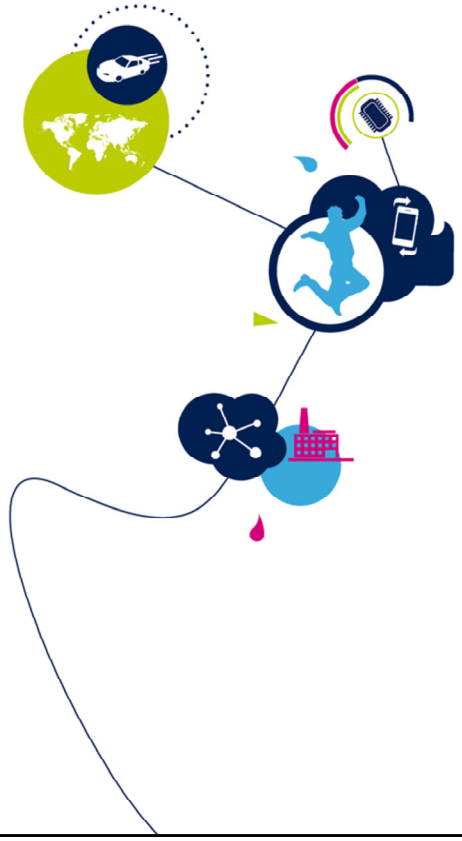
STM32WB- ARM® M0+ Core

ARM Cortex®-M0+ Core

Revision 1.0

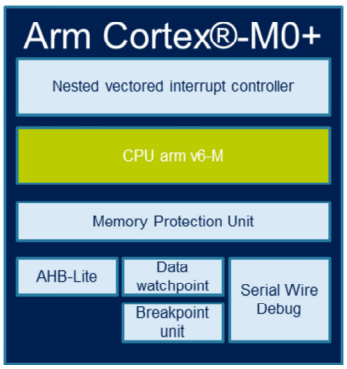


life.augmented

Hello, and welcome to this presentation of the ARM® Cortex®-M0+ core which is embedded in all products of the STM32WB microcontroller family.

Cortex-M0+ processor overview

- ARMv6-M architecture
- Von Neuman architecture, 2-stage pipeline
- Single-issue architecture
- MULTIPLY in 1-cycle

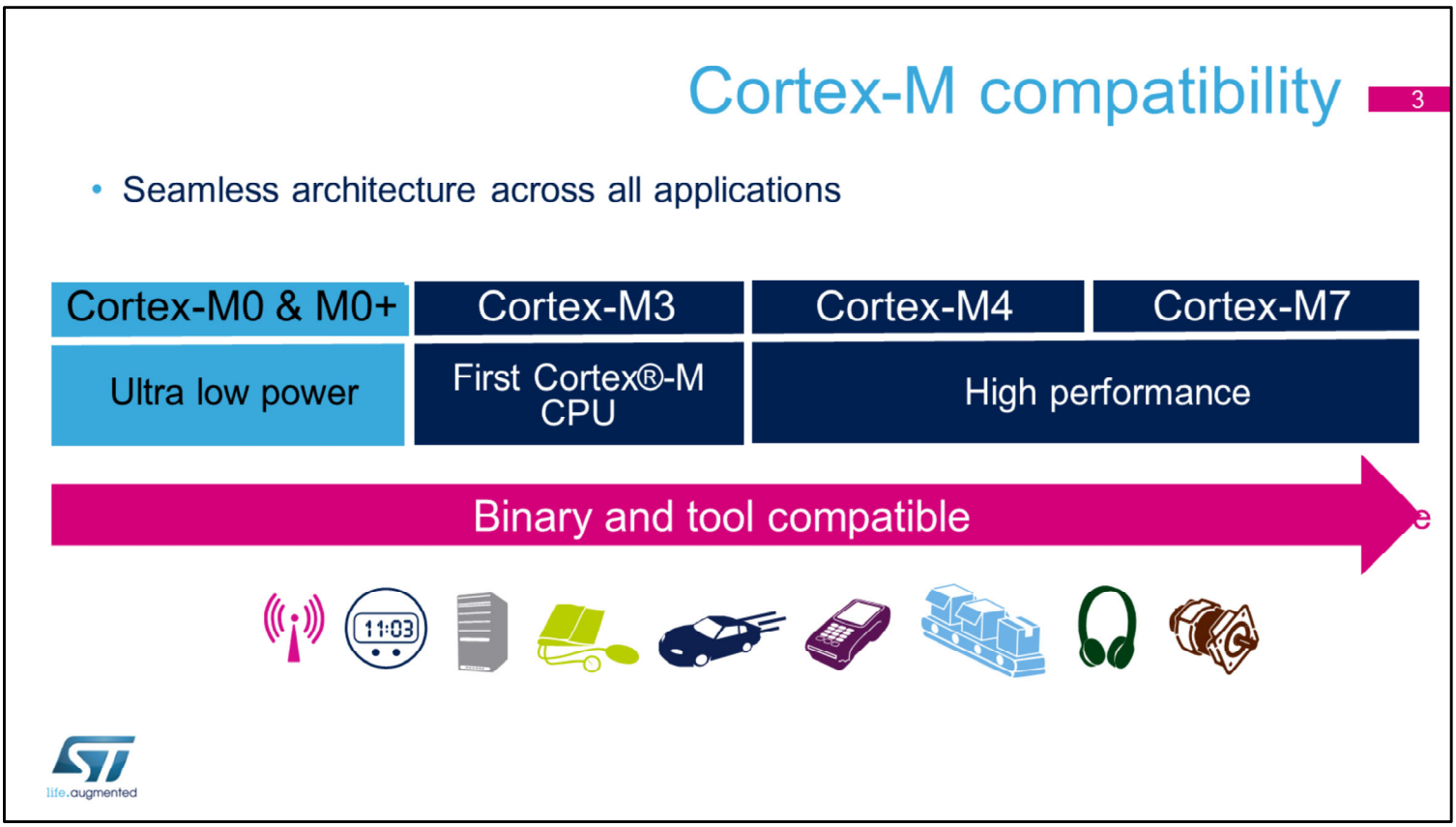


Ultra low power design	Very compact code
Low power consumption and high energy efficiency	Except control instructions and branch and link, all instructions are 16 bits long

life.augmented

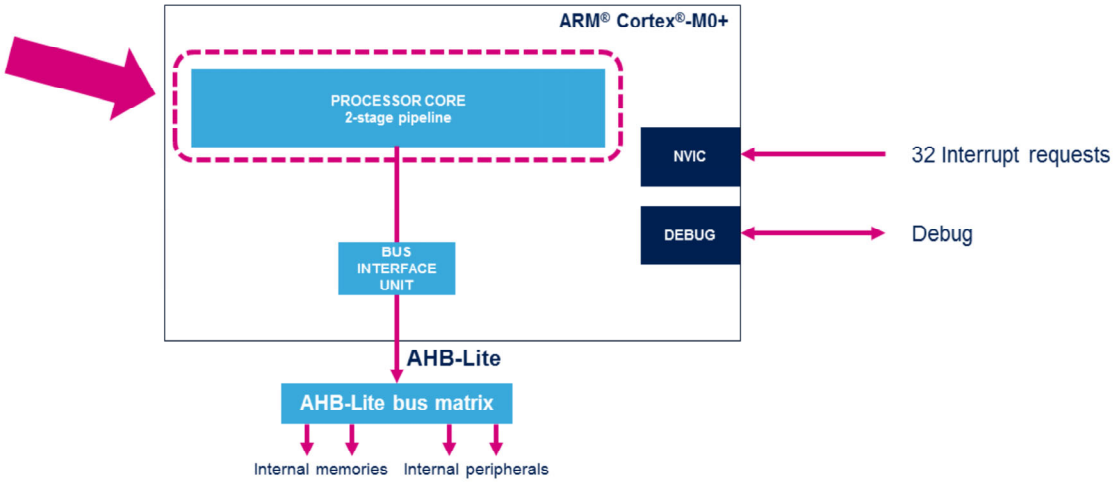
The Cortex®-M0+ core is part of the ARM Cortex-M group of 32-bit RISC cores. It implements the ARMv6-M architecture and features a 2-stage pipeline.

The Cortex®-M0+ has a unique AHB-Lite master port.



STM32WB microcontrollers integrate an ARM® Cortex®-M0+ core in order to benefit from the incomparable performance per milliwatt ratio.
All Cortex®-M CPUs have a 32-bit architecture.
The Cortex®-M3 was the first Cortex®-M CPU released by ARM.
Then ARM decided to distinguish two product lines: high performance and low power, while maintaining the compatibility between them.
The Cortex®-M0+ belongs to the low power product line. It is designed for battery-powered devices, very sensitive to power consumption.

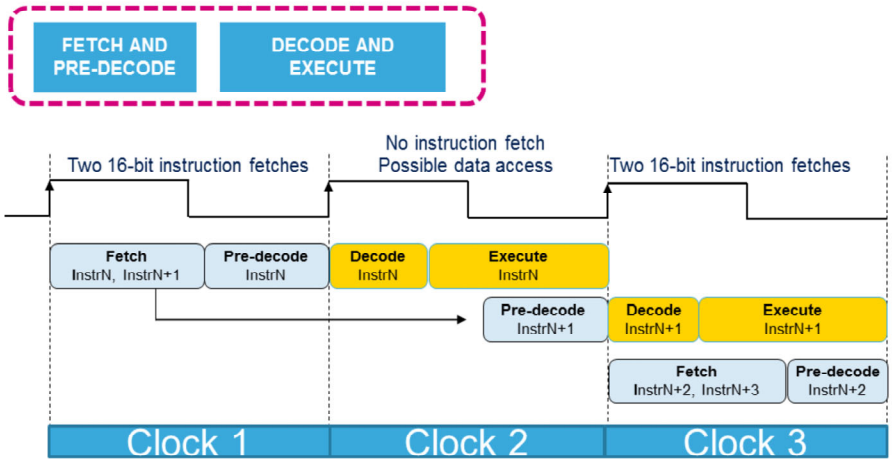
Core architecture overview



life.augmented

The Cortex®-M0+ core delivers more performance than the Cortex®-M0 core thanks to the 2-stage instruction pipeline. Let's start our description of the CPU by the processor core in charge of fetching and executing instructions.

ARM Cortex-M0+ → 2-stage pipeline



life.augmented

Most V6-M instructions are 16 bits long. There are only six 32-bit instructions and most of them are control instructions, rarely used. However the branch and link instruction, which is used to call a sub-program is also 32 bits long, in order to support a large offset between this instruction and the label pointing to the next instruction to be executed.
Ideally one 32-bit access for every two 16-bit instructions, results in less fetches per instruction.
During clock number 2, no instruction fetch occurs. The AHB Lite port is available to execute a data access when instruction N is a load/store instruction.

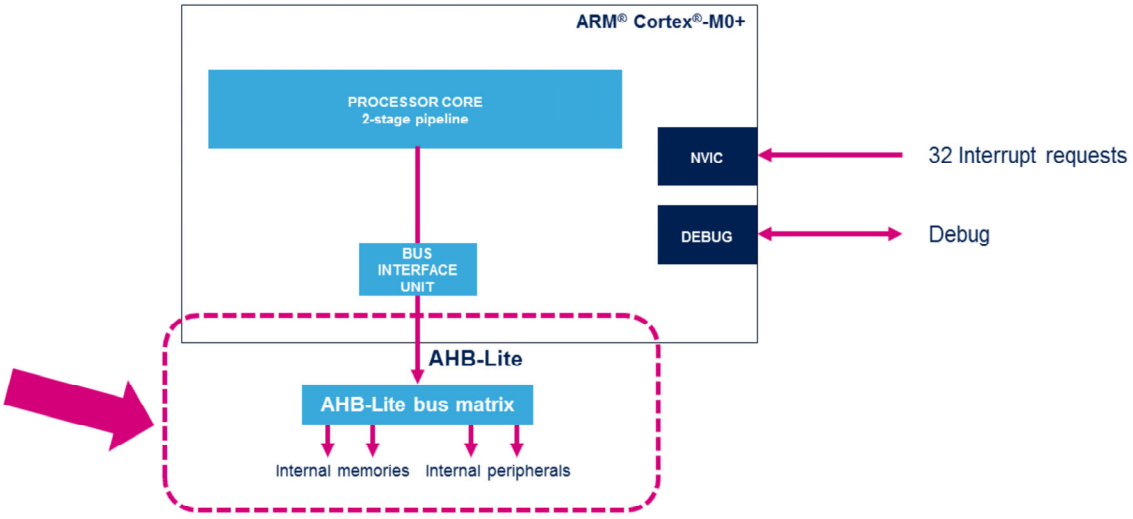
Branch performance

- Cortex®-M0+ core
  - Maximum two 16-bit branch shadow instructions

```
                Inst0
                B           Label       ; Branch to Label
                Inst1                   ; Branch shadow instruction
                Inst2                   ; Branch shadow instruction
...             ...
Label:          InstN
                InstN+1
```

On a given branch, fewer pre-fetched instructions are wasted (thanks to the 2-stage pipeline).
In clock number 1, the processor fetches Inst0 and an unconditional branch instruction.
In clock number 2, it executes Instr0.
In clock number 3, it executes the branch instruction while fetching the two next sequential instructions Inst1 and Inst2 called branch shadow instructions.
In clock number 4, the processor discards Inst1 and Inst2 and fetches InstrN and InstN+1.
Cortex-M0, M3 and M4 implement a 3-stage pipeline: Fetch, Decode and Execute. The number of branch shadow instructions is larger: up to four 16-bit instructions.

Core architecture overview



The Cortex®-M0+ has neither a cache nor internal RAM. Consequently any instruction fetch transaction is steered to the AHB-Lite interface and any data access is steered to the AHB-Lite interface.
Note that the STM32WB implements a SoC-level cache, external to the CPU.
The AHB-Lite master port is connected to a bus matrix, enabling the CPU to access memories and peripherals.
Since transactions are pipelined on AHB-Lite, the best throughput is 32 bits of data or instructions per clock, with a minimum 2-clock latency.

References

- For more details, please refer to the following documentation:
  - STM32G0 Series Cortex®-M0+ processor programming manual (PM0223)
  - ARM website at the following link:
    - http://www.arm.com/products/processors/cortex-m/cortex-m0+-processor.php

life.augmented

For more details, please refer to these application notes and the Cortex®-M0+ programming manual available on www.st.com website.
Also visit the ARM website where you will find more information about the Cortex®-M0+ core.