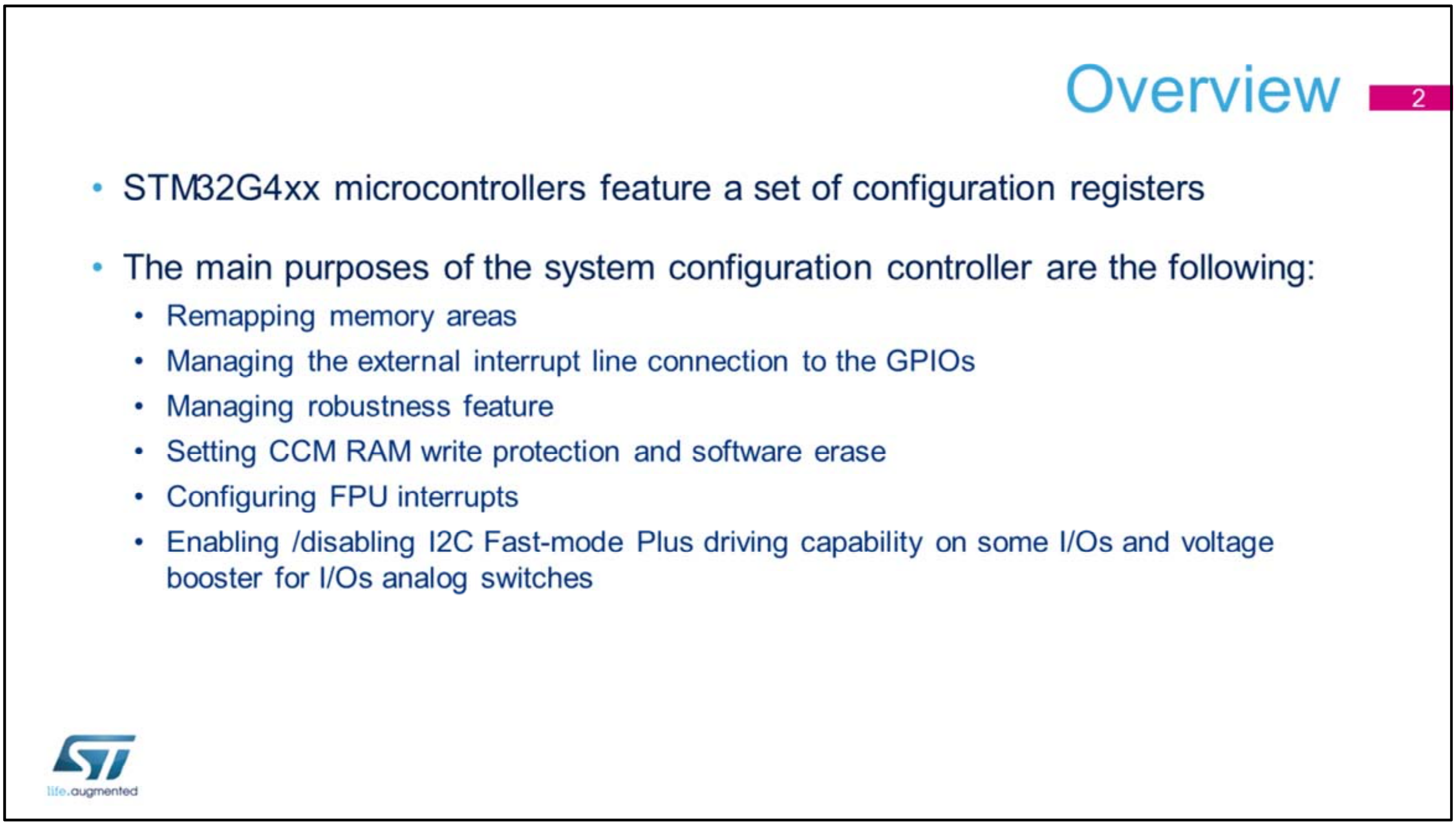
STM32G4 – SYSCFG

System Configuration Controller

Revision 1.0

Hello, and welcome to this presentation of the STM32G4 System Configuration Controller.



Overview

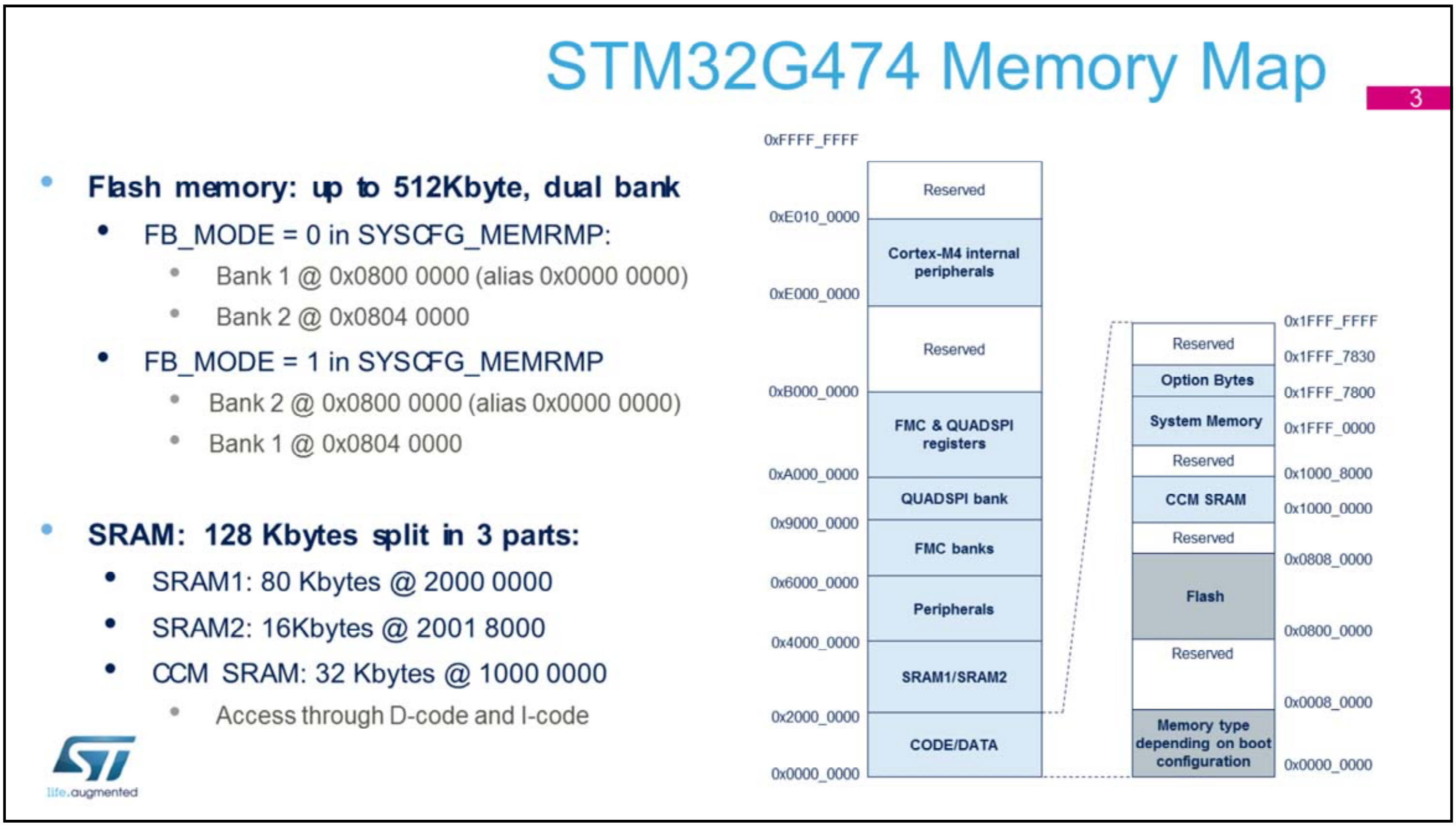
2

- STM32G4xx microcontrollers feature a set of configuration registers
- The main purposes of the system configuration controller are the following:
  - Remapping memory areas
  - Managing the external interrupt line connection to the GPIOs
  - Managing robustness feature
  - Setting CCM RAM write protection and software erase
  - Configuring FPU interrupts
  - Enabling /disabling I2C Fast-mode Plus driving capability on some I/Os and voltage booster for I/Os analog switches

STM32G4 microcontrollers feature a set of configuration registers located in the SYSCFG module.

The System Configuration Controller gives access to the following features:

- Remapping memory areas and selecting the memory accessible at address 0x0000_0000
- Managing the external interrupt line connection to the GPIOs
- Managing robustness feature
- Setting CCM RAM write protection and software erase
- Configuring FPU interrupts
- Enabling/disabling I2C Fast-mode Plus driving capability on some I/Os and voltage booster for I/Os analog switches.



STM32G474 Memory Map

- **Flash memory: up to 512Kbyte, dual bank**
  - FB_MODE = 0 in SYSCFG_MEMRMP:
    - Bank 1 @ 0x0800 0000 (alias 0x0000 0000)
    - Bank 2 @ 0x0804 0000
  - FB_MODE = 1 in SYSCFG_MEMRMP
    - Bank 2 @ 0x0800 0000 (alias 0x0000 0000)
    - Bank 1 @ 0x0804 0000

- **SRAM: 128 Kbytes split in 3 parts:**
  - SRAM1: 80 Kbytes @ 2000 0000
  - SRAM2: 16Kbytes @ 2001 8000
  - CCM SRAM: 32 Kbytes @ 1000 0000
    - Access through D-code and I-code

This slide describes the memory map of the STM32G474 microcontroller.

The differences between the STM32G474 and STM32G431 variants are highlighted, later in this presentation.

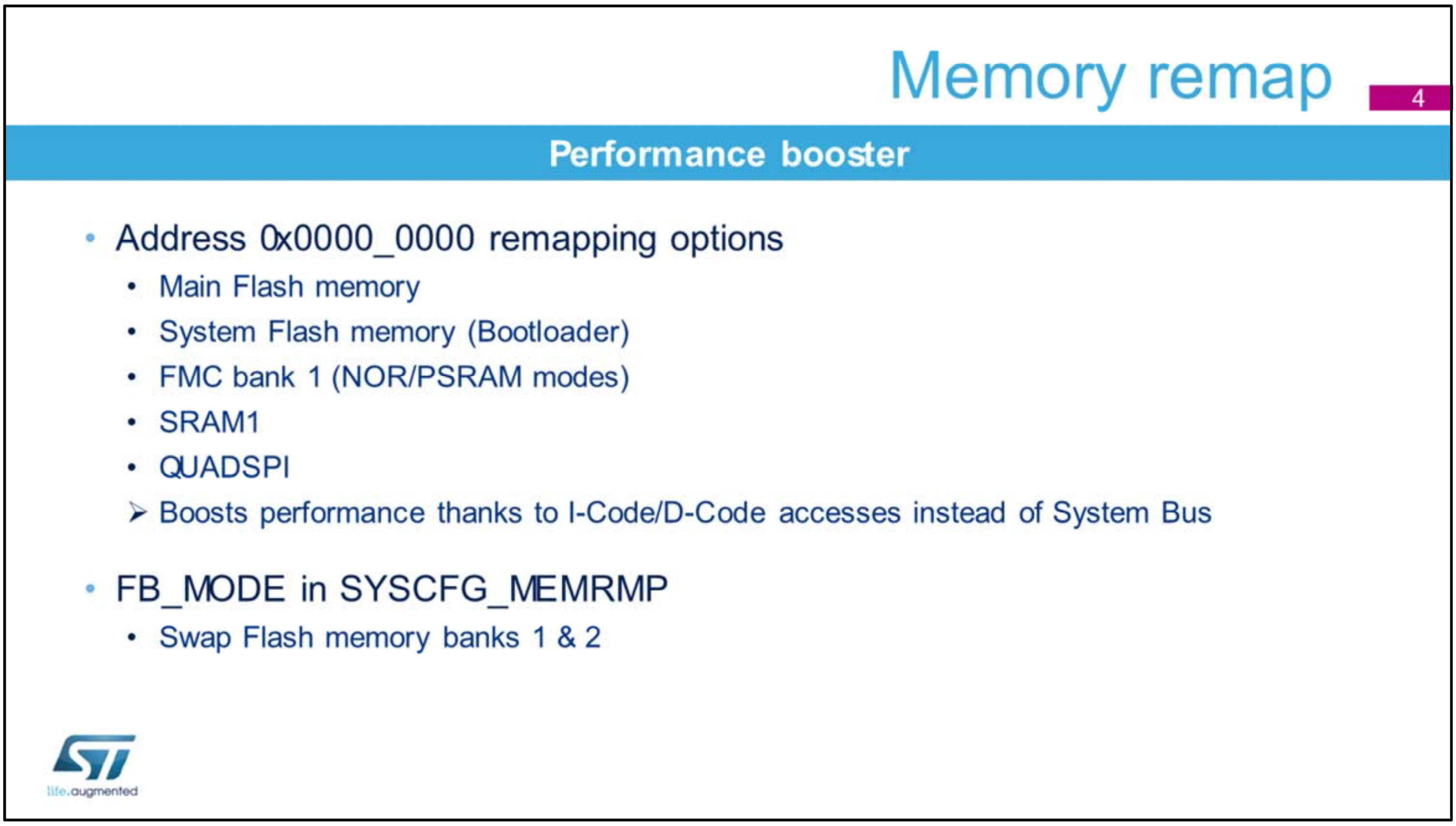
The Flash memory size is up to 512 Kbytes, in a dual-bank configuration.

The FB_MODE bit determines the address mapping of Banks 1 and 2 and also selects which bank is aliased to address 0.

The SRAM total size is 128 Kbytes. It is split into 3 parts:

- The SRAM1 size is 80 Kbytes starting from address 0x2000_0000
- The SRAM2 size is 16 Kbytes starting from address 0x2001_8000
- The Core-Coupled Memory (or CCM) SRAM size is 32Kbytes starting from address 0x1000_0000.

SRAM1 and SRAM2 memories are located in the usual Arm V7-M memory space dedicated to SRAM while the CCM SRAM is accessed through Dcode and Icode AHB buses.
This architecture enables concurrent accesses to CCM SRAM and SRAM1 or SRAM2 memories.



Memory remap

4

Performance booster

- Address 0x0000_0000 remapping options
  - Main Flash memory
  - System Flash memory (Bootloader)
  - FMC bank 1 (NOR/PSRAM modes)
  - SRAM1
  - QUADSPI
  - ➢ Boosts performance thanks to I-Code/D-Code accesses instead of System Bus
- FB_MODE in SYSCFG_MEMRMP
  - Swap Flash memory banks 1 & 2

The memory remap at address 0 boosts up the code execution performance thanks to the dedicated ICODE and DCODE bus accesses, instead of using the System bus.

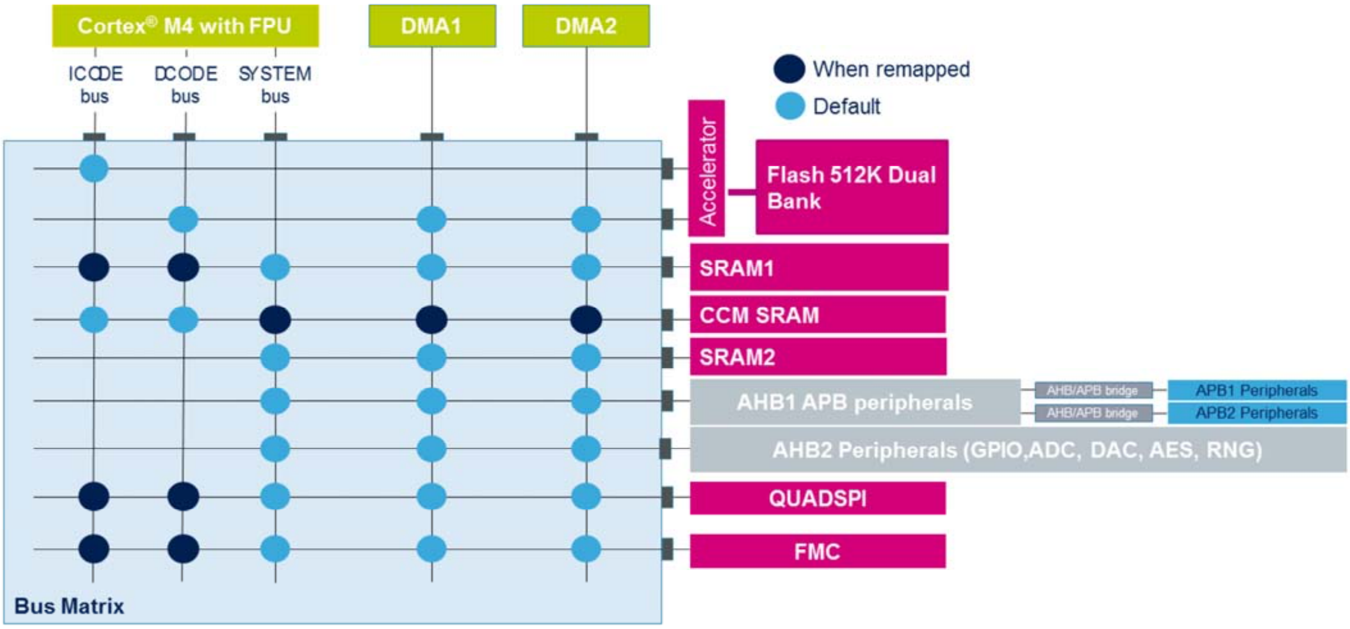
The memory remap at address 0 selects the memory accessible at address 0.

It could be either the main Flash memory, or the system Flash memory, or the FMC bank 1, or the SRAM1, or the QUADSPI.

The FMC bank1 maps external NOR flash or PSRAM memory.

The FB_MODE bit in the System Configuration Remap register allows the swap in between Flash memory banks 1 and 2.

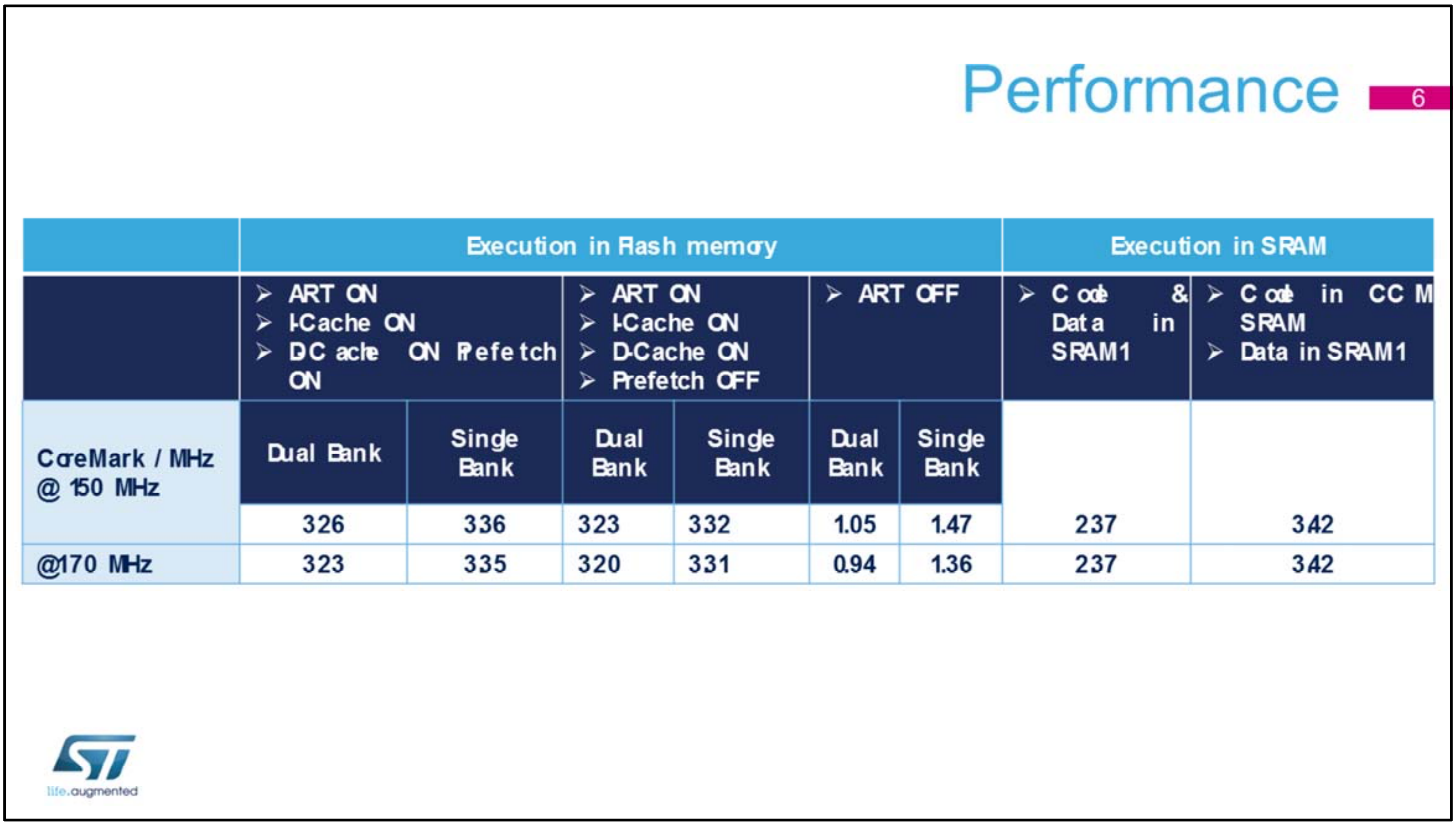
STM32G474x Bus matrix



	ICODE bus	DCODE bus	SYSTEM bus	DMA1	DMA2
Flash 512K Dual Bank (Accelerator)	○				
Flash 512K Dual Bank (Accelerator)		○		○	○
SRAM1	●	●	○	○	○
CCM SRAM	○	○	●	●	●
SRAM2			○	○	○
AHB1 APB peripherals			○	○	○
AHB2 Peripherals (GPIO,ADC, DAC, AES, RNG)			○	○	○
QUADSPI	●	●	○	○	○
FMC	●	●	○	○	○

This figure represents the STM32G4 bus matrix.
The bus masters are at the top of the figure: the Cortex-M4 core with its three AHB master interfaces ICODE, DCODE and SYSTEM and the two DMA controllers.
The bus slaves are at the right of the figure: internal and external memories and peripherals.
The Flash memory is read through the accelerator.
When the Cortex-M4 core accesses data within the CODE or DATA address range, the DCODE bus is used.
When the Cortex-M4 core accesses instructions within the CODE or DATA address range, the ICODE bus is used.
The SRAM1 is accessed by default through the System bus, and can be accessed through the ICODE bus and DCODE bus when it is remapped at address 0, in order to increase performance.
The CCM SRAM memory is always accessed through

the I-bus and D-bus allowing zero-wait-state code execution.
The Quad-SPI and FMC banks can be read and executed through the System bus by default, and can be remapped at 0 to increase performance.
The two DMA controllers can access all memories and peripherals.



Performance

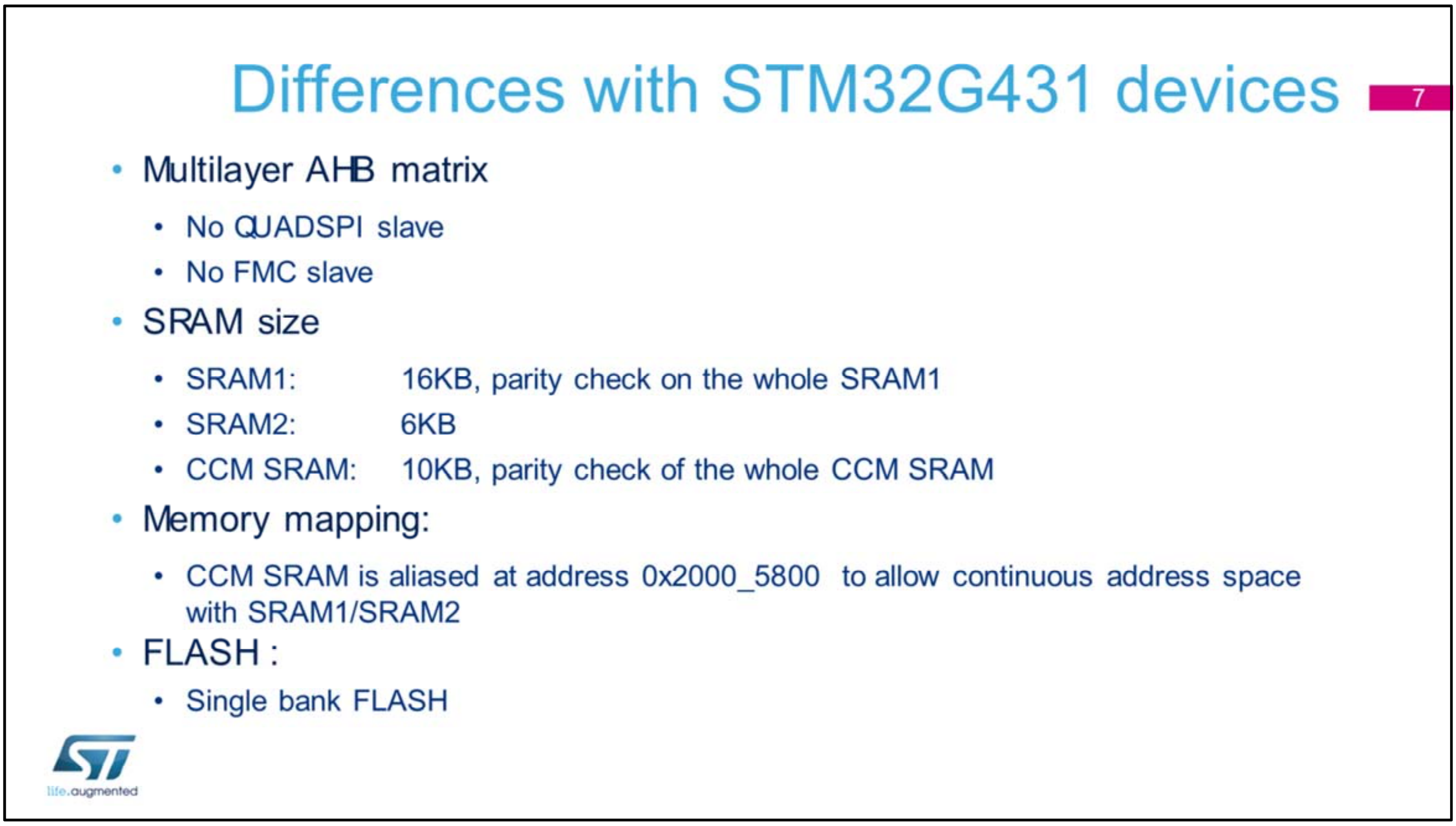
	Execution in Flash memory						Execution in SRAM	
	➢ ART ON ➢ I-Cache ON ➢ D-Cache ON Prefetch ON		➢ ART ON ➢ I-Cache ON ➢ D-Cache ON ➢ Prefetch OFF		➢ ART OFF		➢ Code & Data in SRAM1	➢ Code in CCM SRAM ➢ Data in SRAM1
CoreMark / MHz @ 150 MHz	Dual Bank	Single Bank	Dual Bank	Single Bank	Dual Bank	Single Bank		
	326	336	323	332	1.05	1.47	237	342
@170 MHz	323	335	320	331	0.94	1.36	237	342

This table compares the code execution performance at 150 MHz and 170 MHz while running the EEMBC CoreMark benchmark.

The maximum performance is reached when the code is executed in CCM SRAM with data is located in SRAM1. It is also possible to reach maximum performance with code in SRAM1 and data in SRAM2 if the SRAM1 is remapped at address 0.

When executing from Flash memory the maximum CoreMark performance is reached when the ART accelerator is enabled, and there is almost no loss of performance due to the Flash access time requiring 7 wait states at 150 MHz or 8 wait states at 170 MHz .

Enabling the prefetch buffer yields a slightly higher score: 3.36 CoreMark / MHz in case of Single bank mode.



Differences with STM32G431 devices

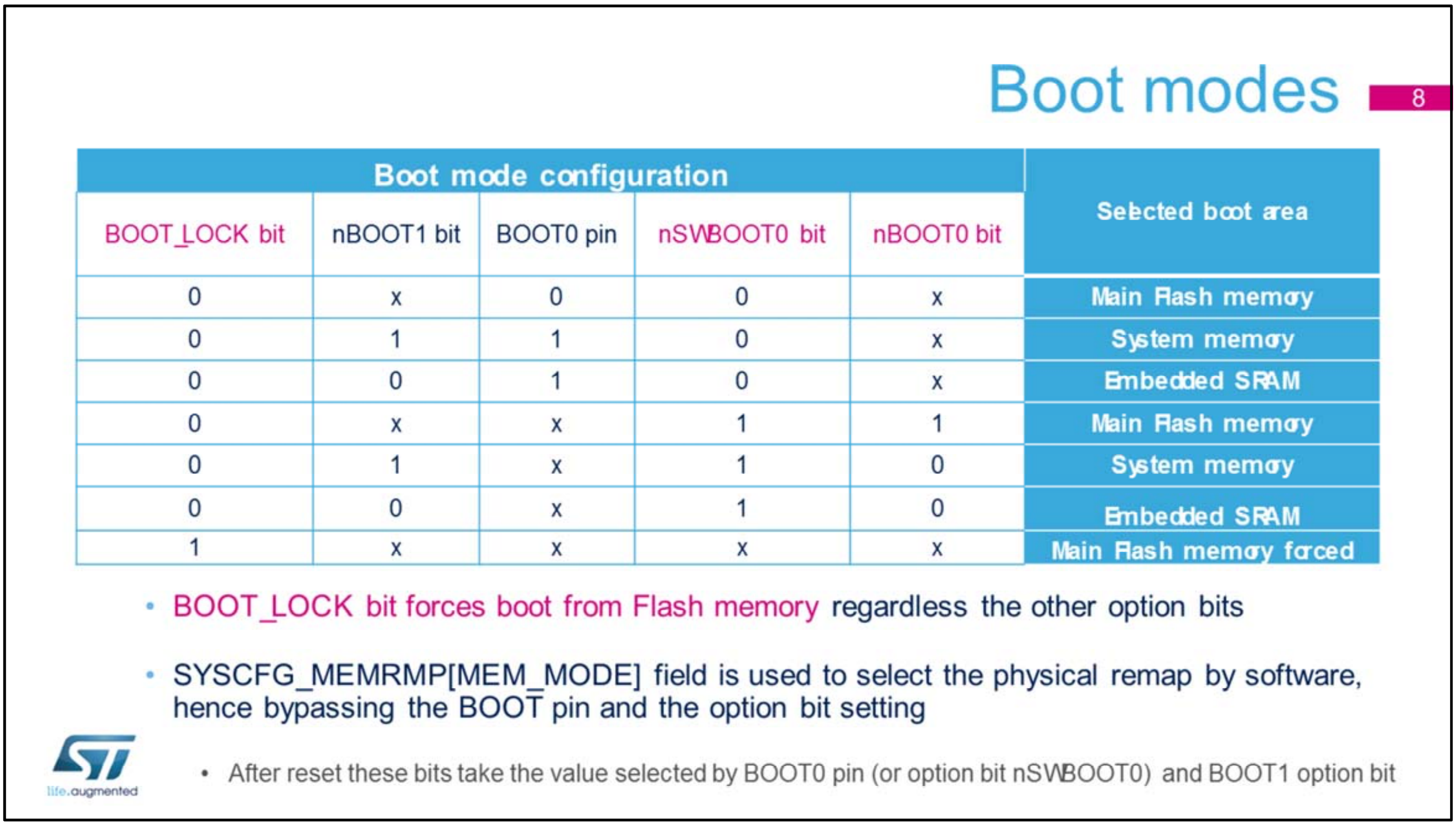
- Multilayer AHB matrix
  - No QUADSPI slave
  - No FMC slave
- SRAM size
  - SRAM1: 16KB, parity check on the whole SRAM1
  - SRAM2: 6KB
  - CCM SRAM: 10KB, parity check of the whole CCM SRAM
- Memory mapping:
  - CCM SRAM is aliased at address 0x2000_5800 to allow continuous address space with SRAM1/SRAM2
- FLASH :
  - Single bank FLASH

This slide presents the key differences between the STM32G431 and STM32G474 microcontrollers.
The STM32G431 line includes neither QuadSPI nor FMC units.

The STM32G431 line has smaller SRAM memories: a 16-Kilobyte SRAM1 and a 10-kilobyte CCM SRAM both supporting parity + a 6-kilobyte SRAM2.

Regarding the mapping, the CCM SRAM is aliased at address 0x2000_5800 to allow continuous RAM address range with SRAM1 and SRAM2 memories.

At last, the STM32G431 has a unique bank of flash memory.



Boot modes

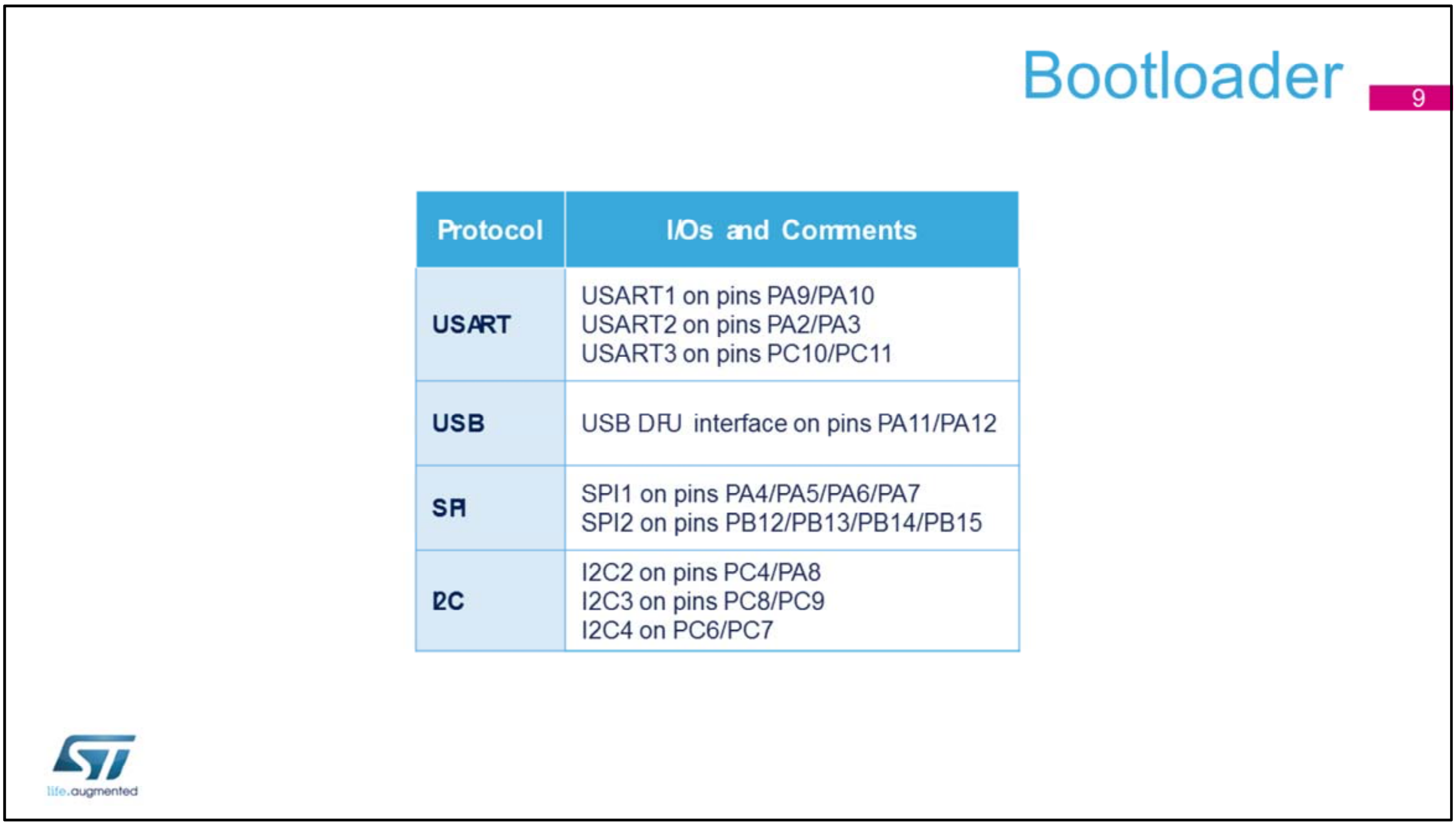
8

Boot mode configuration					Selected boot area
BOOT_LOCK bit	nBOOT1 bit	BOOT0 pin	nSWBOOT0 bit	nBOOT0 bit	
0	x	0	0	x	Main Flash memory
0	1	1	0	x	System memory
0	0	1	0	x	Embedded SRAM
0	x	x	1	1	Main Flash memory
0	1	x	1	0	System memory
0	0	x	1	0	Embedded SRAM
1	x	x	x	x	Main Flash memory forced

- BOOT_LOCK bit forces boot from Flash memory regardless the other option bits
- SYSCFG_MEMRMP[MEM_MODE] field is used to select the physical remap by software, hence bypassing the BOOT pin and the option bit setting
  - After reset these bits take the value selected by BOOT0 pin (or option bit nSWBOOT0) and BOOT1 option bit

There are 3 boot modes which are selected by the BOOT0 pin or the nBOOT0 bit (if the nSWBOOT0 bit is cleared) and by nBOOT1 bit.
When the BOOT0 pin is at a low level, the STM32G4 microcontroller boots from the User Flash memory.
When the BOOT0 pin is at a high level, the nBOOT1 bit determines the boot mode.
When it is high, the boot is done from the system memory, that contains the ST proprietary boot code.
The other option is booting from the SRAM1 memory region.
When the boot lock option bit is high, the boot is forced from the Main Flash memory.
Software can dynamically select which memory is visible at address 0 by programming the MEM_MODE field in the SYSCFG_MEMRP register.
The default value of this field depends on the boot pins

state and related option bytes value, BOOT_LOCK and nSWBOOT0.

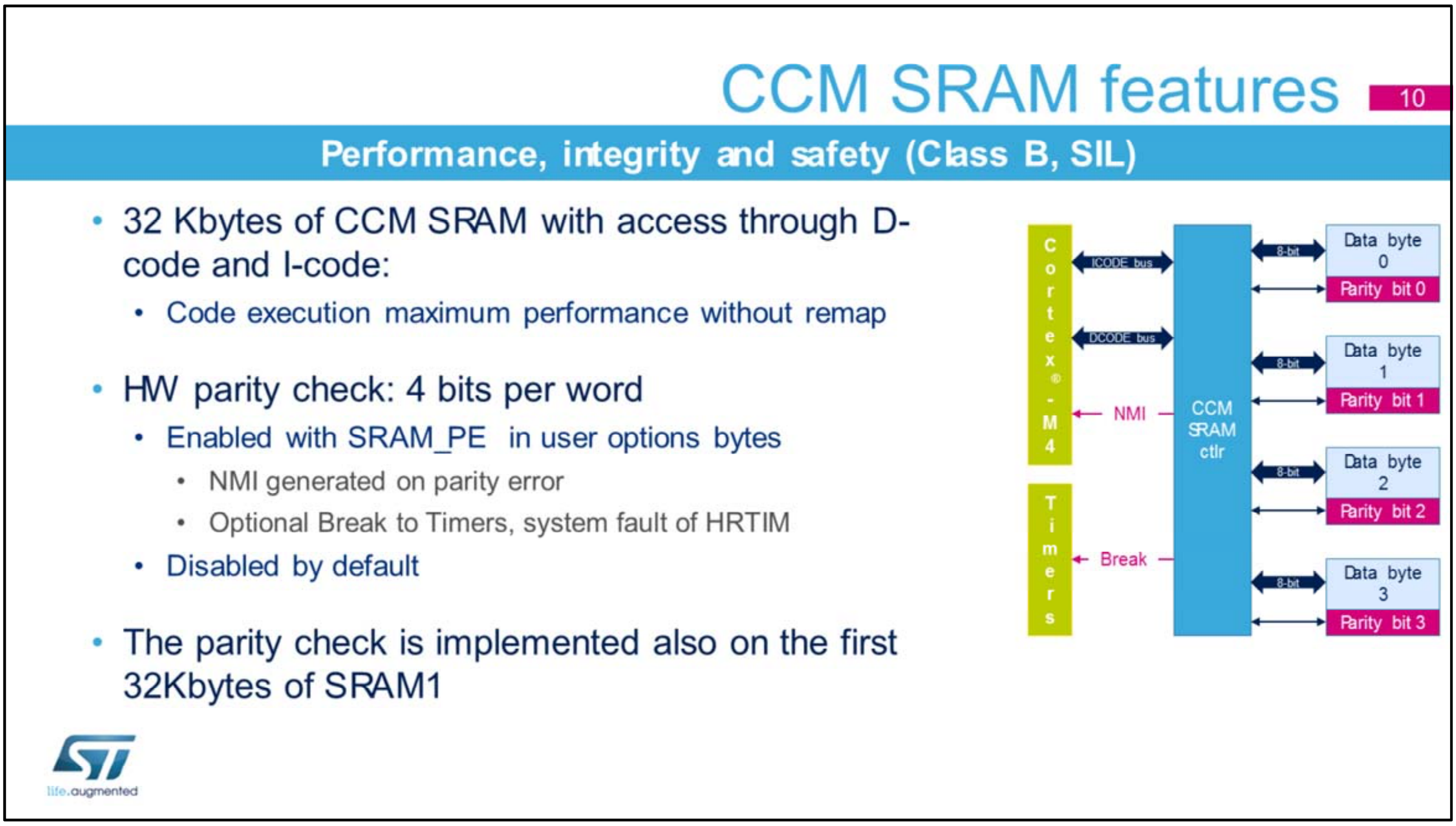


Bootloader

9

Protocol	I/Os and Comments
USART	USART1 on pins PA9/PA10 USART2 on pins PA2/PA3 USART3 on pins PC10/PC11
USB	USB DFU interface on pins PA11/PA12
SPI	SPI1 on pins PA4/PA5/PA6/PA7 SPI2 on pins PB12/PB13/PB14/PB15
I2C	I2C2 on pins PC4/PA8 I2C3 on pins PC8/PC9 I2C4 on PC6/PC7

The on-chip boot-loader allows the user to program the Flash memory with an image downloaded to the STM32G4 through a serial communication peripheral. The supported protocols are USART, USB, CAN, SPI and I²C.



CCM SRAM features

Performance, integrity and safety (Class B, SIL)

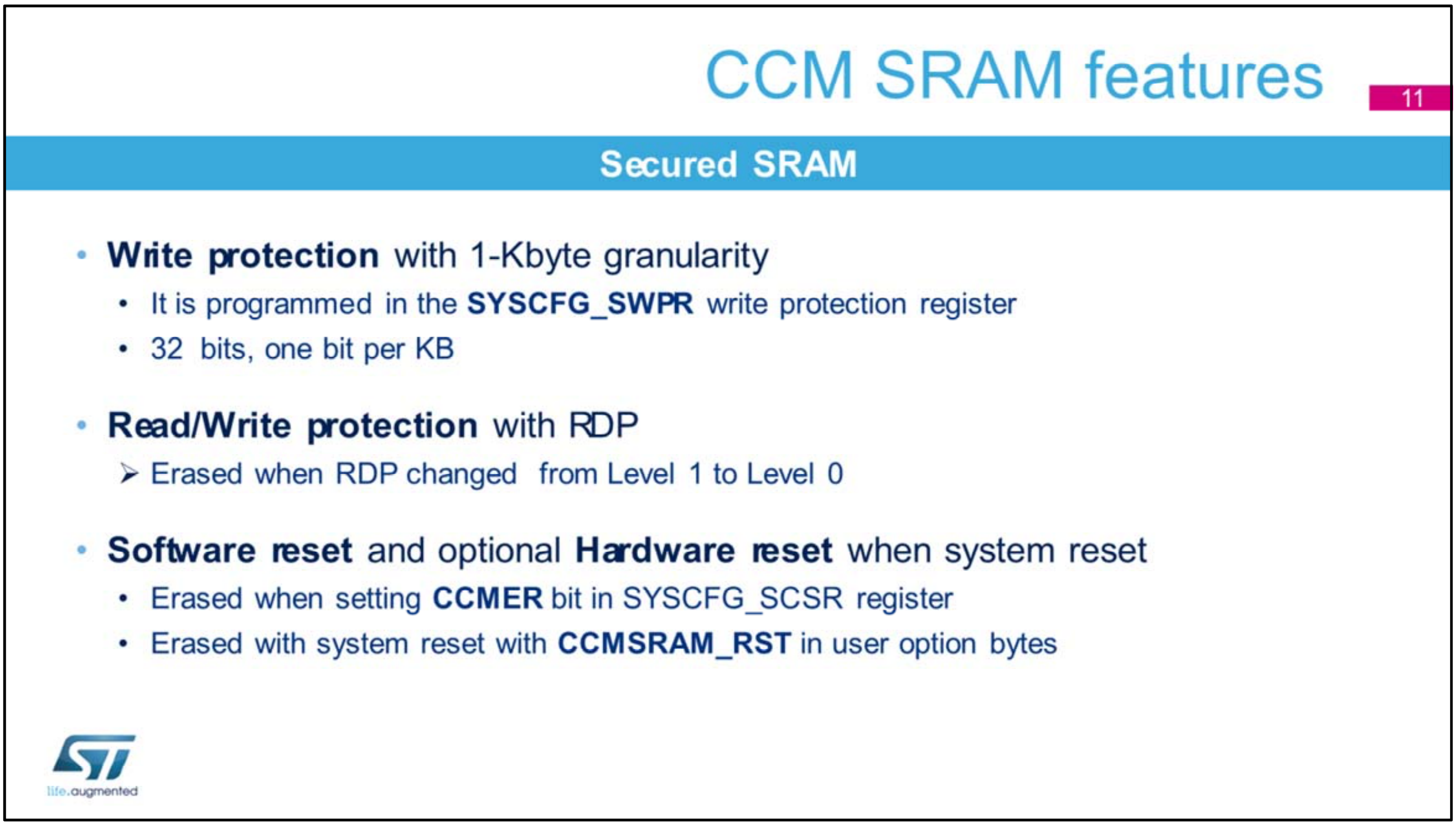
- 32 Kbytes of CCM SRAM with access through D-code and I-code:
  - Code execution maximum performance without remap
- HW parity check: 4 bits per word
  - Enabled with SRAM_PE in user options bytes
    - NMI generated on parity error
    - Optional Break to Timers, system fault of HRTIM
  - Disabled by default
- The parity check is implemented also on the first 32Kbytes of SRAM1

Cortex®-M4
ICODE bus
DCODE bus
NMI
Timers
Break
CCM SRAM ctlr
8-bit
Data byte 0
Parity bit 0
8-bit
Data byte 1
Parity bit 1
8-bit
Data byte 2
Parity bit 2
8-bit
Data byte 3
Parity bit 3

life.augmented

The 32 Kbytes of CCM SRAM is particularly suitable for performance, integrity and safety.
The CCM SRAM is accessed through the DCODE and ICODE buses without any remapping, which enables code execution at zero-wait-states.
The CCM SRAM supports parity check. The Data bus width is 36 bits because 4 bits are available for parity check (1 bit per byte) in order to increase the memory robustness, as required, for instance, by Class B or SIL standards. Class B and SIL are safety standards: Class B is for Home Appliances and SIL for the Safety Integrity Level.
The parity bits are computed and stored when writing into the SRAM. Then, they are automatically checked when reading. If at least one bit fails, a Non-Maskable Interrupt (NMI) is generated. The same error can also be linked to the Break input of the timers and the HRTIM

system fault input.
Note that the parity check is disabled by default.
The lower 32 Kbytes of the SRAM1 also support parity generation and checking.



CCM SRAM features

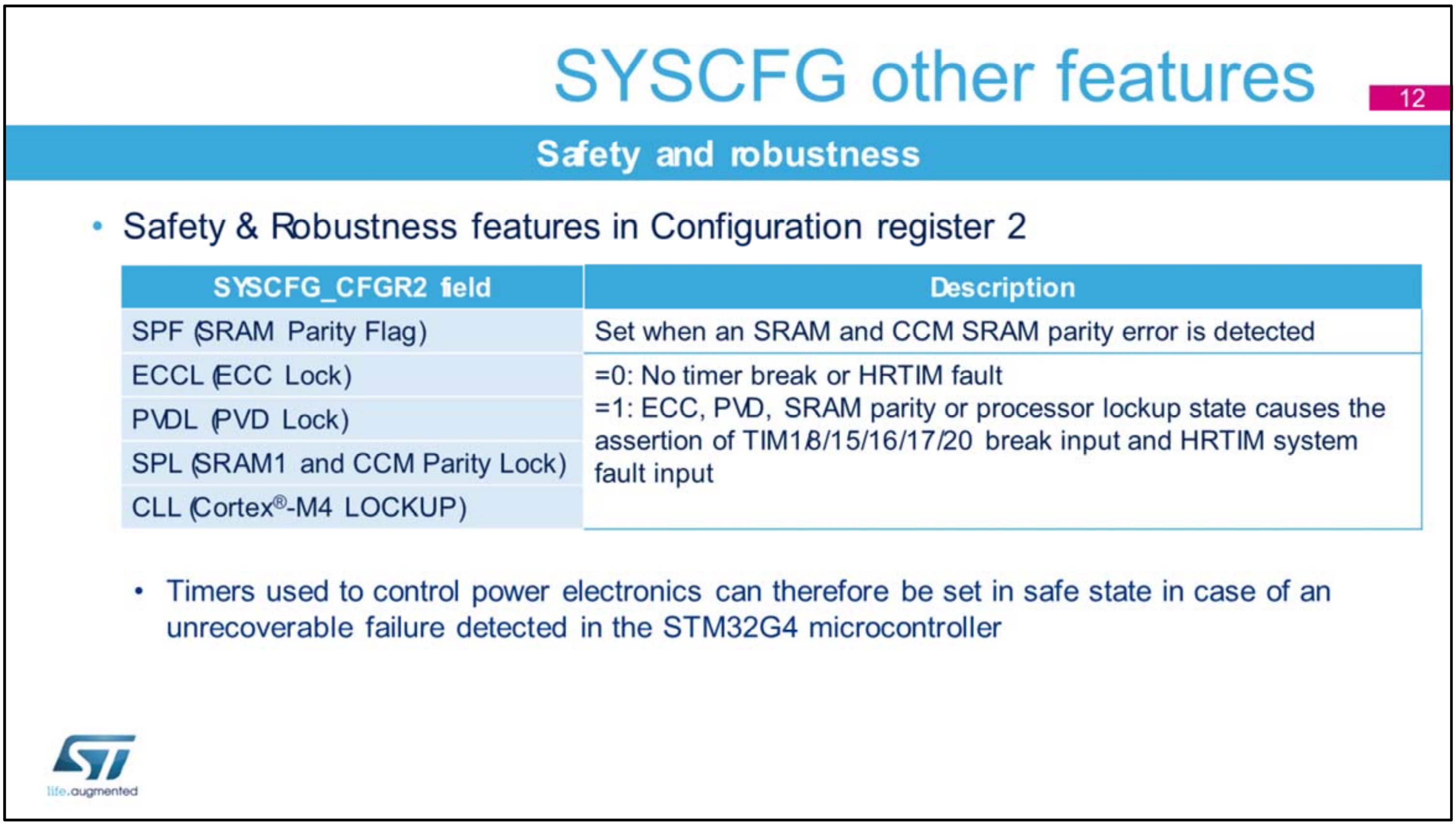
11

Secured SRAM

- **Write protection** with 1-Kbyte granularity
  - It is programmed in the **SYSCFG_SWPR** write protection register
  - 32 bits, one bit per KB
- **Read/Write protection** with RDP
  - ➢ Erased when RDP changed from Level 1 to Level 0
- **Software reset** and optional **Hardware reset** when system reset
  - Erased when setting **CCMER** bit in SYSCFG_SCSR register
  - Erased with system reset with **CCMSRAM_RST** in user option bytes

life.augmented

The CCM SRAM is also suitable for secure applications. It can be write-protected with a 1-Kbyte granularity. It can also be readout-protected via the RDP option byte. When protected, the CCM SRAM as well as the Flash main memory and the backup registers are totally inaccessible in debug mode or when code is running from boot RAM or boot loader. The CCM SRAM is erased when the readout protection is changed from Level 1 to Level 0. The CCM SRAM can be erased by software by setting the CCMER bit in the CCM SRAM System Configuration Control and Status register. The CCM SRAM can also be erased with the system reset depending on the CCMSRAM_RST option bit in the user option bytes.



SYSCFG other features

Safety and robustness

- Safety & Robustness features in Configuration register 2

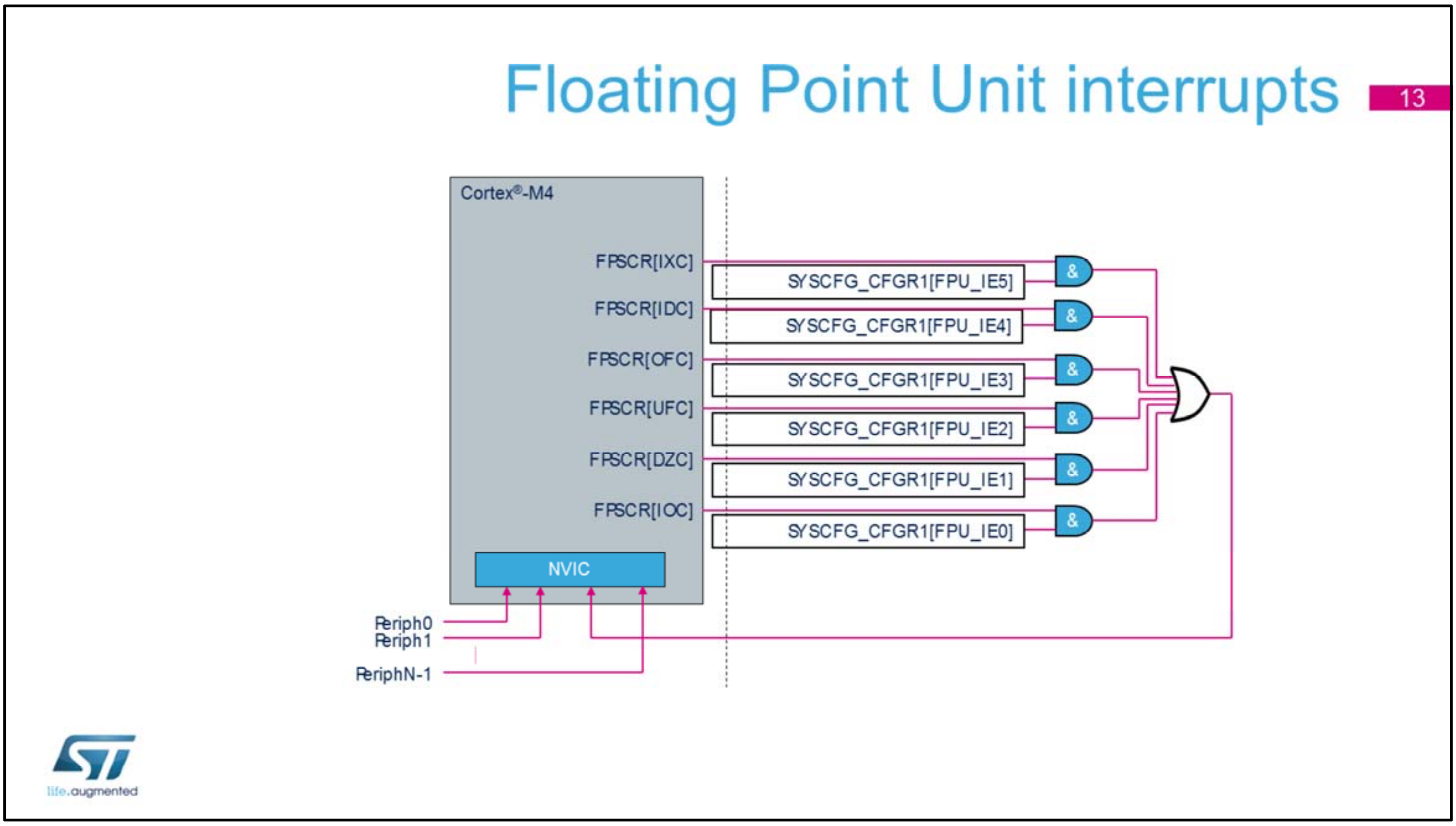
SYSCFG_CFGR2 field	Description
SPF (SRAM Parity Flag)	Set when an SRAM and CCM SRAM parity error is detected
ECCL (ECC Lock)	=0: No timer break or HRTIM fault =1: ECC, PVD, SRAM parity or processor lockup state causes the assertion of TIM1/8/15/16/17/20 break input and HRTIM system fault input
PVDL (PVD Lock)	
SPL (SRAM1 and CCM Parity Lock)	
CLL (Cortex®-M4 LOCKUP)	

- Timers used to control power electronics can therefore be set in safe state in case of an unrecoverable failure detected in the STM32G4 microcontroller

life.augmented

The System Configuration Register 2 contains the control and status bits related to safety and robustness such as the CCM SRAM parity error flag, and the control bits to direct some error detections events to the timers' break inputs.

This allows timer outputs to be placed in a known state during an application crash. Once programmed, the connection is locked until the next system reset. These internal events include a Flash error-code-correction event, a power voltage detector event, SRAM1 and CCM RAM parity error event, and the Cortex M4 hard fault.



Floating Point Unit interrupts

13

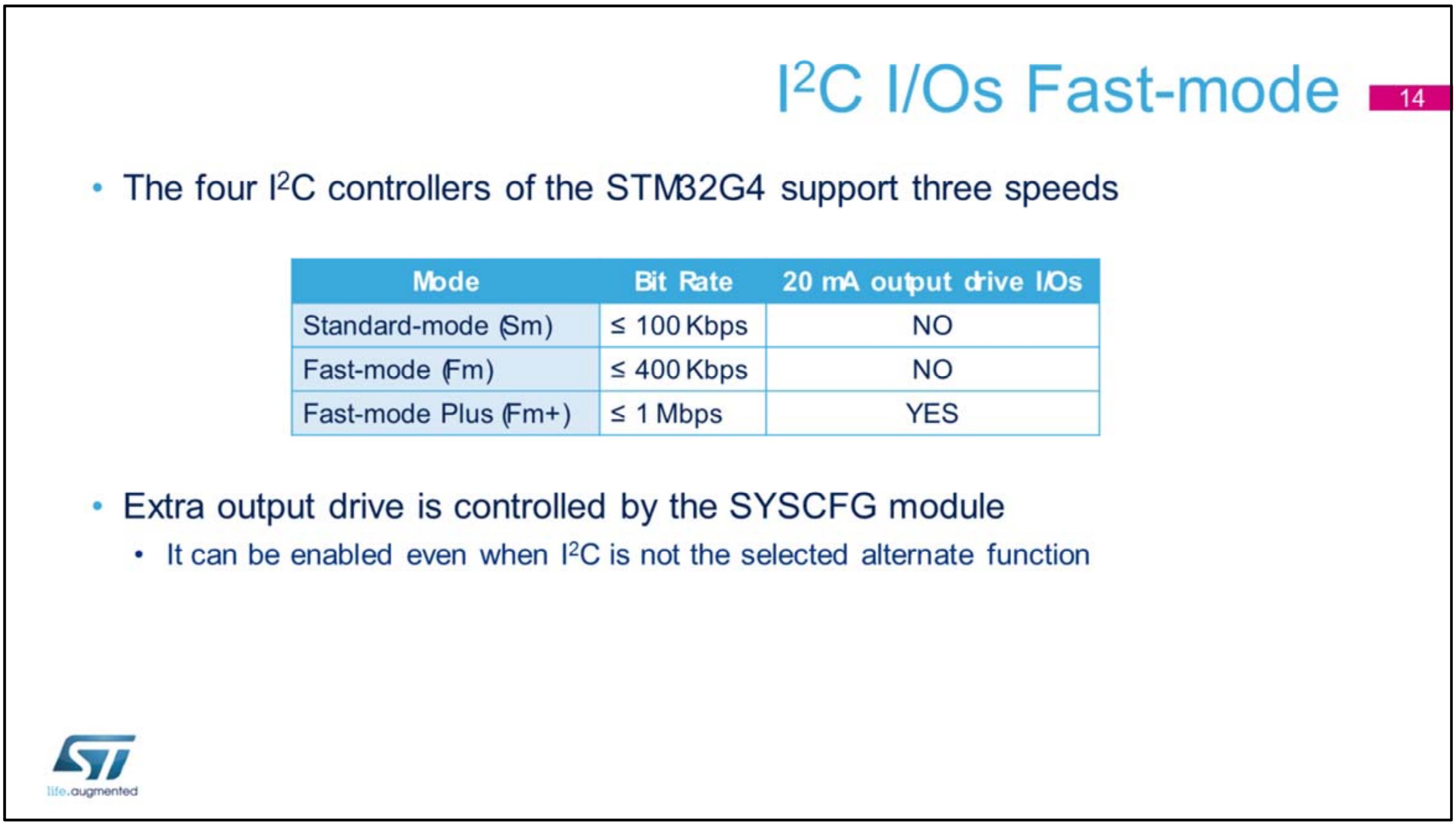
The Floating Point Unit (FPU) present in the Cortex-M4 core sets the cumulative exception status flag in the FPSCR register as required for each instruction, in accordance with the FPv4 architecture.

The FPU does not support user-mode traps. The exception enable bits in the FPSCR read-as-zero, and writes are ignored.

The processor also has six output pins: IXC, FC, OFC, DZC, IDC, and IOC, that each reflects the status of one of the cumulative exception flags.

When the corresponding enable bit in the SYSCFG_CFGR1 register is set, an interrupt is requested when the flag is set.

The interrupt service routine is in charge of determining which flag or flags have been set.



I^{2}C I/Os Fast-mode

14

- The four I^{2}C controllers of the STM32G4 support three speeds

Mode	Bit Rate	20 mA output drive I/Os
Standard-mode (Sm)	≤ 100 Kbps	NO
Fast-mode (Fm)	≤ 400 Kbps	NO
Fast-mode Plus (Fm+)	≤ 1 Mbps	YES

- Extra output drive is controlled by the SYSCFG module
  - It can be enabled even when I^{2}C is not the selected alternate function

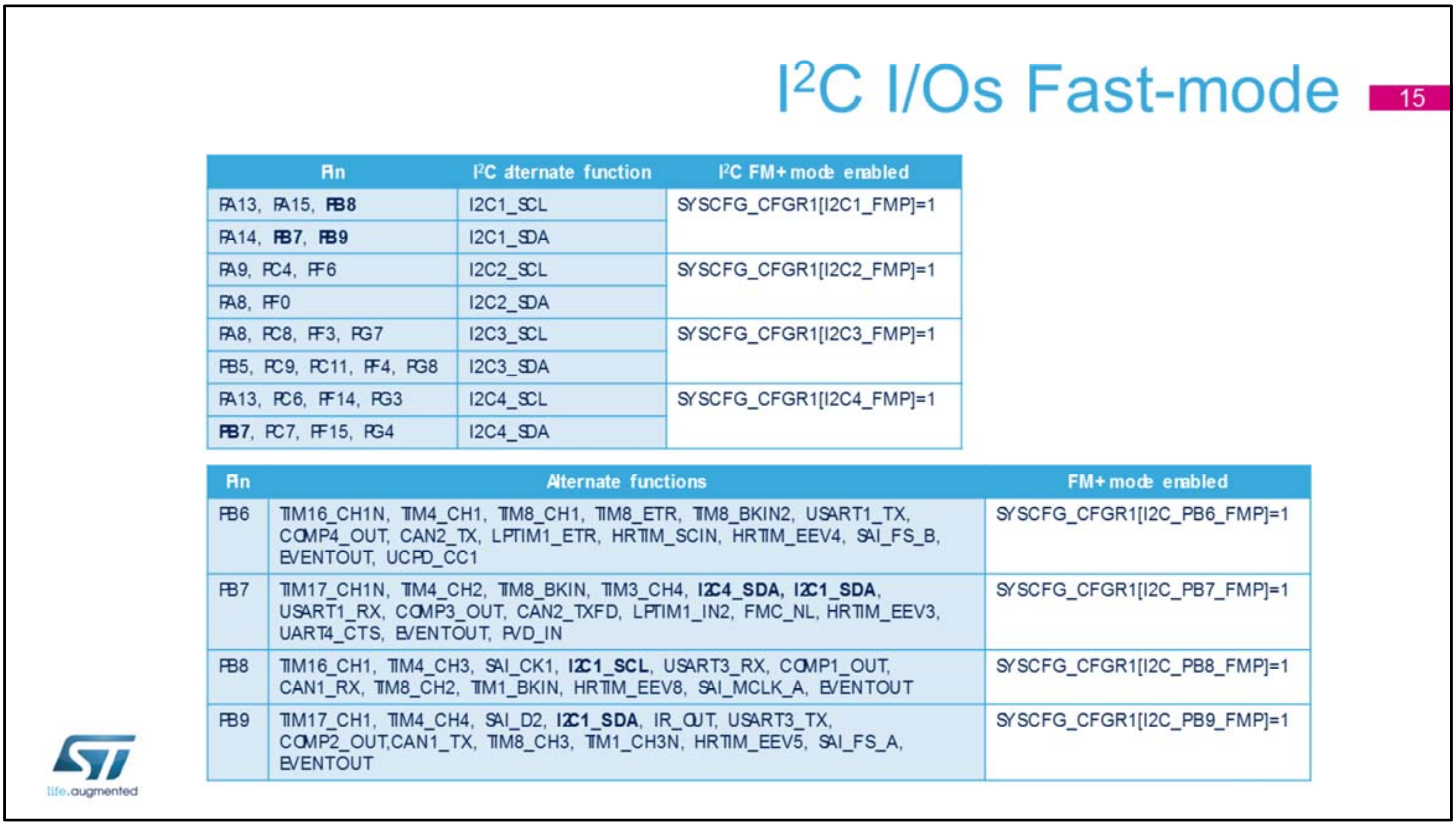
life.augmented

The four I2C controllers embedded in the STM32G4 microcontroller support 3 speeds:

- Standard-mode, the maximum bitrate is 100 Kilobits per second,
- Fast-mode, the maximum bitrate is 400 Kilobits per second,
- Fast-mode Plus, the maximum bitrate is 1 Megabit per second.

Fast-mode Plus requires a high drive capability, which is enabled in the SYSCFG module.

Since high-drive is controlled at pin level, it is also available for the other alternate functions.



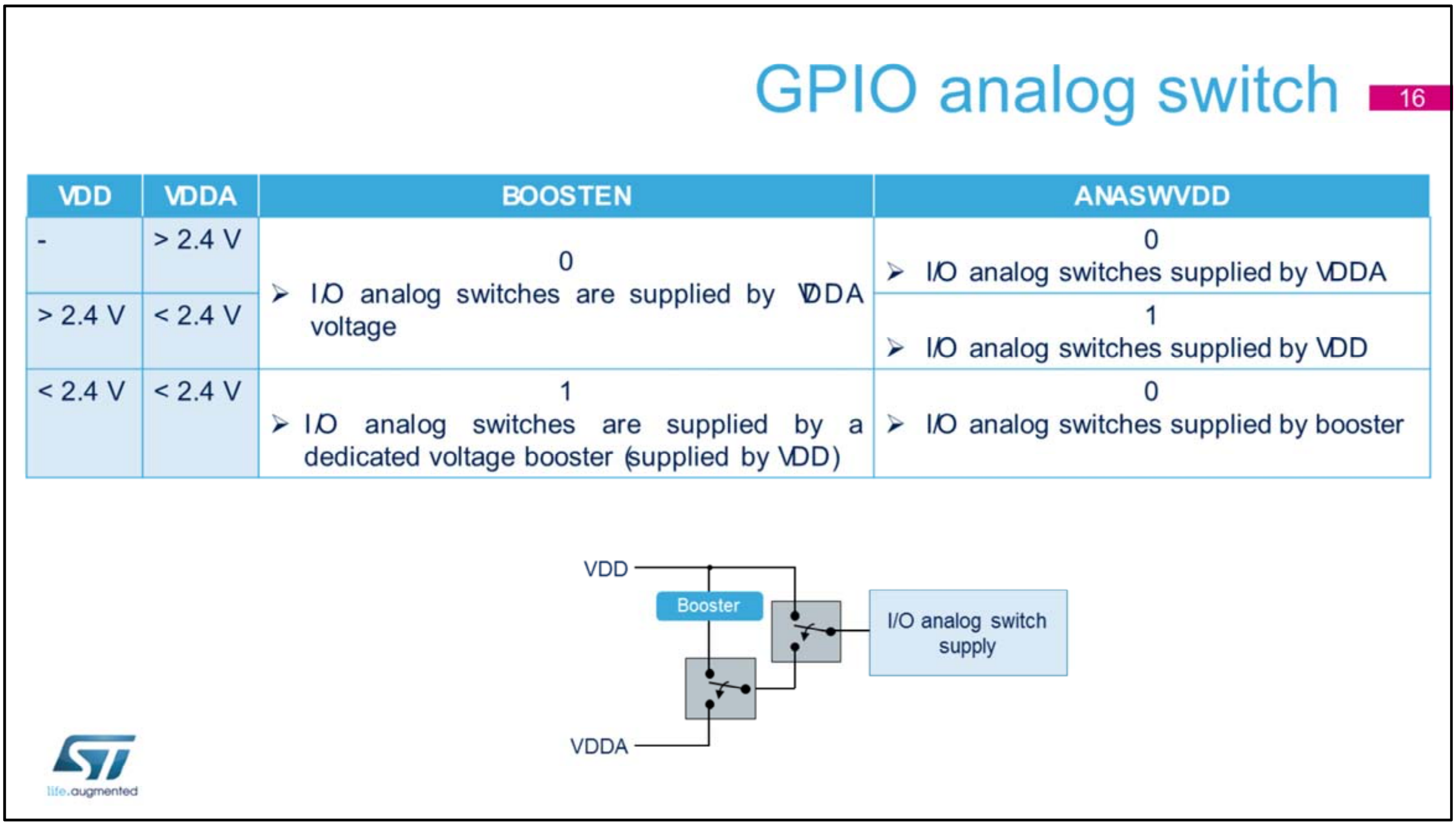
I²C I/Os Fast-mode

Pin	I²C alternate function	I²C FM+ mode enabled
PA13, PA15, **PB8**	I2C1_SCL	SYSCFG_CFGR1[I2C1_FMP]=1
PA14, **PB7**, **PB9**	I2C1_SDA	
PA9, PC4, PF6	I2C2_SCL	SYSCFG_CFGR1[I2C2_FMP]=1
PA8, PF0	I2C2_SDA	
PA8, PC8, PF3, PG7	I2C3_SCL	SYSCFG_CFGR1[I2C3_FMP]=1
PB5, PC9, PC11, PF4, PG8	I2C3_SDA	
PA13, PC6, PF14, PG3	I2C4_SCL	SYSCFG_CFGR1[I2C4_FMP]=1
PB7, PC7, PF15, PG4	I2C4_SDA	

Pin	Alternate functions	FM+ mode enabled
PB6	TIM16_CH1N, TIM4_CH1, TIM8_CH1, TIM8_ETR, TIM8_BKIN2, USART1_TX, COMP4_OUT, CAN2_TX, LPTIM1_ETR, HRTIM_SCIN, HRTIM_EEV4, SAI_FS_B, EVENTOUT, UCPD_CC1	SYSCFG_CFGR1[I2C_PB6_FMP]=1
PB7	TIM17_CH1N, TIM4_CH2, TIM8_BKIN, TIM3_CH4, **I2C4_SDA, I2C1_SDA**, USART1_RX, COMP3_OUT, CAN2_TXFD, LPTIM1_IN2, FMC_NL, HRTIM_EEV3, UART4_CTS, EVENTOUT, PVD_IN	SYSCFG_CFGR1[I2C_PB7_FMP]=1
PB8	TIM16_CH1, TIM4_CH3, SAI_CK1, **I2C1_SCL**, USART3_RX, COMP1_OUT, CAN1_RX, TIM8_CH2, TIM1_BKIN, HRTIM_EEV8, SAI_MCLK_A, EVENTOUT	SYSCFG_CFGR1[I2C_PB8_FMP]=1
PB9	TIM17_CH1, TIM4_CH4, SAI_D2, **I2C1_SDA**, IR_OUT, USART3_TX, COMP2_OUT,CAN1_TX, TIM8_CH3, TIM1_CH3N, HRTIM_EEV5, SAI_FS_A, EVENTOUT	SYSCFG_CFGR1[I2C_PB9_FMP]=1

life.augmented

Each I2C controller has a control bit in the SYSCFG_CFGR1 register to enable fast mode plus driving capability mode.
Each pin PB6, PB7, PB8, PB9 has its own I2C_PB_FMP control bit to activate the fast mode plus driving capability, whatever the selected alternate function.
When FM+ mode is activated on the GPIO pin, the speed configuration of the GPIO, programmed in the GPIOx_OSPEEDR register, is ignored.
In some cases, there are two ways to activate the FM+ mode. For instance, the PB8 pin is configured as I2C1_SCL supports FM+ when the I2C1_FMP or I2C_PB8_FMP bit is set to one in the SYSCFG_CFGR1 register.



GPIO analog switch

VDD	VDDA	BOOSTEN	ANASWVDD
-	> 2.4 V	0 ➢ I/O analog switches are supplied by VDDA voltage	0 ➢ I/O analog switches supplied by VDDA
> 2.4 V	< 2.4 V		1 ➢ I/O analog switches supplied by VDD
< 2.4 V	< 2.4 V	1 ➢ I/O analog switches are supplied by a dedicated voltage booster (supplied by VDD)	0 ➢ I/O analog switches supplied by booster

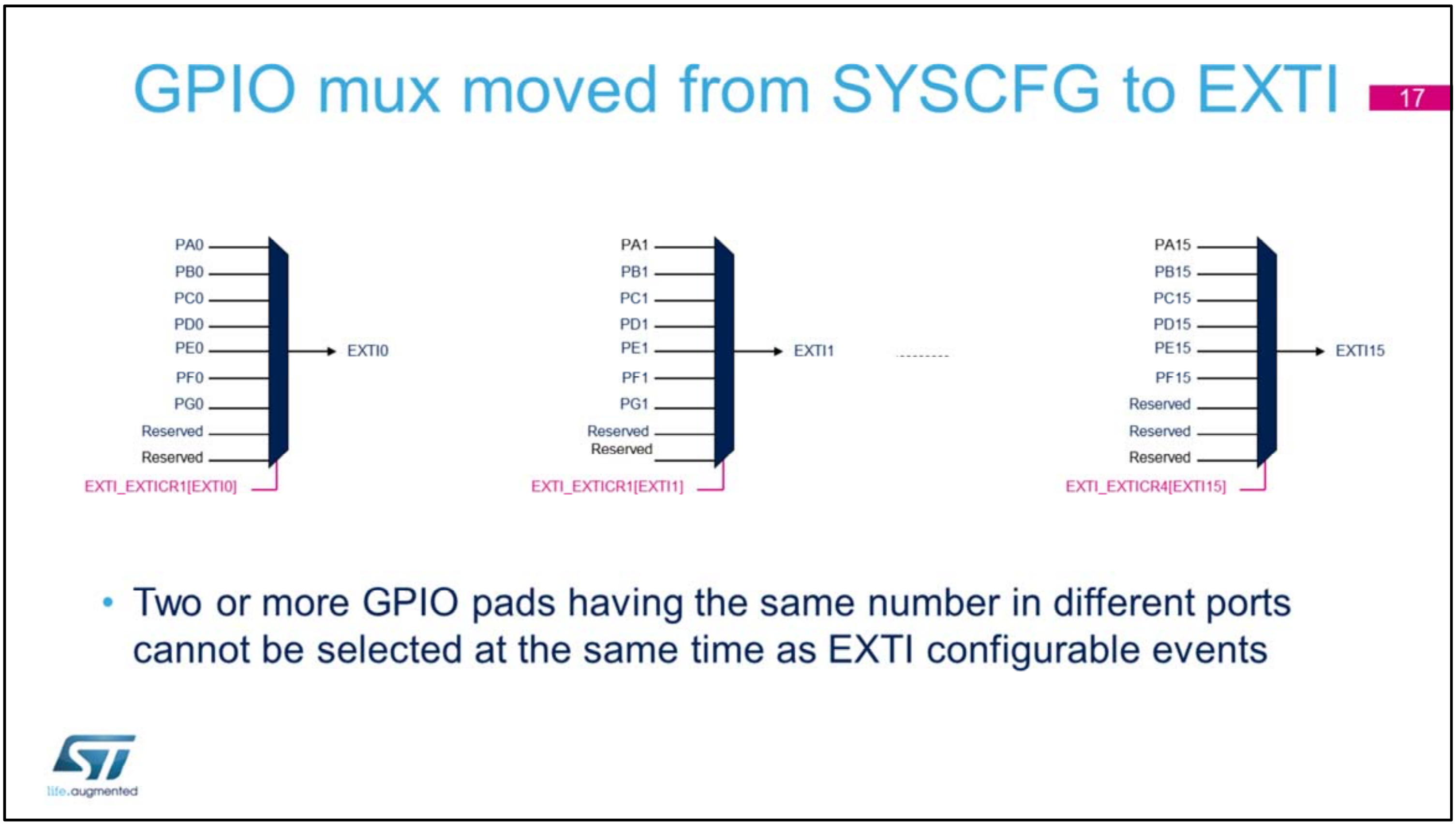
Two bits from the SYSCFG_CFGR1 register are used to select the power supply of the I/O analog switch: BOOSTEN and ANASWVDD.

They have to be initialized according to the voltage of the VDD and VDDA power supplies.

When VDDA voltage is larger than 2.4 V, the I/O analog switch should be powered by VDDA.

When VDDA voltage is lower than 2.4 V and VDD is higher than 2.4 V, the I/O analog switch should be powered by VDD.

When both VDD and VDDA voltages are lower than 2.4 V, the I/O analog switch should be powered by the output of the VDD booster.



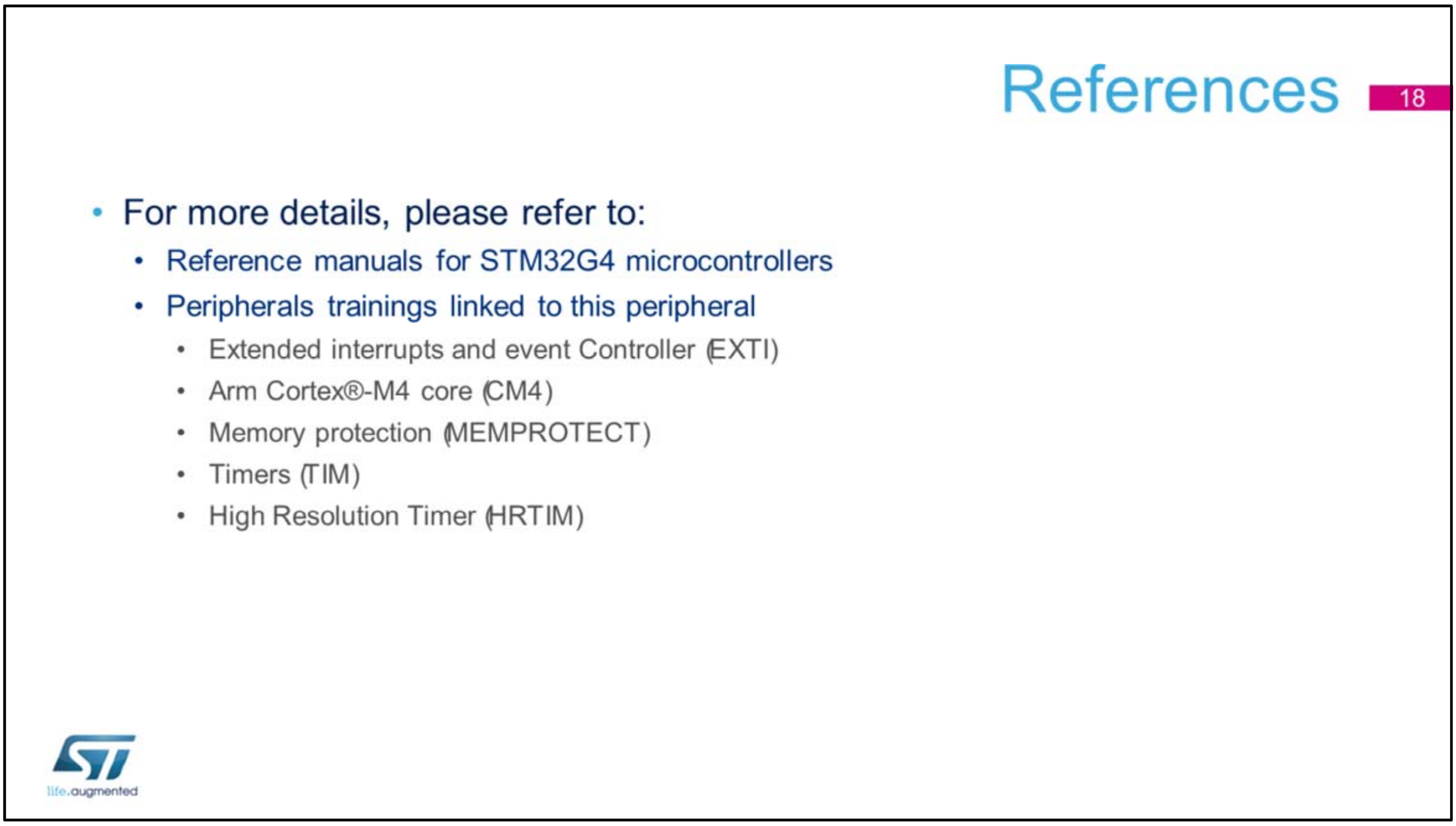
GPIO mux moved from SYSCFG to EXTI

17

- Two or more GPIO pads having the same number in different ports cannot be selected at the same time as EXTI configurable events

life.augmented

The STM32G4 microcontroller has 7 IO ports: Ports A to F are 16-pin wide ports , port G is 11-pin wide port.
Each of the 16 EXTI configurable events related to GPIO ports has an independent multiplexor.
The EXTI multiplexer outputs are available independently from any masks defined in the EXTI_IMR and EXTI_EMR registers.



References

18

- For more details, please refer to:
  - Reference manuals for STM32G4 microcontrollers
  - Peripherals trainings linked to this peripheral
    - Extended interrupts and event Controller (EXTI)
    - Arm Cortex®-M4 core (CM4)
    - Memory protection (MEMPROTECT)
    - Timers (TIM)
    - High Resolution Timer (HRTIM)

life.augmented

For more details about the System Configuration module, refer to the reference manual for STM32G4 microcontrollers.

Refer also to these trainings for more information if needed:

- Extended interrupts and event Controller,
- Arm Cortex-M4 core,
- Timers,
- High-resolution timer.