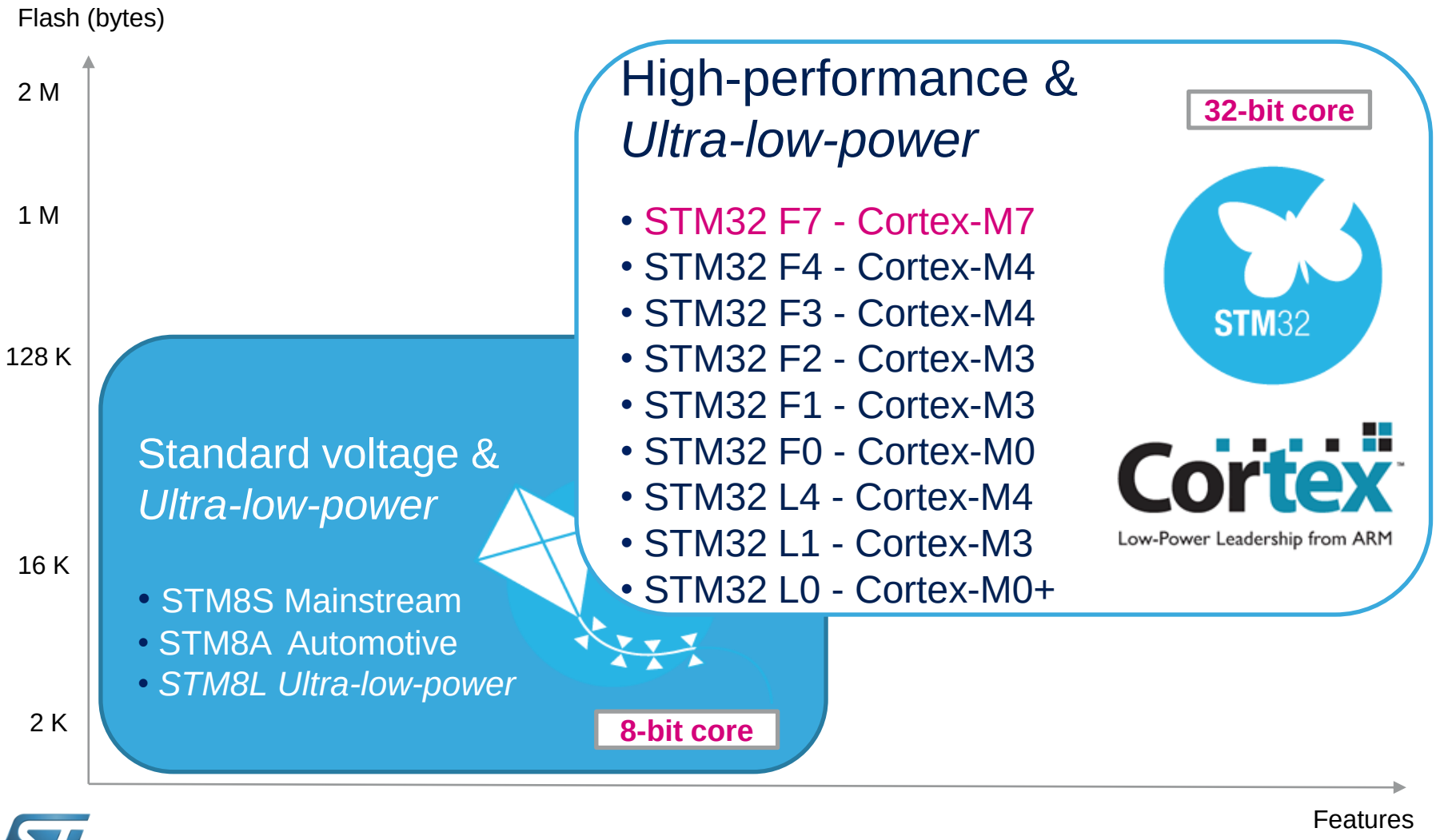
First Cortex-M7 MCU, STM32F7

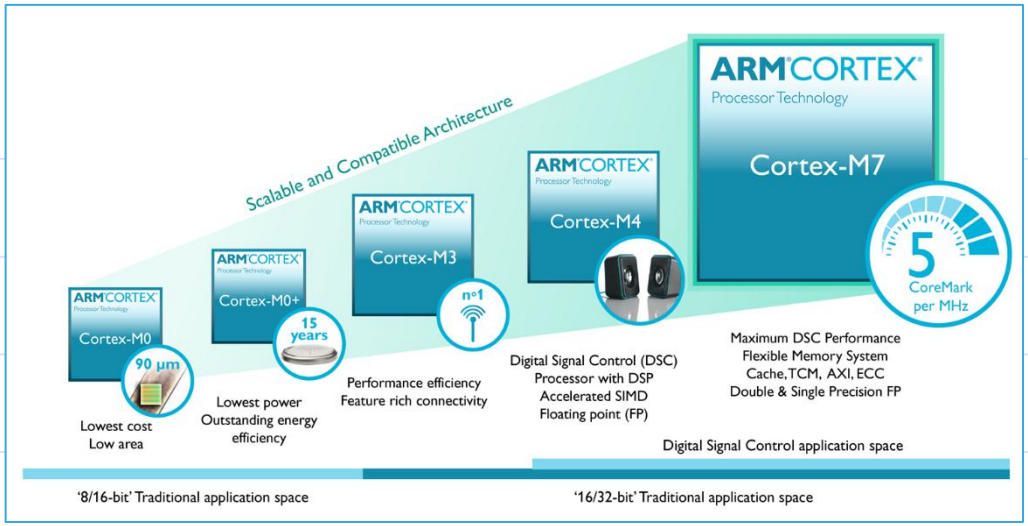
STMicroelectronics

life.augmented

MCUs – new families development focus



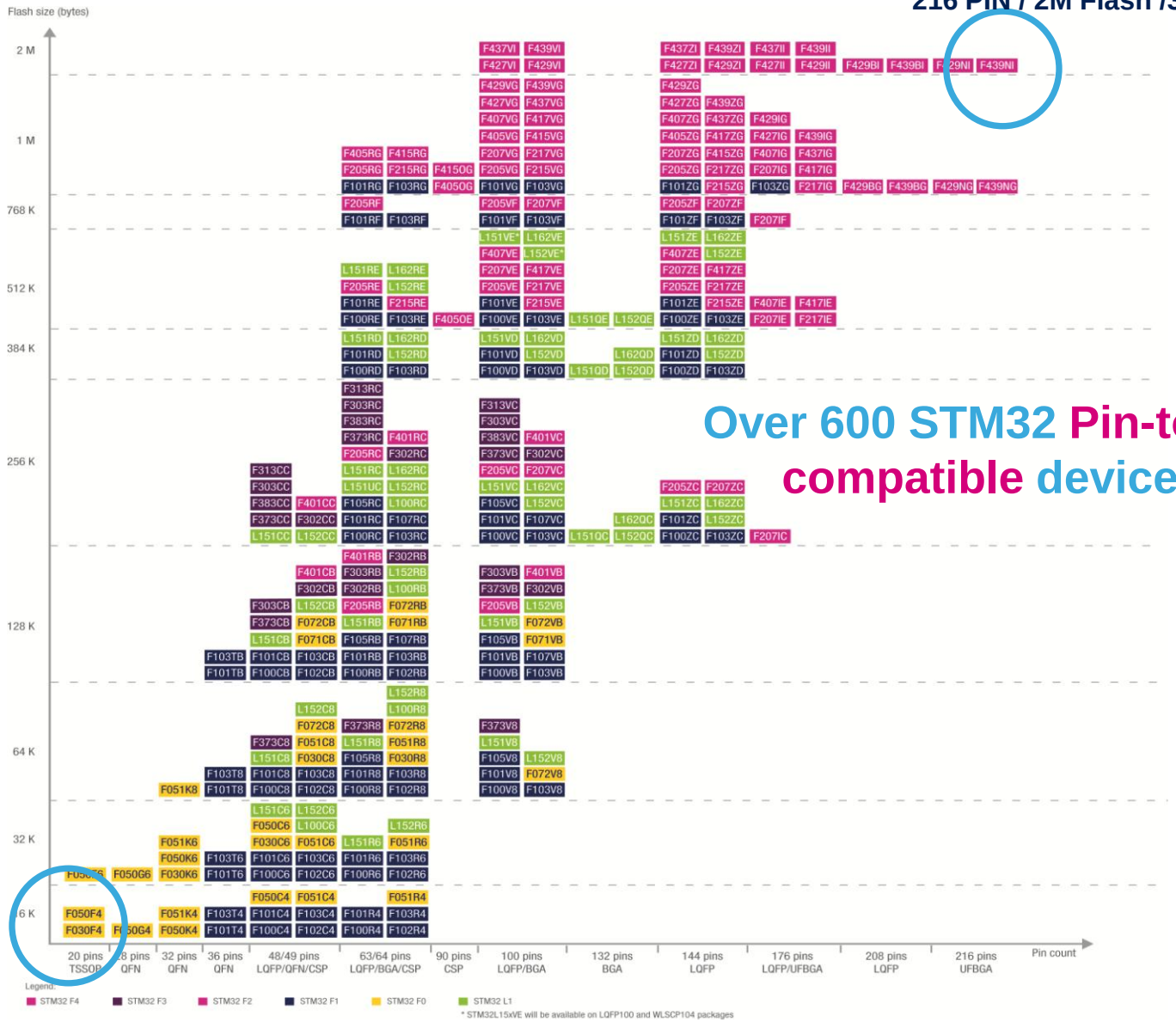
STM32 DMIPS performance



life.augmented

A rich & reliable product base and…

216 PIN / 2M Flash /320KSRAM



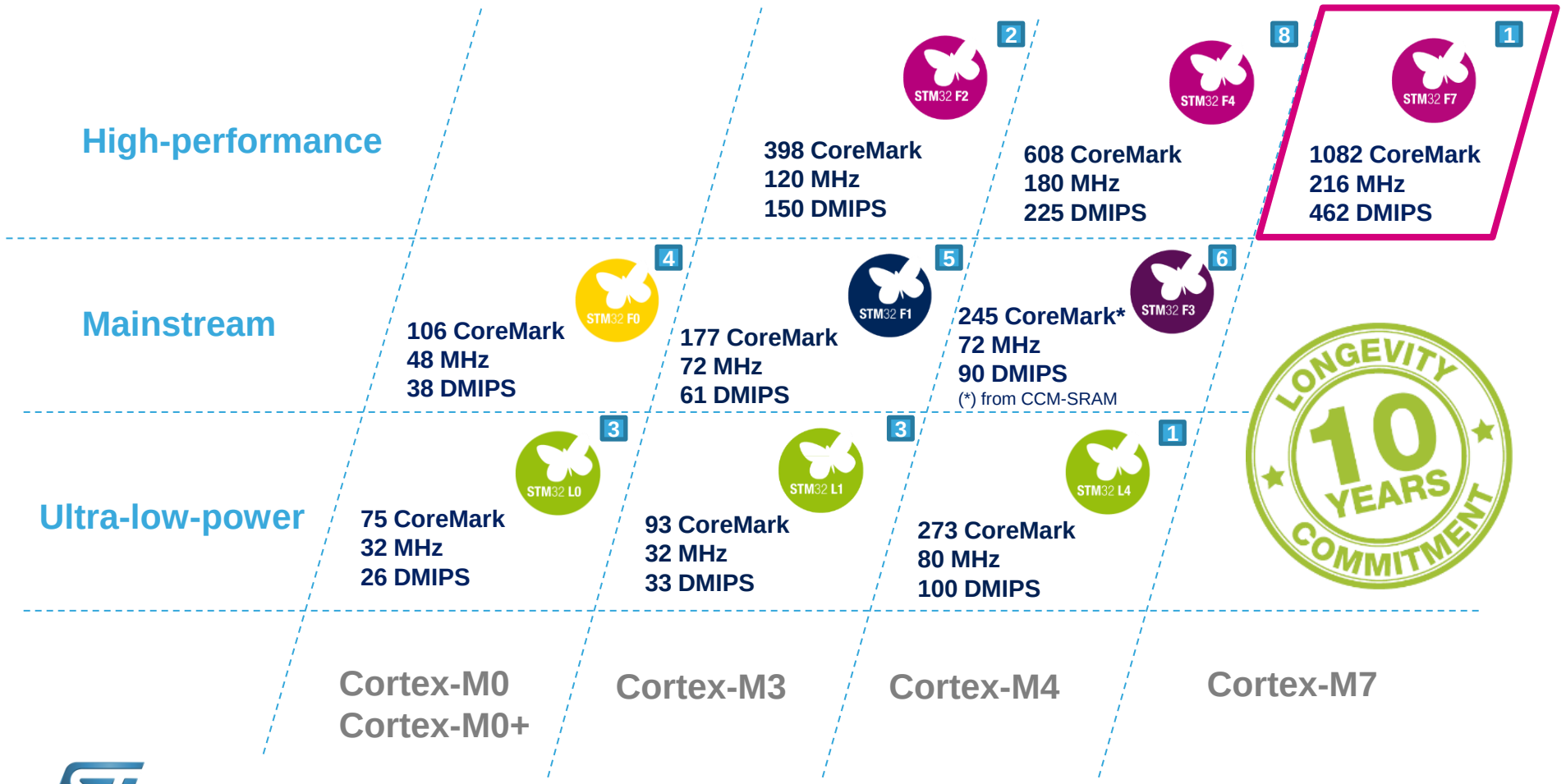
Over 600 STM32 Pin-to-pin compatible devices

20 PIN / 16KFlash

life.augmented

STM32 F7 : continuity in STM32 portfolio

9 product series / 33 product lines



life.augmented

number of lines

STM32 ecosystem

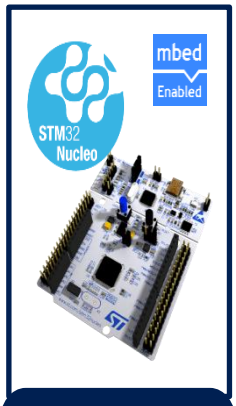
Hardware tools

Software tools

Embedded software

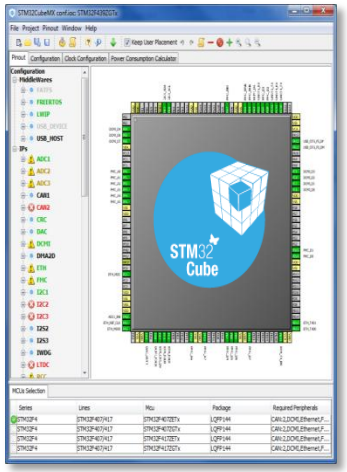
STM32 ecosystem

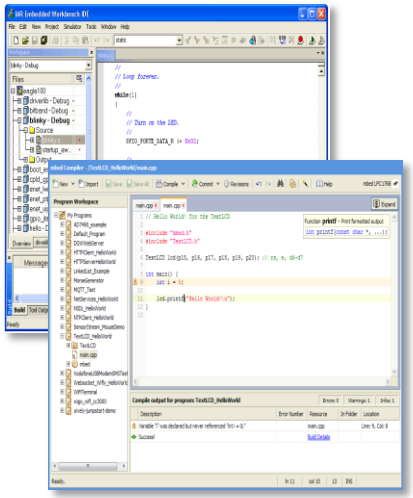
Hardware tools

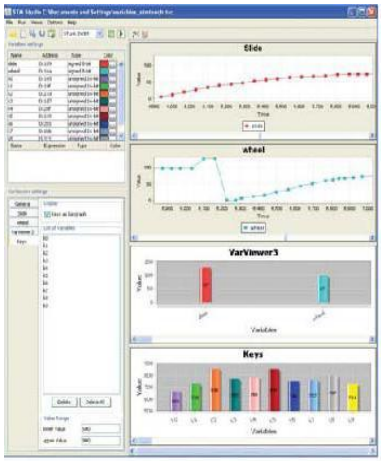


	STM32 Nucleo	Discovery kits	Evaluation boards	3rd parties
Typical use case	Flexible prototyping, Community	Prototyping, Creative demos	Full feature evaluation	From full evaluation to open hardware
Extension possibilities	+ + +	+ +	+ + +	
Connectivity	Arduino™ ST Morpho	ST	ST	

STM32 ecosystem







STM32Cube MX — Generate Code

Partners IDEs — Compile & Debug

STMStudio — Monitor

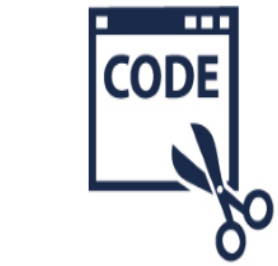
IAR SYSTEMS · atollic · TRACE SYSTEM · EMPROG "Where Software Meets Hardware" · TASKING · KEILABS · SEGGER Embedded Studio

ARM KEIL Microcontroller Tools · CooCox · mbed · AC6 Assistance Conseil Systèmes

Free IDE

Embedded software

STM32 ecosystem



STM32 Snippets*	STM32Cube and Std Libraries	CMSIS and mbed SDK	Virtual machines and models
High optimization Low portability	Average optimization STM32 portability	Low optimization ARM portability	Low optimization Large portability

*on STM32F0 and STM32L0 only

life.augmented

STM32 ecosystem

Apps & social media

- Find more about STM32 products and solutions:

ST MCU Finder mobile application

www.st.com/stmcufinder

Social media

ST Forums on microcontrollers

facebook.com/stm32

youtube.com/STonlineMedia

twitter.com/@ST_World

Mbed.org

ARM mbed

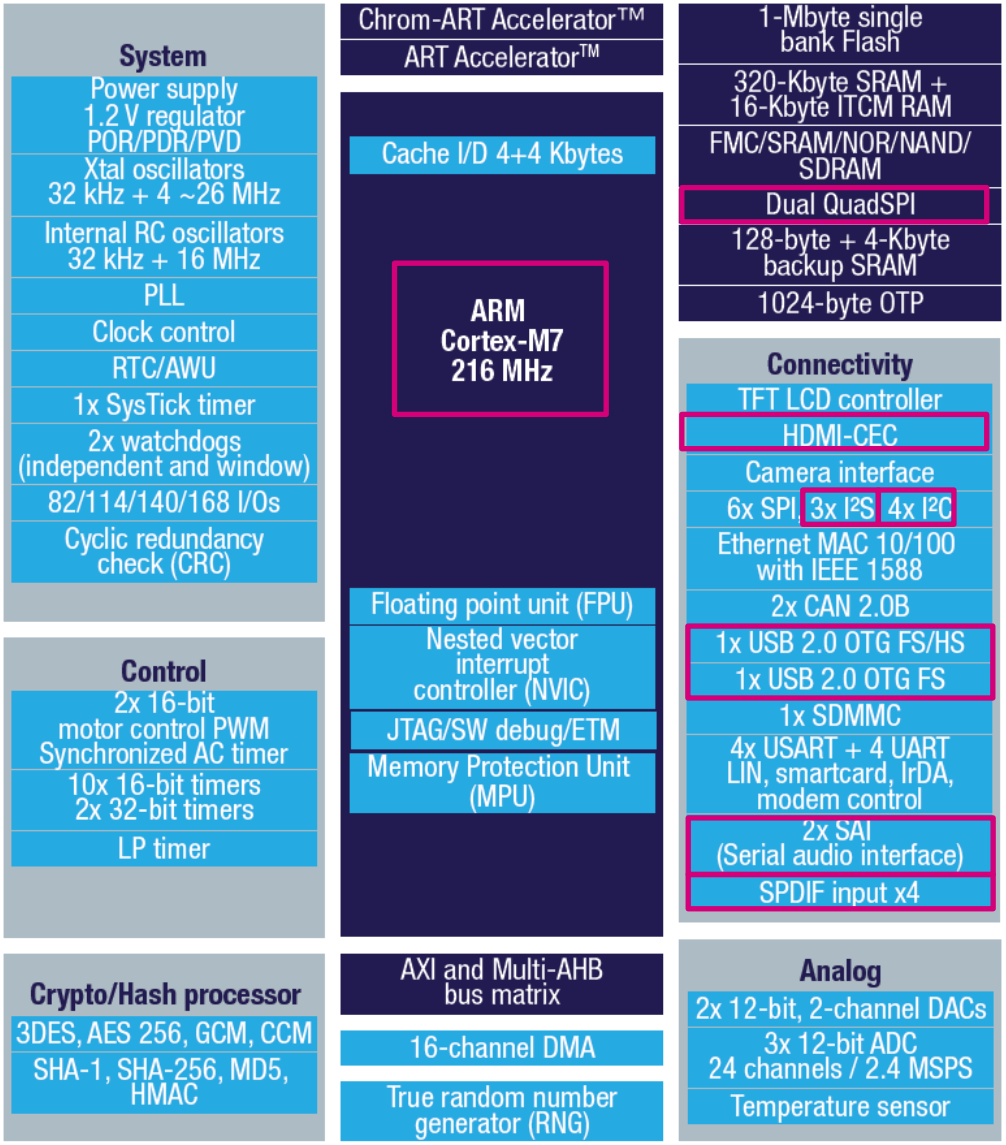
ARM Connected Community

STM32 @ ARM connected community

STM32 F7 block diagram

- NEW core: ARM Cortex-M7
- Up to 216 MHz, 462 DMIPS /1082 CoreMark
- Twice more DSP performance vs Cortex-M4 core
- New generation of Peripherals
- 3xI2S, 4xI2C, USB dedicated supply for 1.8 V operation, CEC, **Quad SPI**, **SPDIF input**, 2xSAI(**SPDIF output**), ...
- Same packages as F429
  - WLCSP143
  - LQFP100,144,176,208
  - BGA 176, 216

STM32 F7

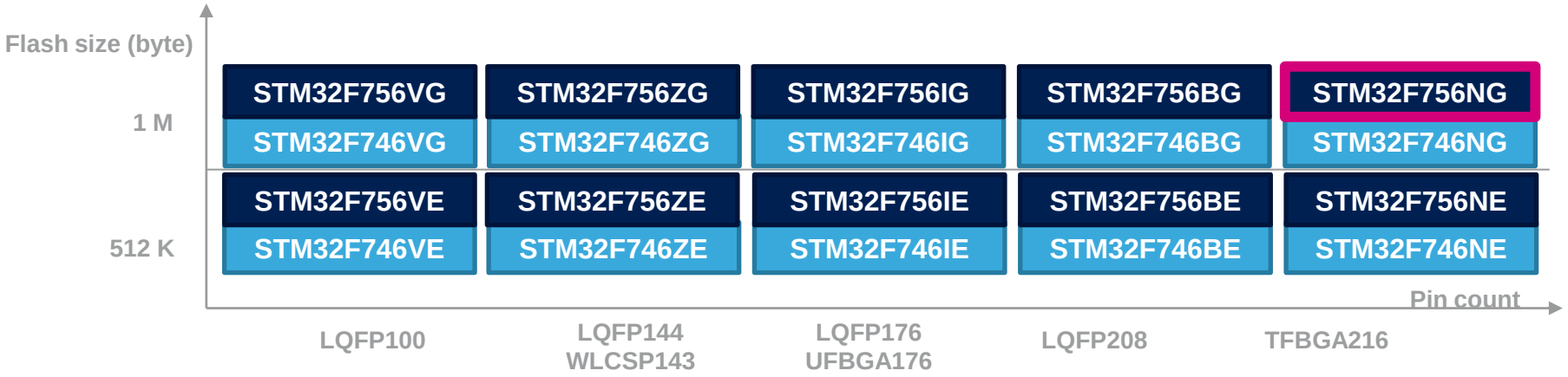


STM32 F7 portfolio

ARM® Cortex®-M7 – 216 MHz

- ART Accelerator™
- L1 cache: 4K+4K data and instruction cache
- Chrom-ART Accelerator™
- Single Precision FPU
- 2 x USB 2.0 OTG FS/HS
- SDIO
- 2 x CAN
- I²S + audio PLL
- 2 x SAI
- 2 x 12-bit DAC
- SPDIF-RX
- 16- and 32-bit timers

Product	FCPU (MHz)	Flash (bytes)	RAM (KB)	Ethernet I/F IEEE 1588	Quad SPI	Camera I/F	FMC	TFT LCD controller
STM32F746*	216	512 K to 1 M	320	•	•	•	•	•
STM32F745	216	512 K to 1 M	320	•	•	•	•	



Legend: without HW crypto/Hash coprocessor; with HW crypto/Hash coprocessor

life.augmented

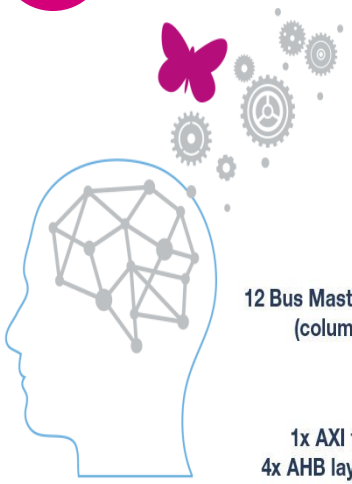
Key words of STM32 F7 series

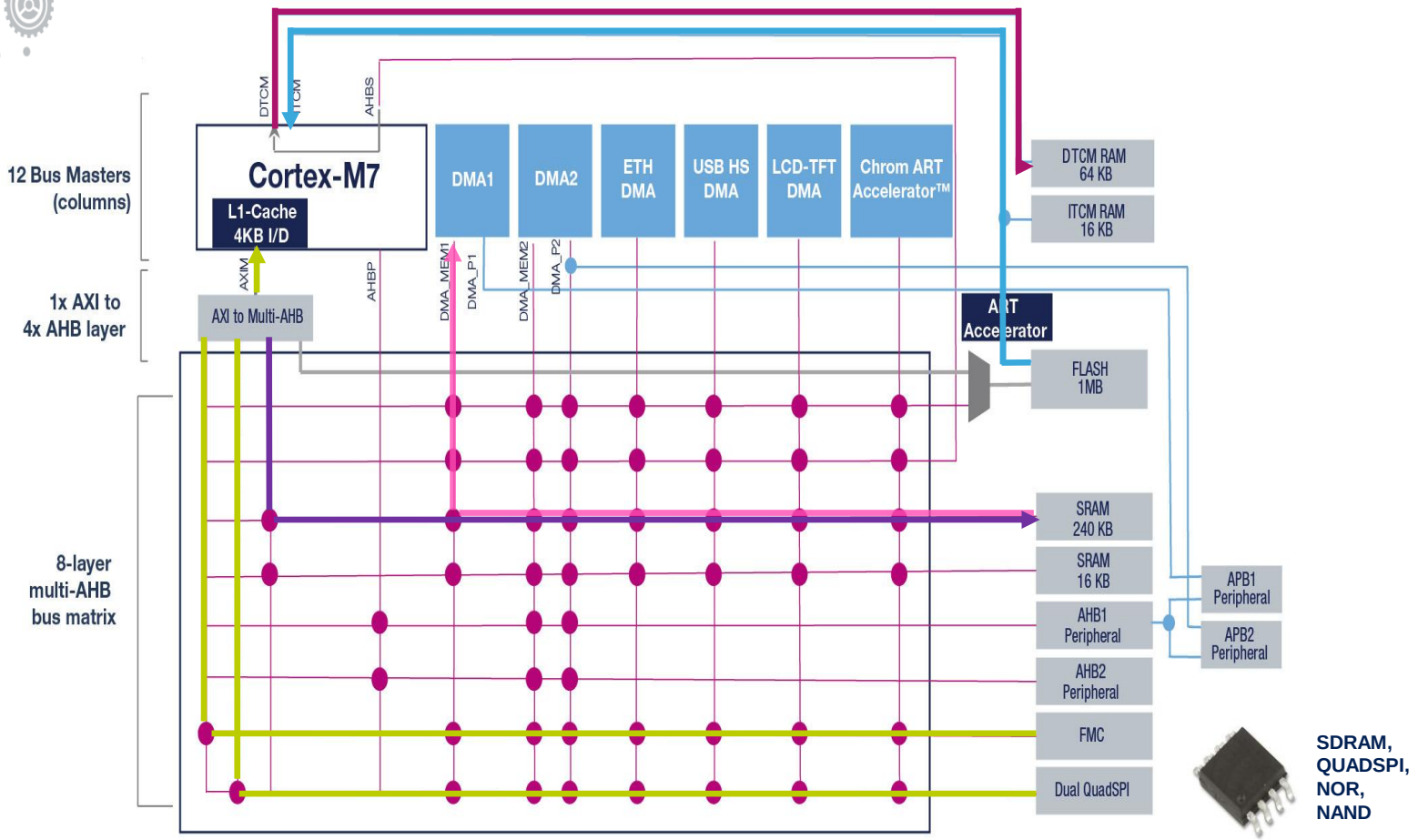
ST is first to sample fully functional Cortex-M7 based 32-bit MCU: STM32 F7 series

- ST is a lead partner of ARM; ST actively participated to the specification of today's new Cortex®-M7 core
- ST is ready with fully functional samples of the STM32 F7 microcontroller in TFBGA216
  - The first product of the new F7 series
- The STM32 F7 product is running several application demonstrations on an STM32 evaluation board.

Full functional product **today**

Smart Architecture – Use case





SDRAM, QUADSPI, NOR, NAND

Legend:

ITCM: Critical Code with deterministic execution

DTCM RAM: Critical real time data (Stack, heap ..)

System SRAM: Concurrent data transfer CPU or DMA

External Memories: Quad SPI, and FMC for data manipulation or code execution

Record Performance

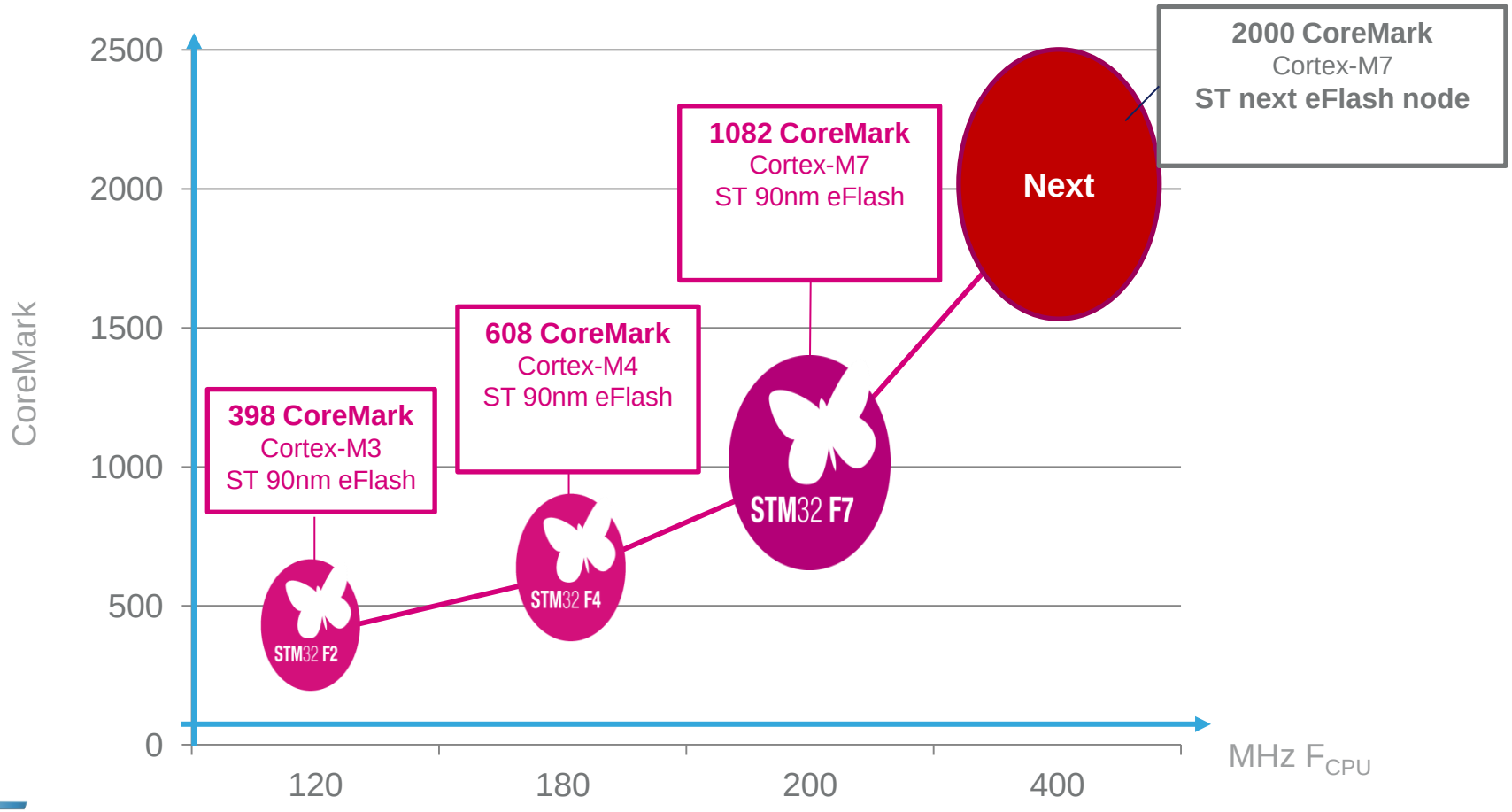
…Unleashed by STM32 F7 Silicon

Benchmark	ARM® data		Measured on STM32 F7 Silicon	
	Cortex-M4	Cortex-M7	Executing from Embedded Flash	Executing from External memory
CoreMark/MHz	3.4	5	5	
DMIPS/MHz	1.25	2.14	2.14	

- And : …
  - Up to twice more DSP performance increase over Cortex-M4
  - ARMv7-M architecture, 100% binary forwards compatibility from Cortex-M4
  - STM32 F7 runs at F_{CPU} = 216 MHz → 5 x 216 ≈ 1082 CoreMark

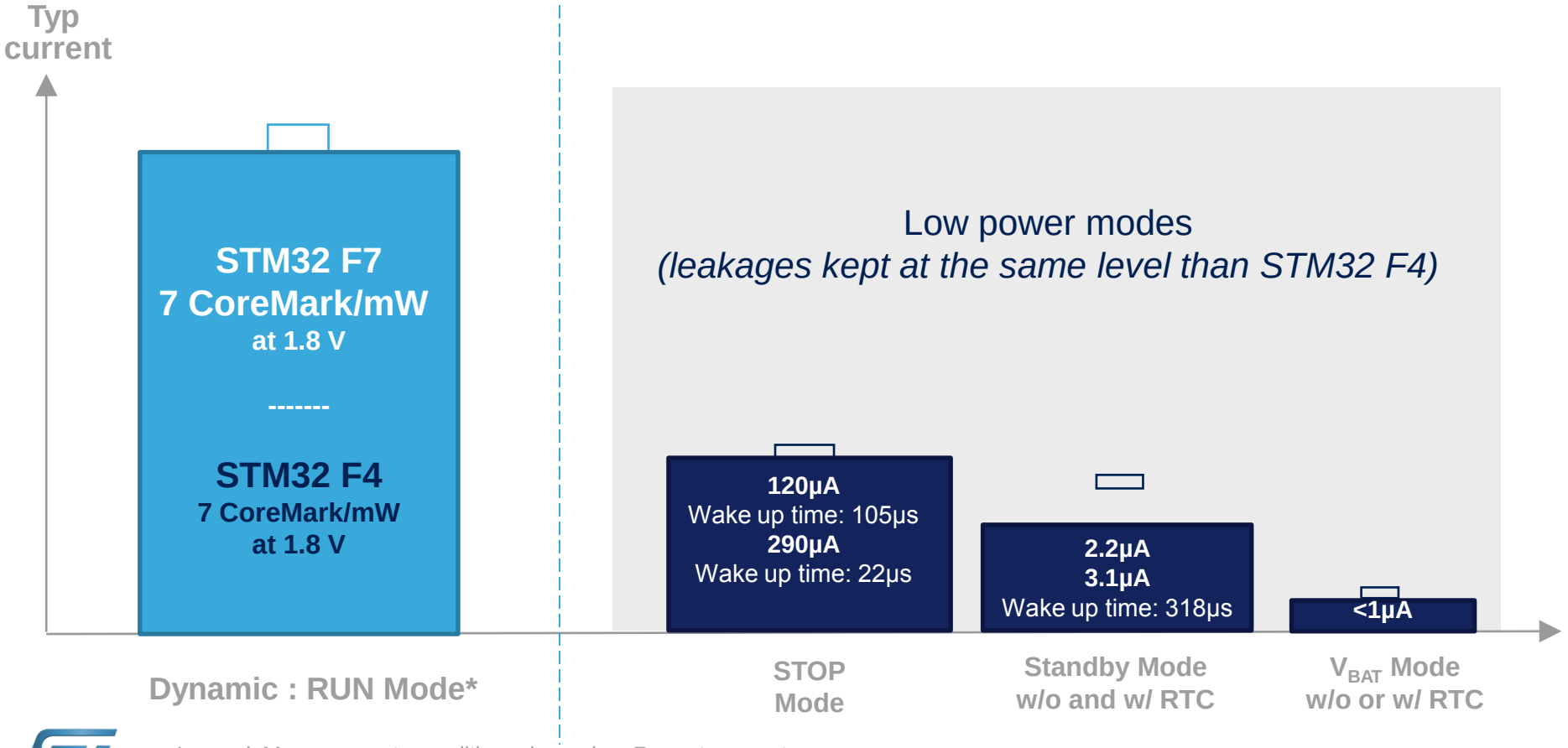
Record Performance

Fully compatible with the STM32 F4 and fully reuses STM32 Ecosystem
Our next step will go for the 2000 CoreMark on the next technology node



Power Efficient

STM32 F7 power efficiency = STM32 F4 power efficiency
STM32 F7 Boosts performance, but does not compromise on power efficiency



Legend: Measurements conditions depend on Room temperature
*Run mode Conditions: CoreMark executed from Flash, peripherals OFF

life.augmented

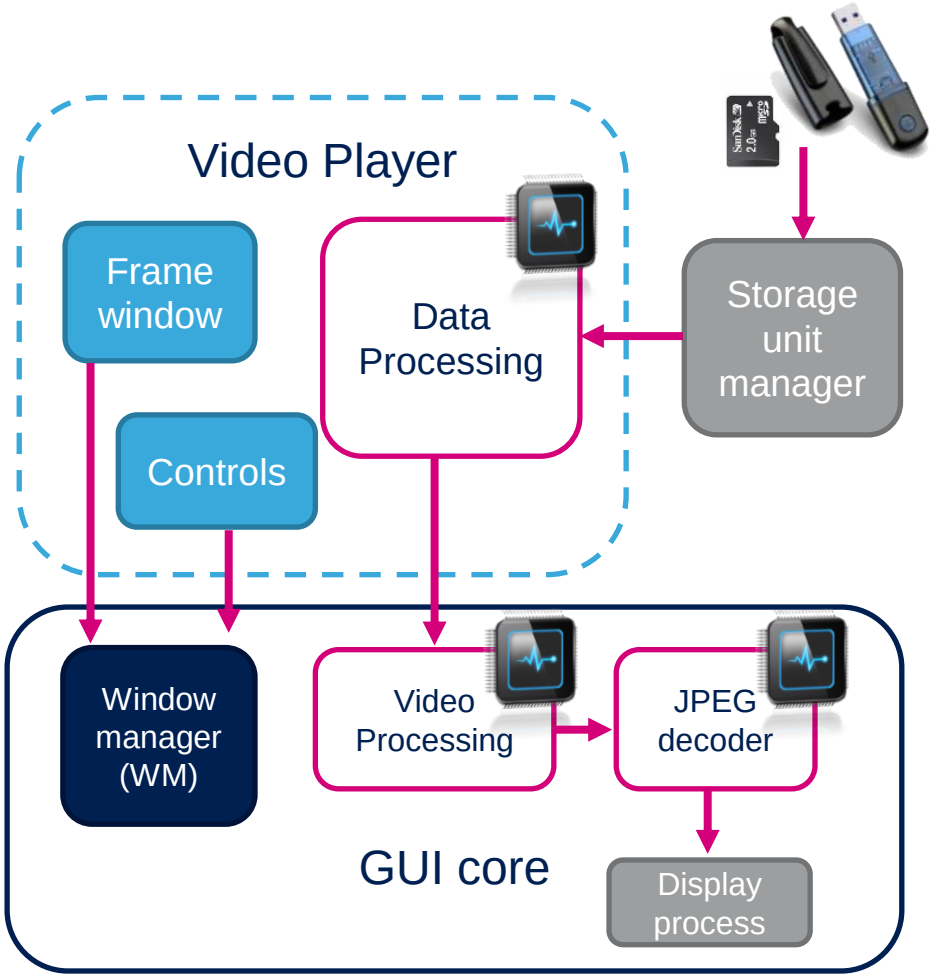
STM32 F7 Demos

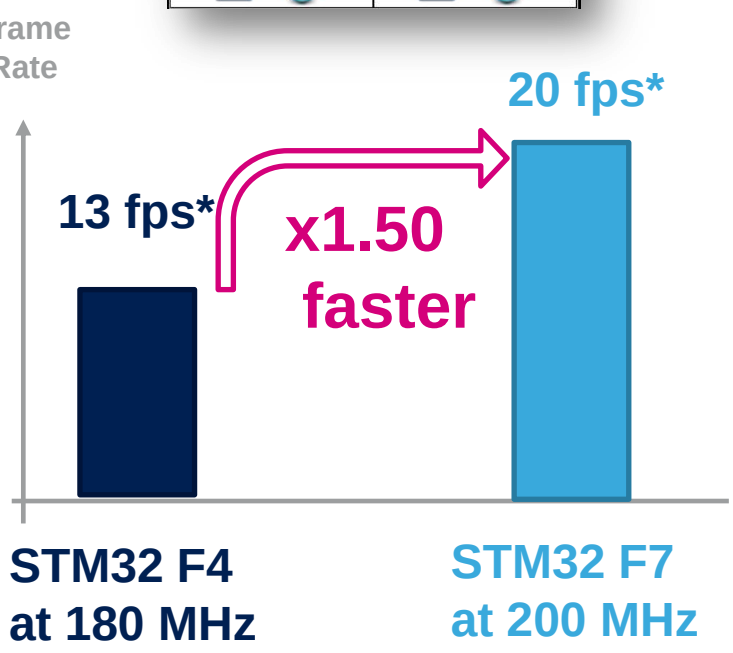
Pin-to-pin compatibility, enables demonstrations on the same board for **STM32 F4** and **STM32 F7** devices.

STM32 benchmark demonstration shows (in 2 demos) increase of performance on the F7 vs. F4 series :

- **Multi-tasking benchmark**: Manages the HMI frames on the display, the buttons on the board, and shows the STM32 core performance to manage 2 computation intensive tasks (decodes and displays simultaneously two MJPEG files from one single USB mass storage device). Key performance enablers: ART Accelerator, L1 cache, ST Chrom-ART Accelerator and bus matrix to send the picture to the LCD.
- **Recursive FPU computation**: Computes and displays the Julia set fractal. Key performance enablers: FPU, ART Accelerator, L1 cache.

Multi-tasking benchmark



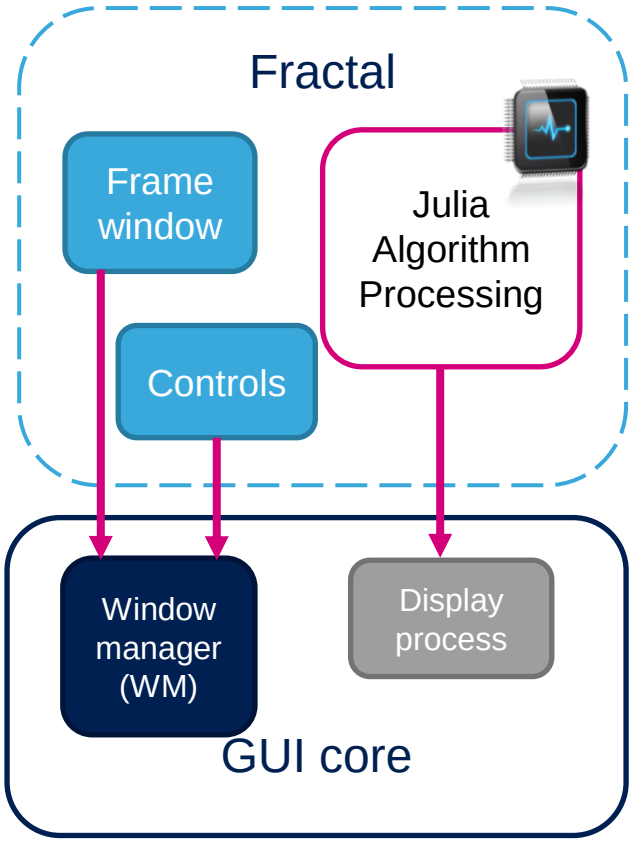


Requires high CPU resources

* : frames per second

life.augmented

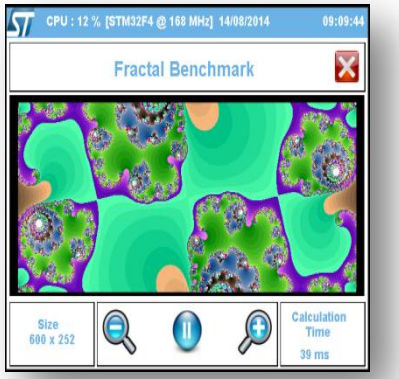
Recursive FPU computation benchmark



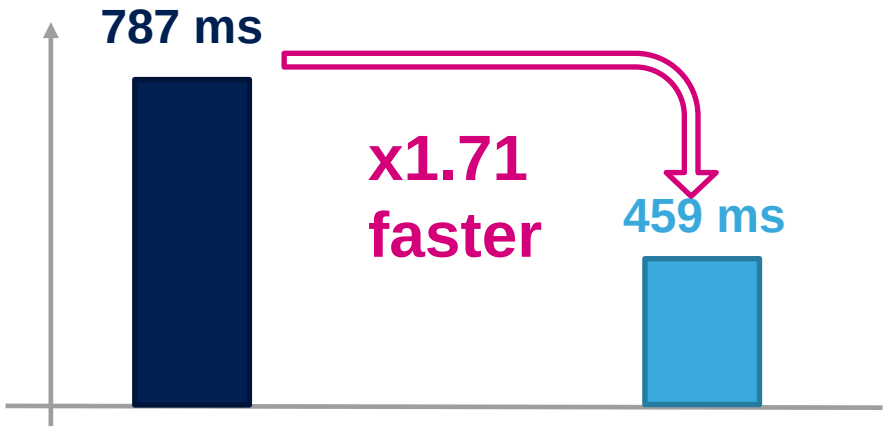
Julia set fractal formula

$$Z_{(n+1)} = Z_n^2 + C$$

$x_{(n+1)} + iy_{(n+1)}$ = $(x_{(n)} + iy_{(n)})^2$ + $c_x + ic_y$



Execution time

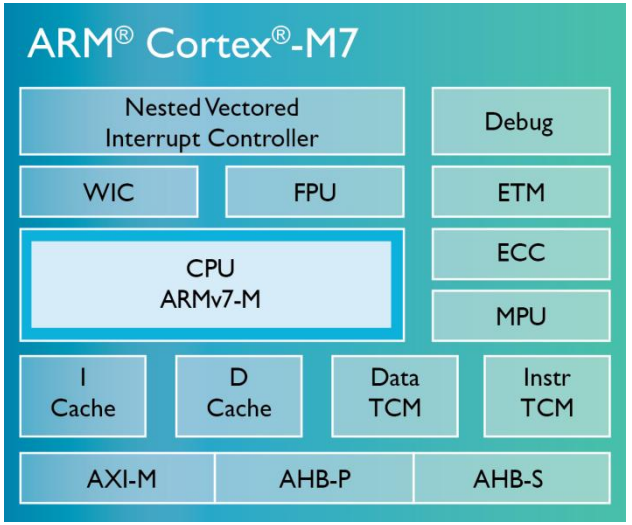


Requires high CPU resources

life.augmented

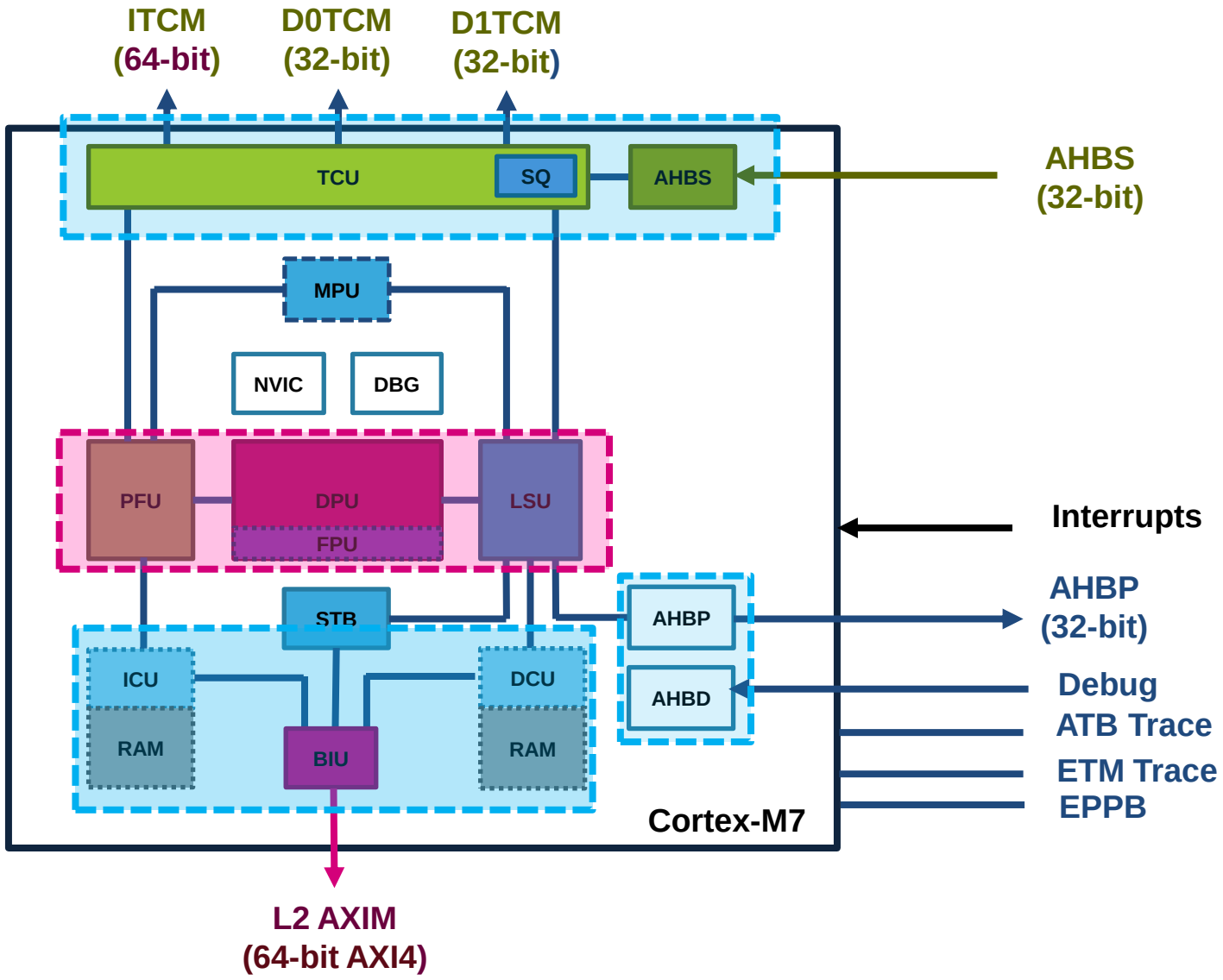
Cortex-M7

Cortex-M7 overview

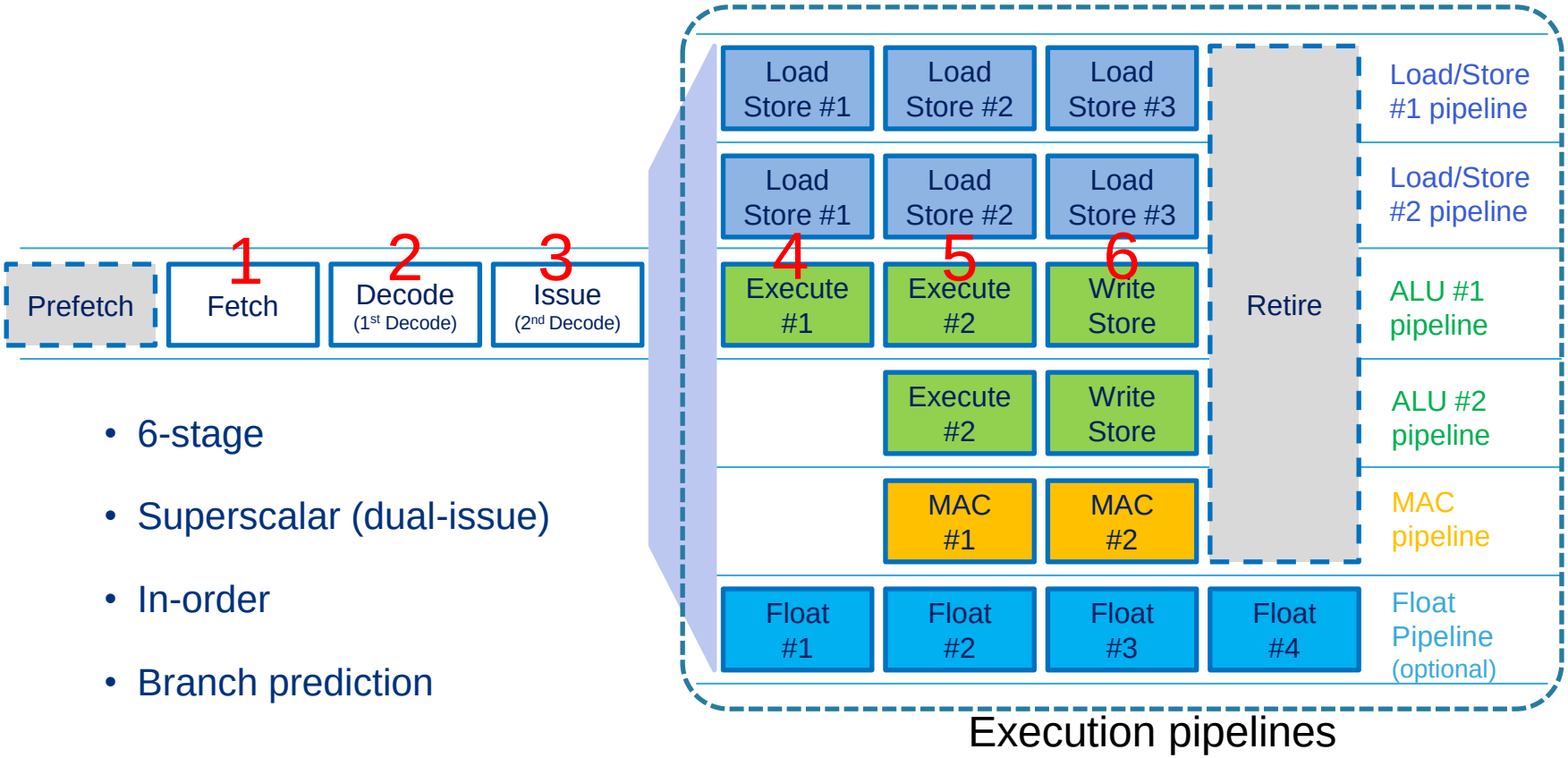


- Performance and configurability
  - Six-stage dual-issue pipeline (superscalar, branch prediction)
  - Powerful DSP instructions and SP/DP Floating Point
  - Flexible system and memory interfaces : TCM, AXI and AHB
  - Harvard caches (4kB of I-cache and 4kB of D-cache)
  - Dedicated peripheral and DMA ports
- Power efficiency
  - Clock gating, support of WIC and Sleep modes same as CM3/CM4
  - Support of multiple power domains and state retention
- Safety
  - Memory ECC (SEC-DED), MPU, MBIST, lock-step operation, full data trace, safety manual
- Debug and Trace
  - ETMv4 instruction and optional data trace support
  - Serial-Wire and JTAG support

Core Architecture



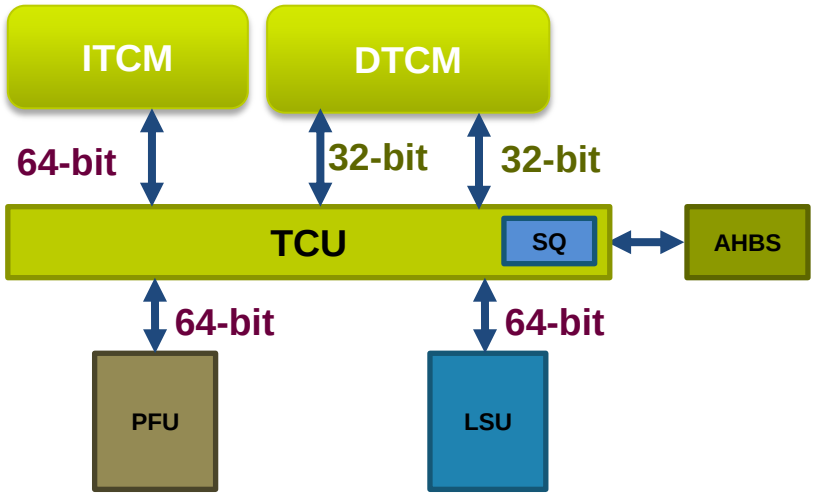
Cortex-M7 Pipeline



- 6-stage
- Superscalar (dual-issue)
- In-order
- Branch prediction

Memory I/F

- A Tightly-Coupled interface Unit (TCU)
  - Memory Interface between TCMs and CPU
- All TCMs:
  - Support wait-states
  - Can be used at boot-up
- Fixed base address
  - ITCM : 0x00000000, DTCM : 0x20000000



- Instruction TCM (ITCM)
  - 64-bit interface single port. Flash connected to this interface needs to be accelerated externally to meet processor fetch bandwidth requirements at expected frequencies
- Data TCM (DTCM)
  - 2 x 32-bit interface: D0TCM and D1TCM is selected according to addresses bit[2]
    - The upper 32-bits of data is on the D1TCM interface and the lower 32-bits of the data is on the D0TCM interface.

Memory I/F

- Bus Interface Unit (BIU)
  - with a configurable AMBA 4 AXI interface that can support a high-performance L2 memory system.
- AXI - Interface to L2 system:
  - L2 Caches (if implemented)
  - On/Off-chip memory
  - Devices & Peripherals
- Single AXI 64-bit interface
  - Buffers implemented to decouple bus transfer from CPU
  - 2 data linefill buffers
  - 1 instruction linefill buffer
  - 1 store buffer
  - 1 write buffer
- An extended AHB-Lite interface to support low-latency system peripherals.

AHB Limitations & AXI concept

- AHB : 1 transaction at a time
  - The request MUST be followed by its corresponding data (bus blocked if memory latency)
  - The request can be a read OR a write
- To allows more transaction in parallel (instruction fetch, data fetch, peripheral accesses …etc…)
  - we need several AHB busses
  - Cortex-M4 has 3 AHB busses
- AXI is decorrelating request and data phase
  - The request is independant from its corresponding data (bus available if memory latency)
  - Equivalent of 1 bus for request &1 for data (& optionally 1 for write acknowledge)
  - Out of ordering is possible if no functional relationship (ex. Ins.Fetch & Data Fetch)
- AXI is separating read and writes
  - Read bus & Write bus are splitted : a read can be launch in the same cycle than a write
  - Total equivalent of 5 busses: Read Request, Read Data, Write Request, Write Data, Write Acknowledge

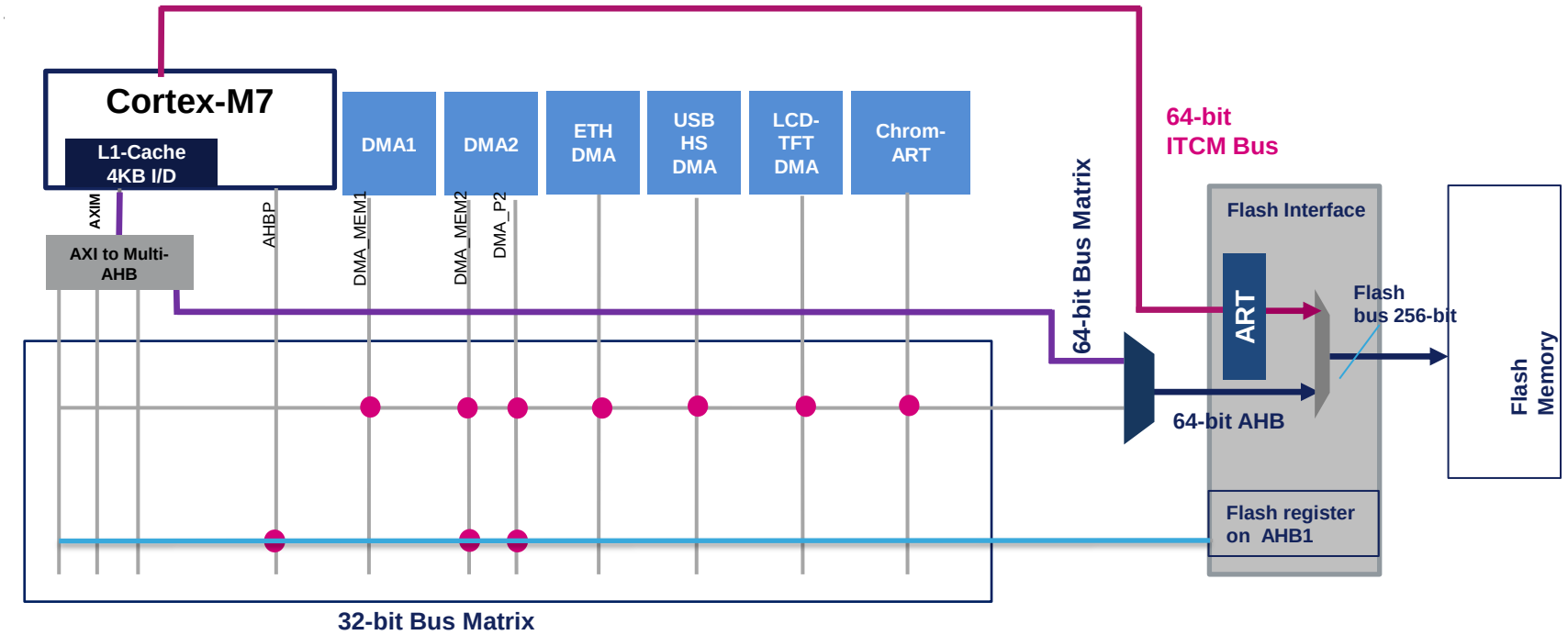
Caches

- Harvard architecture for optimum performance
- I and D both optional, separately configurable sizes (4kB – 64kB each)
- Cache only serve the AXIM region (No caching for the TCMs or AHBP)
- I-cache 2-way associative, D-cache 4-way associative,
- Extensions required to the ARMv7E-M system architecture
  - Addition of cache maintenance operations: Add-on of new registers for cache maintenance.
  - System SW will be impacted by cache inclusion

ART Accelerator

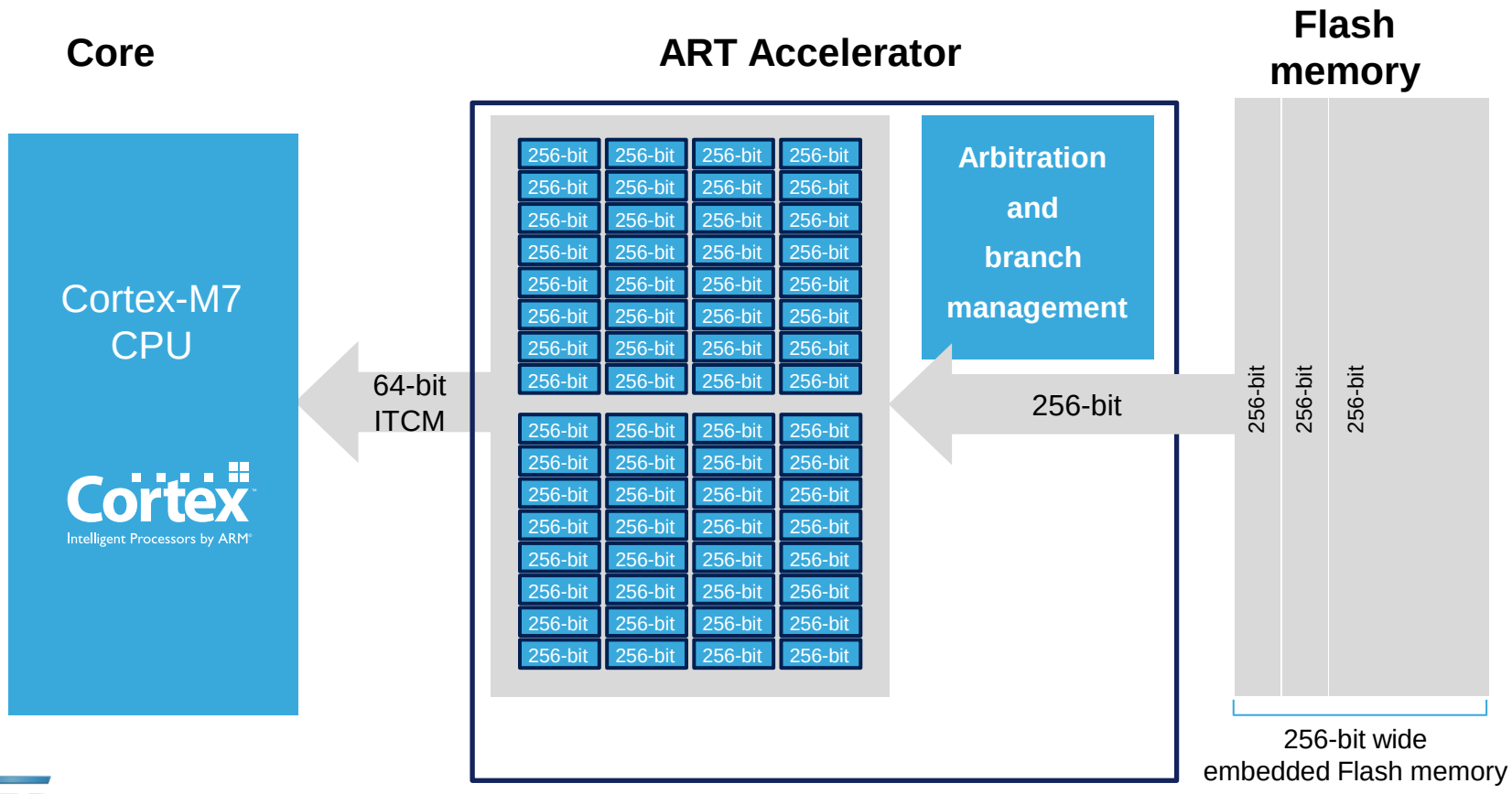
Flash interfaces

- 3 Main Flash interfaces
  - 64-bits ITCM interface
  - 64-bits AXI/AHB interface
  - 32-bits AHB register interface



ART Accelerator

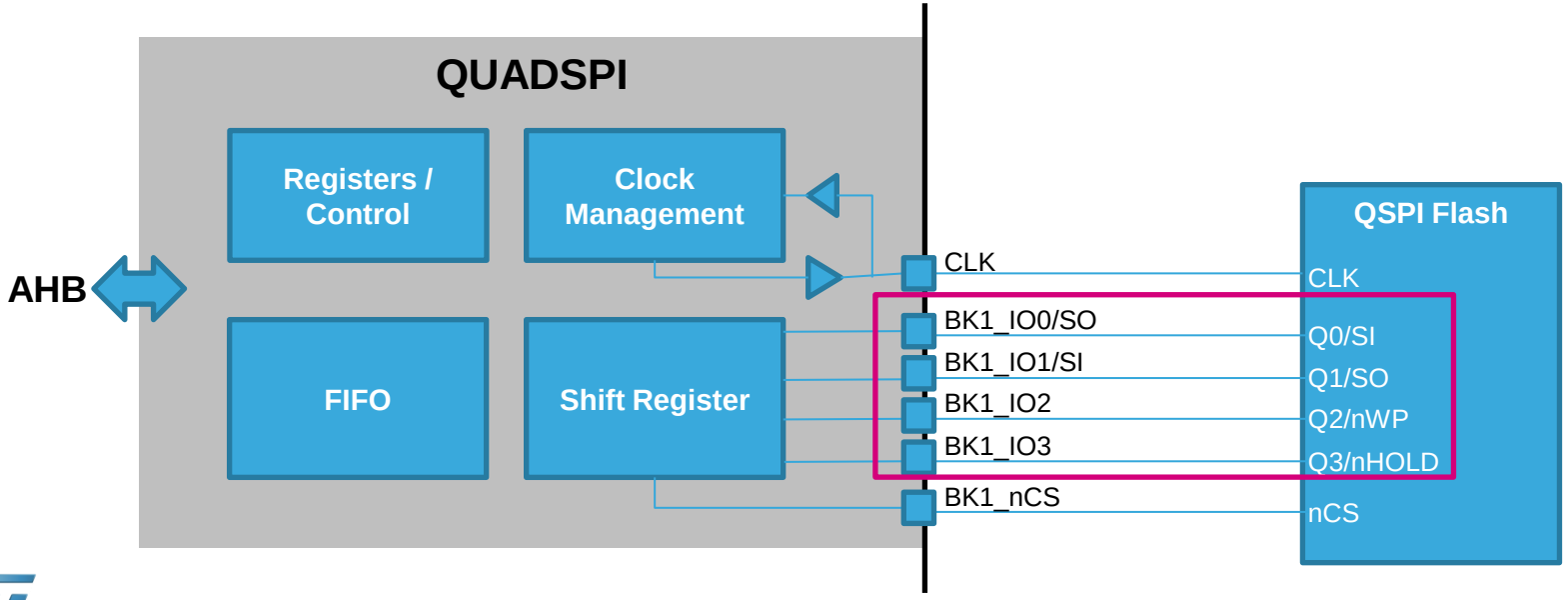
- ART Accelerator™ for F7 series
  - The ART (Adaptive Real-Time) memory accelerator unleashes processing performance equivalent to 0-wait state Flash execution up to 216 MHz for F7 series



QuadSPI

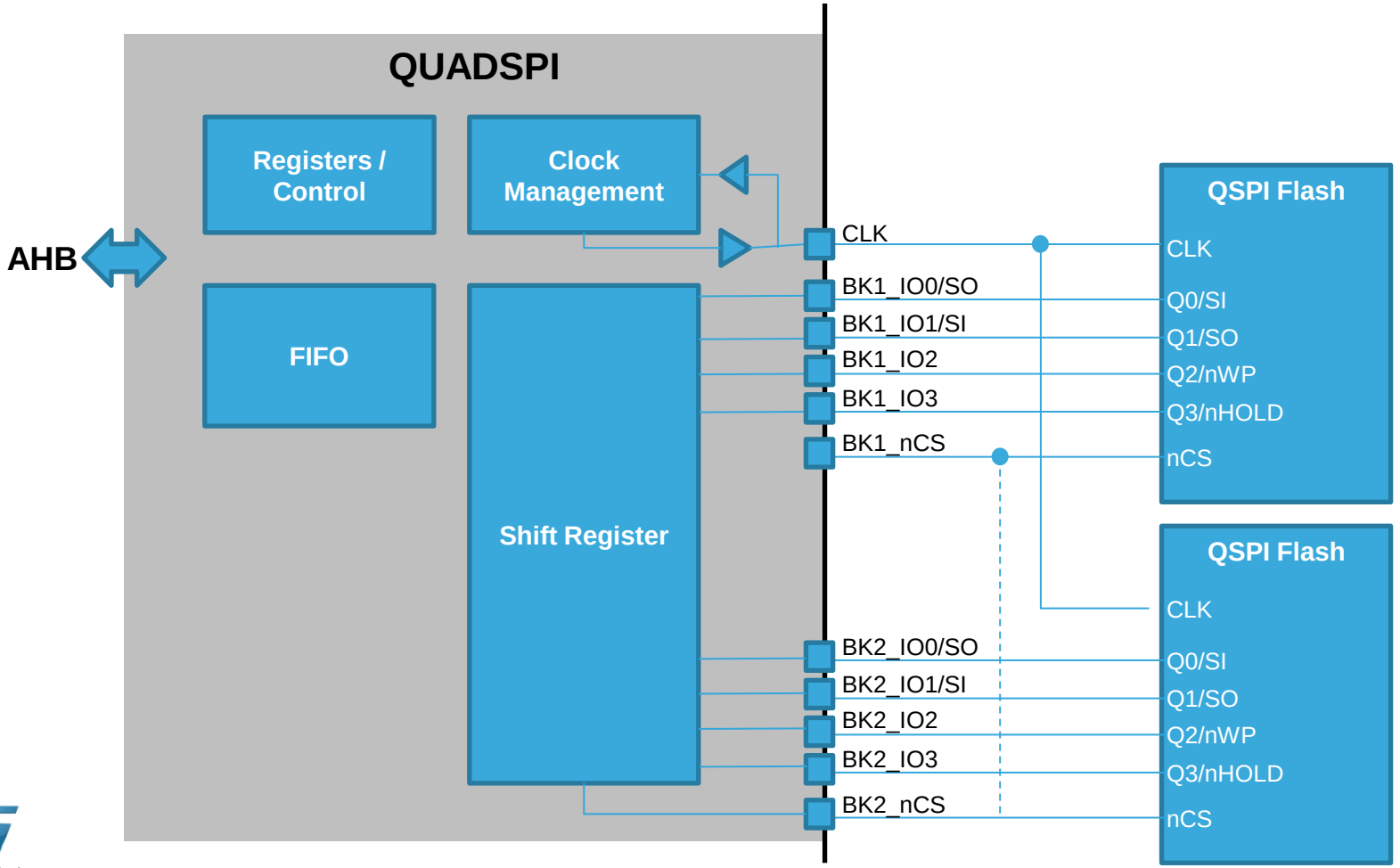
Overview

- Communication interface for single/dual/quad SPI flash memories
- Three operating modes
  - **Indirect** : all the operations are performed through registers (classical SPI)
  - **Status polling** : periodical read of the flash status registers (interrrupt generation)
  - **Memory mapped** : External flash seen as internal for read operations



Dual-quad mode

- Access two flashes in parrallel with the same frame format and the same instruction (8-bit per cycle)

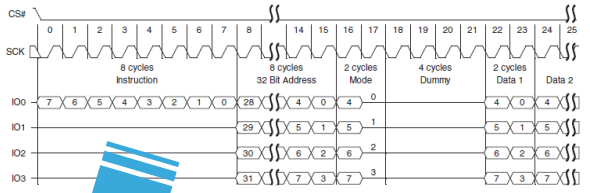


Main features

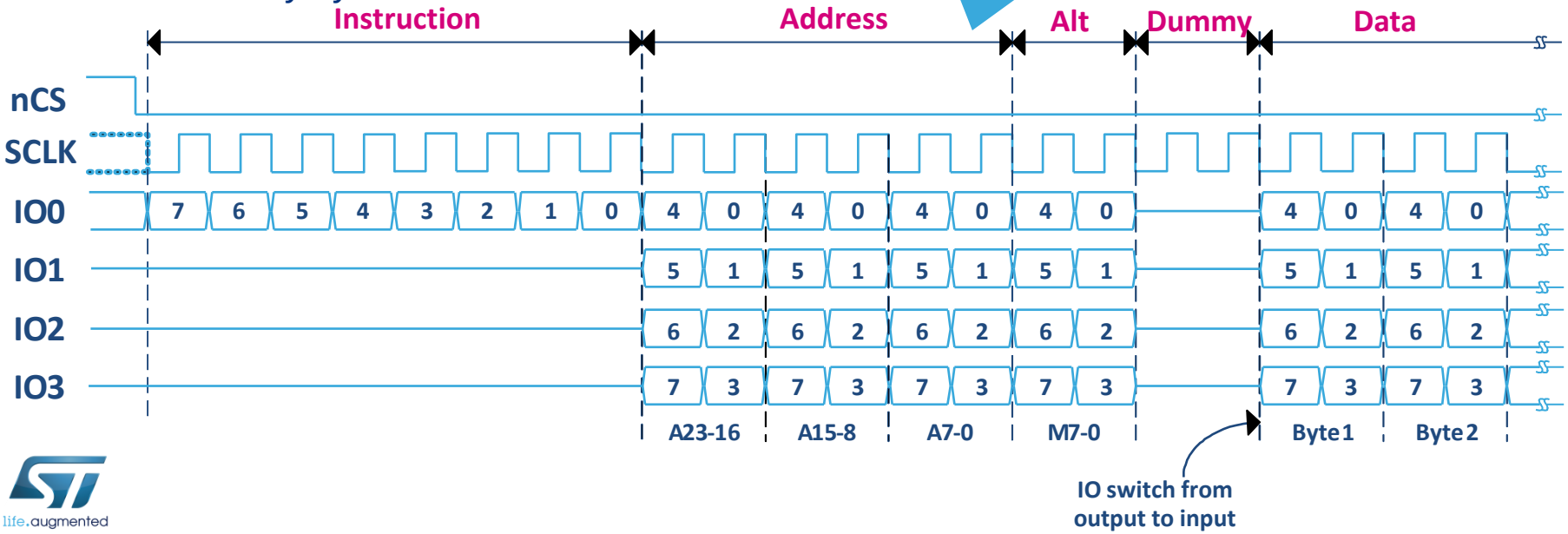
- Three functional modes:
  - Indirect
  - Status-polling
  - Memory-mapped
- Optimized operations
  - Dual-flash mode (8 bits accessing two flash memories in parallel)
  - SDR and **DDR** support
- Fully programmable
  - Opcode for both indirect and memory mapped mode
  - Frame format for both indirect and memory mapped mode
- Integrated FIFO for reception and transmission
  - 8, 16, and 32-bit data accesses are allowed
  - DMA channel for indirect mode operations
- Interrupt generation on FIFO threshold, timeout, operation complete, and access error

Frame format

- Each of the 5 phases is fully configurable
  - Enabled or not
  - Lengh
  - Number of lanes
- Exemple of Read configuration
  - Instruction on 1 lane
  - Address, Alternate & Data on 4 lanes
  - 2 dummy cycles



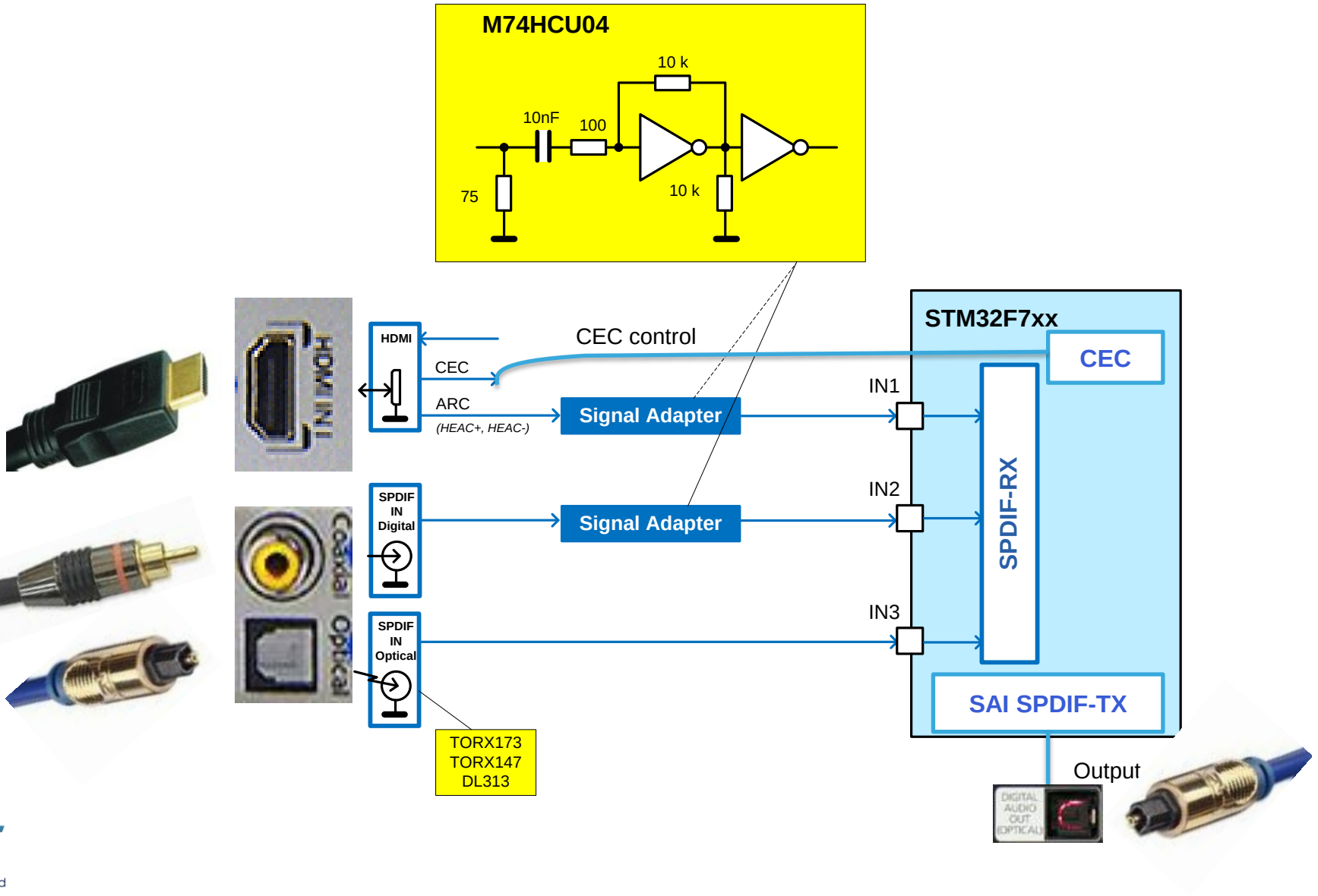
Example : read



Sony/**P**hilips **D**igital **I**nter**F**ace receiver
(SPDIF-RX)

life.augmented

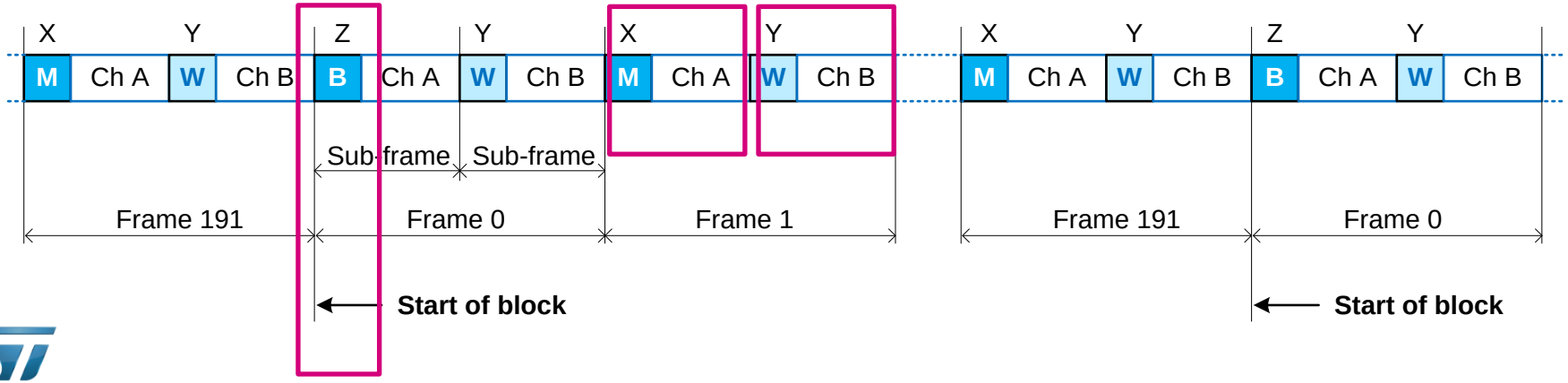
SPDIF-RX Usage



SPDIF Protocol Overview

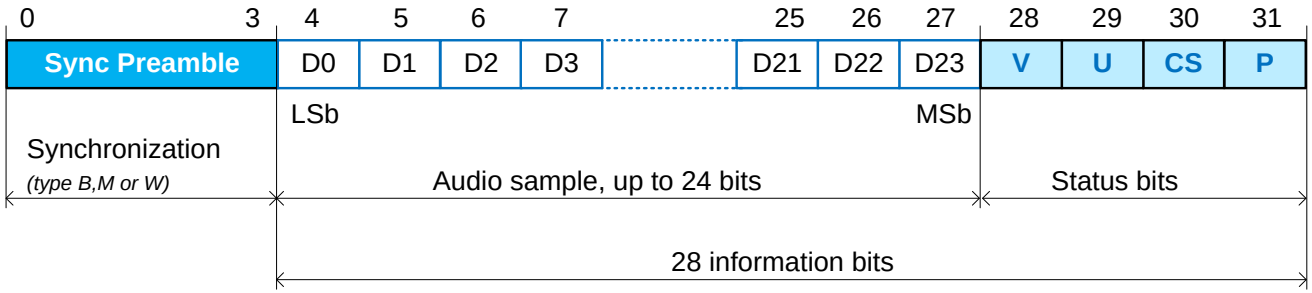
SPDIF IEC60958 (1/3)

- The block structure is used to organize the Channel Status, and User information.
  - Each block contains 192 frames
  - Each frame contains 2 sub-frames
  - Each sub-frame contains 32 bits
  - A preamble allows the detection of the block and sub-frame boundaries
    - Preamble **B** detects the start of new block, and the start of a Channel A
    - Preamble **M** detects the start of a Channel A (when it is not a block boundary)
    - Preamble **W** detects the start of a Channel B



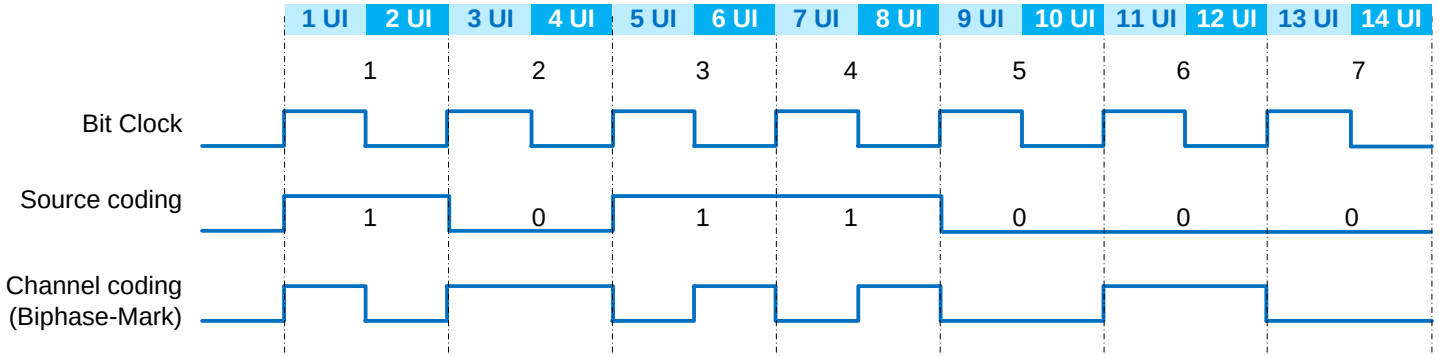
SPDIF IEC60958 (2/3)

- Each sub-frame contains
  - The preamble
  - Up to 24-bit data
  - 4 Status bits
    - **V** is the validity bit, it means that the current sample can be directly converted into an analog signal.
    - **P** is the parity bit of the received sub-frame, it is used to check the received sub-frame
    - **U**: Is the User data channel, each message is composed of 192 bits per sub-frame
    - **CS**: Is the Channel Status, each message is composed of 192 bits per sub-frame (i.e. sampling rate, sample length….)

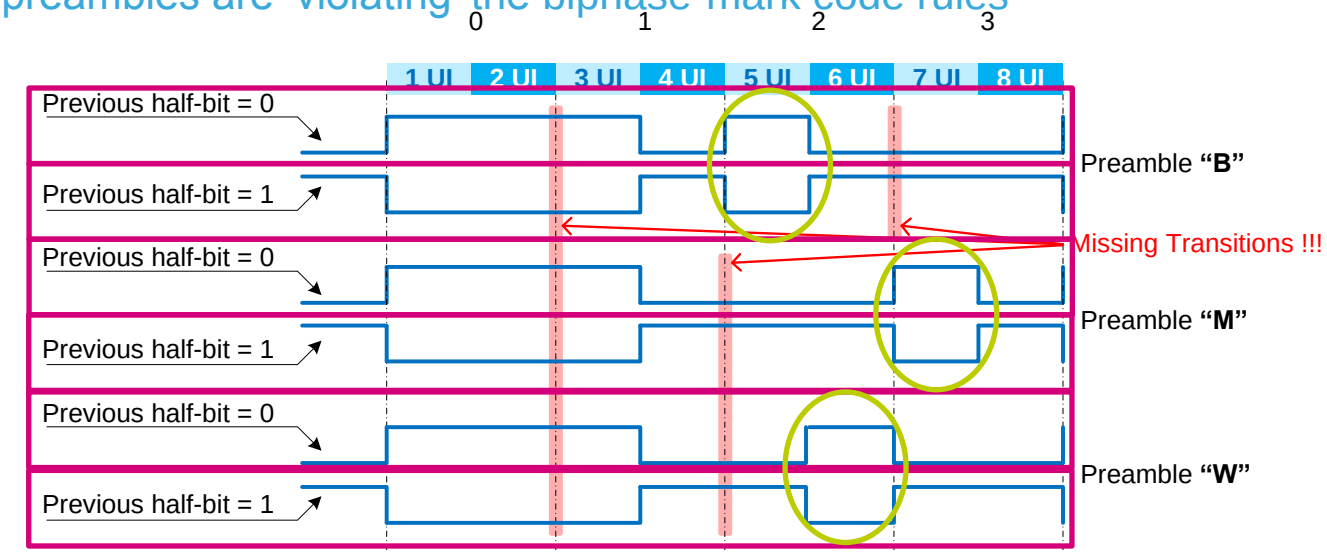


SPDIF IEC60958 (3/3)

- Biphase-mark data encoding



- Preambles
  - The preambles are 'violating' the biphase-mark code rules



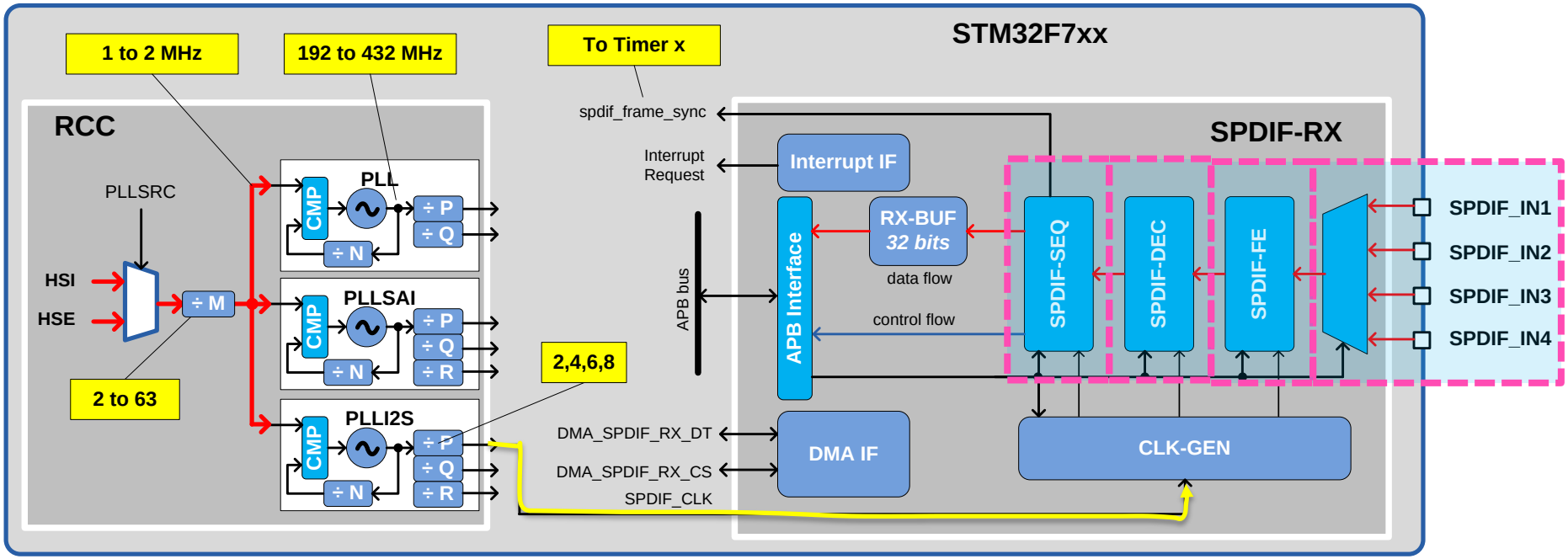
SPDIF-RX Introduction

The SPDIF-RX interface (**S**ony/**P**hilips **D**igital **I**nter**F**ace) is able to receive an S/PDIF flow compliant with the IEC-60958 and IEC-61937.

Features List:

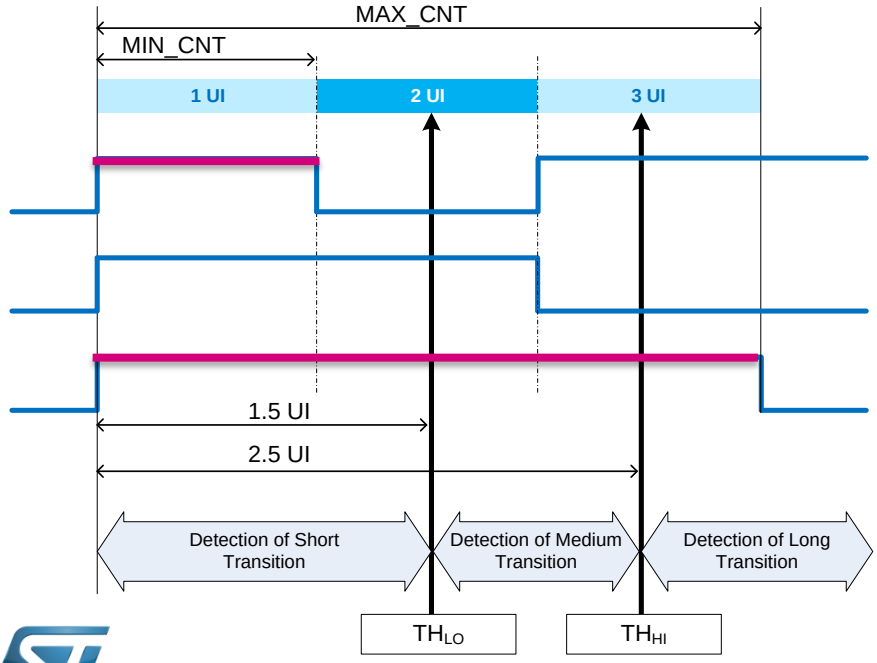
- Possibility to select the audio stream from 1 of the 4 inputs
- Automatic symbol rate detection
- Stereo Stream **up to 192 kHz** supported
- Support Audio IEC-60958 (i.e. non encoded stereo stream)
- Support Audio IEC-61937 (i.e. encoded audio stream such as Dolby Digital)
- Support of DMA interface for:
  - Data stream
  - Control stream
- Interrupt capabilities

SPDIF-RX Block Diagram



Synchronization

- The receiver must be able to estimate the time interval between two transitions and defines if it is:
  - A part of a ‘1’ symbol
  - A ‘0’ symbol
  - A long pulse (preamble)



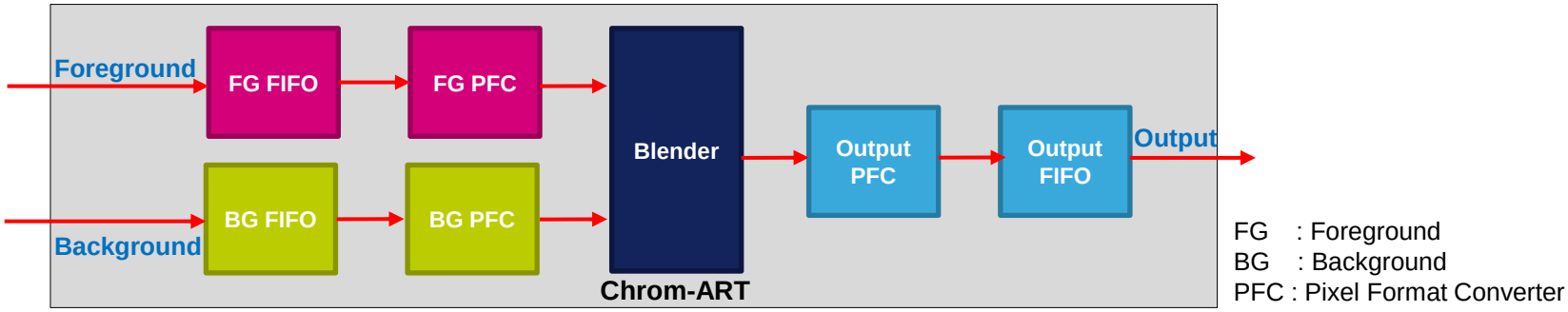
- The receiver estimates two thresholds:
  - THLO and THHI in order to distinguish the transition type.
- If the time interval between two transition is:
  - Lower than THLO → short transition detected
  - Higher than THHI → Long transition detected
  - Between THLO and THHI → Medium transition detected
- The synchronization is performed in two steps:
  - The COARSE SYNC
  - The FINE SYNC

Chrom-ART Accelerator™ (DMA2D)

life.augmented

Chrom-ART

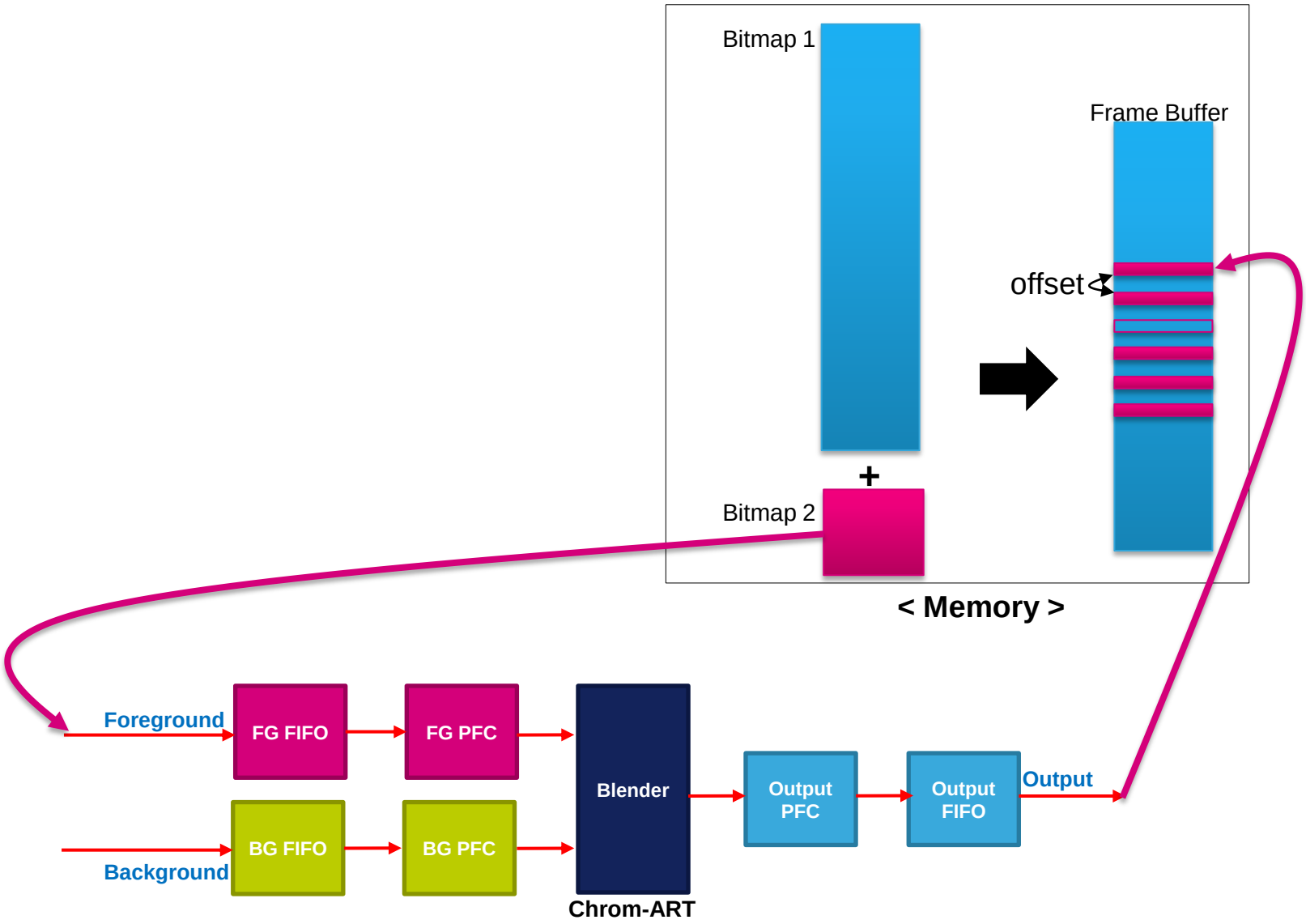
- Provides hardware acceleration for **graphical operations**
  - Graphic oriented 2D DMA
  - Planes blending & pixel format conversion
  - Specific modes for anti aliased fonts



Application benefits

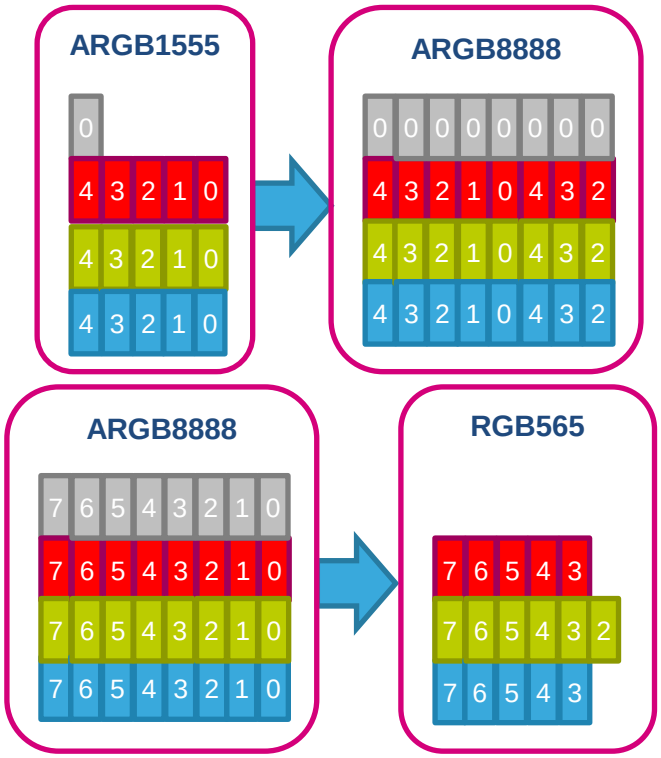
- Offload CPU for graphical operation
- One pixel per cycle calculation
- Integrated pixel format converter & blender
- Simple integration through graphical stack

Rectangle copy

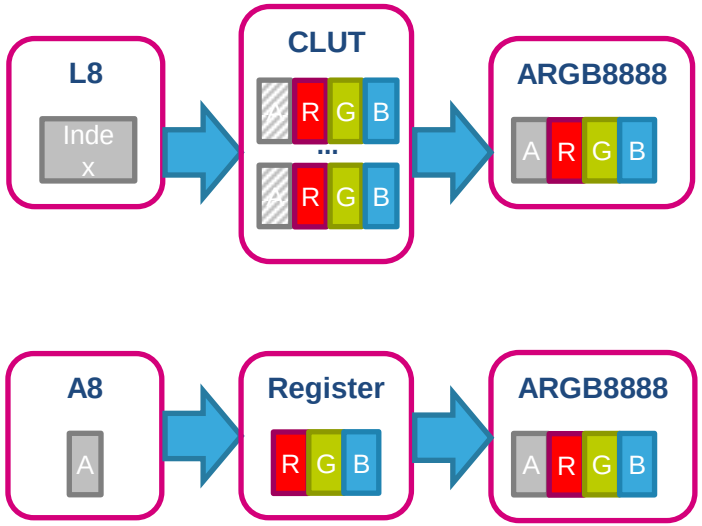


Pixel Format Conversion

- **Direct Mode**
  - ARGB8888, ARGB4444 or ARGB1555
  - RGB888 or RGB565 (with alpha in register)
  - Alpha can be replaced or modulate

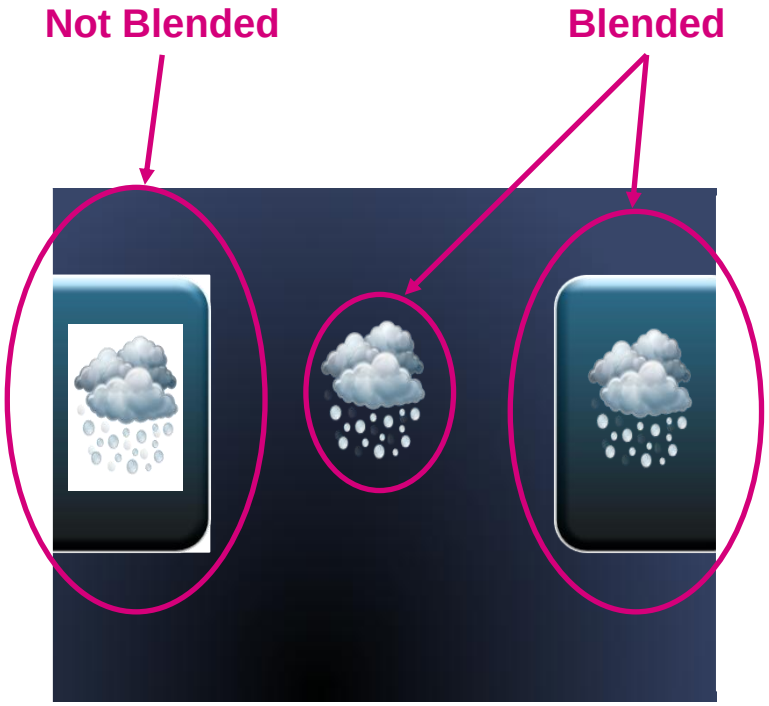


- **Indirect Mode**
  - L8 or L4 and a Color Lock Up Table (**CLUT**)
  - A8 or A4 and a color register (for fonts)
  - Mixed AL88 or AL44

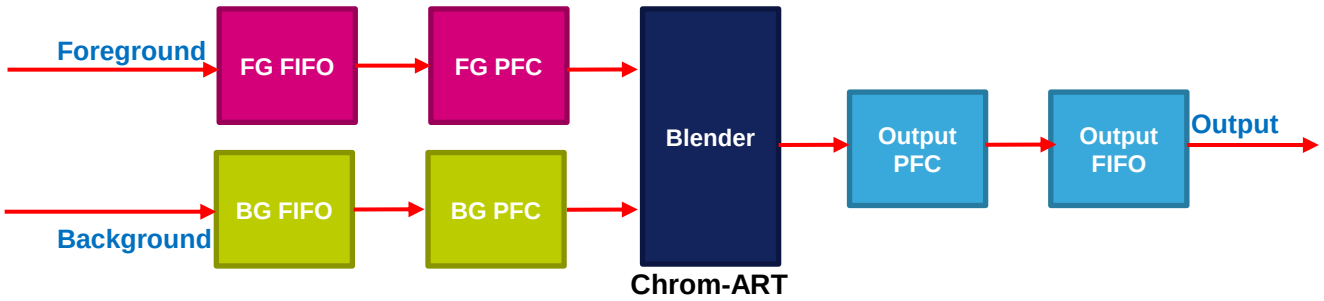


Blending

Fully hardware blending process



- **Hardware Blending**
  - Blend Foreground & Background pixels
  - 1 pixel generated per cycle
  - Native ARGB8888 operation
  - Input data are converted into ARGB8888 by their respective PFC
  - Output data has also its own PFC



Font Special Modes

Efficient support for *anti aliased* bitmap fonts

Aliased

Anti-aliased

- **Using A8 or A4 coding**
  - Only the Alpha channel is stored in the memory
  - The ChromART add a programmed color

Colored

Key features

- **2D DMA with graphical oriented features with 4 operating modes**
  - register-to-memory
  - memory-to-memory
  - memory-to-memory with PFC
  - memory-to-memory with PFC and blending
- **User programmable sources and destination addresses on the whole memory**
- **User programmable sources and destination size and offset**
- **User programmable source and destination color format**
  - Up to 11 color formats supported from 4-bit up to 32-bit per pixel with indirect or direct color coding
  - Internal memories for CLUT storage in indirect color mode with automatic loading
  - Alpha value can be modified (source value, fixed value or modulated value)

STM32F7 Demonstration

TouchGFX demo

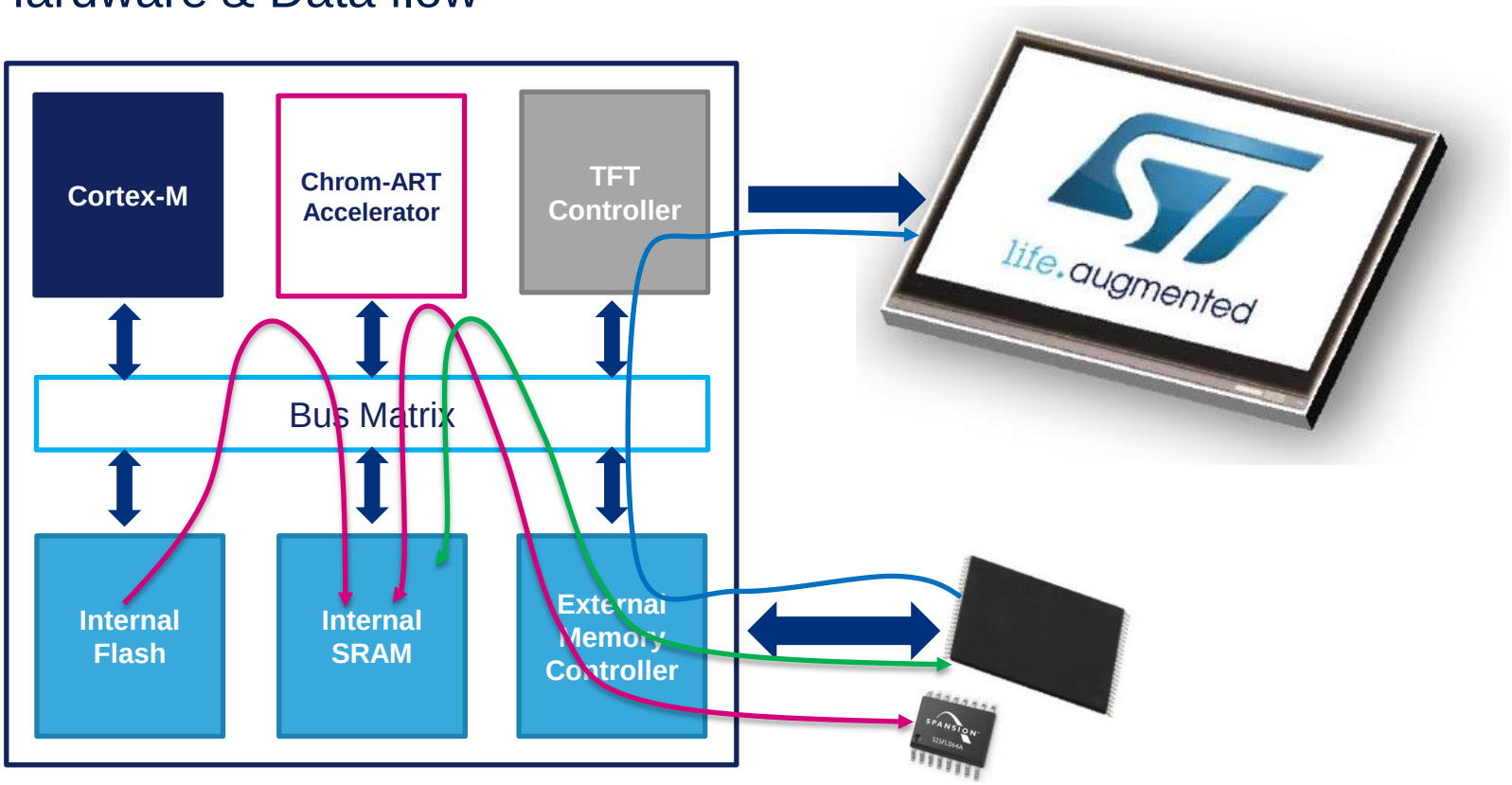
1. Animation*
   - Set of bitmapped image
2. Draggable image*
3. Vector graphics: the drawing is made by the CPU in an offline buffer, and than composed by the hardware accelerator.

- All the image are located in external flash.
- Graphic library is managed by FreeRTOS
- The driver for the F7 is not full optimized.

* Use Chrom-ART accelerator

TouchGFX demo

- Hardware & Data flow



Creation of an object in a memory device by the Chrom-ART

Update/Creation of the frame buffer in the external RAM by the Chrom-ART

TFT controller data flow

Demo

그래픽 Demo는 F7과 F4에서 똑같은 동작을 구현하였습니다.
많은 그래픽 작업이 진행되면서 왼쪽 아래에 표시되는 FPS, 초당 화면update를 비교하면 , F7과 F4의 성능차이를 보실 수 있는데요.. F4은 초당64프레임을 보여주는데 비해 F7은 두배인 128프레임을 보여줍니다. 앞서 설명드린 F7의 성능이 그대로 보여지고 있습니다.

이제 MCU로도 보시는 것과 같은 고급스러운 그래픽 유저 인터페이스를 부담없이 사용할 수 있습니다.

이상으로 STM32F7 웨비나를 마치겠습니다.

감사합니다.