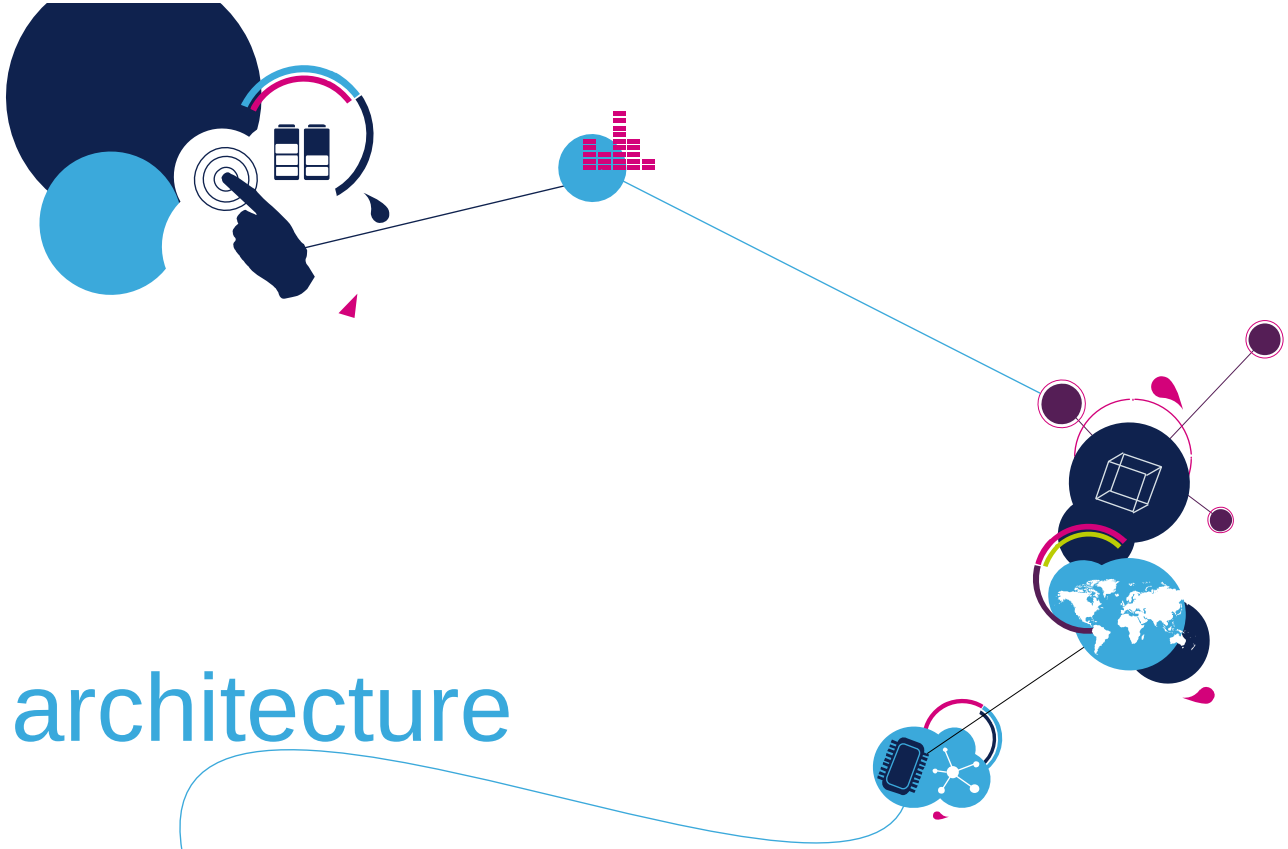


STM32L0 architecture

life.augmented

STM32 L0

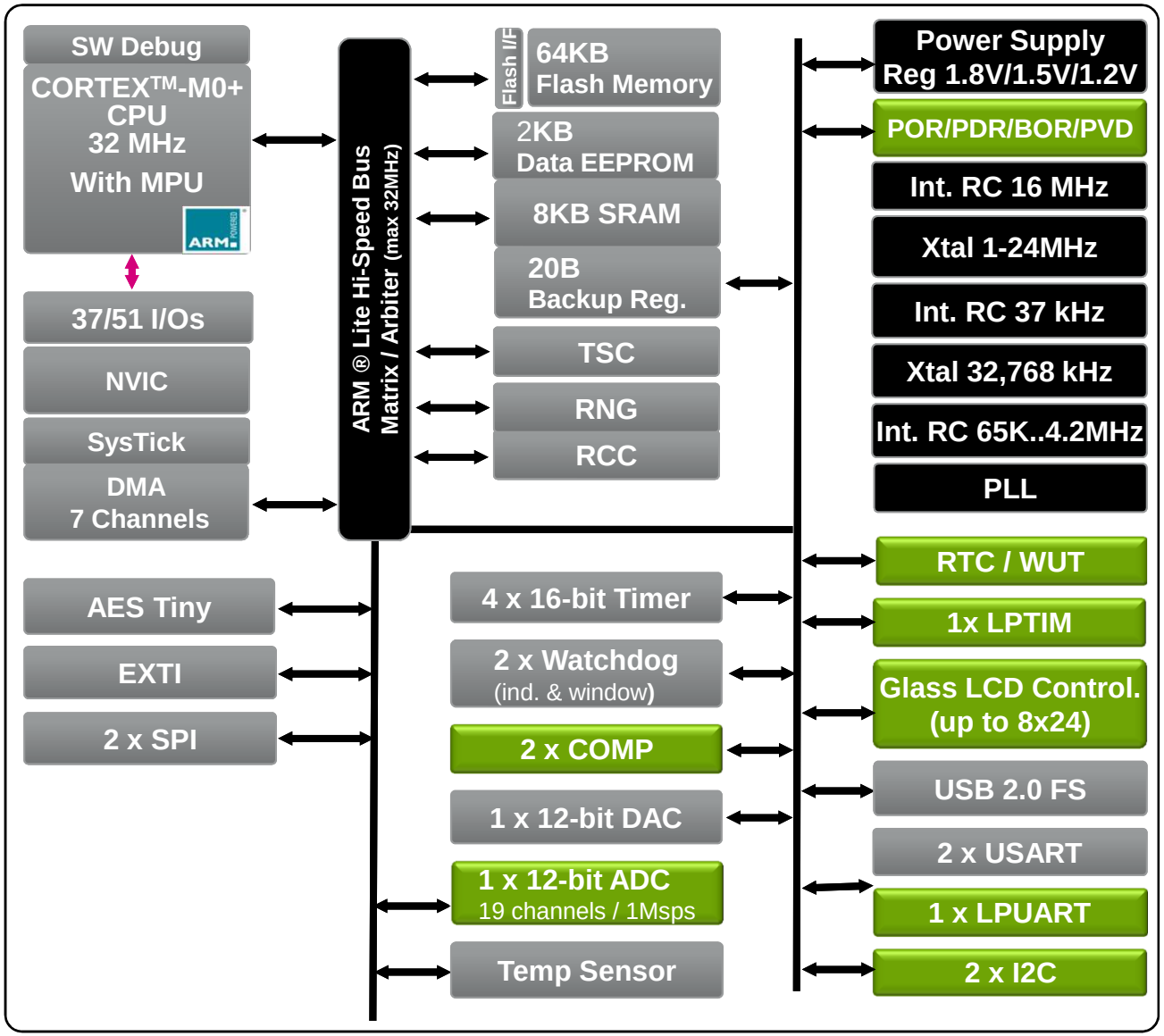
OBJECTIVES

- Introduce STM32L0 internal structure
- Briefly describe each internal component
  - some of them will be explained more in detail later
- Highlight the main features of each peripheral

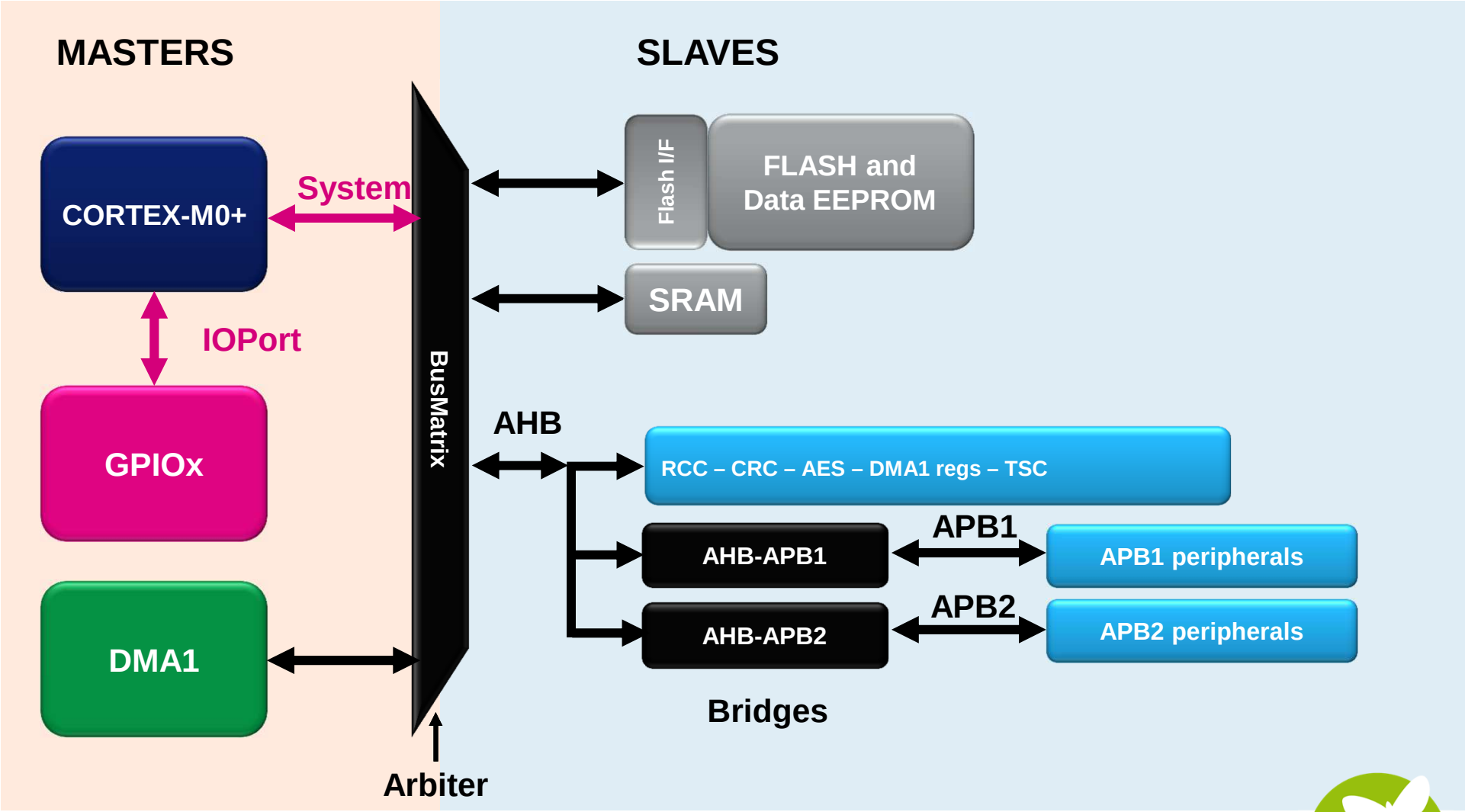
After this presentation you will know what you can find inside STM32L0 today.

Introduction

STM32L06x – 64kB Block Diagram

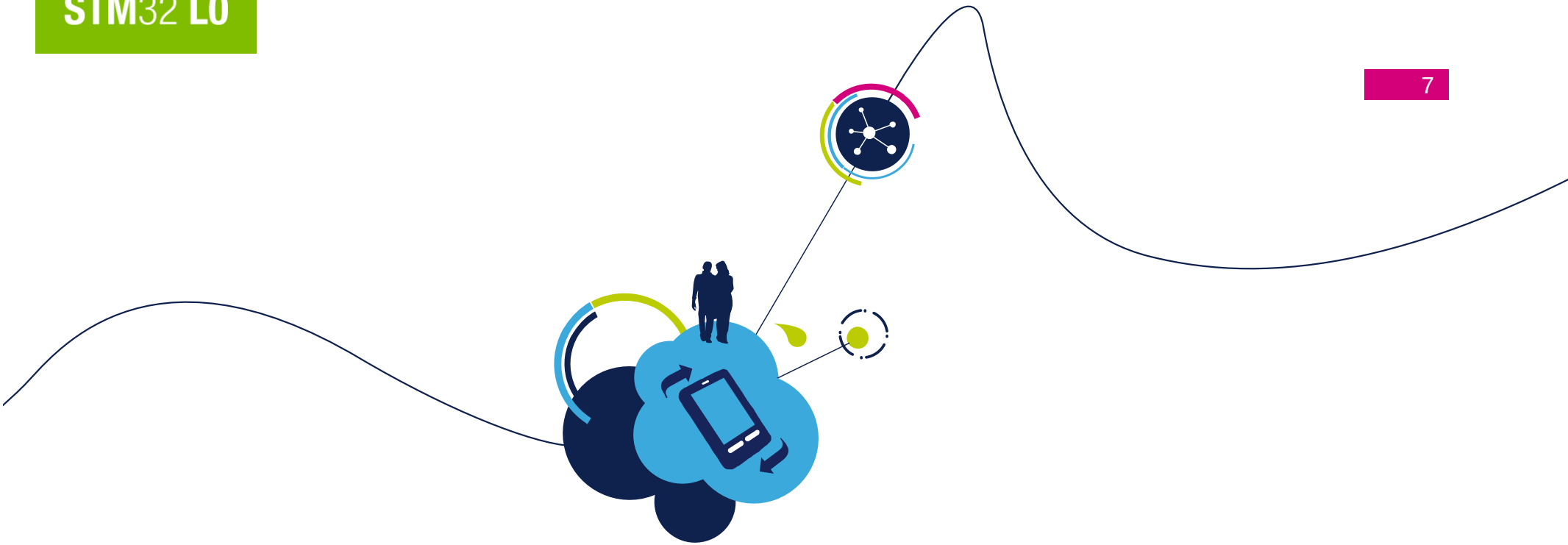


System Architecture



Buses are not overloaded with data movement tasks !

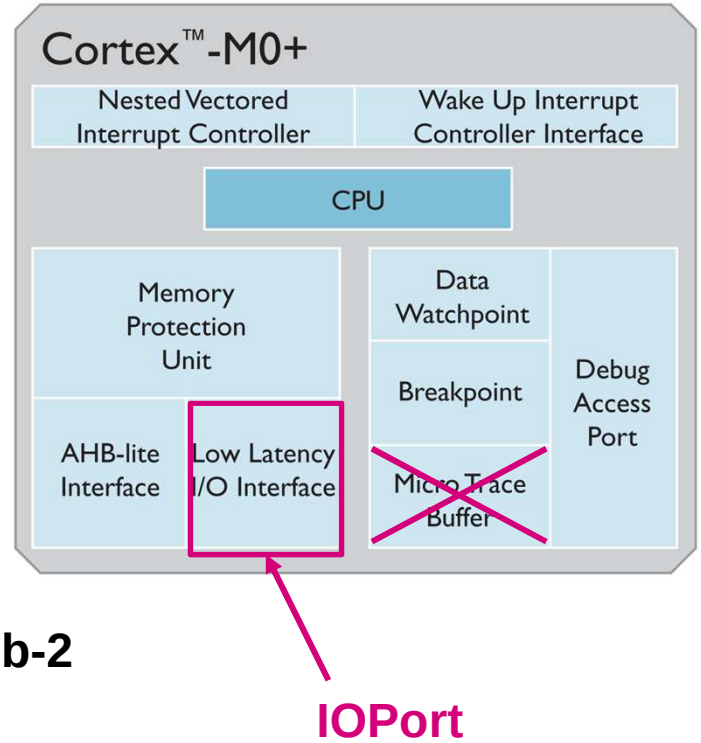
System blocks



System blocks
Core

Cortex-**M0+** Processor Overview

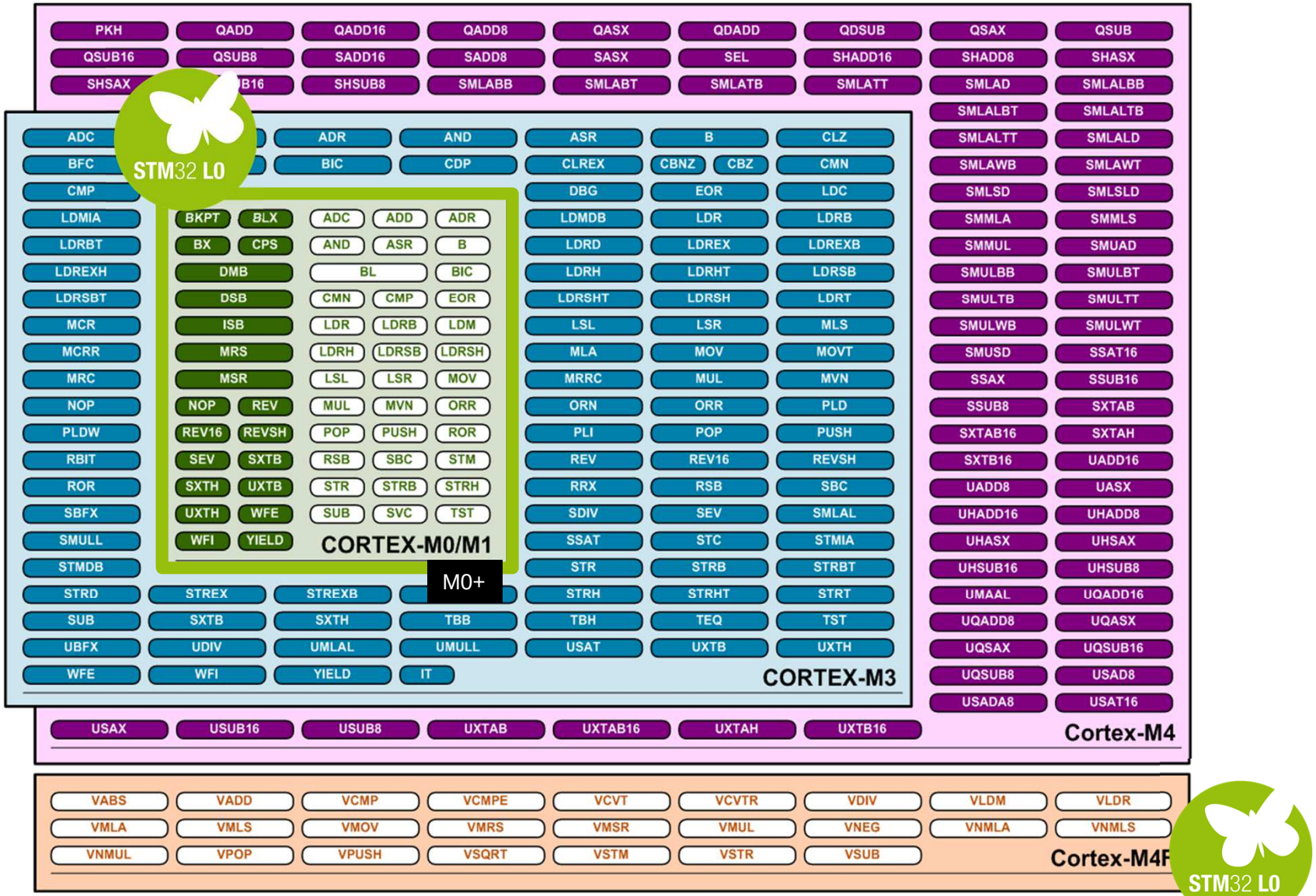
- **ARMv6-M** Architecture
- **von Neumann** architecture, **2-stage pipeline**
- **Integrated** Nested Vectored Interrupt Controller (**NVIC**) for low latency interrupt processing
- Designed to be fully programmed in **C-language**
- Vector Table is a simple list of addresses



- Full **Thumb** Instruction Set and **subset of Thumb-2**
- **Single cycle multiply (optional)**
- **Memory Protection Unit (MPU)*** **(optional)** **, privileged / unprivileged mode***
- Integrated 24-bit System Timer (**SysTick**) for RTOS (optional)

+ interrupt vector table relocation

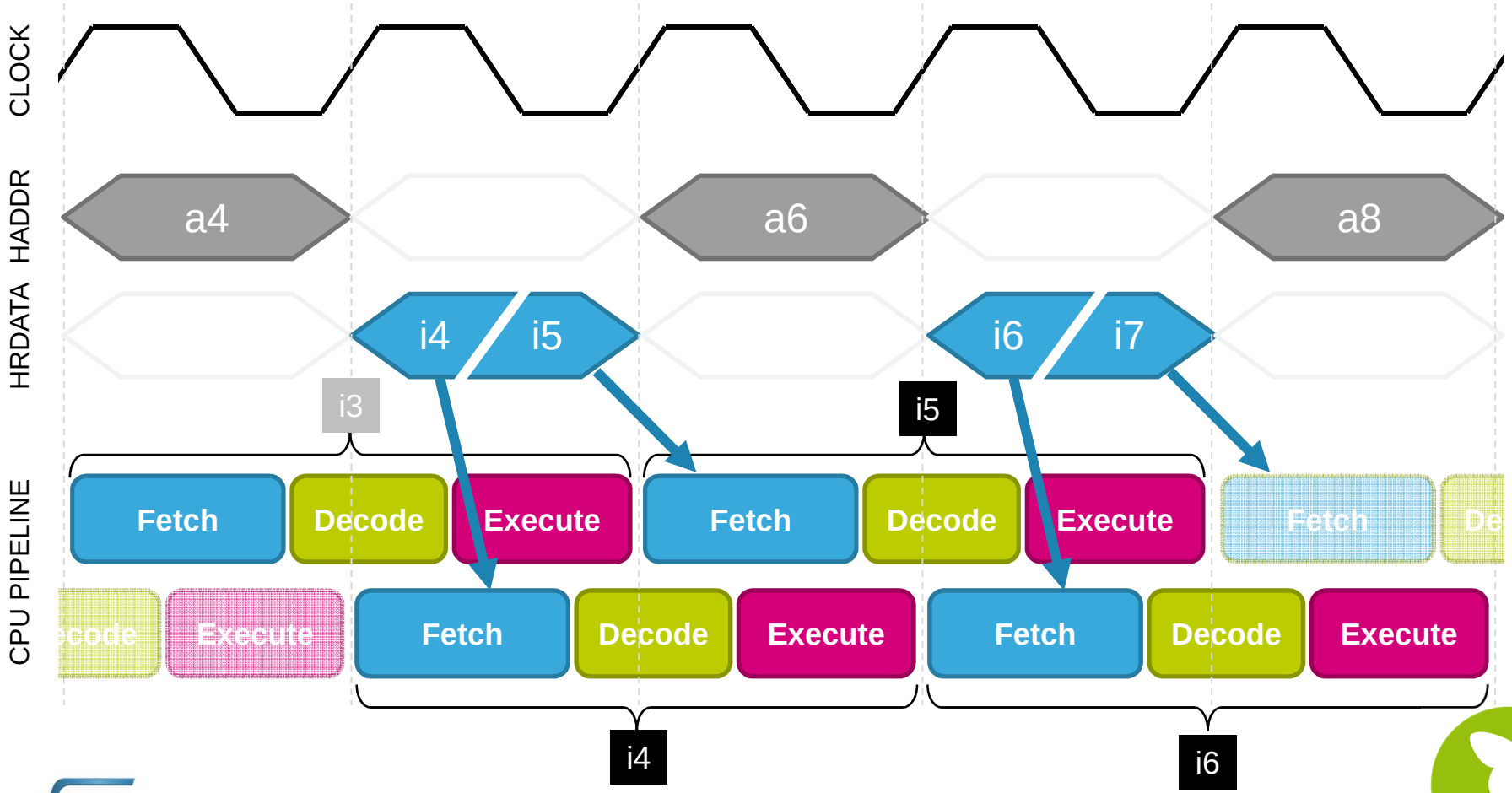
Cortex-M0+ instruction set



life.augmented

Cortex-M0+ pipeline

- Only two stage pipeline for maximum energy efficiency

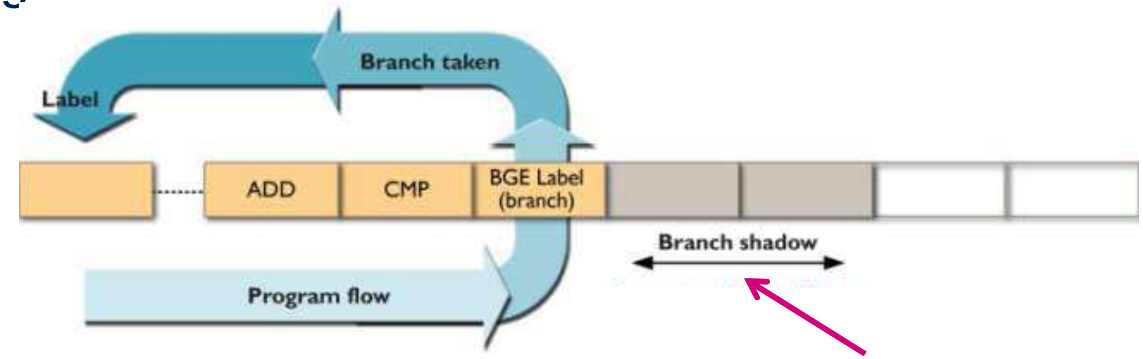


1x 32-bit word = 2x 16-bit wide instructions

Cortex-M0+ Higher dynamic efficiency

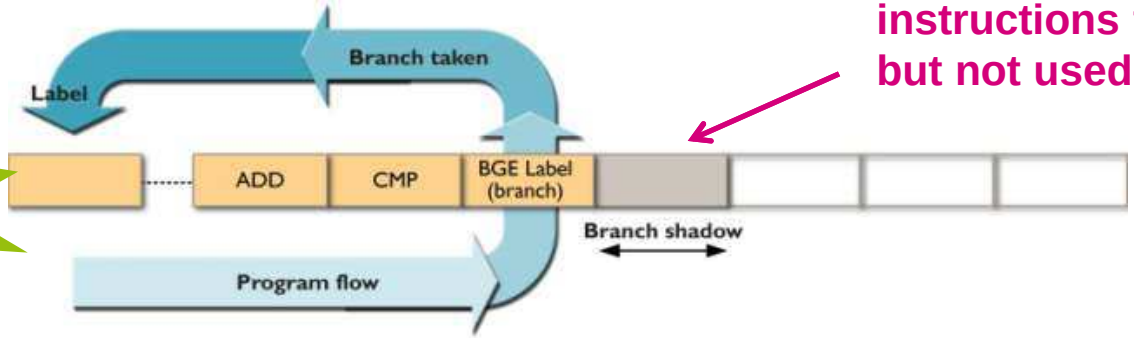
- In pipelined processors, subsequent instructions are fetched while executing current instructions

3-stage pipeline



instructions fetched/decoded but not used

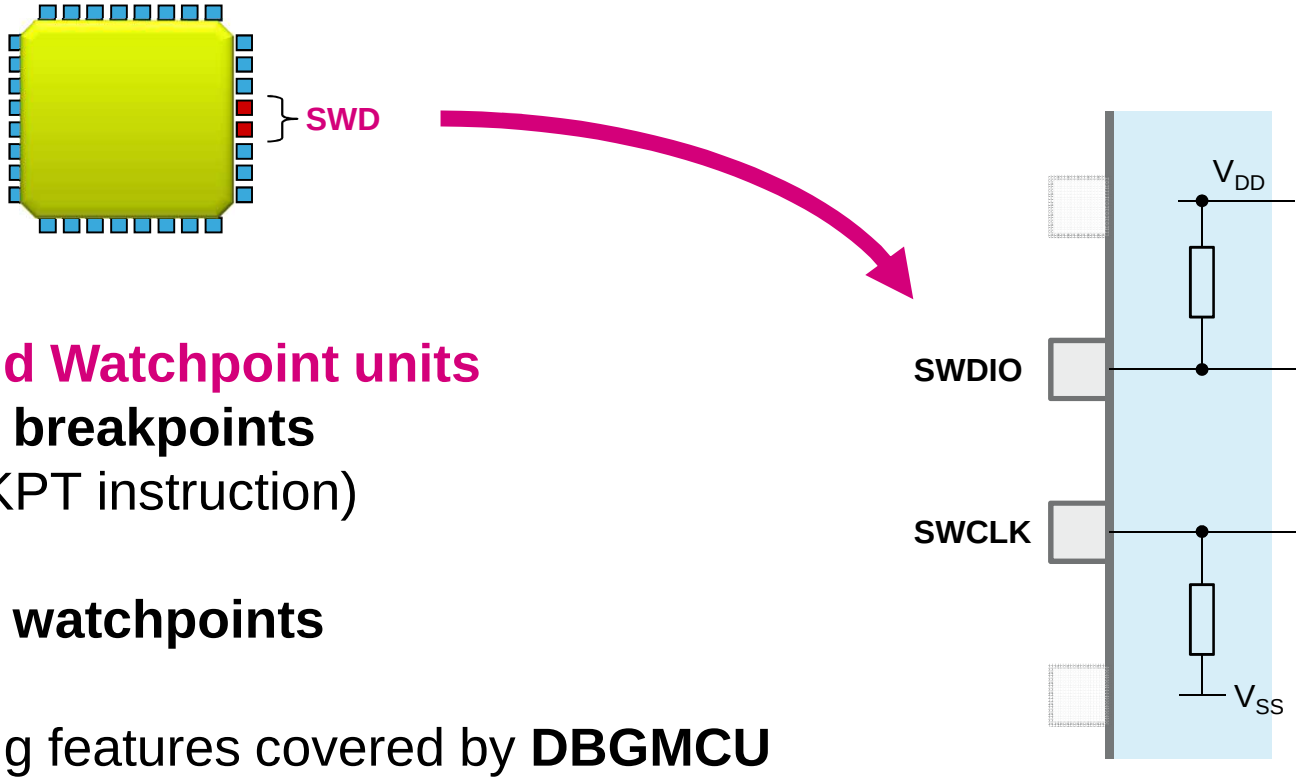
2-stage pipeline



- In **2-stage** pipeline:
  - Branch shadow is reduced and energy is saved!
  - Branch turn-around is 1 cycle faster!

Debug Capabilities

- Serial Wire Debug interface

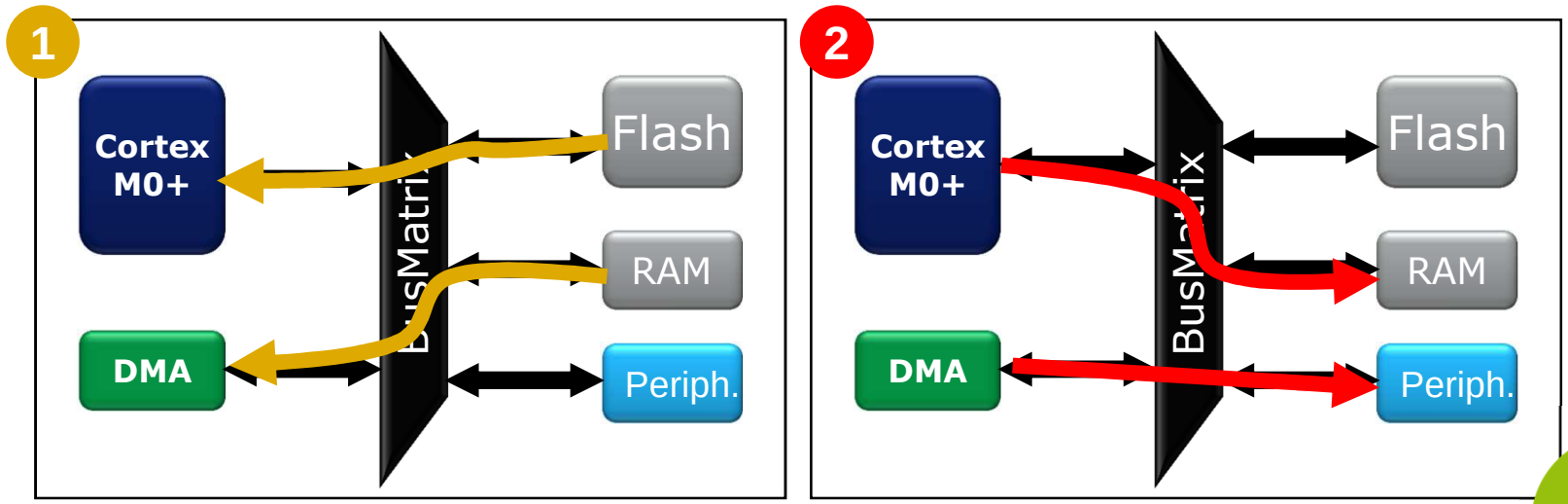


- **Breakpoint and Watchpoint units**
  - 4 hardware **breakpoints** (besides BKPT instruction)
  - 2 hardware **watchpoints**
- Additional debug features covered by **DBGMCU** peripheral

System blocks
DMA

DMA Controller Features

- Up to **7** independently configurable **channels** (requests)
- 4 configurable levels of priority
- Independent source and destination transfer size (**byte / half word / word**)
- Support for **circular buffer** management
- **Half-Transfer** and Transfer complete **event**s
- Programmable number of data to be transferred: up to **65536 (16-bit counter)**

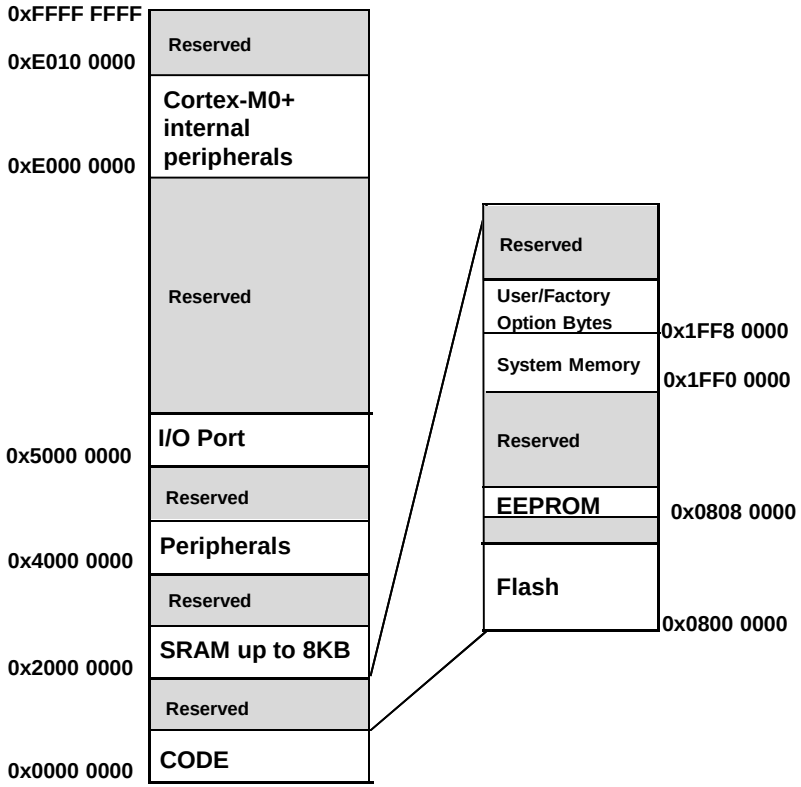


System blocks
Internal Memories

life.augmented

Memory Mapping and Boot Modes

- **Addressable memory space of 4 Gbytes**
- **RAM : up to 8 Kbytes**
- **FLASH : up to 64 Kbytes**
- **Data EEPROM: up to 2 Kbytes**

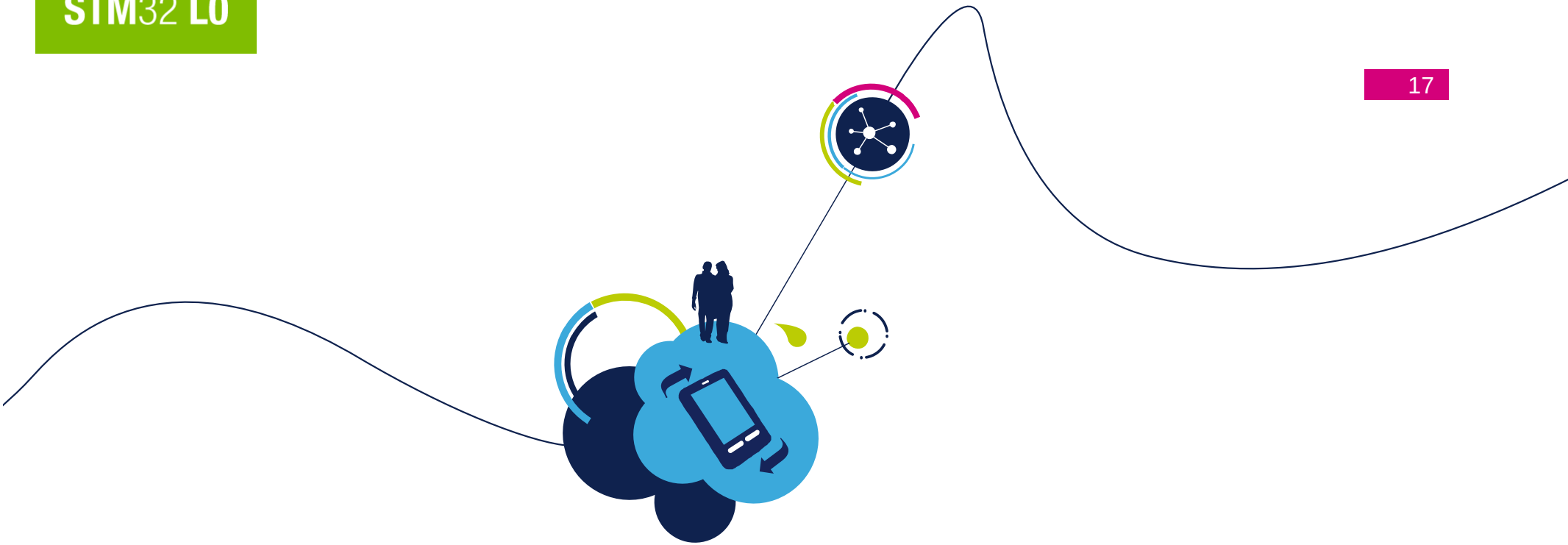


- **Boot modes**
  Depending on the Boot configuration, Embedded Flash Memory, System Memory or Embedded SRAM Memory is aliased at @0x00. The System Memory and Embedded SRAM memory can be remapped also at @0x0 using a dedicated software bits.
  BOOT0 is read on dedicated pin BOOT0
  BOOT1 is an option register bit

BOOT Mode Selection		Boot Mode	Aliasing
nBOOT1	BOOT0		
x	0	User Flash	User Flash is selected as boot space
0	1	SystemMemory	SystemMemory is selected as boot space
1	1	Embedded SRAM	Embedded SRAM is selected as boot space

- **SystemMemory**: contains the Bootloader used to re-program the FLASH through USART1/2 or SPI1/2

- **Boot from SRAM** In the application initialization code you have to Relocate the Vector Table in SRAM using the NVIC Exception Table and Offset register.



System blocks
Reset and Clock Control (RCC)

The Gear Box

- clock tree consists of **6 clock sources** + **1xPLL**
  - **Dynamic Internal Voltage Scaling**: optimize consumption according to speed you need!
  - **Consumption down to few µA only with still running CPU !**

MSI
Internal 65kHz to 4.2MHz

Multi-Speed Internal clock : Default RUN mode
- Low to Medium frequency, Ultra-Low consumption
- Default Clock Source (2.1MHz after reset)

HSI16
Internal @ 16MHz
+/-0.5%

High-Speed Internal 16MHz clock : Performance mode
- Up to 32MHz (PLL): 33.3 DMIPS

HSE
External 1-24MHz

High-Speed External clock : Crystal / ext. signal
- USB 48MHz clk with single 16MHz crystal + **PLL**.

PLL
Up to 32MHz

HSI48
Internal 48MHz

High-Speed Internal 48MHz clock : USB and RNG
- "Synchronizable" 48MHz oscillator for USB enabling crystal less operation
- RNG seed clock source

LSI
Internal @ 37kHz

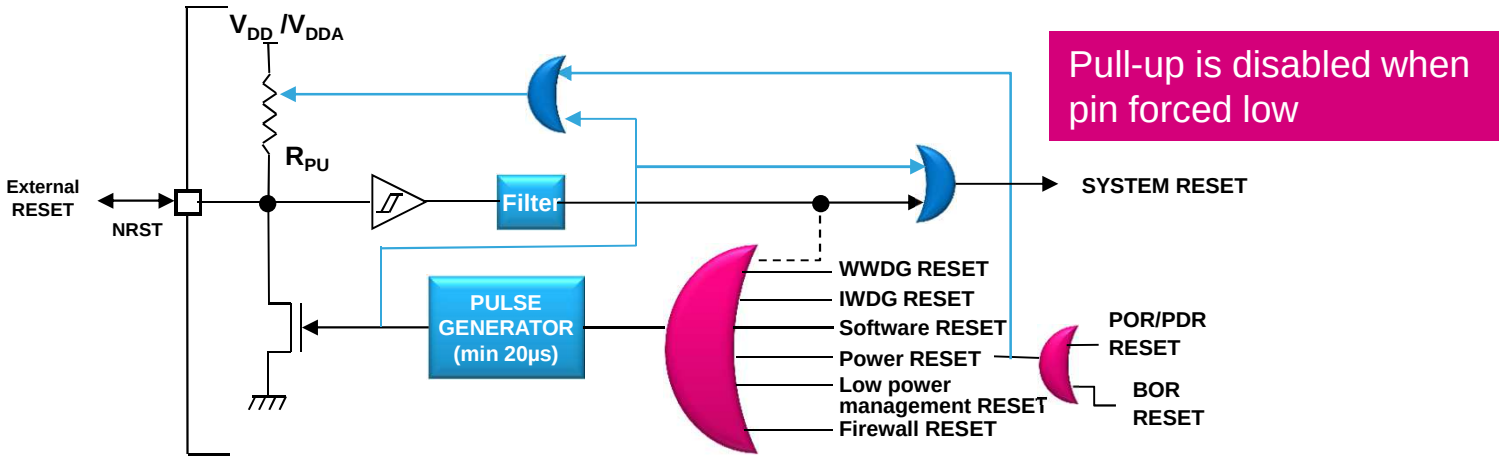
Low Speed Internal clock : Security clock
- Used for Independent Watchdog security and RTC

LSE
External @ 32kHz

Low Speed External clock (32.768KHz)
- Mainly used for precise RTC
- Could be used to calibrate HSI & MSI

Configurable drive level

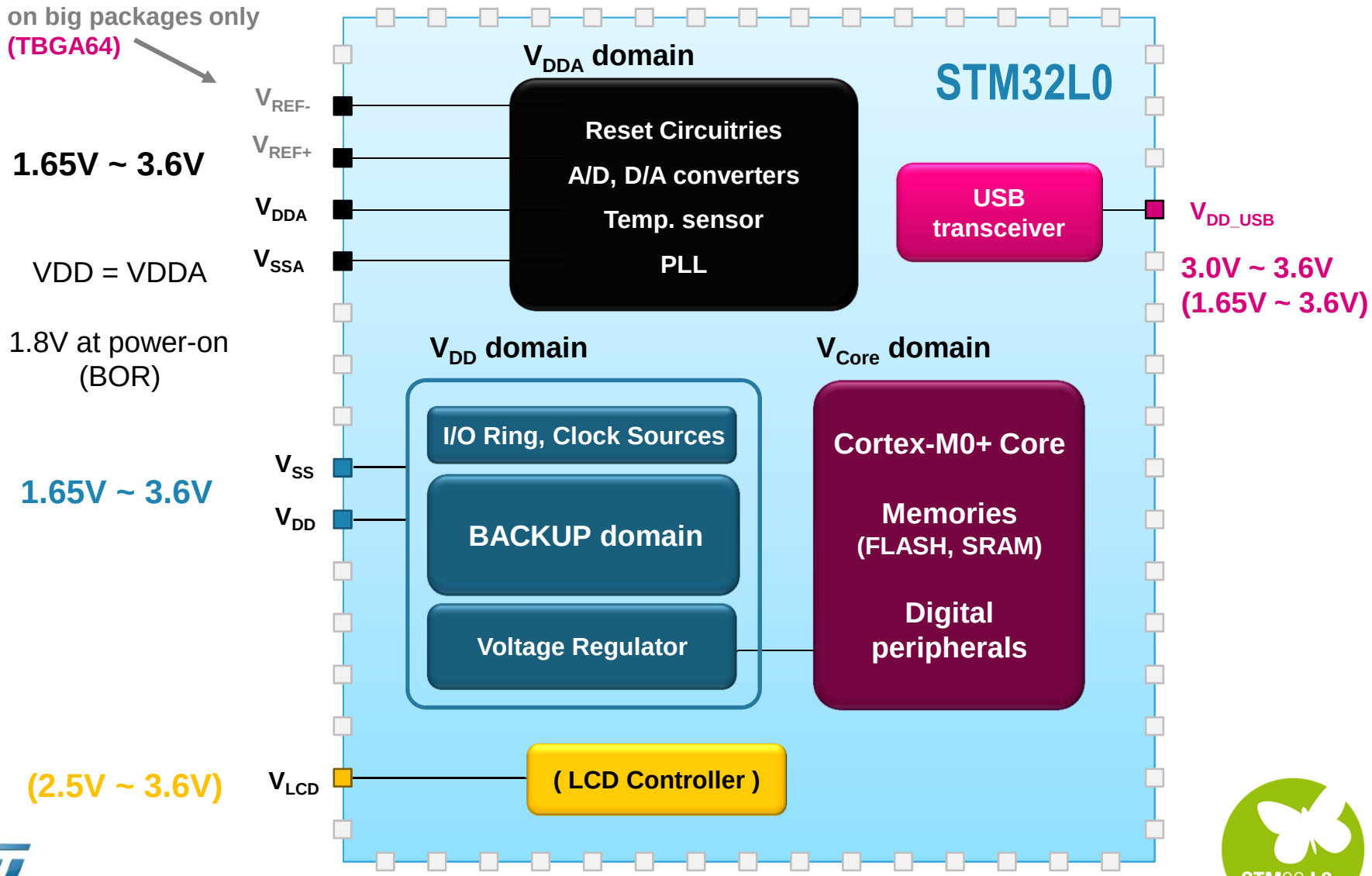
Internal Reset Circuitries



- Power-on-Reset / Power-down-Reset circuitry (**POR/PDR**):
  - For devices operating from 1.65 to 3.6 V, there is no BOR and the reset is released when V_{DD} goes above POR level (1.5V) and asserted when V_{DD} goes below PDR (1.5V) level (no hysteresis)
- Brown-out-Reset circuitry (**BOR**): (enabled by default, can be disabled)
  - Configurable level from 1.8V up to 2.9V (100mV hysteresis), if enabled → POR/PDR have no effect
- Programmable Voltage Detector (**PVD**)
  - Configurable level from 1.9V up to 3.1V (100mV step), no reset, can generate interrupt

System blocks
Power Control (PWR)

Power Supply Domains



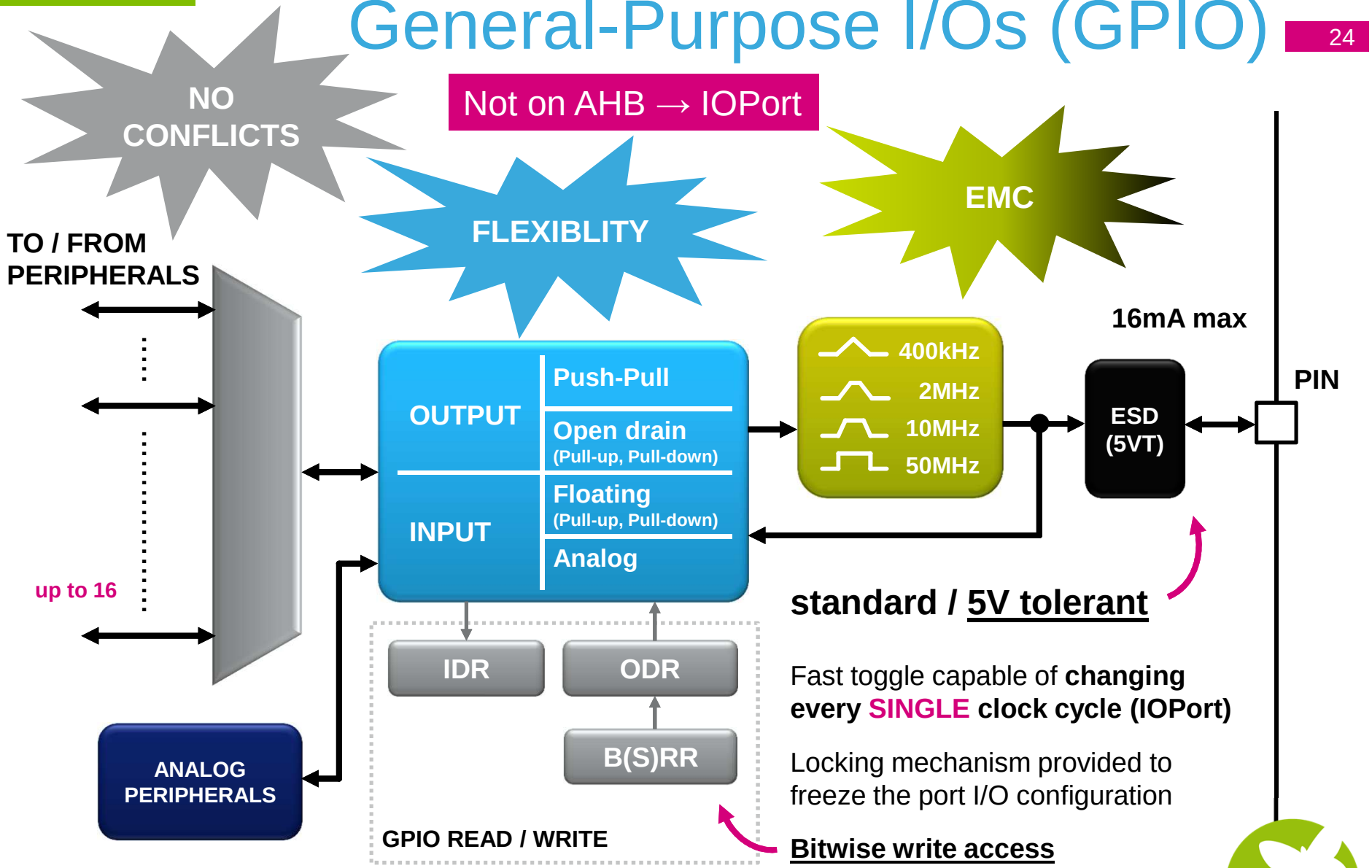
For Battery Powered Application

- Functional down to **1.65V**
  - What for?
    - Battery powered radio, fire alarms, motion detectors, other sensors…
  - Extend application life time vs. 1.8V standard V_{DD} Range
  - **1.65V but no compromise on performance**
    - @ V_{DD} = 1.65V freq. running still @ 16MHz
  - …But what is functional @**1.65V**?
    - Memories and Core (even Flash/EEPROM can be programmed)
    - GPIOs
    - Comparators
    - Communication peripherals (USART, SPI, I2C)
    - Timers, RTC
    - Capacitive touch…
  - (Only some high-speed peripherals and some analog peripherals need higher voltage)

System blocks
GPIO

General-Purpose I/Os (GPIO)

Not on AHB → IOPort



standard / 5V tolerant

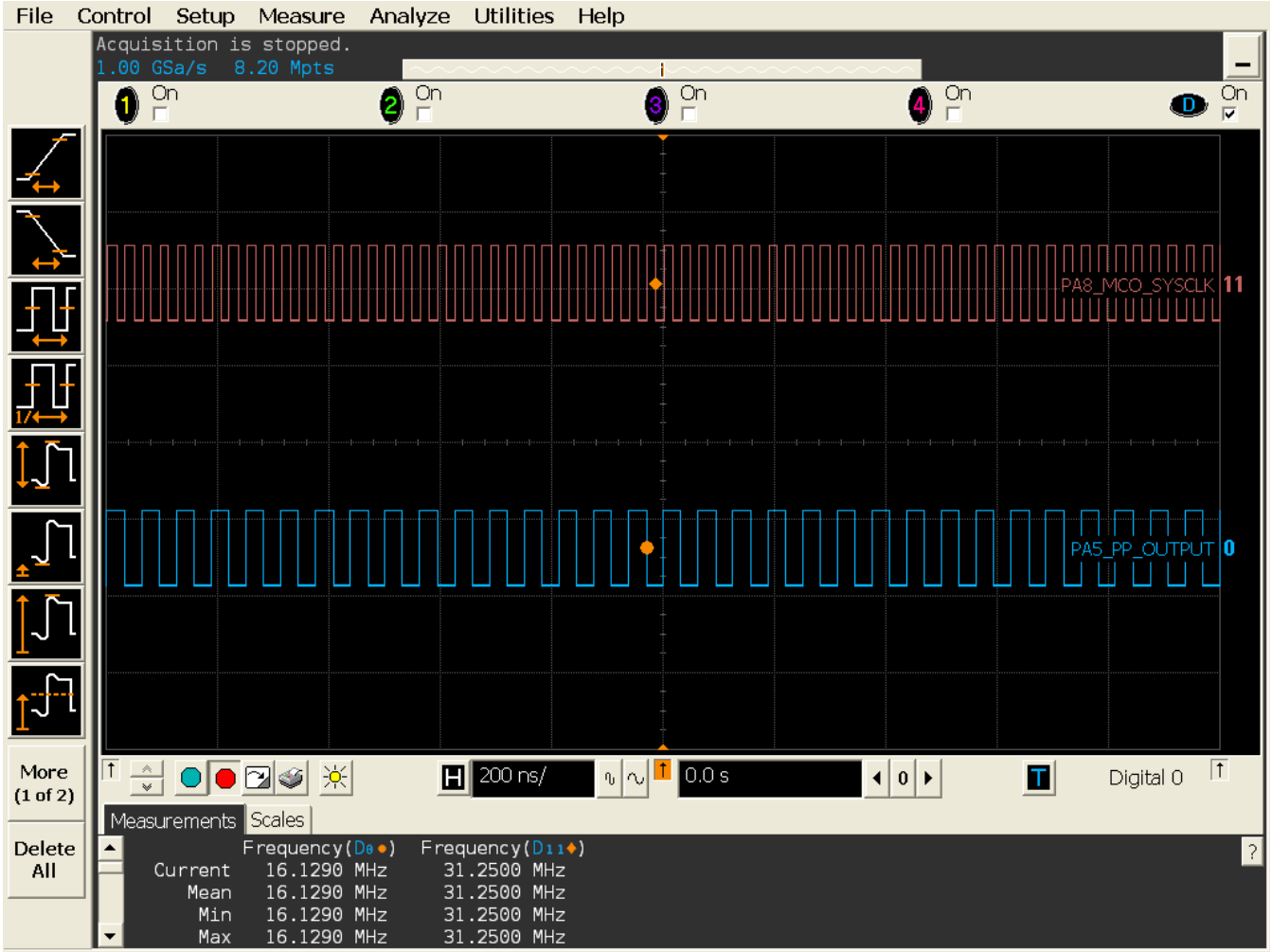
Fast toggle capable of **changing every SINGLE clock cycle (IOPort)**

Locking mechanism provided to freeze the port I/O configuration

Bitwise write access

All the GPIO's can be configured to generate **interrupt on external event** (up to 16 lines at time)

16MHz SW I/O toggling



PA8_MCO

32MHz SYSCLK (HSI16 + PLL)

PA5 as Push-Pull output

16MHz toggling generated by consecutive writes to BRR and BSRR

16MHz I/O toggling

Compiler optimizations to be enabled for SPEED

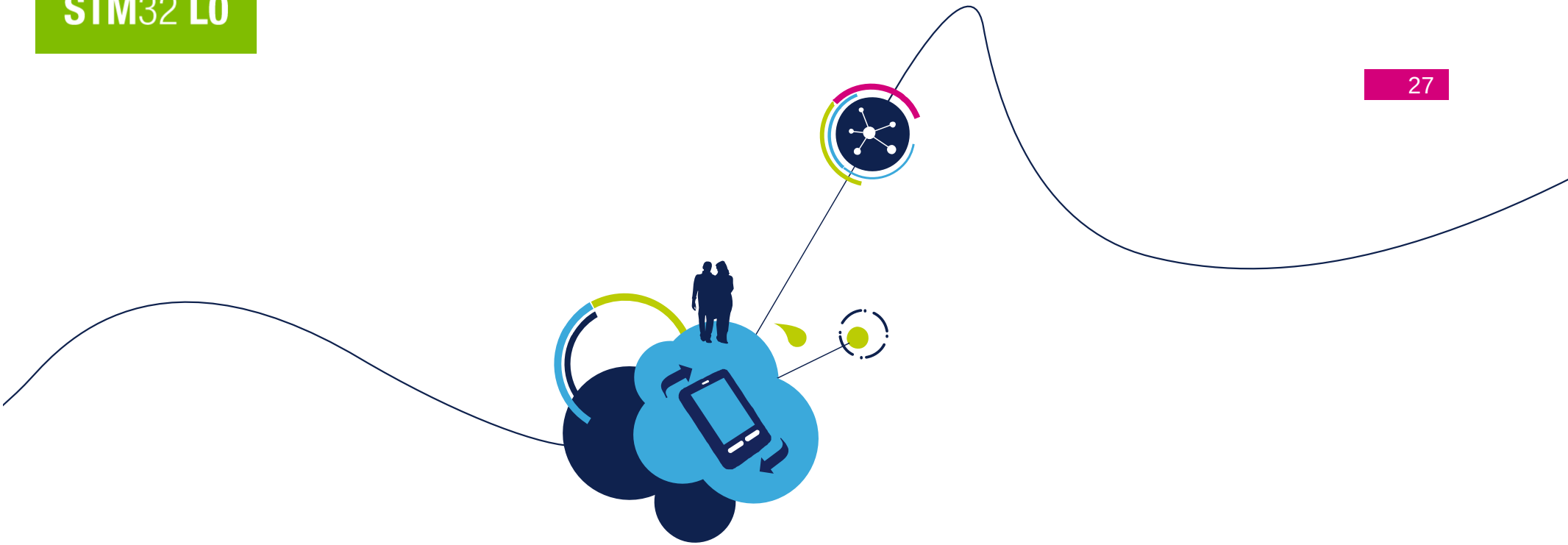
```
98
99    /* Configure PA5 */
100   /* GPIO periph clocks enable */
101   __GPIOA_CLK_ENABLE();
102
103   /* Set all unused GPIO pins as analog in
104   GPIO_InitStructure.Pin = GPIO_PIN_5;
105   GPIO_InitStructure.Speed = GPIO_SPEED_HI
106   GPIO_InitStructure.Mode = GPIO_MODE_OUTP
107   GPIO_InitStructure.Pull = GPIO_NOPULL;
108   GPIO_InitStructure.Alternate = GPIO_AF0_
109   HAL_GPIO_Init(GPIOA, &GPIO_InitStructure
110
111   /* Infinite loop */
112   while (1)
113   {
114     GPIOA->BSRR = GPIO_PIN_5; /* 1 */
115     GPIOA->BRR = GPIO_PIN_5;
116     GPIOA->BSRR = GPIO_PIN_5; /* 2 */
117     GPIOA->BRR = GPIO_PIN_5;
118     GPIOA->BSRR = GPIO_PIN_5; /* 3 */
119     GPIOA->BRR = GPIO_PIN_5;
120     GPIOA->BSRR = GPIO_PIN_5; /* 4 */
121     GPIOA->BRR = GPIO_PIN_5;
122     GPIOA->BSRR = GPIO_PIN_5; /* 5 */
123     GPIOA->BRR = GPIO_PIN_5;
124     GPIOA->BSRR = GPIO_PIN_5; /* 6 */
125     GPIOA->BRR = GPIO_PIN_5;
126     GPIOA->BSRR = GPIO_PIN_5; /* 7 */
127     GPIOA->BRR = GPIO_PIN_5;
128     GPIOA->BSRR = GPIO_PIN_5; /* 8 */
```

```
GPIO_InitStructure.Mode = GPIO_MODE_OUTPUT_PP;
  0x8000ab8: 0x6048          STR       R0, [R1, #0x4]
GPIO_InitStructure.Pull = GPIO_NOPULL;
  0x8000aba: 0x2000          MOVS      R0, #0
  0x8000abc: 0x6088          STR       R0, [R1, #0x8]
GPIO_InitStructure.Alternate = GPIO_AF0_SPI1;
  0x8000abe: 0x6108          STR       R0, [R1, #0x10]
HAL_GPIO_Init(GPIOA, &GPIO_InitStructure);
  0x8000ac0: 0x25a0          MOVS      R5, #160
  0x8000ac2: 0x05ed          LSLS      R5, R5, #23
  0x8000ac4: 0x0028          MOVS      R0, R5
  0x8000ac6: 0xf7ff 0xfee1   BL        HAL_GPIO_Init
  0x8000aca: 0x2020          MOVS      R0, #32
  GPIOA->BSRR = GPIO_PIN_5; /* 1 */
??main_0:
  0x8000acc: 0x61ac          STR       R4, [R5, #0x18]
  GPIOA->BRR = GPIO_PIN_5;
  0x8000ace: 0x8528          STRH      R0, [R5, #0x28]
  GPIOA->BSRR = GPIO_PIN_5; /* 2 */
  0x8000ad0: 0x61ac          STR       R4, [R5, #0x18]
  GPIOA->BRR = GPIO_PIN_5;
  0x8000ad2: 0x8528          STRH      R0, [R5, #0x28]
  GPIOA->BSRR = GPIO_PIN_5; /* 3 */
  0x8000ad4: 0x61ac          STR       R4, [R5, #0x18]
  GPIOA->BRR = GPIO_PIN_5;
  0x8000ad6: 0x8528          STRH      R0, [R5, #0x28]
  GPIOA->BSRR = GPIO_PIN_5; /* 4 */
  0x8000ad8: 0x61ac          STR       R4, [R5, #0x18]
  GPIOA->BRR = GPIO_PIN_5;
```

1 toggle

Example can be found in the STM32CubeL0 package.

(…STM32L053R8-Nucleo\Examples\GPIO\GPIO_IOToggle_MaxFrequency)



System blocks
Watchdogs

Watchdogs

- Independent Watchdog (IWDG) → **IWWDG**
  - Dedicated low speed clock (LSI)
  - HW and SW way of enabling
  - IWDG clock still active if main clock source fails
  - Timeout values @37kHz: **108us …28s**
  - **Window Functionality**

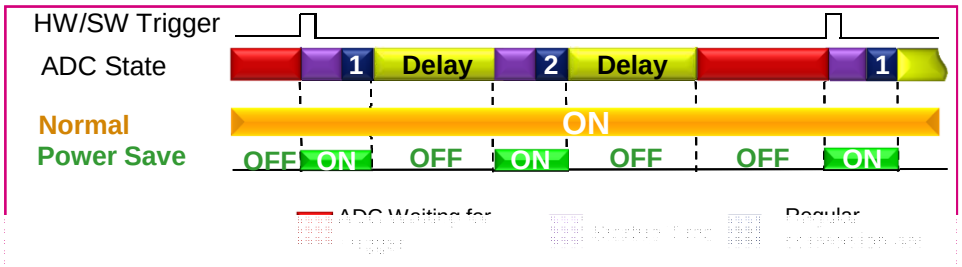
- Window Watchdog (WWDG)
  - Configurable Time Window
  - Can detect abnormally early or late application behavior
  - Conditional Reset
  - WWDG Reset flag
  - Timeout value @32MHz: **128us … 65.54ms**

Analog peripherals

Analog to Digital Converter (ADC)

- ADC conversion rate 1.14 MSPS and 12-bit resolution (**0.87us @16MHz**)
  - **Up to 16-bit resolution with internal HW oversampler**
- Available in Performance and Low-power Run with CPU clk @ 32kHz
- ADC supply requirement: 1.8V to 3.6 V (from 2.4V full speed available)
- Up to **19 multiplexed channels** (16 external + 3 internal)
- Single and continuous conversion modes
- **Programmable sampling time**
- Hardware Delay insertion between conversions
- Programmable Conversion resolution : **12**, 10, 8 or **6-bit**
- **Analog Watchdog** on high and low thresholds

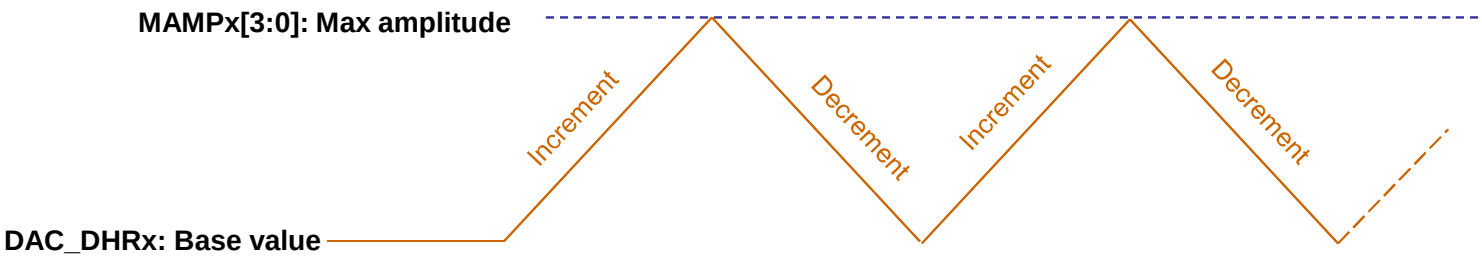
DMA



Digital to Analog Converter (DAC)

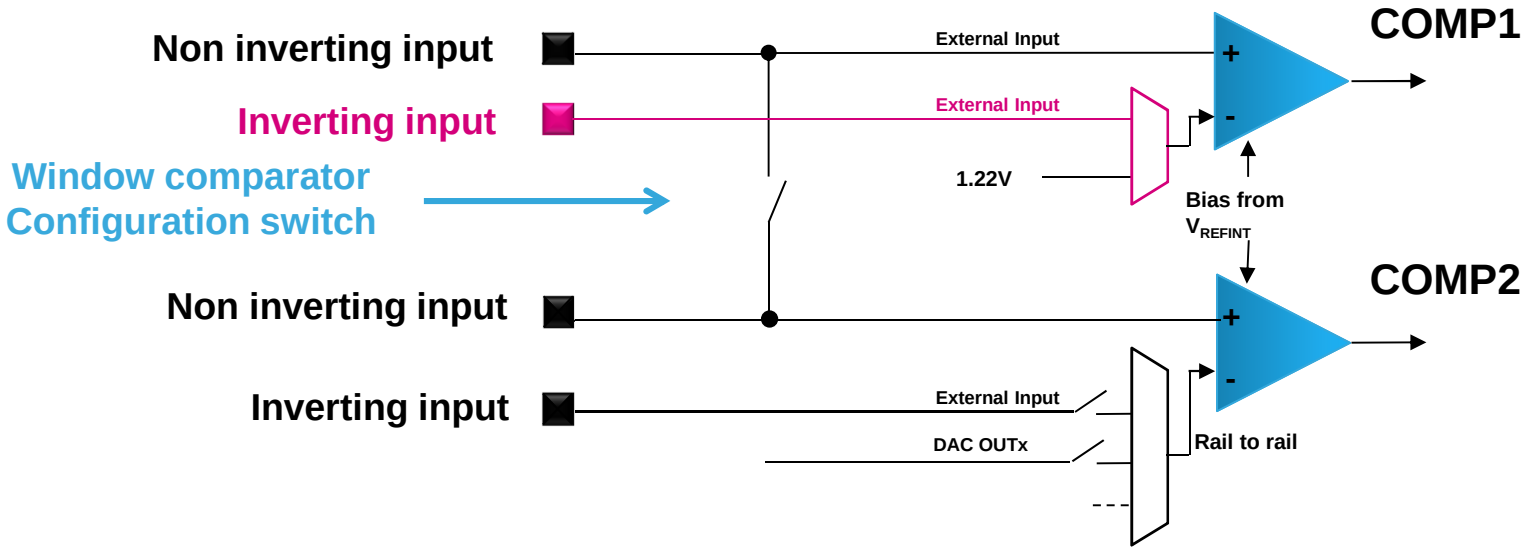
- **One DAC** converter
- **8-bit** or **12-bit** monotonic output (left or right data alignment)
- Independent or simultaneous conversions
- **External triggers for conversion (Timers)**
- Conversion range: 0.5mV (**0.2V**) to VDDA-1LSB (**VDDA-0.2V**)
- **Noise-wave** and **Triangular-wave** generator
- **Integrated buffer** to reduce the output impedance

DMA



Analog Comparators (COMPx)

- two zero-crossing comparators COMP1 and COMP2 sharing the same current bias
- COMP1 with fixed internal reference voltage **/ external** threshold
- COMP2 has Rail-to-Rail inputs with selectable threshold
- Can be combined into a **window comparator**



COMP1 and COMP2 inputs and outputs are available on GPIO

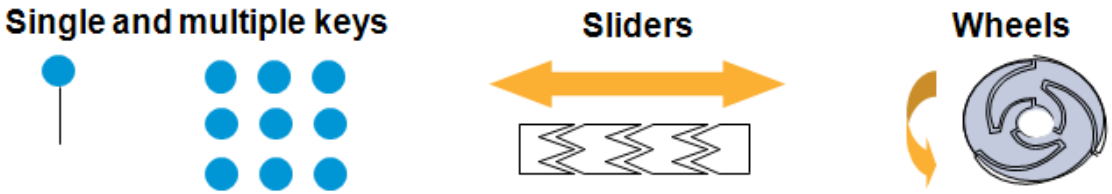
LCD Main features

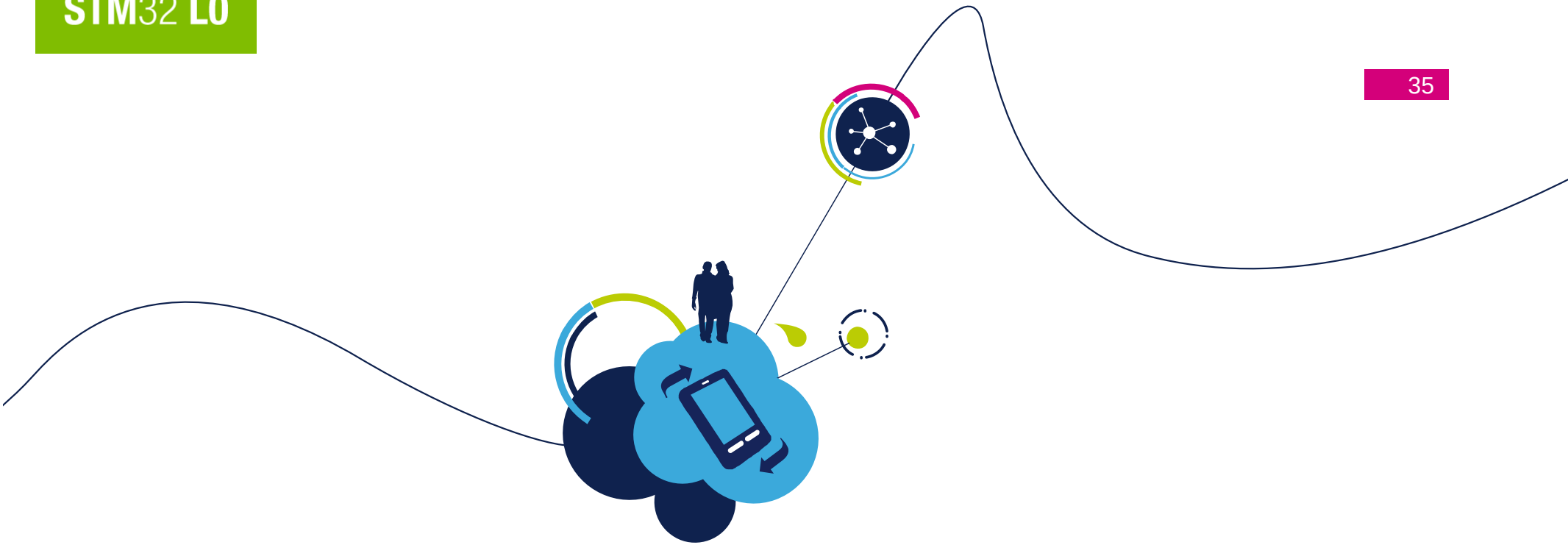
- High Flexibility Frame Rates
- Drive up to **224 (8x28)** or **128 (4x32)** picture elements (pixels)
- Programmable duty and bias
  - Duty: Static, 1/2, 1/3, 1/4, 1/8
  - Bias: Static, 1/2, 1/3, 1/4
- Low Power Waveform to reduce consumption
- External (VLCD) or internal (**STEP-UP**) voltage source
- Double buffer memory
- Contrast Control whatever power supply voltage source
- Blinking programmable pixels and frequency
  - 1, 2, 3, 4, 8 or all pixels at programmable frequency
  - Adjustable blink frequency: 0.5 Hz, 1 Hz, 2 Hz or 4 Hz
- Unused segments and common pins can be used as I/O

Frame ~30 Hz to ~100 Hz

Touch Sensing Controller

- Proven and robust **surface charge transfer acquisition principle**
- **One sampling capacitor for up to 3 capacitive sensing channels** to reduce the system components
- Supports up to **24 capacitive sensing channels** split over **8** analog I/O groups
- Up to **8 capacitive sensing channels can be acquired in parallel** offering a very good response time
  - 1 counter per analog I/O group to store the current acquisition result
- **Full hardware management** of the charge transfer acquisition sequence
  - No CPU load during acquisition
- **Spread spectrum** feature to improve system robustness in noisy environments (minimum step of 20.8ns)





Timers

Real-Time Clocks (RTC)

- Daylight saving compensation programmable by software
- (Smooth – 0.954ppm resolution)
- The RTC clock source can be any of the following:
  - LSE oscillator clock
  - LSI oscillator clock
  - HSE 1MHz max (HSE divided by ***/32*** in clock controller).
- Up to **2** TAMPERs

General Purporse Timers

- **3 x 16-bit timer**
  - with autoreload
  - 8 CAPCOM units

Output Compare / Input Capture

PWM Output / Input

One Pulse Mode

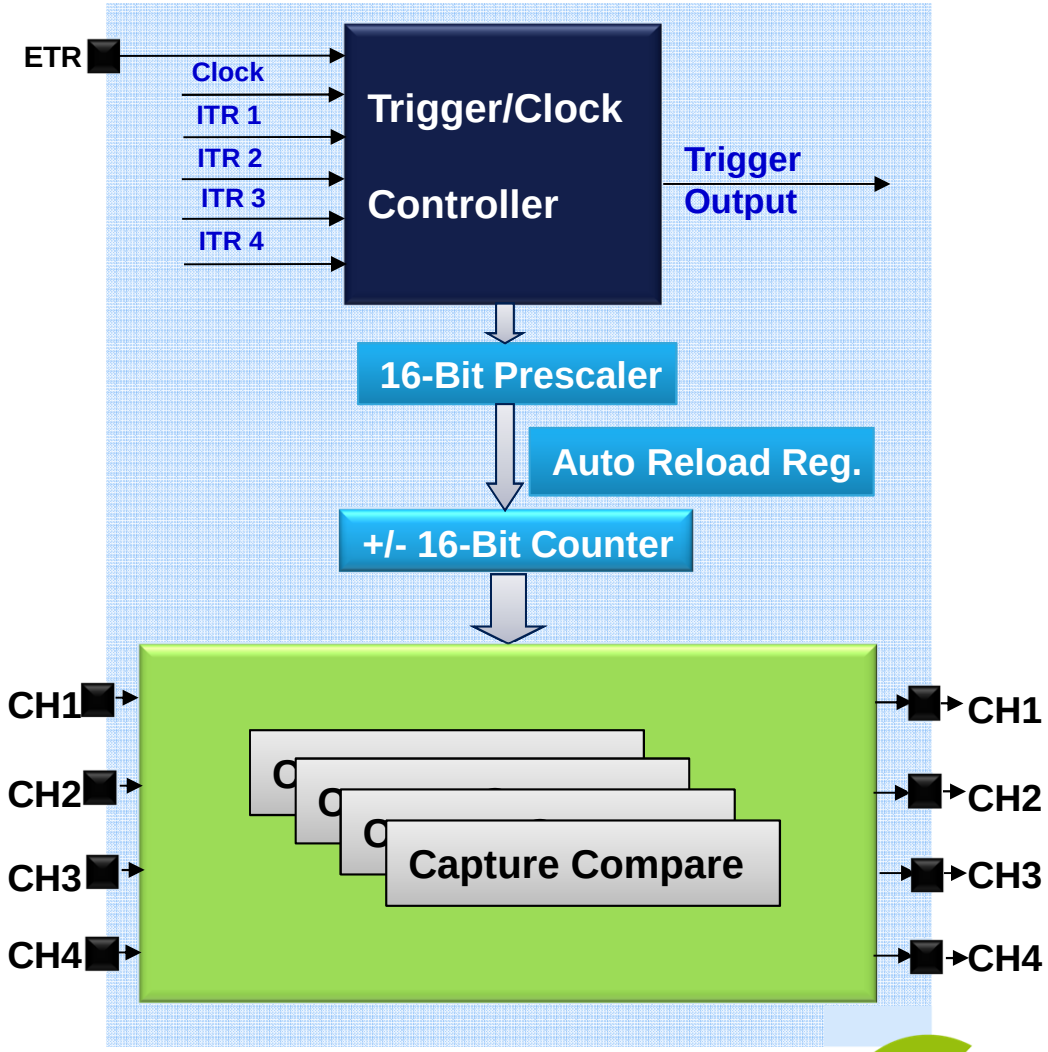
Encoder interface

Synchronization

(Master/Slave, with external trigger)

- **1 x 16-bit basic timer**
  - with autoreload (DAC support)

- **1 x 16-bit Low-Power timer** (LPTIM)



Low-Power Timer

- Asynchronous running capability

- Ultra low power-consumption

- Timeout function for wakeup from low power modes

Communication Peripherals

USART

2x USARTs + 1x LPUART

Frame	• **7**, 8, 9 DATA bits • **0.5**, 1, 1.5, 2 STOP bits • Even, odd, none PARITY • Oversampling /8 and /16 (default)
Modes	• Asynchronous ▪ LIN ▪ SmartCard (T=0**, T=1**) ▪ IrDA SIR ENDEC ▪ Multiprocessor communication ▪ Half duplex ▪ **Basic MODBUS** • Synchronous (CLK line)
Other	• DMA support • HW flow control (RTS, CTS lines) • Programmable data order (MSB/LSB) • **Wake-Up from STOP mode** • **Swappable Tx/Rx pin, Driver Enable (for RS-485)**

4Mbps

DMA

I²C

- **I²C Version 3.0** compatibility
- Standard-Mode, Fast-Mode (up to **400 kHz**), **Fast-Mode+** (up to **1 MHz**)
- Slave and master modes with multi-master capability
- 7-bit and 10-bit addressing mode, dual addressing capability
- Programmable timing, **optional clock stretching**
- **Easy to use event management,** 1-byte buffer with DMA capability
- SMBus ver. 2.0 and PMBus ver 1.1 standards compatibility
- Programmable analog and digital **noise filters**
- **Wakeup from STOP** on address match

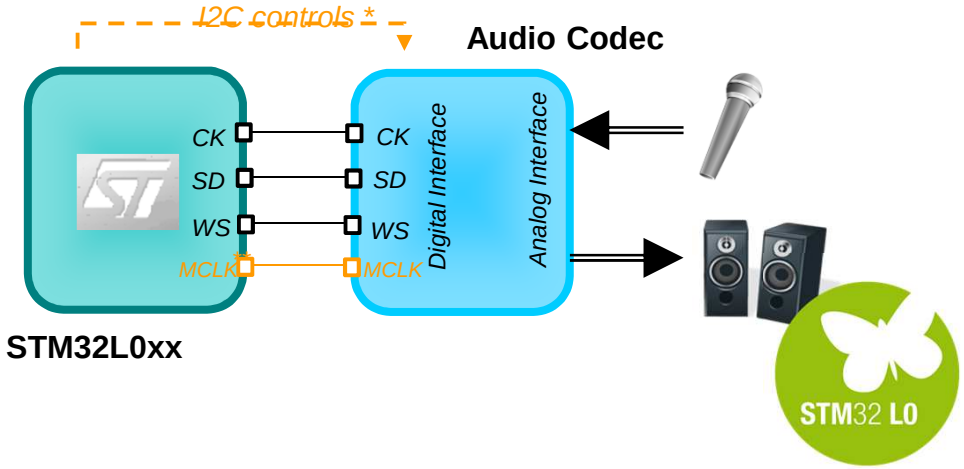
DMA

SPI / I²S

- **SPI**
  - Speed up to **16 MHz bitrate**
  - Full-duplex (3 wires), half-duplex (2 wires) or simplex synchronous transfers (2 wires, unidirectional data line)
  - **8-bit or 16-bit** data size selection
  - MASTER or SLAVE operation, Multi-master mode capability
  - **NSS management** by HW or SW for both MASTER and SLAVE modes
  - **CRC calculation** and check for reliable communication

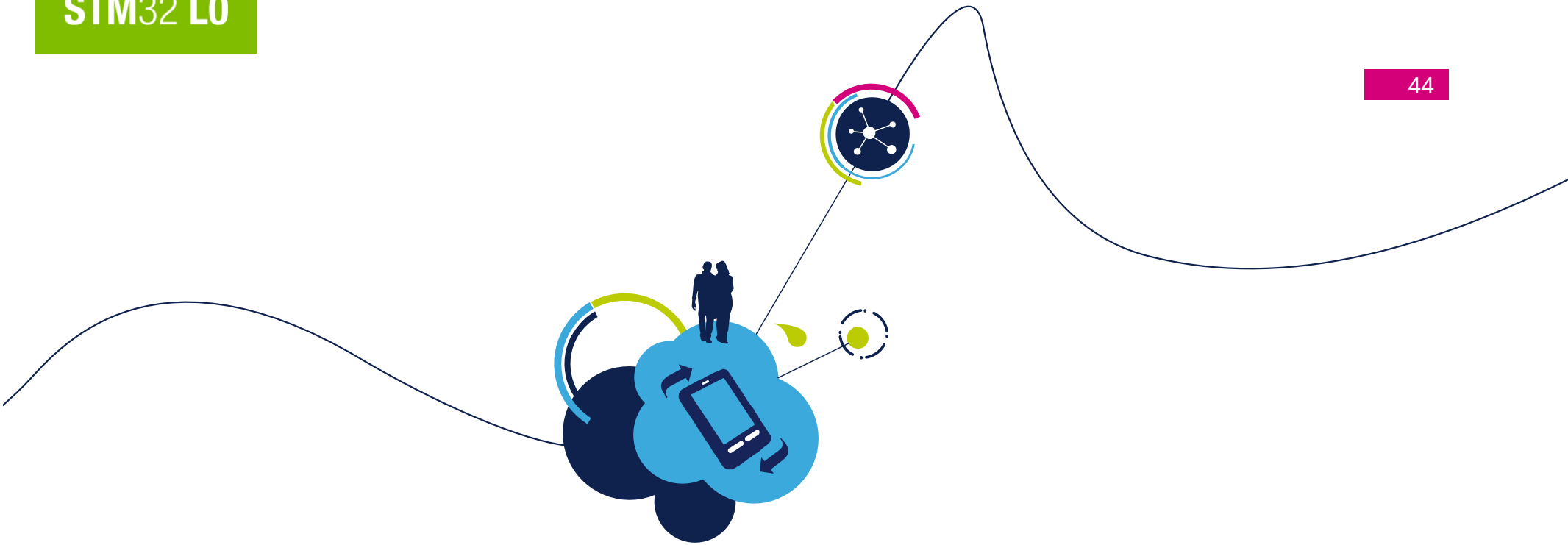
- **I²S**
  - **Up to 192kHz, 32-bit**
  - I²S Philips
  - Left-Justified / Right-Justified
  - PCM standard

DMA



USB Interface

- ***Crystal-less**** USB 2.0 FS interface (12Mbit/s) with D+/D- resistors
  - * Integrated on-chip 48 MHz oscillator with clock recovery system (CRS)
    No external resonator/crystal needed (cost saving is in range of 0.10$)
- **Link Power Management** (LPM) and **Battery Charger Detection** (BCD) V1.2 compliant
- USB FS Device Library with intuitive USB device class drivers API
  - Examples and demo based on a set of 6 classes (Audio, CCID, CDC, HID, VCP, MSC)
  - Easy development of applications using USB full speed transfer types (control, interrupt, bulk and isochronous)
- Device Firmware Upgrade on the field over USB (internal bootloader)
- USB VID/PID sublicensing service for free



Security peripherals

Safety and Security features Summary

- True EEPROM embedded → guaranteed robustness:
  - **Derived from Automotive**
  - ECC for Flash, EEPROM and Backup registers
  - Working temp **-40°C to 105°C**
  - Cycling: 10K on Flash / 300K on EEPROM (each block of 128-bit)
  - Data retention: 30 years at 85°C / **10 years at 105°C**
  - Flash operation and programming capability down to 1.65V
- Read-Out Protection: SWD fuse memory protection
- Sector protection (4kb): Read (**PcROP**) or Write
- Firewall internal memory interface to secure selected Code/Data
- MPU, privilege/unprivilege modes, Watchdogs, Registers locking
- Clock Security System (CSS) for both HSE and LSE, if enabled:
  - In case of HSE failure **Clock Security System (CSS)** will switch to MSI
  - In case of LSE failure, wakeup from low-power mode is generated (Interrupt request can be generated)

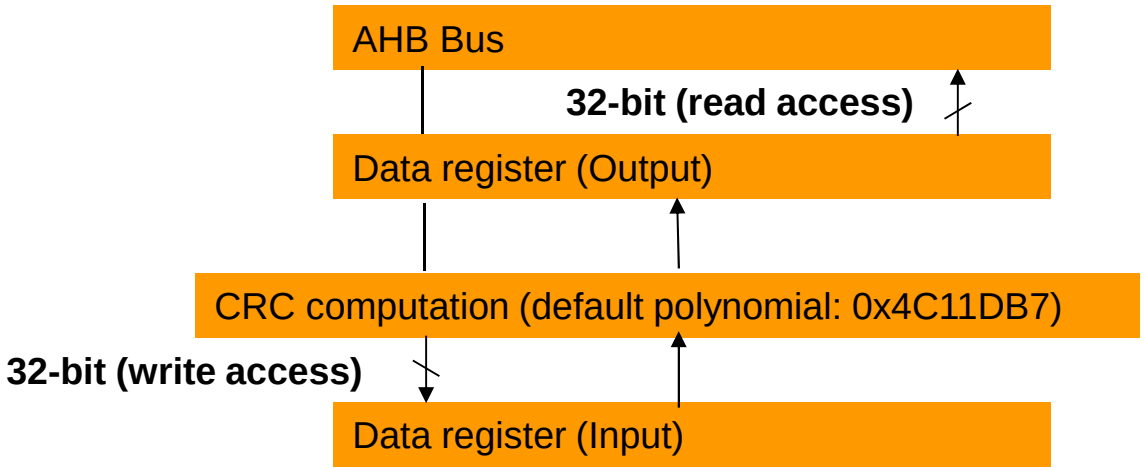
CRC – Features

CRC-based techniques are used to verify data transmission or storage integrity

- **Fully programmable polynomial with programmable size (7, 8, 16, 32 bits)**
- Alternatively, you can use default CRC-32 (Ethernet) polynomial: 0x4C11DB7

$$X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^{8} + X^{7} + X^{5} + X^{4} + X^{2} + X + 1$$

- Single input/output 32-bit data register (**8, 16 or** 32-bit data)
- CRC computation done in 4 AHB clock cycles (HCLK) **(for 32-bit data)**
- General-purpose 8-bit register (can be used for temporary storage)



Advanced Encryption AES

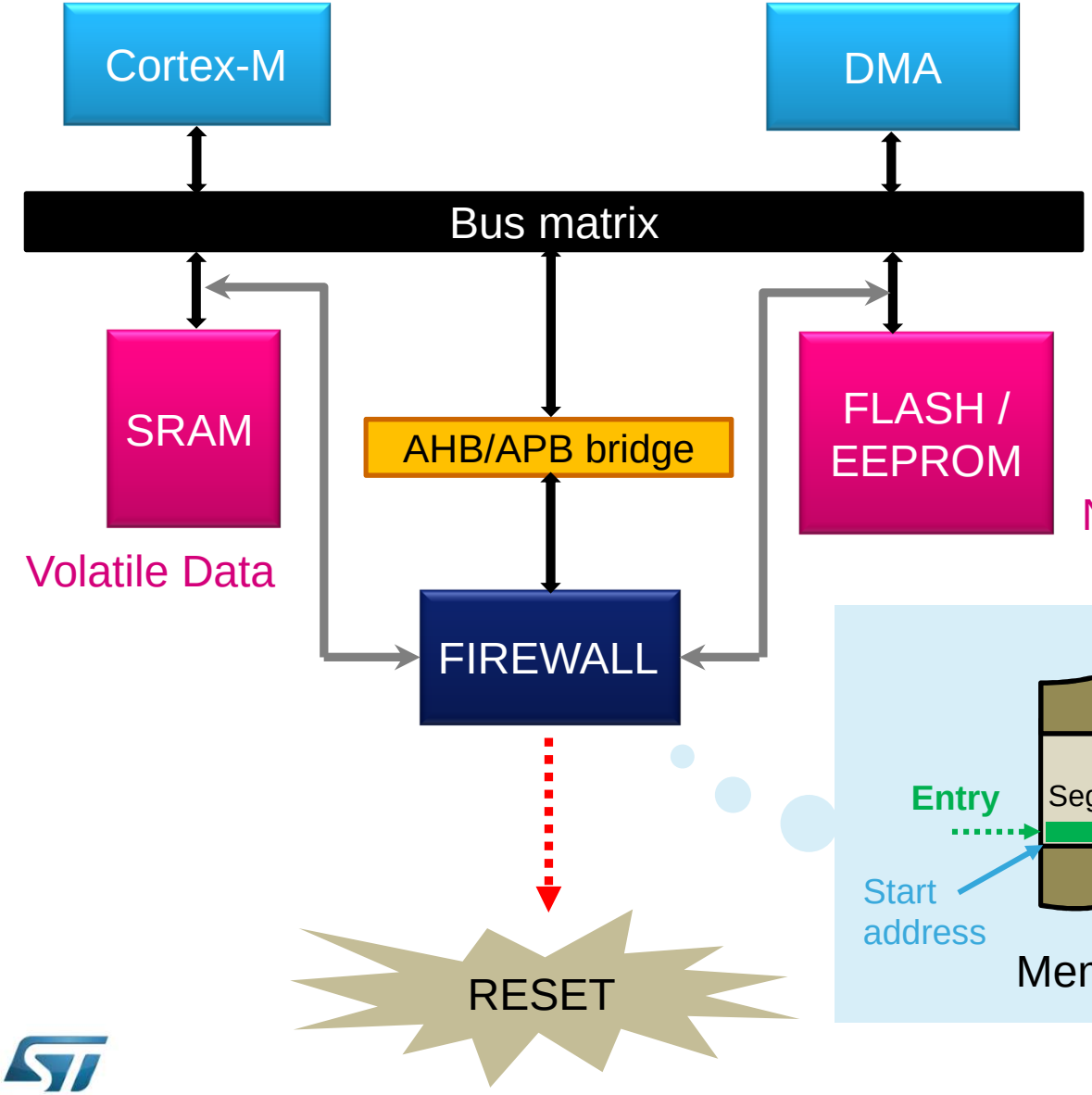
- Supports encryption and decryption using:
  - 128-bit key
  - 128-bit data blocks
- Supported modes:
  - Electronic CodeBook mode (ECB) – default
  - Cipher block chaining (CBC)
  - Counter (CTR) mode
- Dedicated 2 DMA channels:
  - AES_IN – channel 1 or channel 5
  - AES_OUT – channel 2 or channel 3

213 clock cycles for one 128-bit data block

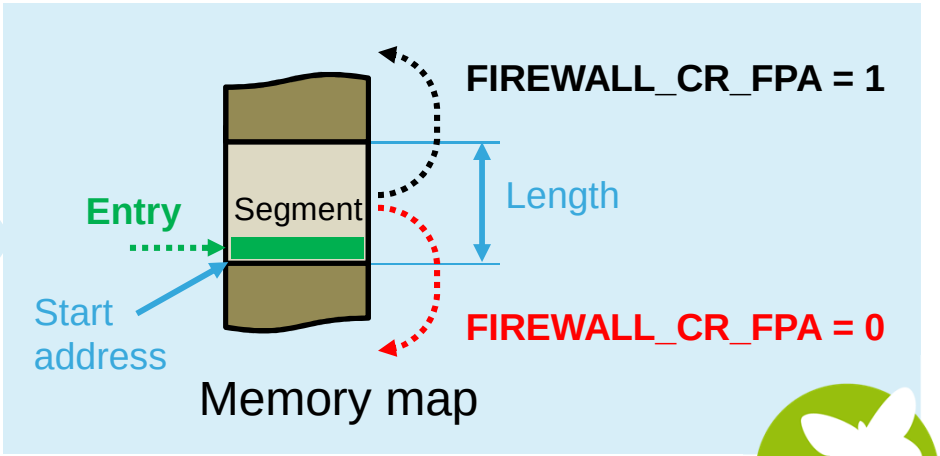
Random Number Generator (RNG)

- Based on a continuous analog noise (True RNG)
- Generates 32-bit random numbers
- Clocked by a dedicated clock (PLL48CLK or **HSI48**)
- 40 periods of the clock signal between two consecutive random numbers
- Can be disabled to reduce power-consumption
- Provide a success ratio of more than 85% to FIPS 140-2 (Federal Information Processing Standards Publication 140-2) tests for a sequence of 20 000 bits

Firewall



- To secure selected region of memory against Read-Out
- Separation of the Main application and some TOP security library/stack



Thank you

www.st.com/stm32l0