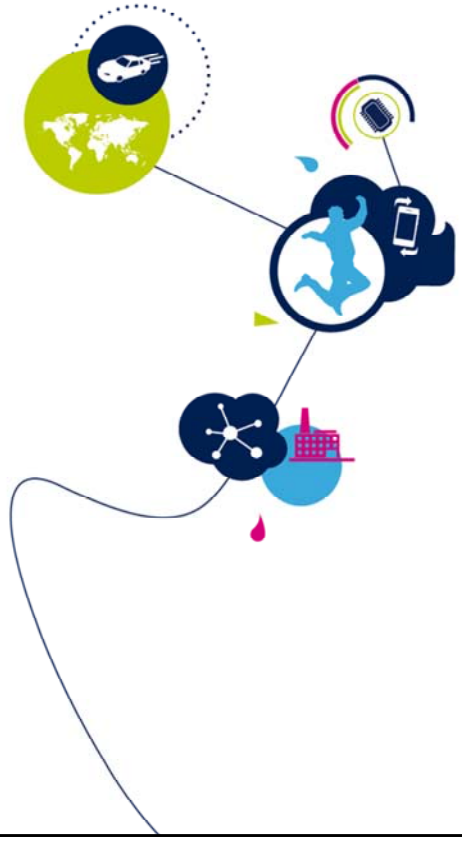
STM32L5 - Flash

Embedded Flash memory

Revision 0.1

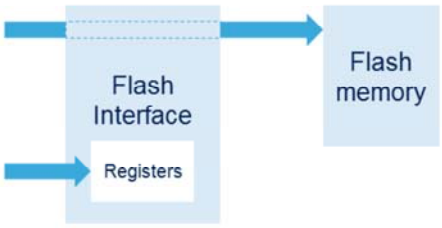


life.augmented

Hello, and welcome to this presentation of the embedded Flash memory which is included in all products of the STM32L5 microcontroller family.

Overview

- STM32L5 embeds up to 512 Kbytes of Flash memory with dual-bank architecture
- The Flash memory interface manages all accesses (read, programming, erasing), memory protection, security and option byte programming



life.augmented

Application benefits

- High-performance and low-power
- Read-while-write capability
- Small erase granularity
- Short programming time
- Dual-bank booting
- Security and protection

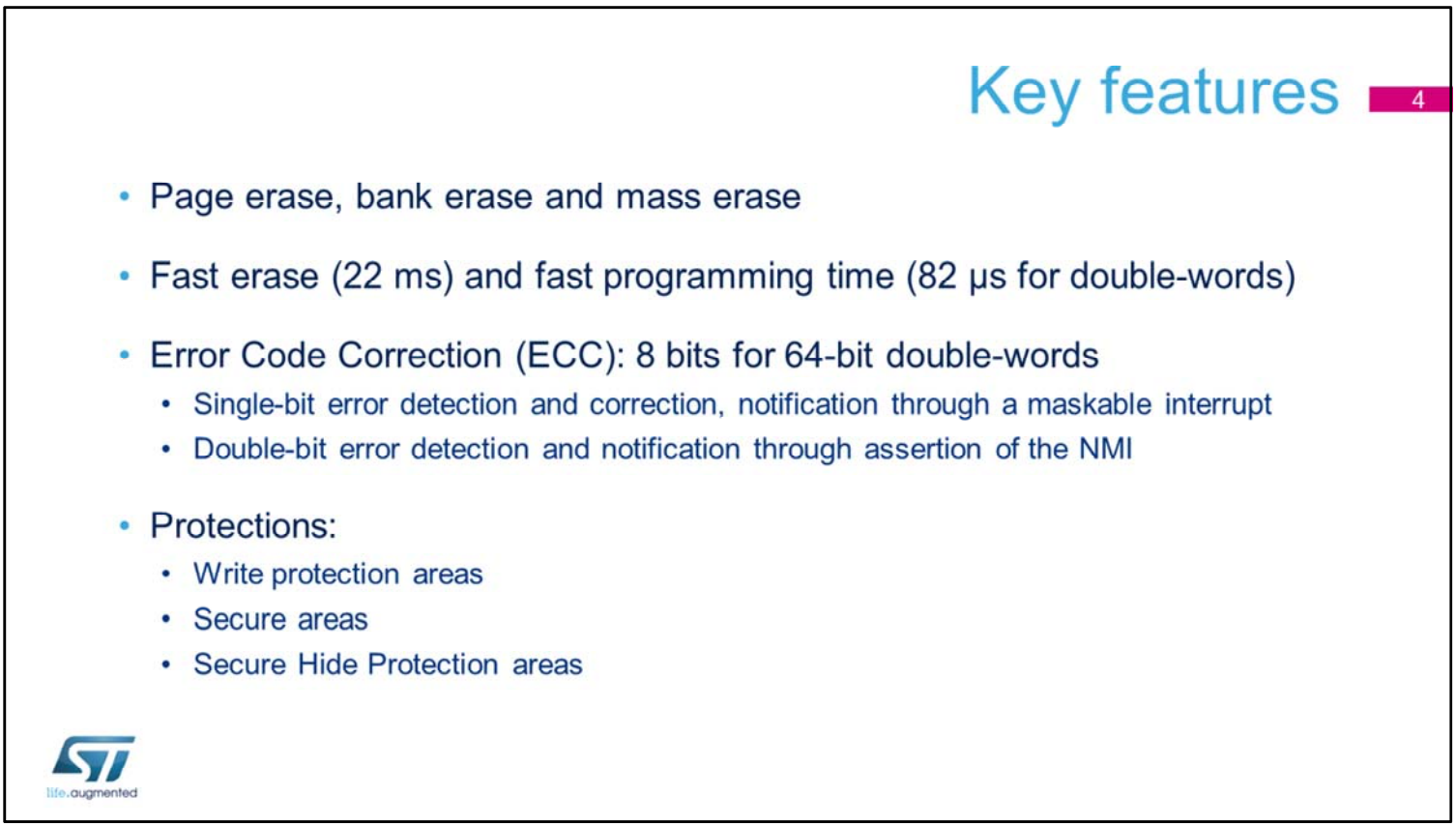
The STM32L5 microcontrollers embed up to 512 Kilobyte of Flash memory with dual-bank architecture.
The Flash memory interface manages all memory accesses (read, programming and erasing) as well as memory protection, security and option byte programming.
Applications using this Flash memory interface benefit from its high performance together with low-power access.
It supports read-while-write, has a small erase granularity, a short programming time and allows dual-bank booting.
It provides various security and protection mechanisms for code and data, read and write accesses.

STM32L5 flash properties

	FLASH_OPTR[DBANK]=0 (Single bank)	FLASH_OPTR[DBANK]=1 (Dual bank)
Size	512 KB	
Number of banks	1	2
Data width	128-bit	64-bit
Page size	4-Kbyte	2-Kbyte
Flash organization	128 pages	128 pages
Write Protect areas (WRPs)	4	2 per bank
Secure areas	2	1 per bank
Secure Hide Protection areas	2	1 per bank

life.augmented

The Flash memory size is 512 Kbytes.
The Number of banks is 1 or 2, depending on the DBANK option bit.
Note that read-while-write capability (or RWW) is only supported when the dual-bank architecture is active.
This feature enables the programming or erasing of one bank while executing code from the other bank.
The page size which provides the minimum erase granularity is 4 KB with a single bank and 2 KB with dual banks.
The number of pages is 128.
Regarding protection features, the flash interface offers 4 write protect areas in single bank mode, 2 per bank in dual bank mode
Two secure areas can be defined, each of them supporting a secure hide protection area.
Access to the hide protection area can be denied by setting a control bit.



The Flash memory supports page erase, bank erase and mass erase.
A page, bank or mass erase operation requires only 22 ms, and the programming time is only 82 µs for a double-word.
An 8-bit ECC code is appended to a programmed double-word. When reading the code is checked to detect and correct single-bit errors and detect double-bit errors.
In the case of an uncorrectable error, the Flash memory controller asserts the Non-Maskable Interrupt (NMI) to the Cortex®-M33.
The following protection mechanisms are supported:

- Write protection areas, used to protect against unwanted write operations
- Secure areas only accessible in secure state
- Secure hide protection areas that can be programmed as non-accessible after a control bit is set.

Flash memory organization (1/2)

5

The Flash memory is organized as follows:

- A Main memory block containing 128 pages
  - In single bank mode, page size is 4-KB
    - Each page consists of 8 rows of 512 bytes
  - In dual bank mode, page size is 2-KB
    - Each page consists of 8 rows of 256 bytes
- An Information block containing:
  - 16 KB for system memory which is reserved for use by ST and contains the **bootloader**
  - 10 KB for Root Secure Services (RSS)
  - 512 bytes OTP (one-time programmable) area for user data
  - 4 KB of option bytes for user configuration

life.augmented

The main memory contains 128 pages.
In single-bank mode, the page size is 4 KB, each page consisting of 8 rows of 512 bytes.
In dual-bank mode, the page size is 2 KB, each page consisting of 8 rows of 256 bytes.
In addition to the main Flash memory, the STM32L5 supports:

- A System memory of 32 Kbytes containing the ST bootloader that is used to reprogram the Flash memory through one of the following interfaces: USART, USB (DFU), I2C or SPI
- 10 Kbyte for Root Secure Services
- 512 bytes of OTP memory that can be used to store user data that must not be erased or modified. If one bit is zero, the entire double-word can no longer be written, even with the value zero.
- 4 Kbytes of option bytes containing default settings to configure IPs in the system-on-chip. They are

automatically loaded after a power-up reset.

Flash memory organization (2/2)

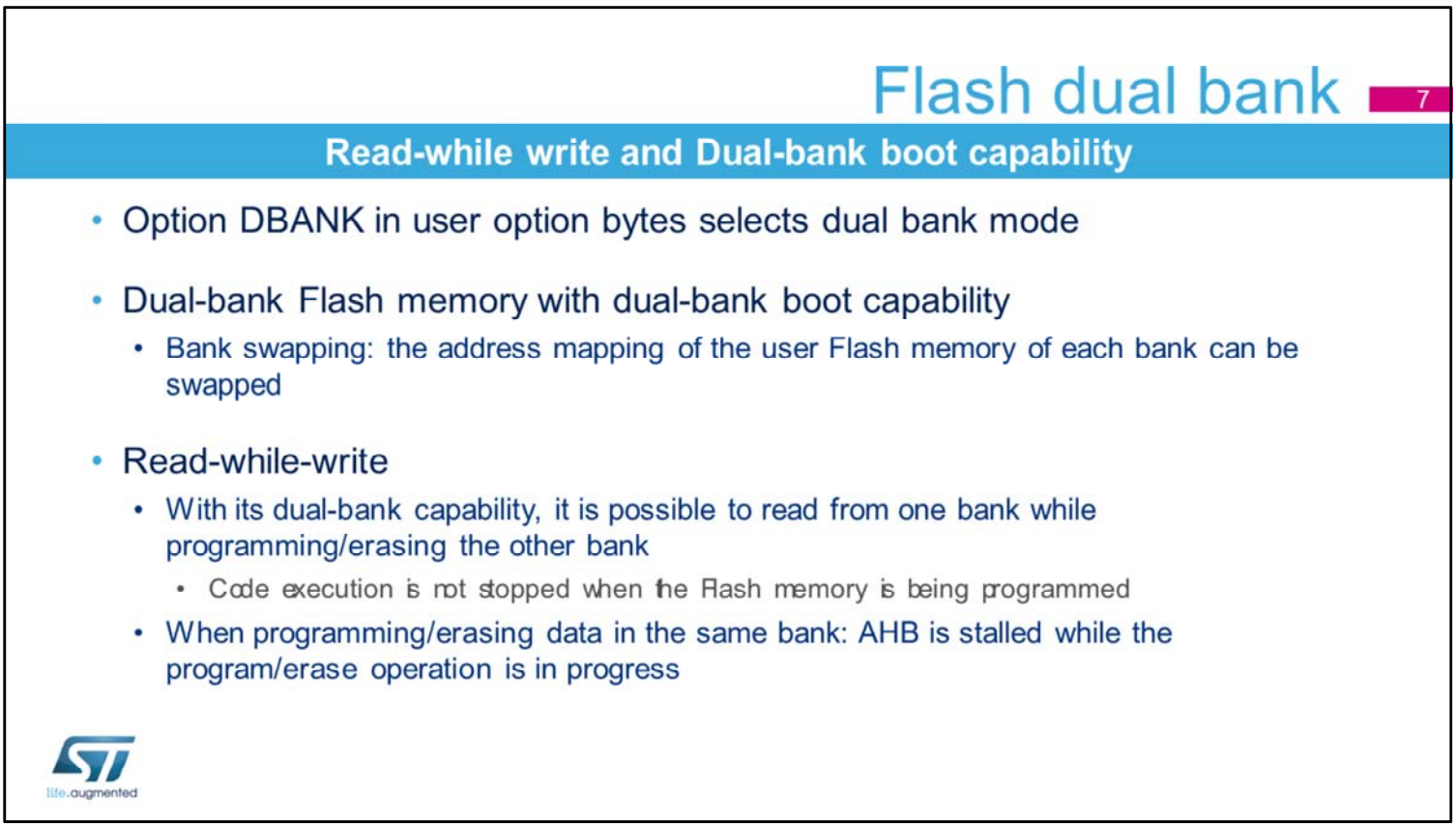
Flash area		Flash memory address (Cat 3, dual-bank)	Size	Name
Main memory	Bank 1	0x0800_0000– 0x0800_07FF	2 KB	Page 0
		…	…	…
		0x0803_F800– 0x0803_FFFF	2 KB	Page 127
	Bank 2	0x0800_4000– 0x0800_07FF	2 KB	Page 0
		…	…	…
		0x0807_F800– 0x0807_FFFF	2 KB	Page 127
Non-Secure Information block		0x0BF9_0000– 0x0BF9_3FFF	16 KB	System memory
		0x0BFA 0000– 0x0BFA_01FF	512 B	OTP area
Secure Information block		0x0FF8_0000– 0x0FF8_1FFF	8 KB	RSS
		0x0FF8_2000– 0x0FF8_27FF	2 KB	RSS library

Operation	Granularity
Programming	8-Byte
Erase	Mass, bank and page
Secure memory	Page
Secure Hide Protection	
Write protection	

life.augmented

The table on the left details the memory organization based on a Main Flash memory area and two information blocks when dual-bank architecture is enabled.
The non-secure information block contains the system memory and the OTP area, while the secure information block contains the RSS and RSS library.
The table on the right details the granularity of the Flash memory operations:

- Programming is done on 8-bytes
- Erase is done either globally (mass erase) or with bank or page granularity.
- The secure memory, write protection and secure hide protection, is aligned on pages.



The DUALBANK (DBANK) option is used to select either single bank or dual bank mode.
The Flash memory can be configured to support two banks, with read-while-write and dual-bank boot capability, able to boot from either Bank 1 or Bank 2.
The SWAP-BANK option, in the user option bytes, is used to swap Bank 1 and Bank 2 addresses.
By enabling the dual bank mode, read-while-write is supported. This feature permits a read operation to be performed on one bank while an erase or program operation is performed on the other bank.
A protection mechanism prevents masters accessing a bank while a program or erase operation is in progress in that bank.

Flash read access

165 DMIPS at 110 MHz

- 8-KB instruction cache enables a linear performance versus frequency, regardless of the Flash memory access time

Wait states (WS) (FLASH latency)	HCLK (MHz)		
	V_{CORE} Range 0	V_{CORE} Range 1	V_{CORE} Range 2
0 WS (1 CPU cycle)	≤ 20	≤ 20	≤ 8
1 WS (2 CPU cycle)	≤ 40	≤ 40	≤ 16
2 WS (3 CPU cycle)	≤ 60	≤ 60	≤ 26
3 WS (4 CPU cycle)	≤ 80	≤ 80	-
4 WS (5 CPU cycle)	≤ 100	-	-
5 WS (6 CPU cycle)	≤ 110	-	-

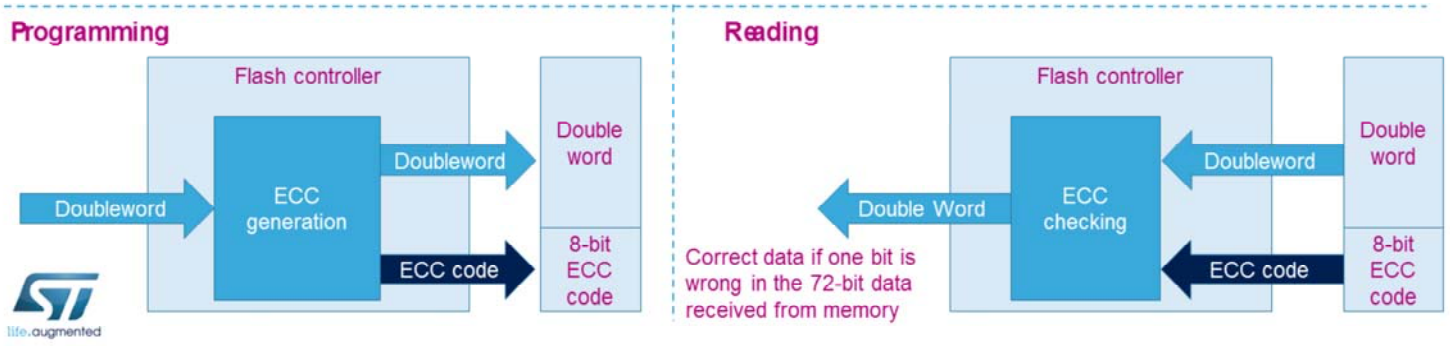
life.augmented

In order to read the Flash memory, it is required to configure the number of wait states to be inserted in a read access, depending on the clock frequency. The number of wait states also depends on the voltage scaling range.
In range 0, the flash memory can be accessed up to 110 MHz, with 5 wait states. It can be accessed with 0 wait states up to 20 MHz.
In range 1, the flash memory can be accessed up to 80 MHz, with 3 wait states.
In range 2, the flash memory can be accessed up to 26 MHz, with 2 wait states.
Thanks to the Instruction cache, the program can be executed with 0 wait states independent of the clock frequency. This provides an almost linear performance in relation to the frequency with a benchmark result of 165 Dhrystone MIPS at 110 MHz.

Flash memory features (1/2)

Robust memory integrity and safety

- **ECC** (Error Code Correction): 8 bits long for a 64-bit word
  - Single error correction: ECCC bit set in FLASH_ECCR, optional interrupt generation
  - Double error detection: ECCD bit set in FLASH_ECCR => NMI
  - **Failure address saved in the FLASH_ECCR register**



Data in Flash memory words are 72-bits wide: eight bits are added per each double word (64 bits). The ECC mechanism supports:

- Single error detection and correction
- Double error detection

The Programming granularity is 64 bits (really 72 bits including 8-bit ECC), 144 bits when single bank mode is used (two times 72 bits).

When one error is detected and corrected, the ECCC flag (ECC correction) is set in the Flash ECC register (FLASH_ECCR). An interrupt can be generated.

When two errors are detected, the ECCD flag (ECC detection) is set in the Flash ECC register (FLASH_ECCR). In this case, an NMI is generated

Programming/erase time

Short programming and erasing time & small page size
→ Advantage for data EEPROM emulation

Parameter	Typical value
64-bit programming time	82µ s
One row (256 bytes) programming time	2.61 ms
One page (2 Kbytes) programming time	20.91ms
Bank programming time	2.68 s
Page (2 Kbytes) erase time	22.02 ms
Mass erase time	22.13 ms

- Program and erase operations are only possible in voltage scaling range 0 and 1

The programming time of a row is equal to 82 µs multiplied by 32 double words.
The programming time of a page is equal to 82 µs multiplied by 256 double words
The Mass erase time, meaning a 512-Kbyte erase operation, takes approximately the same time as a page erase.

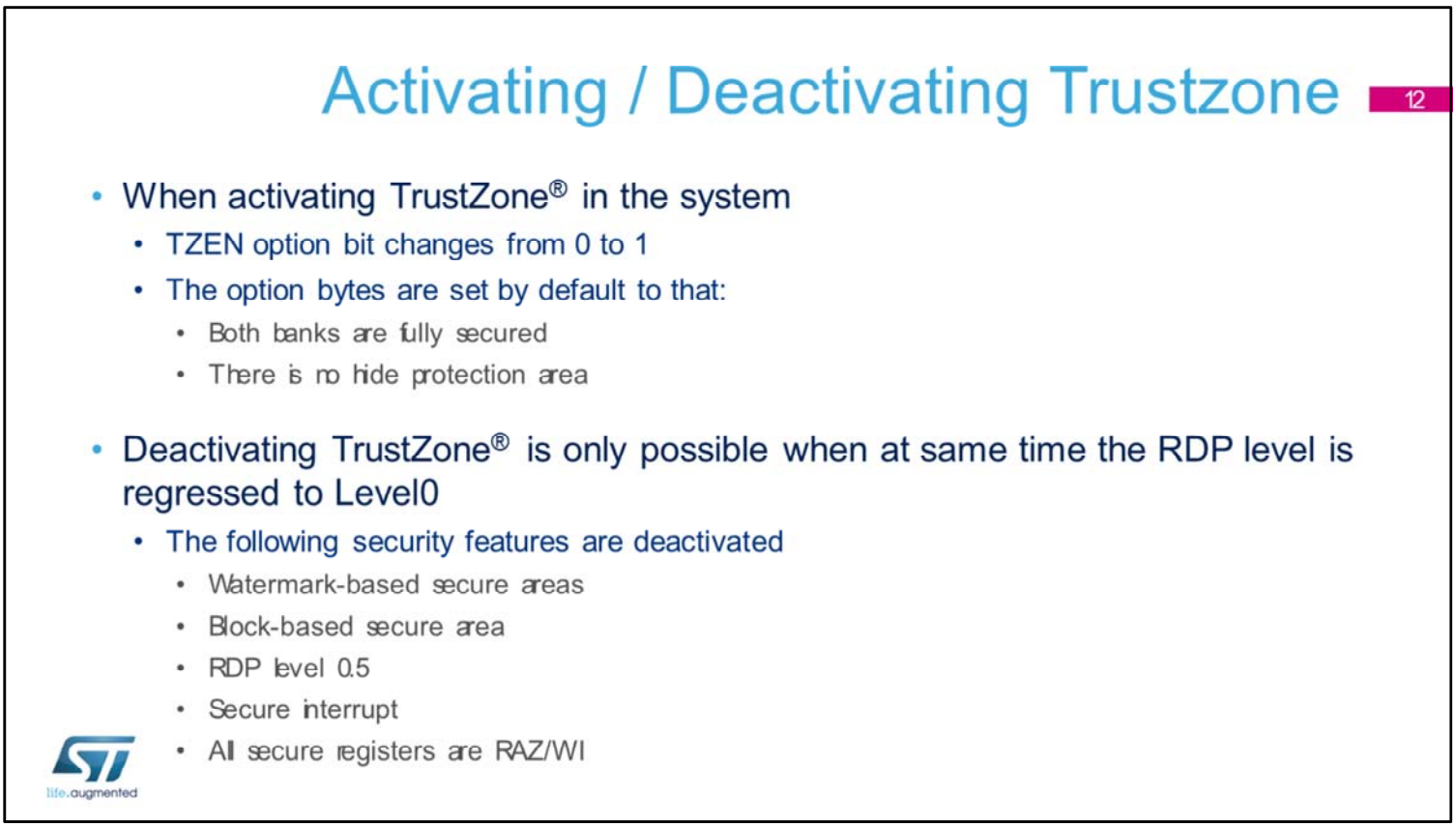
Flash memory retention

- Design expectation

Endurance	10 Kcycles minimum @ -40 to +105° C
Data retention	30 years after 10 Kcycles at 55° C 15 years after 10 Kcycles at 85° C 10 years after 10 Kcycles at 105° C 30 years after 1 Kcycle at 85° C 15 years after 1 Kcycle at 105° C 7 years after 1 Kcycle at 125° C

life.augmented

Each program / erase operation can degrade the Flash memory cell.
After an accumulation of program / erase cycles, memory cells can become non-functional, causing memory errors.
Endurance is the maximum number of erase/programming sequences that the Flash memory can support without affecting its reliability.
Data retention is defined as retaining a given data pattern for a given amount of time.
The retention depends on the number of program/erase cycles and also on the temperature.



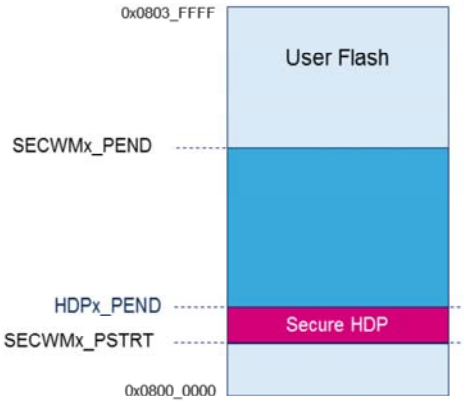
The global TrustZone system security is activated by setting the TZEN option bit in the FLASH_OPTR register.
By default, All Flash memory is secure.
When the TrustZone is active, additional security features are available:

- Secure watermark-based user options bytes defining secure, hide protection areas
- Secure or non-secure block-based areas can be configured on-the-fly after reset
- An additional readout protection: RDP level 0.5
- Erase or program operation can be performed in secure or non-secure mode with associated configuration bit.

When the TrustZone is disabled, these features are deactivated and all secure registers are read as zero, write ignored.

Secure areas (WM) - non-volatile settings (option bytes)

- Secure watermark areas
  - For each of them, the start and end addresses are defined in secure option bytes
- Secure Hide protection areas
  - Start address is the same as that of the secure area
  - End address is defined in secure option bytes



When TrustZone security is active, a part of the Flash memory can be protected against non-secure read and write accesses.
Up to two different non-volatile secure areas can be defined by option bytes and can only be read or written by a secure access :

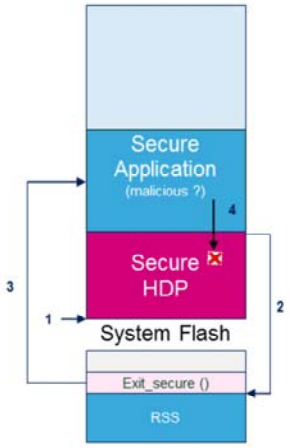
- In single-bank mode, two areas can be selected with a page granularity
- In dual-bank mode, one area per bank can be selected with a page granularity.

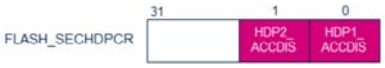
Each mode supports a secure hide protection area, starting at the same start page offset and ending at a programmable end page offset.
The contents of the secure hide protection area is marked as non-accessible after the corresponding HDP_ ACCDIS bit is set to one.

Hide Protection area (aka sticky)

- Secure Hide Protect (HDP) Memory Properties
  - Enables isolation of secure boot code & data (secrets) from (secure) application code
  - HDP area is activated by setting the HDPEN option bit
  - HDP size and HDPEN can only be modified while the HDPx_ ACCDIS bit is reset
  - Once the HDPx_ ACCDIS is set, no further operations are permitted in the HDP zone (size setting / read / write / erase)
    - Any page belonging to the HDP area can only be erased by the HDP code itself

1. System Boots and execute HDP area (sensitive code)
2. Call HDP exit function (immutable in RSS lib)
   - ➢ Disables / Hides secure HDP area until the next reset
3. Exit function will branch to the (secure) application code
4. (secure) firmware is no longer permitted to access the securable HDP area



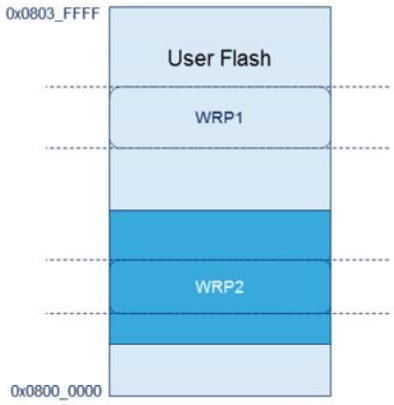


The secure HDP area is part of the Flash watermark-based secure area.
It enables isolation of the secure boot code and data secrets, such as authentication and cryptographic keys from the secure application code.
The HDP area is activated by programming the end page offset and setting the HDP enable bit.
Access to the hide protection area can be denied by setting the HDP ACCess DISable bit in the FLASH_SECHDPCR register.
When this bit is set, data reads, writes and instruction fetches on this hide protection area are denied.
The HDP ACCess DISable bit can be only cleared by a system reset.
The figure on the right explains the typical usage of the HDP area:
1- The System boots and executes code in the HDP area
2- The HDP exit function present in the RSS lib is called

once the secure boot is completed. This function sets the HDPx_ACCDIS bit.
3- The HDP exit function branches to the secure application
4- If the secure application attempts to branch to, or read data from the HDP area, the access is denied and an error is signaled.

Write protection areas - non volatile settings (option bytes)

- 4 independent WRP areas in single bank mode
  - 2 per bank in dual bank mode
- Start and End addresses defined in option bytes
  - Always aligned on number of pages
  - Write protection attribute orthogonal to other settings (Secure / HDP)



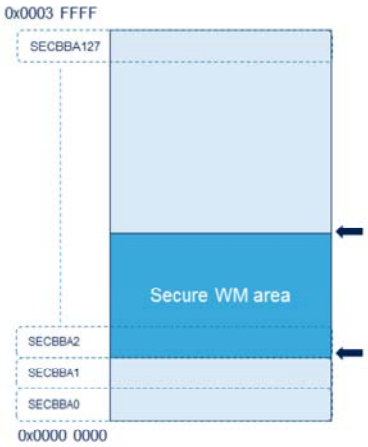
life.augmented

Four write protection areas are supported: two per bank when DBANK=1 and four for full memory when DBANK=0.
Program and erase operations are prohibited in write protection areas. Consequently, a software mass erase cannot be performed if an area is write-protected.
Each area is defined by a start page offset and an end page offset relative to the physical Flash bank base address.
The Write protection attribute is orthogonal to secure and HDP settings.

Block-based security attribute

16

- Any flash page can be set as secure/non-secure thanks to dedicated secure registers in the flash interface (FLASH_SECBB1R1-4/ FLASH_SECBB2R1-4)
  - At reset these registers are cleared (non-secure)
- Setting a page as secure, which already belongs to the secure watermark area, will have no effect



Any page can be programmed on-the-fly as secure or non-secure using the block-based configuration registers. One bit per page enables the secure software to dynamically configure a page as being secure or non-secure.
In dual-bank mode: FLASH_SECBB1R one to four registers are used to configure the security attribute for pages in bank1 and FLASH_SECBB2R one to four registers are used to configure the security attribute for pages in bank2.
In single-bank mode: the FLASH_SECBB1Rx registers are used to configure the security attribute for pages in the entire Flash memory.
When the page security attribute is set for a page, the security attribute is the same as the secure watermark-based area.
A secure page is only accessible by a secure access.
If the page security bit is set for a page already included in a secure watermark-based area, the page keeps the watermark-based protection security attributes.

To modify a page’s block-based security attribute, it is recommended to:

- Check that there is no ongoing Flash operation on that page
- Add an ISB instruction after modifying the page security attribute

Rules for modifying Secure watermark and HDP areas

	Option byte locked
HDPx_ACCDIS=1	SECWMx_PSTRT[6:0], SECWMx_PEND[6:0]
	HDPx_PEND[6:0], HDPxEN

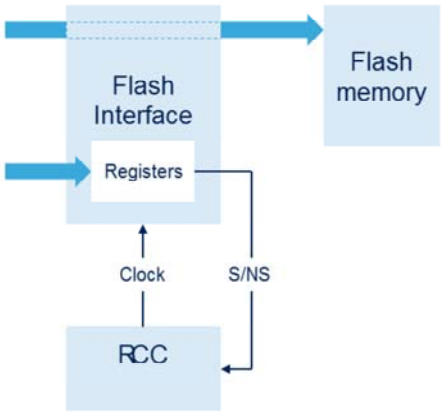
- Option bytes listed above, can only be modified when the HDPx_ACCDIS bit is cleared
  - When it is set, options bytes listed in the table above are locked and cannot be modified until next system reset
  - An attempt to modify one of the these option bytes while HDPx_ACCDIS bit is set will lead to the cancellation of byte modifications without an error flag

life.augmented

When the access disable bit is set for HDP area 1 or 2, the HDP setting can no longer be changed: HDP end page offset and enable bit.
The secure area configuration is also locked: secure start page offset and end page offset.
This locking remains active until the next reset.
If the user tries to modify one of the these option bytes while HDP ACCess DISable bit is set, the option bytes modification is discarded without error flag.

Flash secure state

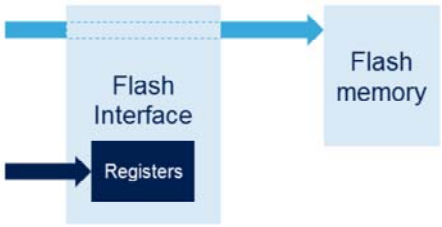
- Flash Interface Secure attribute
  - The Flash port becomes secure as soon as a secure area exists (watermark or block-based)
  - This will make the flash source clock secure (RCC)
- Overriding the secure attribute
  - The secure attribute can be inverted thanks to a specific bit (SECINV) in the FLASH_SECCR register
  - The source clock may remain non-secure while the flash interface has a secure attribute
- FLASH I/F is a trustzone aware IP, containing secure / non-secure registers



The Flash is secure when at least one secure area is defined either by watermark-based option bytes or block-based security registers.
This will lead to control the source clock of the flash as secure.
It is possible to override the Flash security state using the SECINV bit in the FLASH_SECCR register.
The source clock may remain non-secure while the flash interface has a secure attribute.
The flash interface is a trustzone aware IP, containing both secure and non-secure register

Flash peripheral port

- Flash Interface Privilege / unprivileged access is controlled by the FLASH_PRIVCFGR register
  - PRIV =0, all Flash registers can be read and written by both privileged or unprivileged accesses
  - PRIV =1, all Flash registers can be read and written by privileged accesses only
    - Unprivileged access to a privileged registers is RAZ/WI

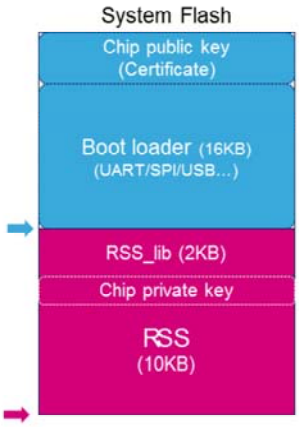


life.augmented

The Flash registers can be read and written by privileged and unprivileged accesses depending on PRIV bit in the FLASH_PRIVCFGR register.
When the PRIV bit is reset, all Flash registers can be read and written by both privileged or unprivileged accesses.
When the PRIV bit is set, all Flash registers can be read and written by privileged accesses only.
Unprivileged access to a privileged register is read as zero, write ignored.

Root Security Services (RSS)

- SYSTEM FLASH (Information block)
  - Immutable (similar to ROM)
  - Root Security Services
    - RSS_boot (sticky property- HDP like)
      - Unique entry point
      - Provides set of security services available at reset
    - RSS_lib
      - Multiple entry points (Trusted ST APIs)
      - Provides set of security services callable by User code
  - Boot Loader
    - Unique entry point
    - Classic bootloader functions
  - Provisionning
    - Pair of chip public/private key
    - Certificate (genuine STM32) + UID



The root secure services (RSS) are embedded in a Flash memory area called the secure information block, programmed during ST production.
The RSS enables, for example, the secure firmware installation (SFI) thanks to the RSS extension firmware (RSSe SFI).
This feature allows customers to protect the confidentiality of the firmware to be provisioned into the STM32 device when the production is subcontracted to a third party.
RSS is available on all devices, once TrustZone has been enabled via the TZEN option bit.
It is composed of the RSS boot and the RSS library, provided by ST.
A pair of public and private keys is provisioned in the RSS area to enable the user image authentication, as well as a certificate and a unique ID.
The boot loader is also designed by ST but unlike RSS, it can be used when TrustZone is disabled.

User option bytes

- The user option bytes are loaded:
  - After a Power reset (POR/BOR or exit from Standby/Shutdown)
  - When the OBL_LAUNCH bit is set in the Flash control register (FLASH_CR)

Options	Description
PA15_PUPEN	USB power delivery dead-battery enabled/disabled - TDI pull-up deactivated/activated
nBOOT0	BOOT0 taken from the option bit nBOOT0 or taken from PB8/BOOT0 pin
nSWBOOT0	
NSBOOTADD0/NSBOOTADD1	Non-secure boot memory address 0 and 1
SRAM2_RST	SRAM2 erased/not erased when a system reset occurs
DBANK	Selection between single bank mode with 128 bit data read width and dual bank mode with 64 bits data read width
DB256K	Dual-bank on 256 Kbytes Flash memory devices
SWAP_BANK	Bank 1 and bank 2 address are swapped/not swapped
BOR_LEV[2:0]	Brown-out reset threshold level
nRST_STOP; nRST_STDBY; nRST_SHDW	Reset / No reset generated when entering STOP/STANDBY/SHUTDOWN mode
WWDG_SW; DWG_SW IWDG_STOP; IWDG_STDBY	HW/SW window watchdog / independent watchdog Independent watchdog counter is frozen / not frozen in STOP/STANDBY mode

life.augmented

Option Bytes are used to configure the system-on-chip before starting the Cortex®-M33.
They are automatically loaded after a power reset or on request by setting the OBL_LAUNCH bit in the FLASH_CR register. The later allows a new configuration to be made without resetting the device.
This slide and the next two describe the various fields in the Option Bytes.

User option bytes (Protection and security)

Options	Description
TZEN	Global TrustZone security enable/disable
BOOT_LOCK	When set, the boot is always forced to the base address value programmed in the SECBOOTADD0 option bytes regardless the boot selection option When set, it cannot be cleared
SECBOOTADD0/NSBOOTADD1	Non-secure boot memory address 0 and 1
SECWM1_PSTRT/SECWM1_PEND	Start and End page of first secure area
HDP1_PEND, HDP1EN	End page of first hide protection area HDP first area enable/disable
SECWM2_PSTRT/SECWM2_PEND	Start and End page of second secure area
HDP2_PEND, HDP2EN	End page of second hide protection area HDP second area enable/disable
RDP[7:0]	Readout protection level
WRP1A_PSTRT[6:0] WRP1A_PEND[6:0] WRP1B_PSTRT[6:0] WRP1B_PEND[6:0] WRP2A_PSTRT[6:0] WRP2A_PEND[6:0] WRP2B_PSTRT[6:0] WRP2B_PEND[6:0]	Bank 1 Write protection area A start page Bank 1 Write protection area A end page Bank 1 Write protection area B start page Bank 1 Write protection area B end page Bank 2 Write protection area A start page Bank 2 Write protection area A end page Bank 2 Write protection area B start page Bank 2 Write protection area B end page

life.augmented

When TZEN is set, TrustZone is active.
Bootlock forces the system to boot from the Main Flash memory regardless of the other boot options.
The setting of secure areas and secure hide protection areas is done through option bytes.
The readout protection level enables the readout protection for the entire Flash memory:

- Level 0: no protection
- Level 0.5: non-secure debug only
- Level 1: read protection
- Level 2: no debug.

Readout protection is fully described in the presentation related to memory protections.
The Write protection start page and end page offsets are also programmed in option bytes.

Interrupts

Interrupt vector	Interrupt event	Description
FLASH_S	Secure end of operation	Set by hardware when one or more Flash memory secure operations (programm / erase) has been completed successfully
	Secure operation error	Set by hardware when a Flash memory secure operation (program / erase) completes unsuccessfully
	Secure read error	
FLASH	Non-secure End of operation	Set by hardware when one or more Flash memory non-secure operations (programm / erase) has been completed successfully
	Non-secure operation error	Set by hardware when a Flash memory non-secure operation (program / erase) completes unsuccessfully
	ECC correction	Set by hardware when one ECC error has been detected and corrected
NMI	ECC non-correctable error	Set by hardware when two ECC errors have been detected

The Flash memory controller supports many interrupt sources, listed in this slide.
Two maskable interrupt request signals are used to report a flash event to the NVIC: FLASH_S for secure events and FLASH for non-secure events.
An interrupt can be asserted upon successful end of operation.
An interrupt can be asserted when an error occurs during a program / erase operation. The next slide details the various operation error causes.
A single-bit error correction is also a non-secure interrupt source.
When two bit errors are detected on a flash memory read, the Non Maskable Interrupt is asserted.

Operation error cause	Description
Write protection error	Set by hardware when an address to be erased/programmed belongs to a write-protected part (by WRP, or RDP) of the Flash memory.
Size error	Set by hardware when the size of the access is a byte or half-word during a program or a fast program sequence.
Programming error	Set by hardware when a double-word address to be programmed contains a value different from 0xFFFF_FFFF before programming except if the data to write is 0x0000_0000.
Programming sequence error	Set by hardware when a write access to the Flash memory is performed by the code while PG has not been set previously. Set also by hardware when PROGERR, SIZERR, PGAERR, WRPERR, PGSER or OPTWERR is set due to a previous programming error.
Programming alignment error	Set by hardware when the data to program cannot be contained in the same double word (64-bit) Flash memory in case of standard programming or if there is a change of page during fast programming.
Option write error	Set by hardware when the option bytes are written with an invalid configuration.

This table indicates the sources of operation errors. Two status registers, used by software to identify the cause of the operation errors, are implemented: secure and non-secure.
A write protection violation occurs when an attempt to write to a write-protected area is detected.
A Size error occurs when the data to be programmed is not word-aligned.
A Programming sequential error occurs when a program operation is attempted without having previously erased the location in Flash memory.
A programming alignment error occurs when a complete double word is not provided before initiating a standard program operation.
An option write error occurs when the option bytes are written with an invalid configuration.

Low-power modes

Consumption optimization when executing from SRAM

- Flash clock can be gated off in Run/Low-power run and/or in Sleep/Low-power sleep modes
  - Flash clock is configured in the Reset and Clock Controller (RCC)
  - Flash clock is enabled by default
- Flash can be configured in Power-down mode during Sleep/Low-power sleep modes
- Flash can be configured in Power-down mode during Run/Low-power run modes

life.augmented

The Flash memory's consumption can be reduced when the code is not executed from Flash.
The Flash clock can be gated off in Run and low-power run modes. It can also be configured to be gated off in Sleep and low-power sleep modes. The Flash clock is configured in the Reset and Clock controller. It is enabled by default.
The Flash memory can be configured in Power-down mode during the Sleep and low-power sleep modes.
It can also be configured in power-down mode during Run and low-power run modes, when the code is executed from SRAM.
Gating the clock and putting the Flash memory in Power-down mode significantly reduces power consumption.

Low-power modes

Mode	Description
Run	Active Flash memory clock can be disabled if code is executed from SRAM and the Flash memory can be put in Power-down mode
Sleep	Active Flash memory clock can be disabled if code is executed from SRAM and the Flash memory can be put in Power-down mode
Low-power run	Active Flash memory clock can be disabled if code is executed from SRAM and the Flash memory is in Power-down mode
Low-power sleep	Active Flash memory clock can be disabled if code is executed from SRAM and the Flash memory can be put in Power-down mode
Stop 0/ Stop 1/ Stop 2	Flash memory clock off Contents of peripheral registers are kept Flash memory can be put in Power-down mode
Standby	Powered-down The Flash memory interface must be reinitalized after exiting Standby mode
Shutdown	Powered-down The Flash memory interface must be reinitalized after exiting Shutdown mode

life.augmented

The Flash memory module supports the following low power capabilities:

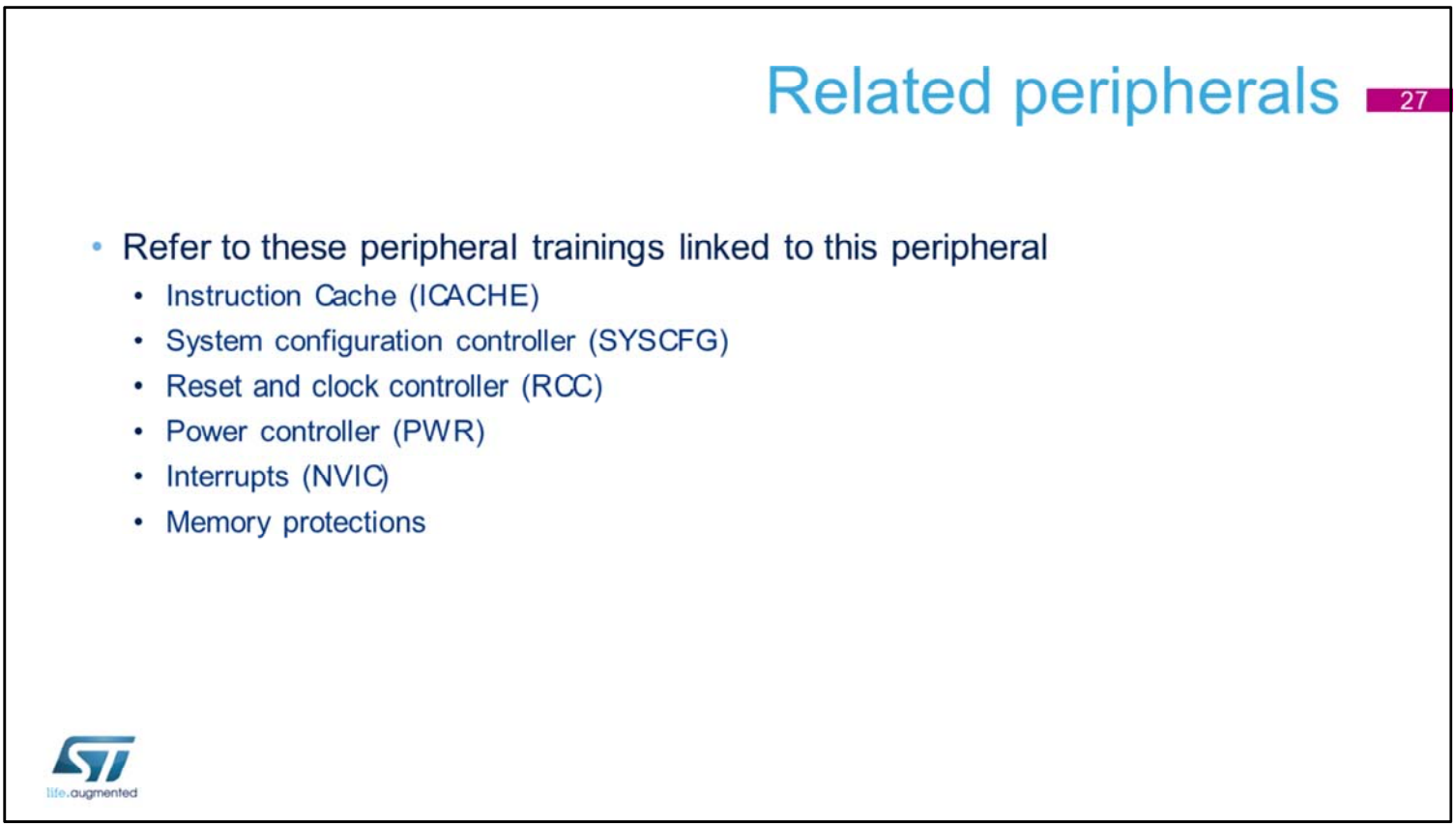
- Clock gating
- Flash memory power-down mode
- Power gating of the entire module: Flash memory and controller.

In Run, Sleep, Low Power Run and Low Power Sleep modes, clock gating and power-down is supported. It can be used when code is executed from SRAM.

In Stop0, Stop1 and Stop2, the clocks are gated and Flash memory can enter Power-down mode.

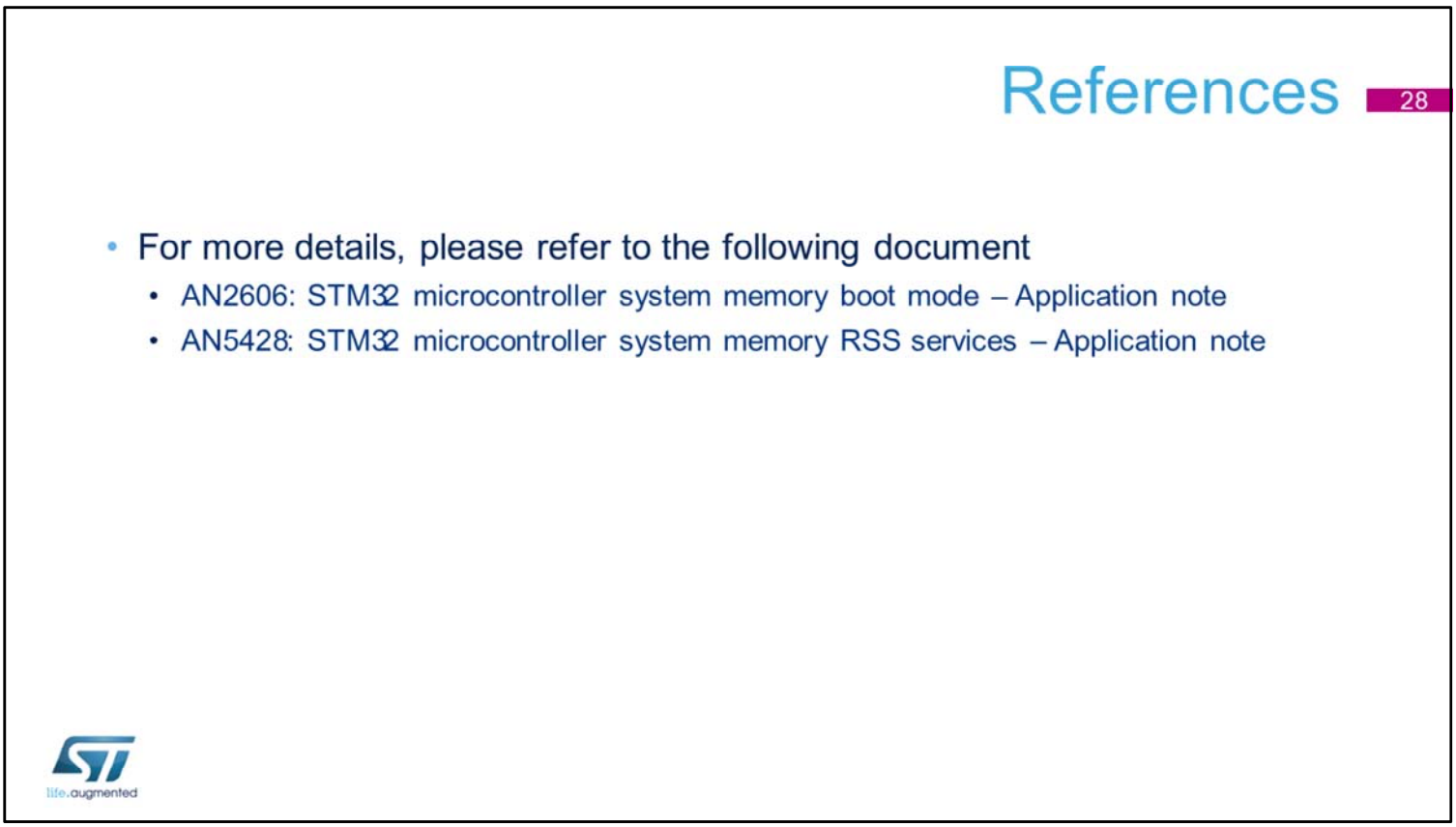
In Shutdown mode, the power of the Flash memory module is gated, for both the Flash memory and controller.

Gating the clock and putting the Flash memory in Power-down mode significantly reduces power consumption.



The Flash memory module has relationships with the following other modules:

- Instruction Cache (ICACHE)
- System configuration controller (SYSCFG)
- Reset and clock controller (RCC)
- Power controller (PWR)
- Interrupts (NVIC)
- Memory protections.



For more details, please refer to application note:

- AN2606 about the STM32 microcontroller system memory boot mode.
- AN5428 about the STM32 microcontroller system memory RSS services