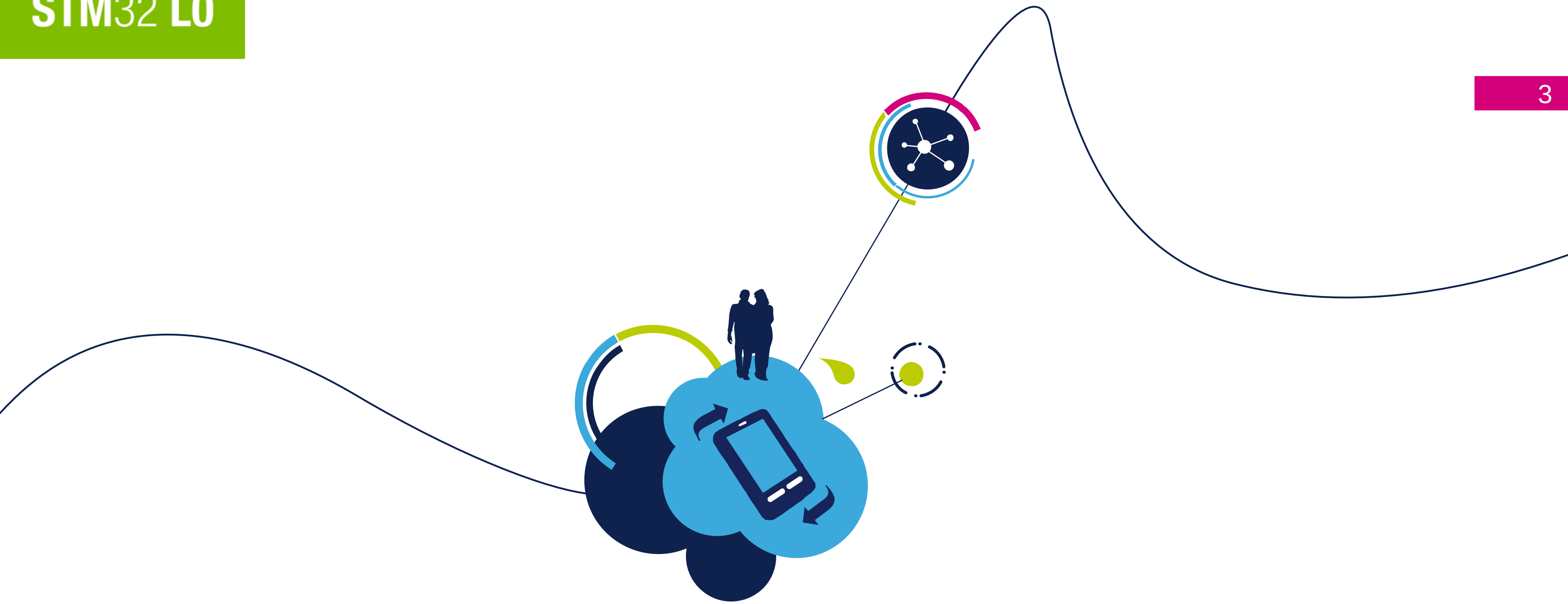
STM32L0 Low-Power Modes

OBJECTIVES

- Remind ARM Cortex-Mx Low-Power modes architecture
- Introduce STM32L0 Low-Power modes implementation
- Discuss the differences and possible application
  - Check the consumption (exercises)

After this part of workshop you will know the differences and capabilities of STM32L0 Low-Power modes.



Low-Power Modes

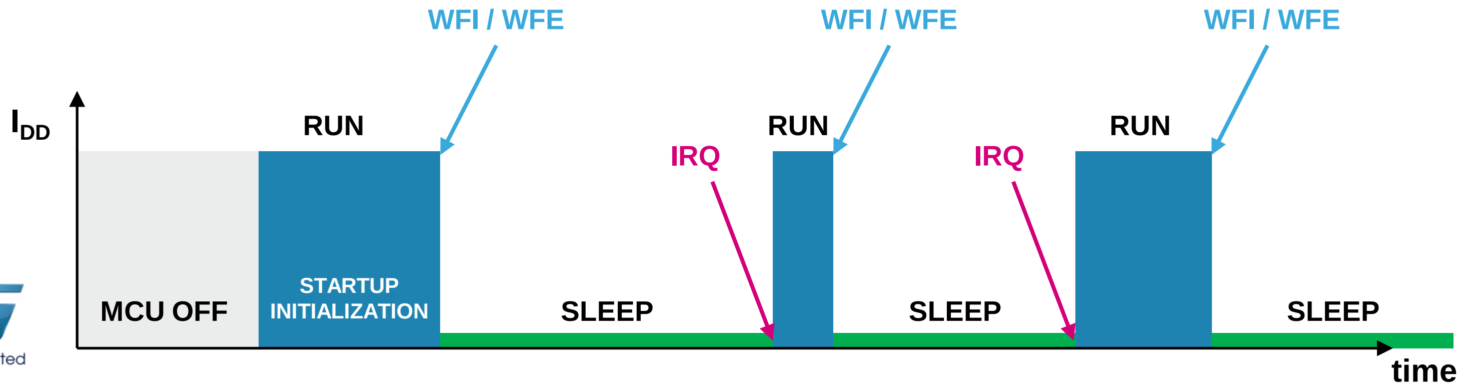
ARM Cortex-Mx Support

ARM Cortex-Mx Low Power Features

- Cortex-Mx low-power modes:
  - **SLEEP**
    - Stops only processor clocks → higher power consumption, but the shortest wakeup time
  - **DEEP SLEEP**
    - System clock stopped, PLL and FLASH turned off, SRAM and registers are still powered
- Entry mechanisms:
  - **WFI** (Wait for Interrupt) instruction
  - **WFE** (Wait for Event) instruction
  - **SLEEP ON EXIT**

SLEEP NOW !

- **MCU enters SLEEP mode as soon as the WFI / WFE instruction is executed**
- **WFI**
  - Exit: any peripheral interrupt, which vector is enabled in the Nested Vectored Interrupt Controller (NVIC)
  - Wake-up: Additional Interrupt entry latency
- **WFE**
  - Exit: An *event* can be an interrupt enabled in the peripheral control register but NOT in the NVIC or an EXTI line configured in event mode
  - Wake-up: No time wasted in interrupt entry/exit

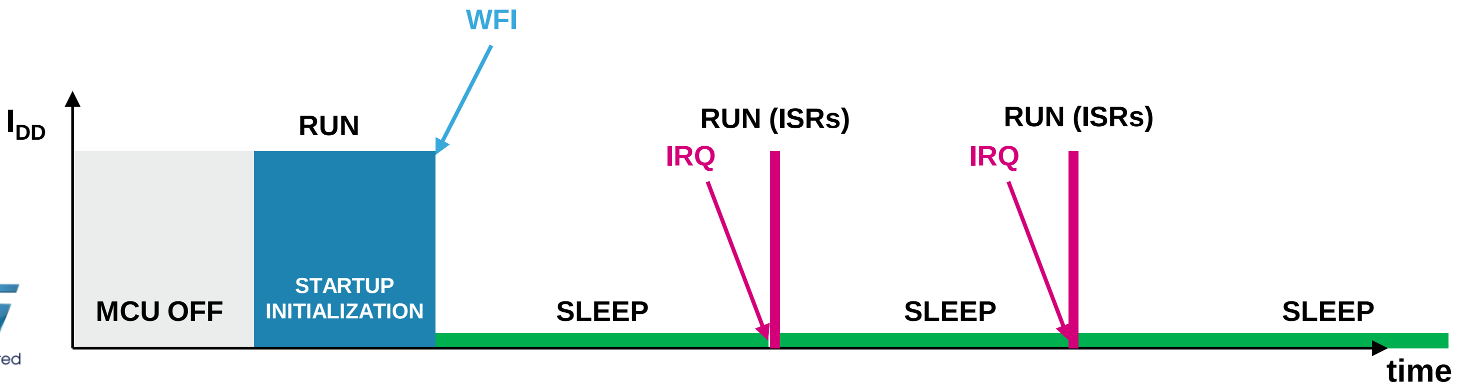


SLEEP ON EXIT !

- **MCU enters SLEEP or DEEP SLEEP mode as soon as it exits the lowest priority interrupt service routine (ISR)**

- Controlled by SLEEPONEXIT bit in Cortex-Mx System Control Register of System Control Block (SCB_SCR)

- Exit: any peripheral interrupt acknowledged by the Nested Vectored Interrupt Controller (NVIC)
- Wake-up: The processor state is not un-stacked → faster wake up on interrupt



Cortex Low-Power Leadership from ARM

life.augmented

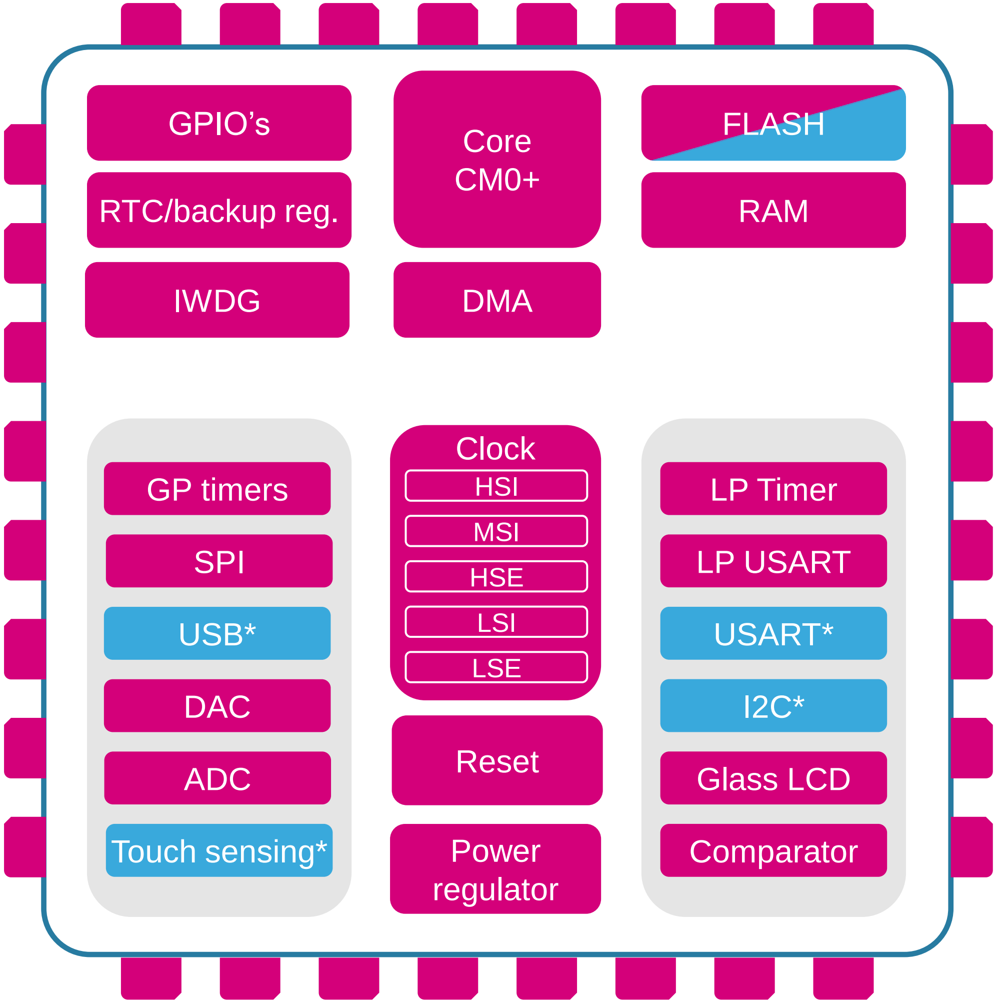
Low-Power Modes

STM32L0 implementation

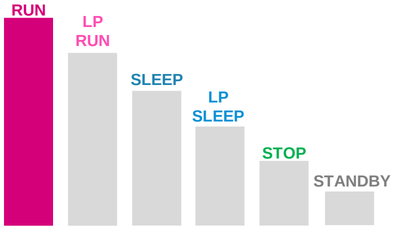
System Operation Modes

- **RUN**
  - clocks gating, FLASH can be put in power-down mode
- **SLEEP**: CPU stopped
  - optional automatic clocks gating, FLASH can be put in power-down mode
- **LP RUN**: power regulator in LP mode, 131kHz Max
  - clocks gating, FLASH can be put in power-down mode
- **LP SLEEP**: CPU stopped + power regulator in LP mode, 131kHz Max
  - optional automatic clocks gating , FLASH can be put in power-down mode
- **STOP**: power regulator in LP mode + all clock stopped except LSI (LSE)
  - **HSI can be kept running, FLASH can be kept in power-down mode after Wake-Up**
  - **LPTIM, LPUART and some other peripherals are available in STOP mode**
- **STANDBY**: V_{Core} switched OFF

RUN Mode

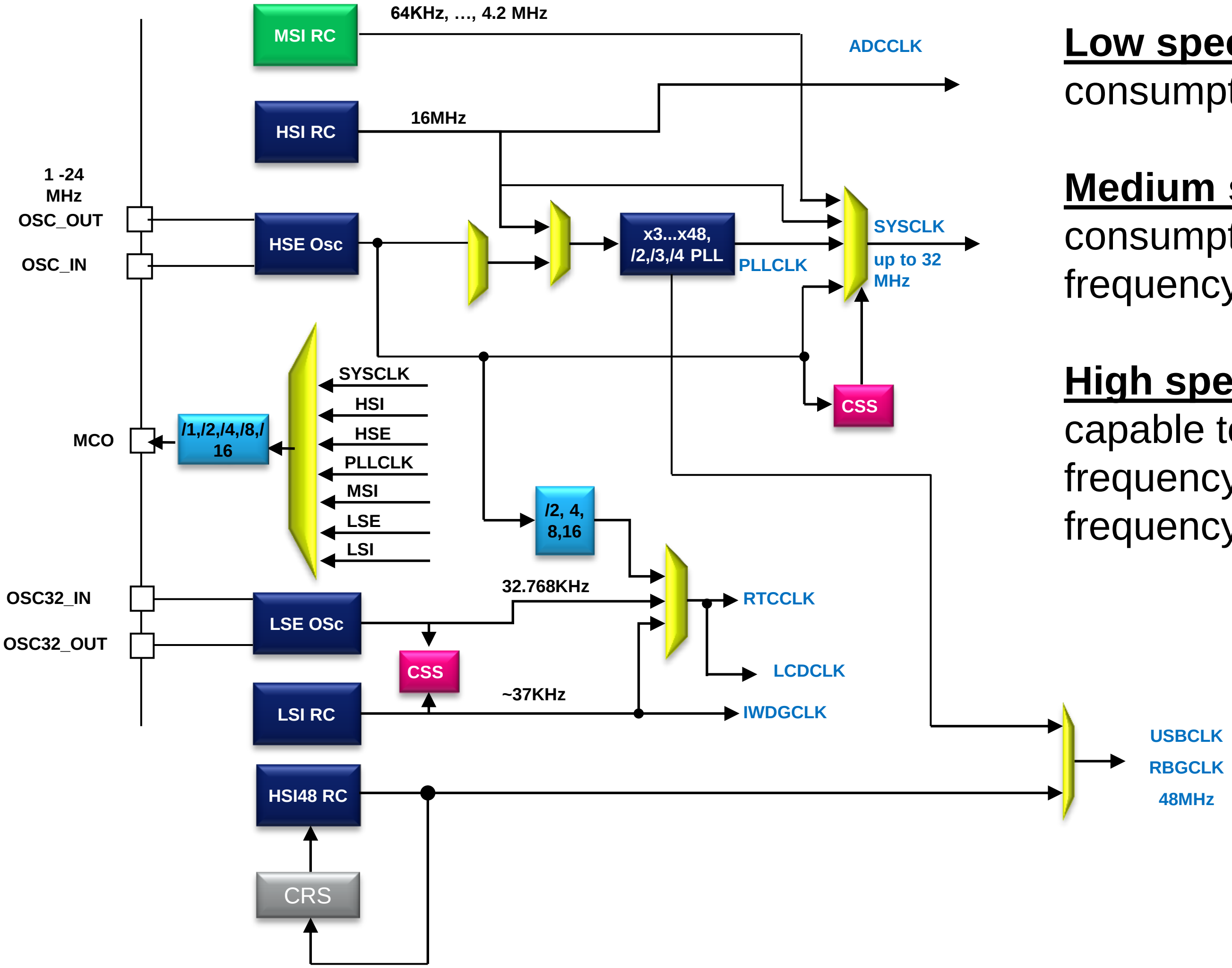


- **Everything can be ON**
- **Any peripheral clocks can be gated (* examples)**
- **After reset peripherals clock are disabled**
- **Code can RUN from RAM and FLASH be OFF**



STM32L0 Clock Subsystem

Clock controller: the “Gearbox”



Low speed path: for low consumption while CPU is static

Medium speed path: for low consumption while CPU is at low frequency

High speed path: for performance capable to scale to the desired level frequency (PLL)- change on the fly of frequency

Clock Sources Parameters

Clock Source	Frequency	Conso	Precision 25°C/0-85°C	Wakeup time
➡ MSI (default)	65kHz-4.2MHz (2.1MHz default)	0.7-15µA	±0.5% / ±3%	3.5 µs (Vcore Range 1/2)
HSI	16MHz	100µA	±1% / ±2.5%	3.7 µs
HSE external crystal	1-24MHz	~500µA	~±0.01% (100ppm)	1ms
HSE external clock	1-32MHz		N/A	N/A
PLL	2-32MHz	~350µA	N/A	100µs (2MHz input)
LSI	37kHz	0.4µA	50%	200µs
LSE external crystal	32.768kHz (typically)	~0.3µA	~0.002% (20ppm)	~1s
LSE external clock	1-1000kHz		N/A	N/A

+ Clock Source Wake-Up time

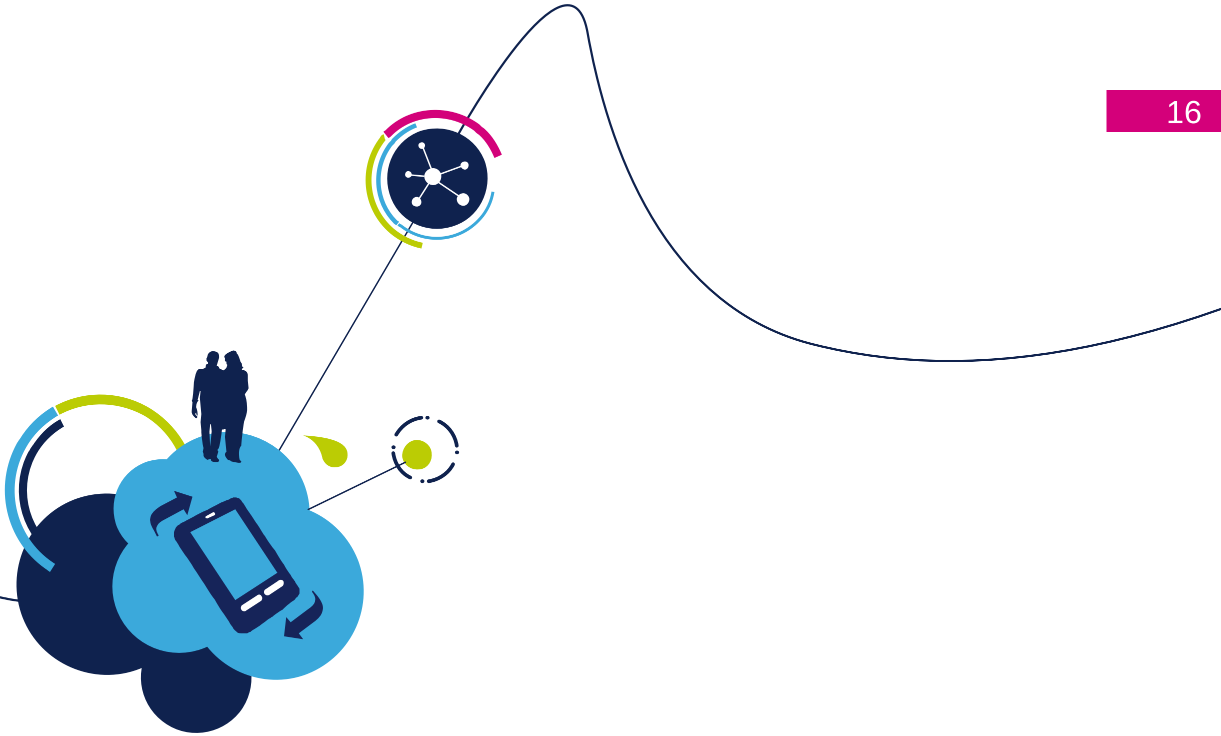
LSE usually woken-up only once after power-on

Peripheral clocks gating

- The clock tree toward each register increase consumption
- So clock toward each peripheral register group can be gated (Default is gated)
- Consumption (**L1 data**):

(L0 data coming soon)

	range 1	range 2	range 3	LP Sleep and run	
Condition:	32MHz	16MHz	4MHz	65kHz	
GPIOA	7	6	5	6	
GPIOB	7	6	5	6	
CRC	0.5	0.5	0.5	1	
DMA1	18	15	13	18	
SYSCFG & RI	3	2	2	3	µA/MHZ
TIM2	13	11	9	11	
TIM6	4	4	4	4	
LCD	4	3	3	4	
WWDG	3	2.5	2.5	3	
USB	15	7	7	7	
PWR	3	3	3	3	
DAC	6	5	4.5	5	
………..					
ALL	279	221	219	215	



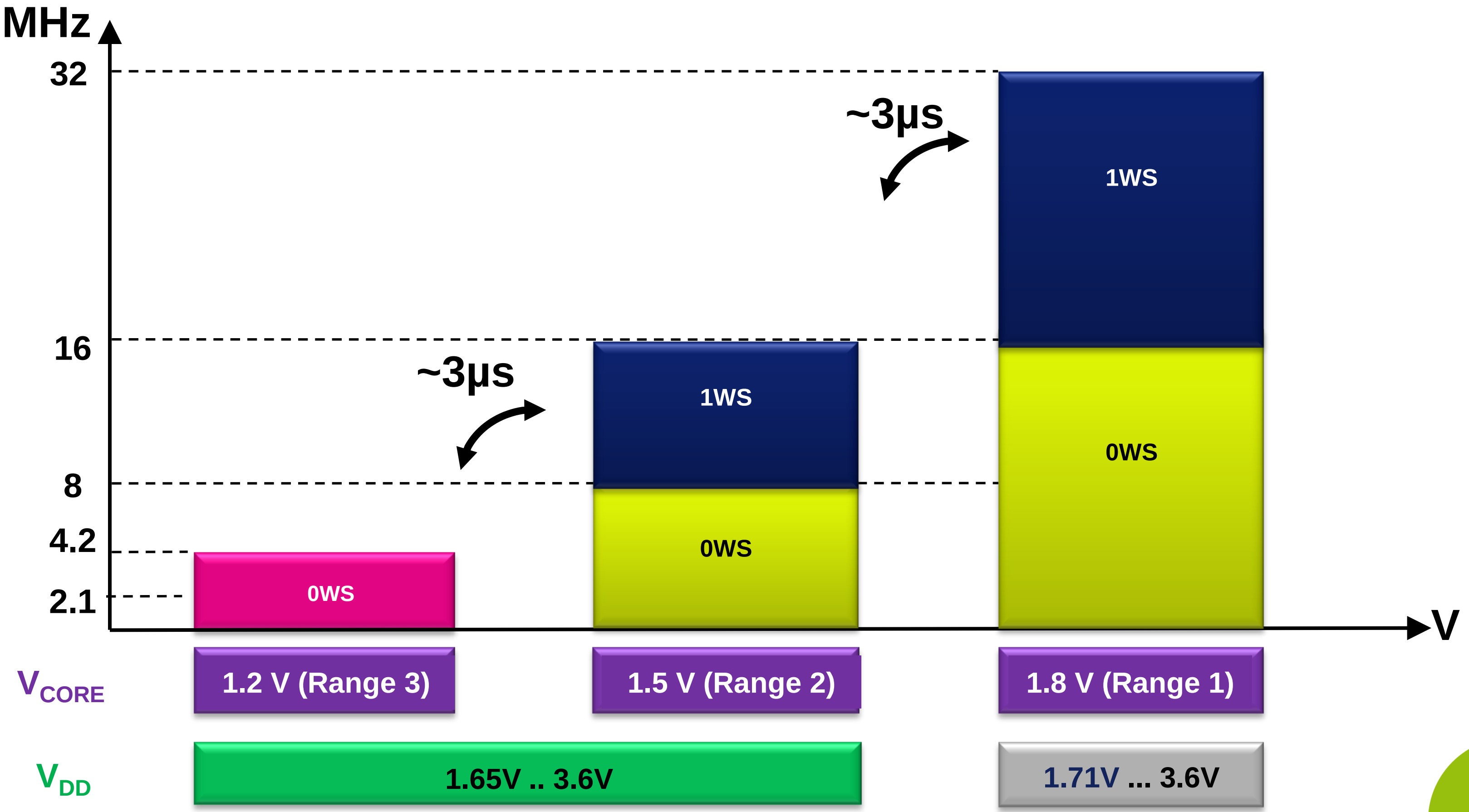
STM32L0 Power Supply Subsystem

Voltage Regulator

- The Voltage regulator has three different modes
  - Main voltage regulator mode (MVR)
    - – no output current limitation
    - – consumption of the regulator is **8µA** (typ)
  - Ultra low power regulator mode
    - – output current limited
    - – consumption of the regulator is **0.1µA** (typ)
  - Regulator OFF mode
    - – no output current
  - Time for the regulator to **wakeup is 3.5µs (max)**

V_{CORE} Dynamic Voltage Scaling

- Voltage scaling optimizes the product efficiency (Consumption vs Performance)
- User selects a Range (voltage scaling) according to :
  - External V_{DD}
  - DMIPS performance required (=w/ or w/o Wait State)
  - Max power consumption



Can go down to 1.71V !

Functionality dependent on Power Supply

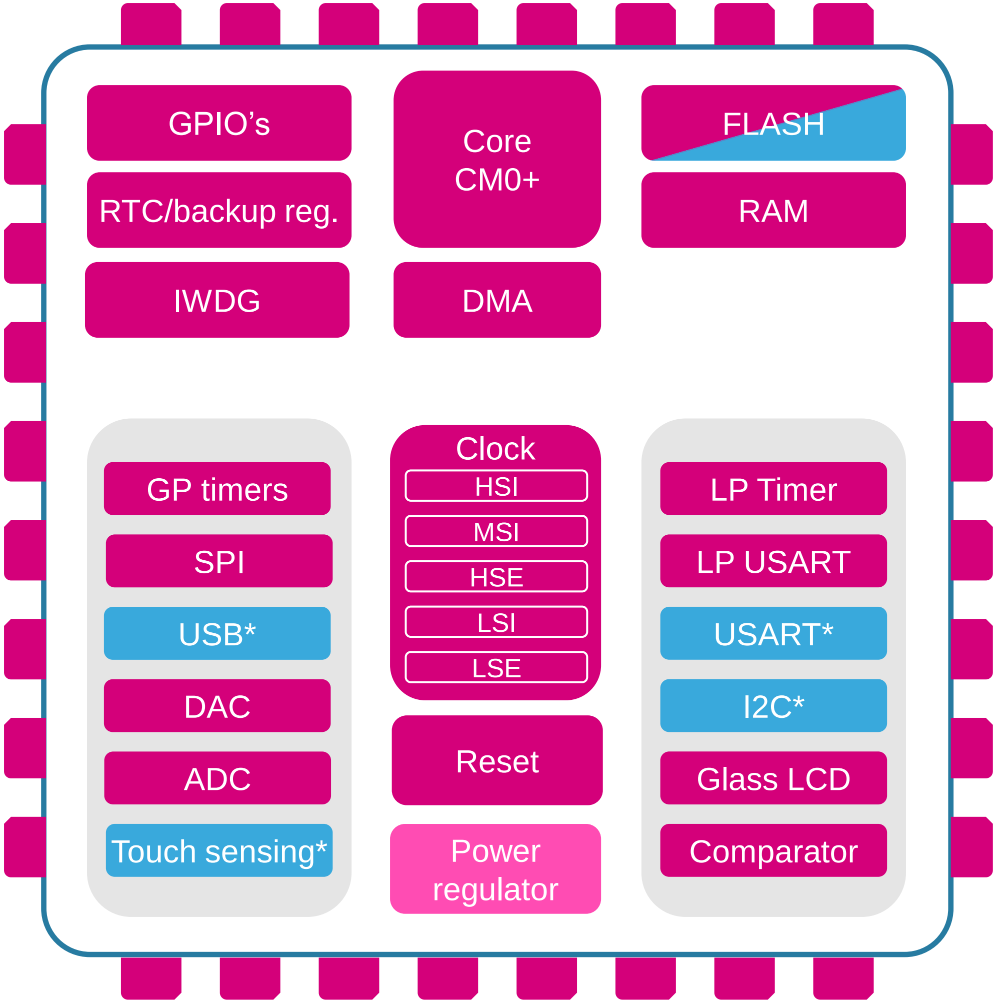
1.8V min still at Power-On-Reset

Functionality dependent on the operating power supply range				
V_{DD} = V_{DDA}	V_{Core}	f_{CPU} (max. CPU frequency)	ADC (max. sampling rate)	USB
1.65~1.71V	**Range 3 – 1.2V**	**4.2MHz (0 WS)**	N/A	N/A
	Range 2 – 1.5V	16MHz (1 WS) 8MHz (0 WS)		
1.71~1.8V	**(Range 3 – 1.2V)*** (Range 2 – 1.5V)* **Range 1 – 1.8V**	**32MHz (1ws)** **16MHz (0ws)**	N/A	**Functional****
1.8~2.0V			0.5 MSPS	
2.0~2.4V			0.5 MSPS	
2.4~3.6V			1 MSPS	

*** For V_{Core} Range 2 and Range 3, the lower CPU frequency limits apply always**

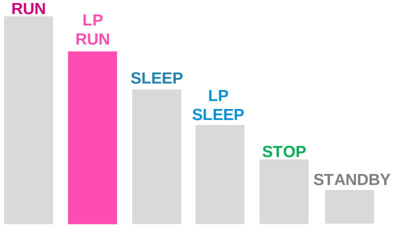
**** USB transceiver requires external VDD_USB>=3V**

LP RUN Mode



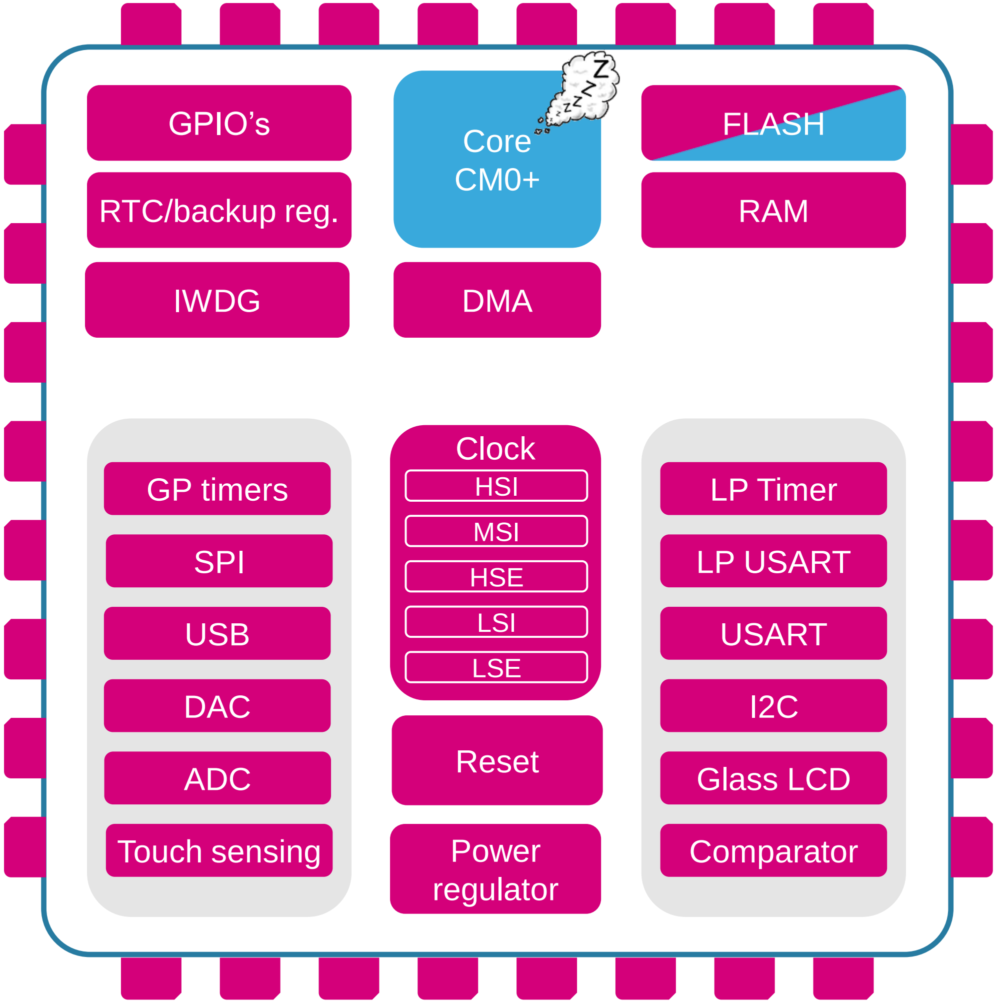
- Everything can be ON
- Any peripheral clocks can be gated (* examples)
- After reset peripherals clock are disabled

- **Power regulator switched to Low Power mode**

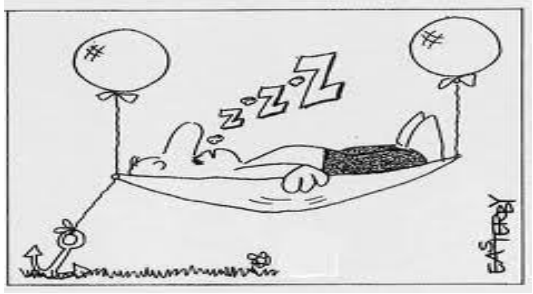


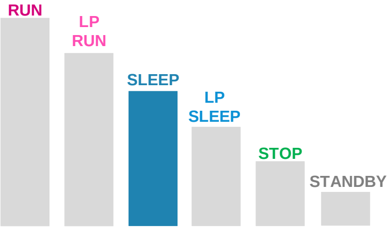
life.augmented

SLEEP Mode

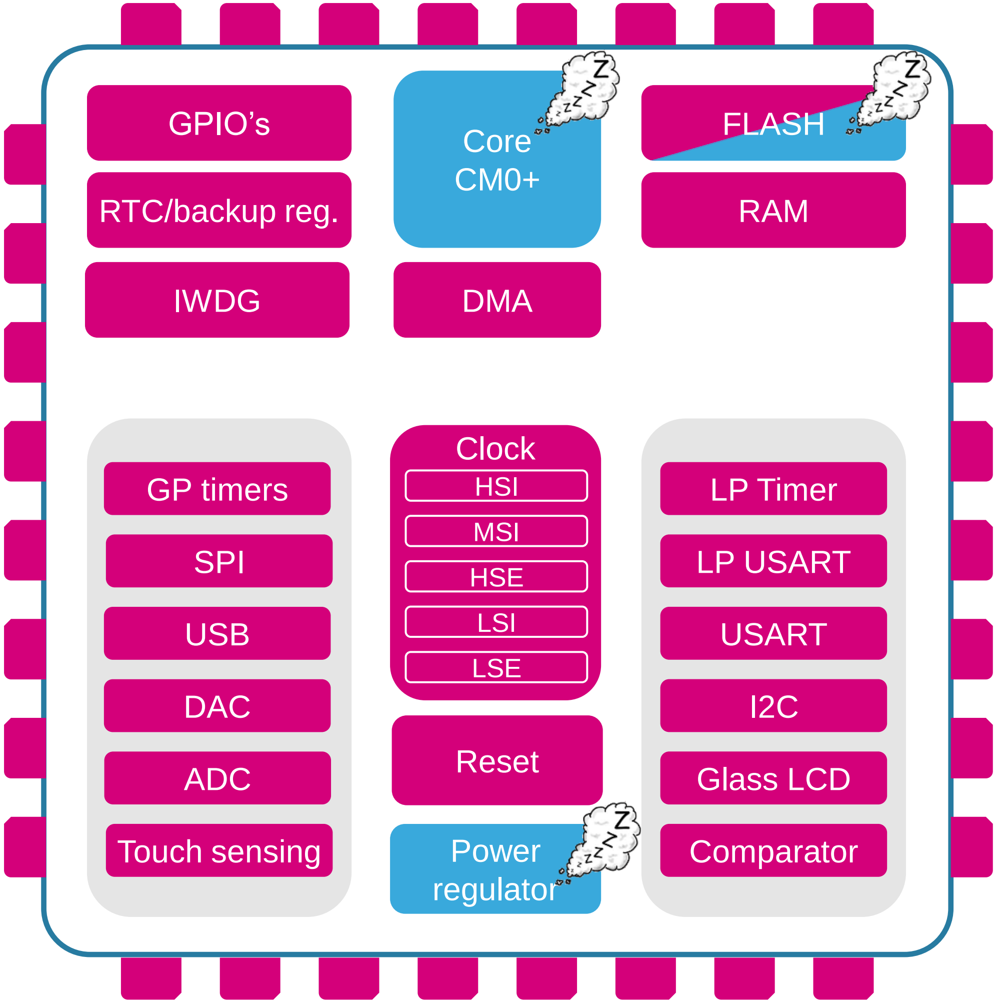


- **Core is stopped**
- **Peripherals are running**

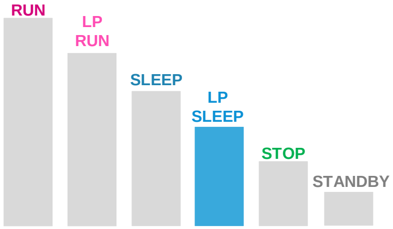




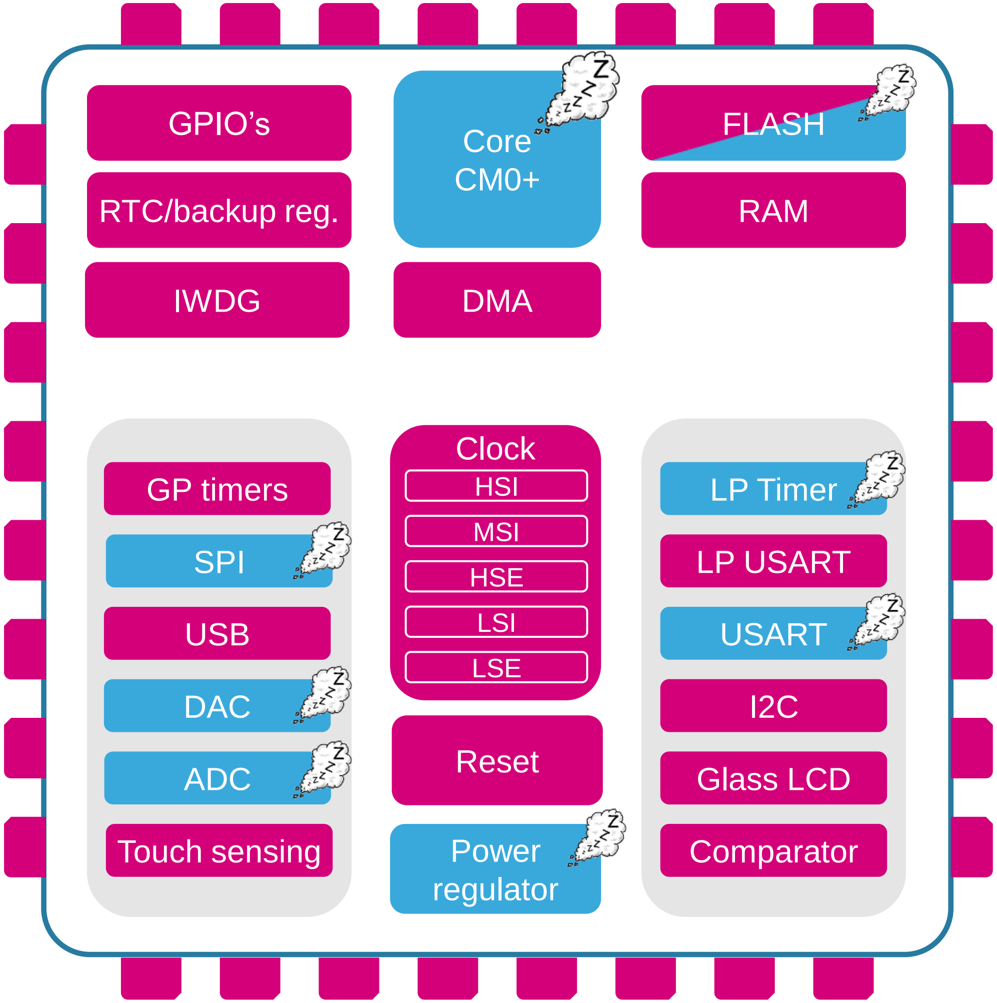
LP SLEEP Mode



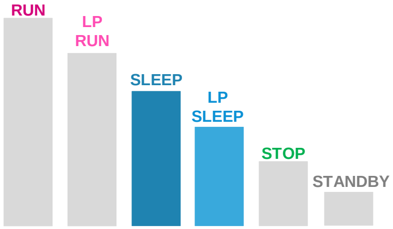
- Core is stopped
- Peripherals are running
- **Power regulator is in Low Power mode**
- **FLASH can be in Power Down mode**
- **VREFINT can be OFF**



SLEEP Modes

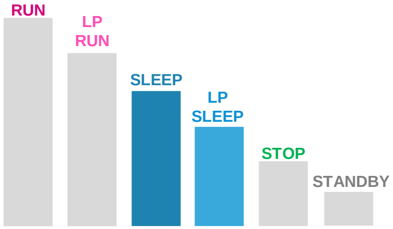


- Core is stopped
- Peripherals are running
- Power regulator is in Low Power mode
- FLASH can be in Power Down mode
- VREFINT can be OFF
- **Peripherals can be gated automatically when entering SLEEP mode**

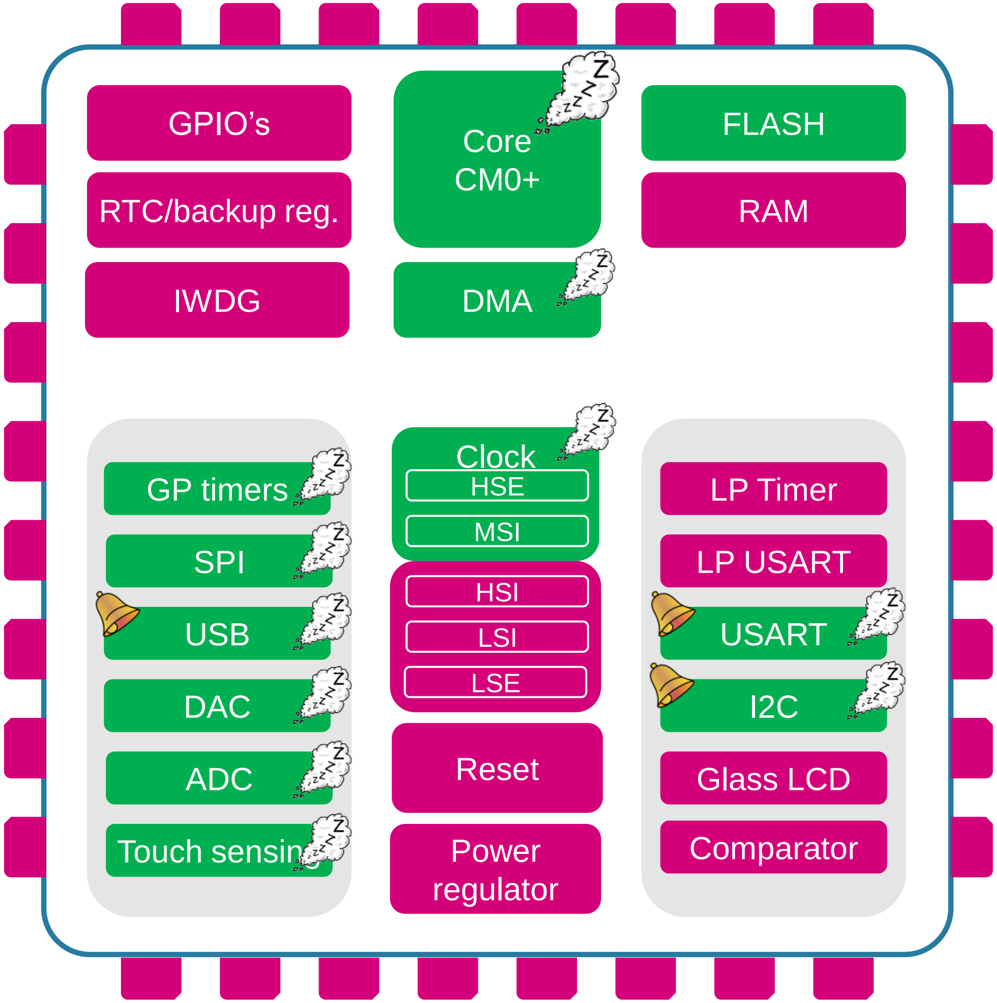


SLEEP Modes

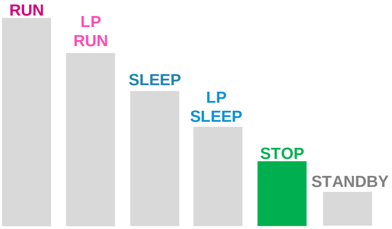
- **Entry & Exit**
  - **WFI** (Wait For Interrupt)
    - Exit: any peripheral interrupt acknowledged by the Nested Vectored Interrupt Controller (NVIC)
  - **WFE** (Wait For Event)
    - Exit: as soon as the event occurs ➔ No time wasted in interrupt entry/exit
- **Two entry options** for SLEEP mode
  - **Sleep Now**: MCU enters SLEEP mode as soon as WFI/WFE instruction are executed
  - **Sleep on Exit**: MCU enters SLEEP mode as soon as it exits the lowest priority ISR



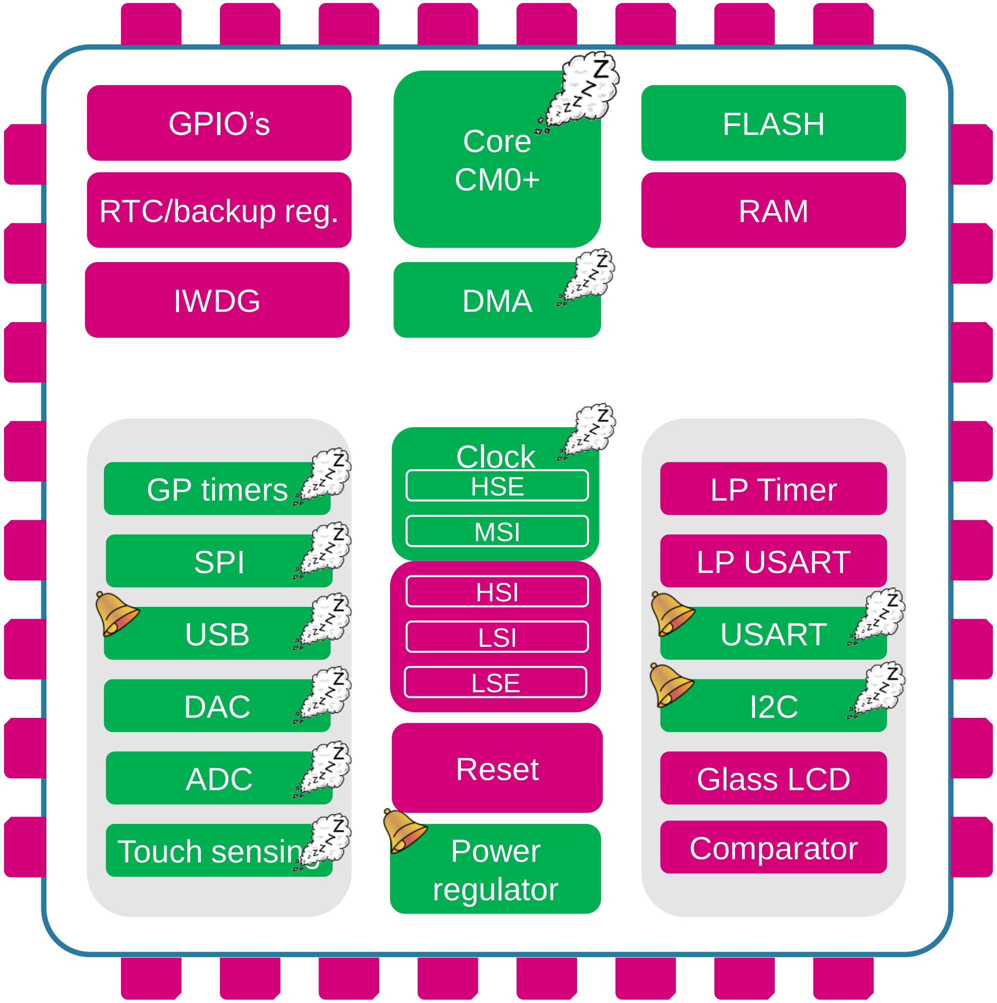
STOP Mode



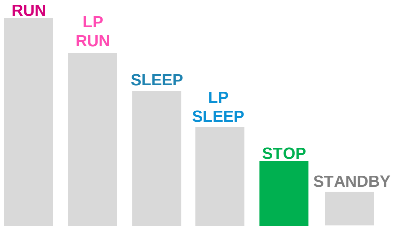
- Core is stopped
- HSE, MSI clocks are OFF
- SRAM and registers content is preserved
- Peripherals with HSI, LSI, LSE clock option can be ON
- GPIO's keep their setup



STOP Mode



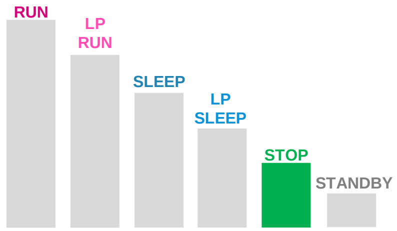
- Core is stopped
- HSE, MSI clocks are OFF
- SRAM and registers content is preserved
- Peripherals with HSI, LSI, LSE clock option can be ON
- GPIO's keep their setup
- **Power regulator is in Low Power mode**
- **FLASH is in Power Down mode**
- **VREFINT, BOR can be OFF**



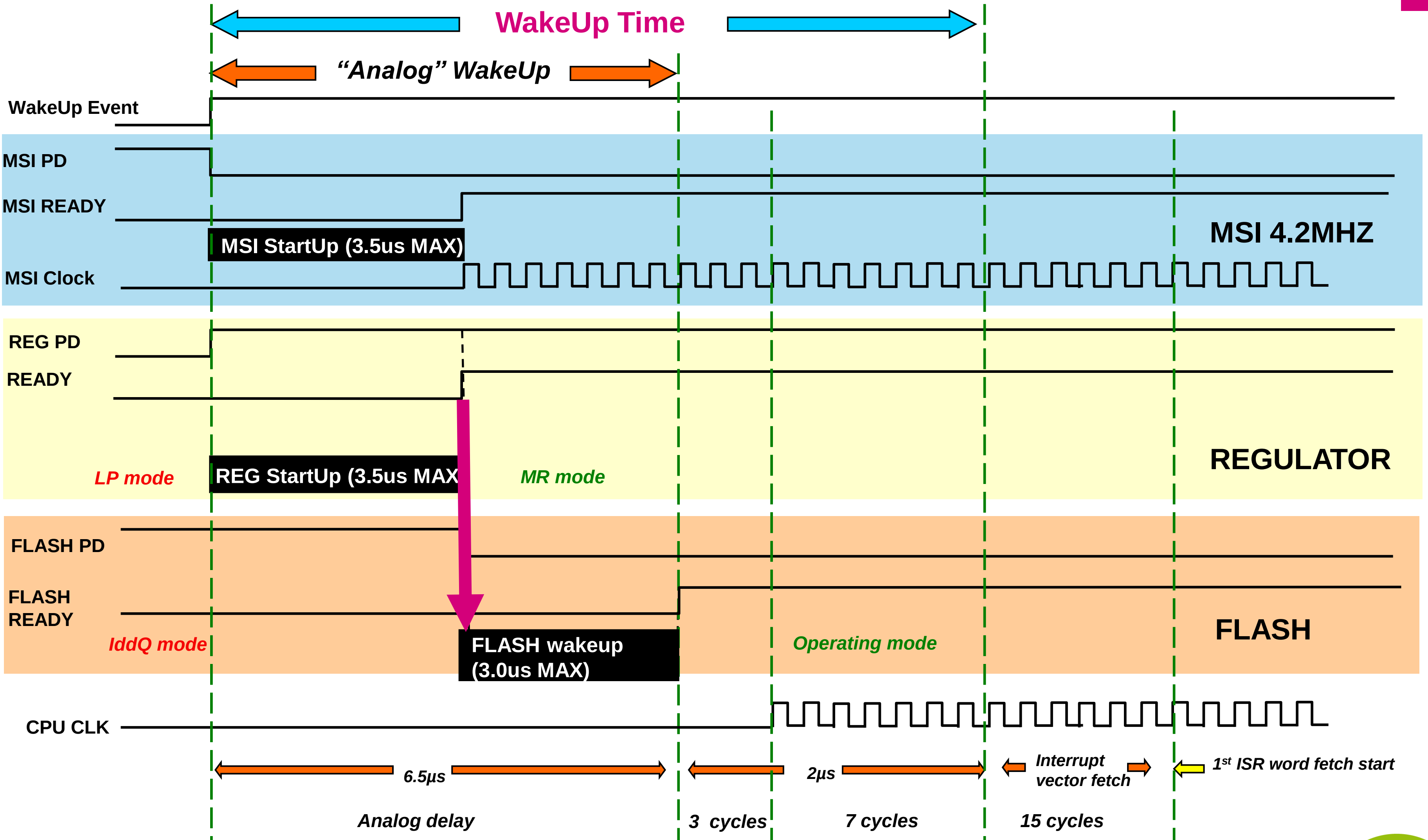
STOP Mode

- Entry by **WFI** or **WFE** while **SLEEPDEEP bit is set** and **PDDS bit is reset**
- Exit by **any EXTI line** configured in interrupt mode or in event mode
- **EXTI lines sources**
  - Any of the 16 external lines (any IO)
  - RTC alarms, RTC Tamper, RTC Time Stamp, RTC Wakeup, Comparators 1&2 events, USB wake-up, PVD
- After resuming from STOP the clock configuration returns to its reset state (**MSI**, **HSI16** or **HSI16/4** used as system clock)

Wake-up time from Stop	on MSI at 4.2MHz	on HSI at 16MHz
Wake-up to Flash	**7.8 µs**	**5 µs**
Wake-up to RAM	**3.5 µs**	**3.7 µs**



Wake-Up from STOP Timing Diagram



WakeUp Time

"Analog" WakeUp

WakeUp Event

MSI PD

MSI READY

MSI StartUp (3.5us MAX)

MSI Clock

MSI 4.2MHZ

REG PD

READY

LP mode

REG StartUp (3.5us MAX

MR mode

REGULATOR

FLASH PD

FLASH READY

IddQ mode

FLASH wakeup (3.0us MAX)

Operating mode

FLASH

CPU CLK

6.5µs

2µs

Interrupt vector fetch

1st ISR word fetch start

Analog delay

3 cycles

7 cycles

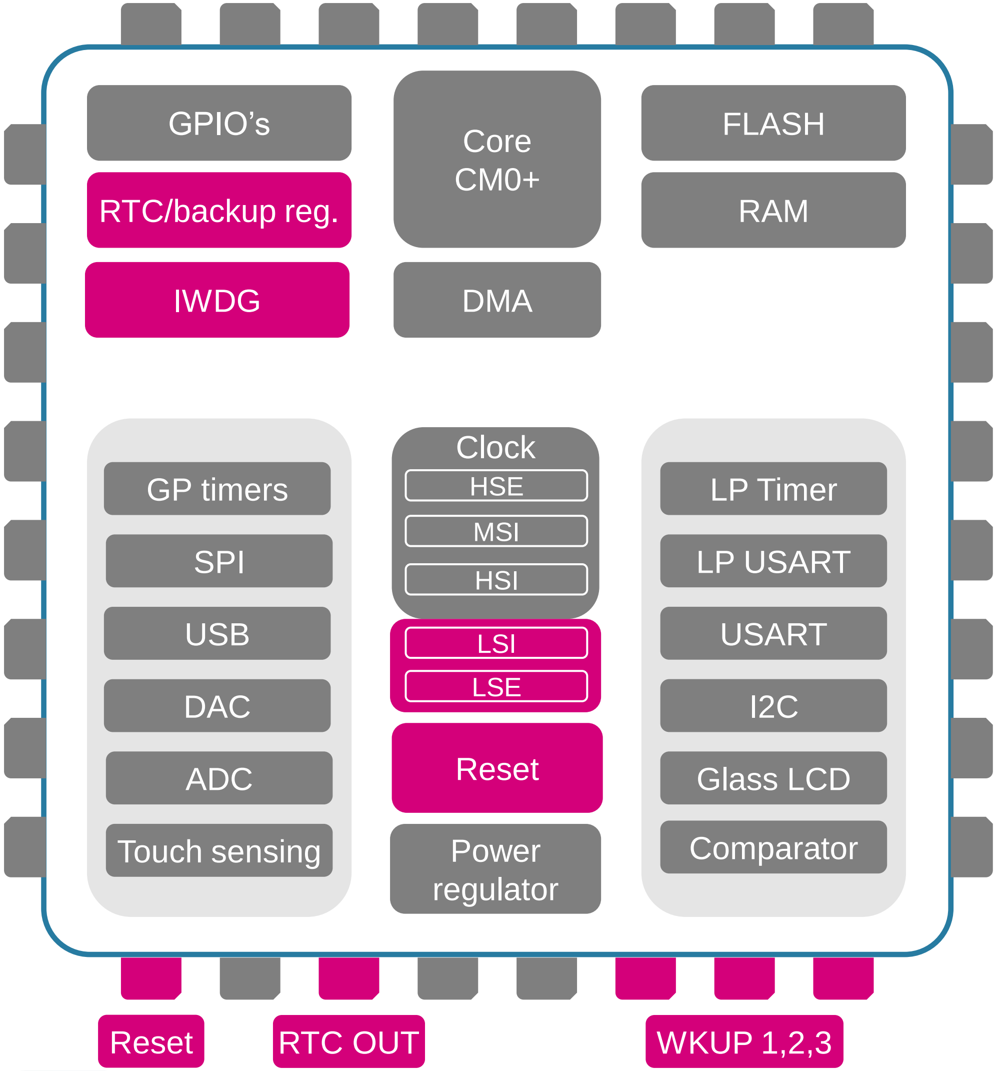
15 cycles

Can be fully eliminated if:

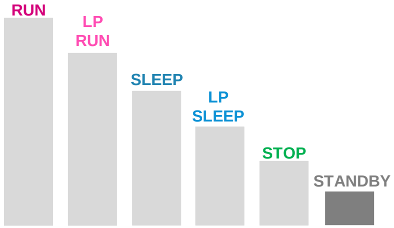
- HSI as default clock source
- Regulator kept in NORMAL mode
- Code executed from SRAM

Isn't it Another Low-Power mode?

STANDBY Mode



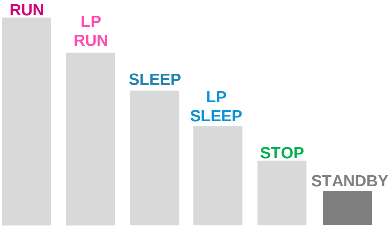
- Core and all peripherals are OFF, **except RTC and IWDG if enabled**
- HSE, MSI, HSI clocks are OFF, **LSI LSE can be ON**
- SRAM and registers content is lost, **except RTC, and standby circuitry**
- GPIO's are in high Z, **except Reset, RTC OUT and WKUP 1,2,3**



STANDBY Mode

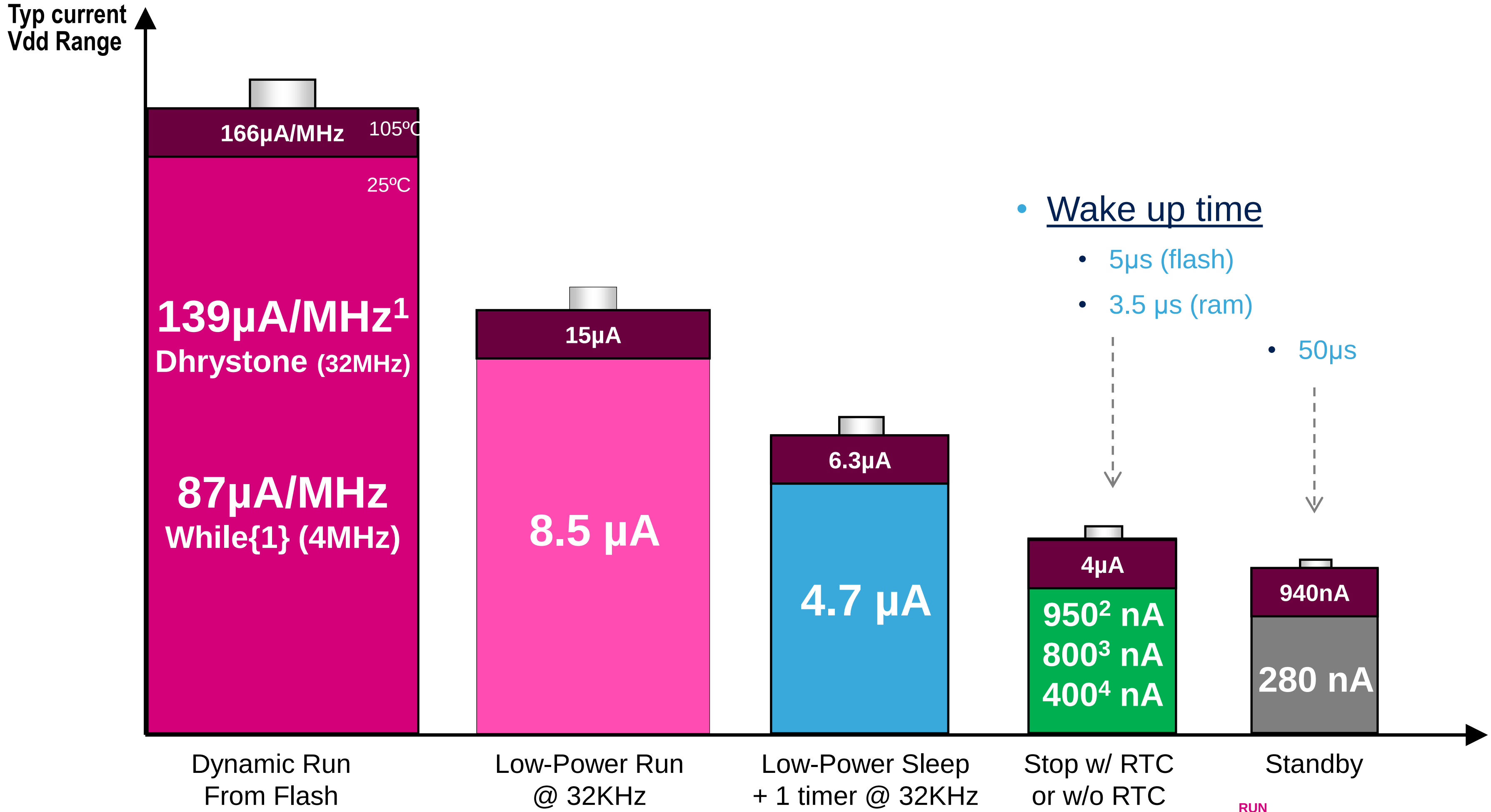
- Entry by **WFI** or **WFE** while **SLEEPDEEP and PDDS bits are set**
- **Wake up sources**
  - WKUP1 or WKUP2 pins rising edge
  - RTC alarm A, RTC alarm B, Wakeup Timer, Tamper event, TimeStamp
  - LSECSS ☹, IWDG reset ☺, External reset in NRST pin ☺☺
- After wake-up from STANDBY mode, program execution will restart in the same way as after a RESET **(startup sequence implementation is crucial)**

Wake-up time from STANDBY mode	on MSI at 2.1 MHz
STANDBY with V_{REFINT} ON	**50 µs**
STANDBY with V_{REFINT} OFF	**2.5 ms**



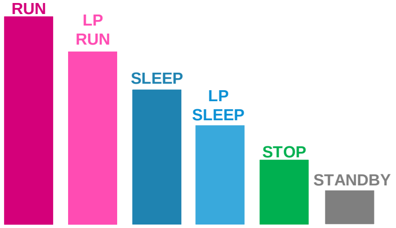
life.augmented

Low power modes summary



- Wake up time
  - 5µs (flash)
  - 3.5 µs (ram)
  - 50µs

1. Dhrystone power consumption value executed from Flash with VDD=3.3V
2. **STOP mode consumption with : Full Ram retention + RTC + Low-power Pulse counter**
3. STOP mode consumption with : Full Ram retention + RTC
4. STOP mode consumption with : Full Ram retention



References

- Datasheet
  - 3.1 Low power modes
  - 3.4.3 Voltage regulator
  - 3.5 Clock management
  - 6.3.4 Supply current characteristics
- Reference Manual
  - 6 Power control (PWR)
  - 7 Reset and clock control (RCC)
- Application note
  - AN4445 STM32L05x ultra-low-power features overview

Debug capability in LP modes

- Depends on **DBGMCU** Control Register configuration

- DBG_SLEEP = 1
  - In this case, when entering Sleep mode, previously configured system clocks are kept running and used

- DBG_STOP = 1
  - In this case, when entering STOP mode, clocks are provided by the internal RC oscillator which remains active in STOP mode

- DBG_STANDBY =1
  - In this case, the digital part is not unpowered and the clocks are provided by the internal RC oscillator which remains active.

- DBG_STANDBY, DBG_STOP, DBG_SLEEP **= 0**: **No clock, no debug**

If the debug mode is kept in STANDBY/STOP/SLEEP, the consumption is higher than in non debug state.

ENABLE CONNECT UNDER RESET !

Thank you

www.st.com/stm32l0