XMC4000 System

XMC™ microcontrollers
September 2016

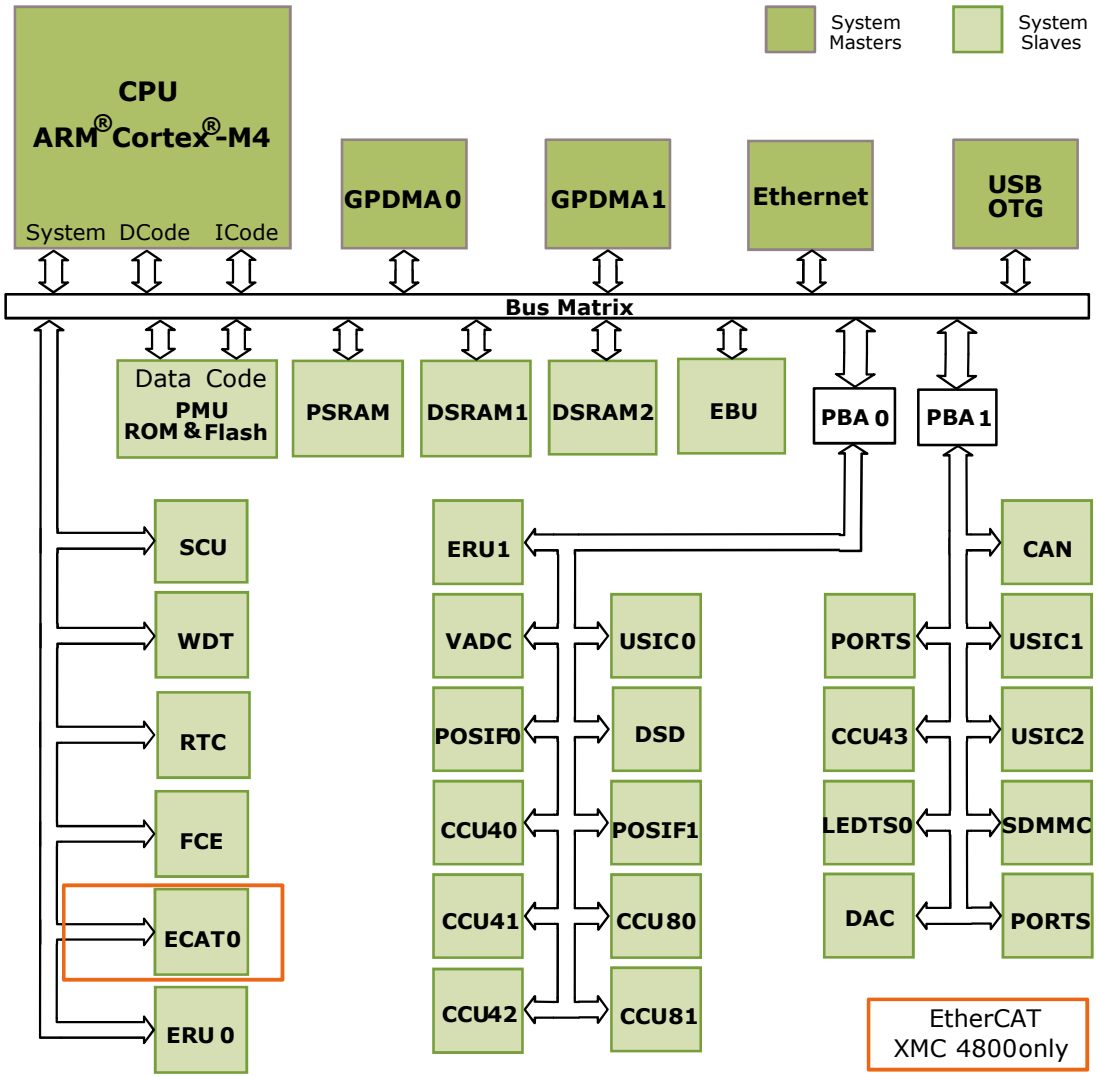
Agenda

1. XMC4000 and ARM® Cortex®-M4
2. Memories
3. Bus system
4. Interrupt system
5. Clock
6. Reset
7. Power
8. Ports

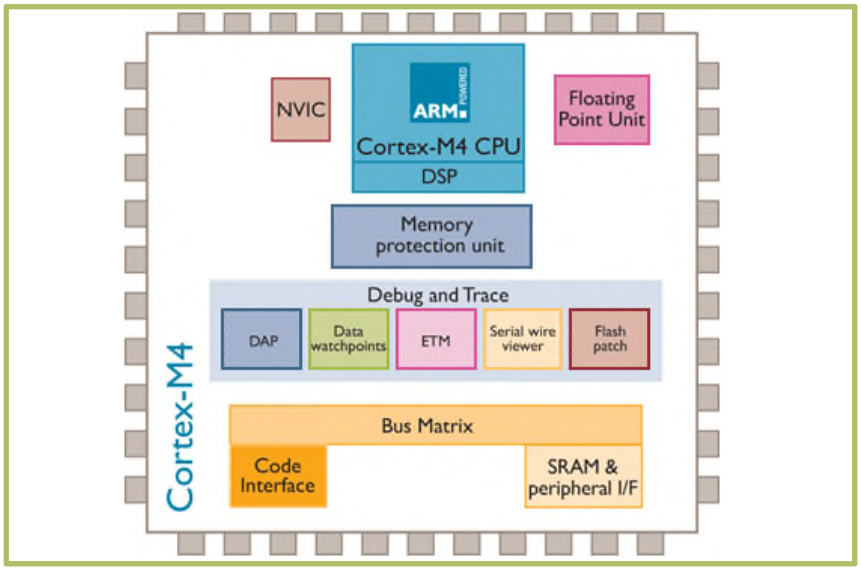
Agenda

1. XMC4000 and ARM® Cortex®-M4
2. Memories
3. Bus system
4. Interrupt system
5. Clock
6. Reset
7. Power
8. Ports

XMC4000 block diagram



Architecture of Cortex®-M4



Highlights

Cortex®-M4 is a high efficiency processor core with 1.25 DMIPS/MHz. Due to the Thumb-2 instruction set it reaches excellent code density.
Its optimised interrupt controller allows use in hard real-time applications.
Widespread standard core offers a broad variety of software tools and components.

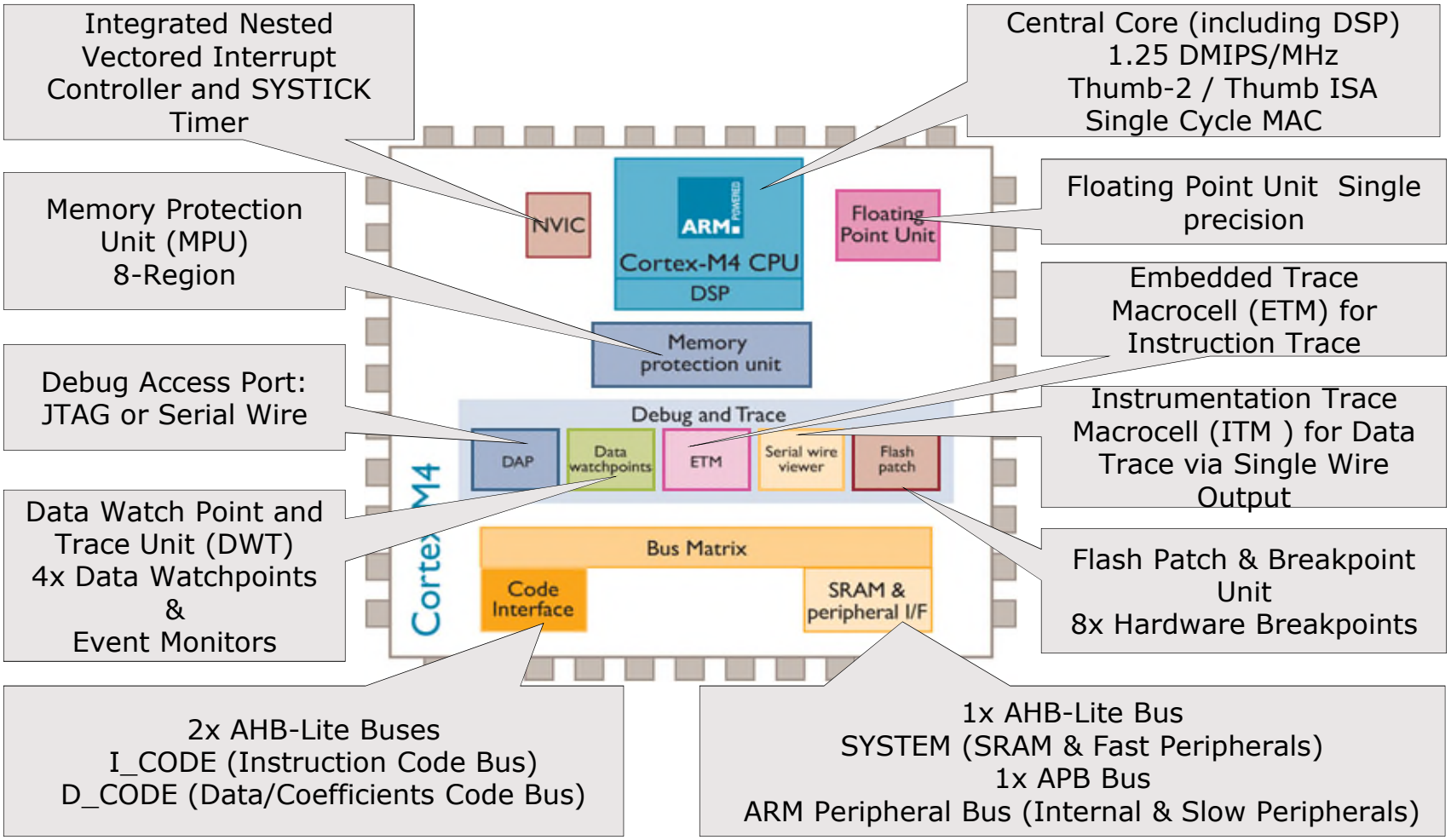
Key feature

- Standard core
- DSP instruction set
- Single precision floating point unit

Customer benefits

- Availability of third party tools, software libraries and engineering force
- Offers computing performance for real-time applications
- Avoids complex scaling and allows direct programming in C-language

Architecture of Cortex®-M4 Standard core (1/2)



Source: http://www.arm.com

Architecture of Cortex®-M4 Standard core (2/2)

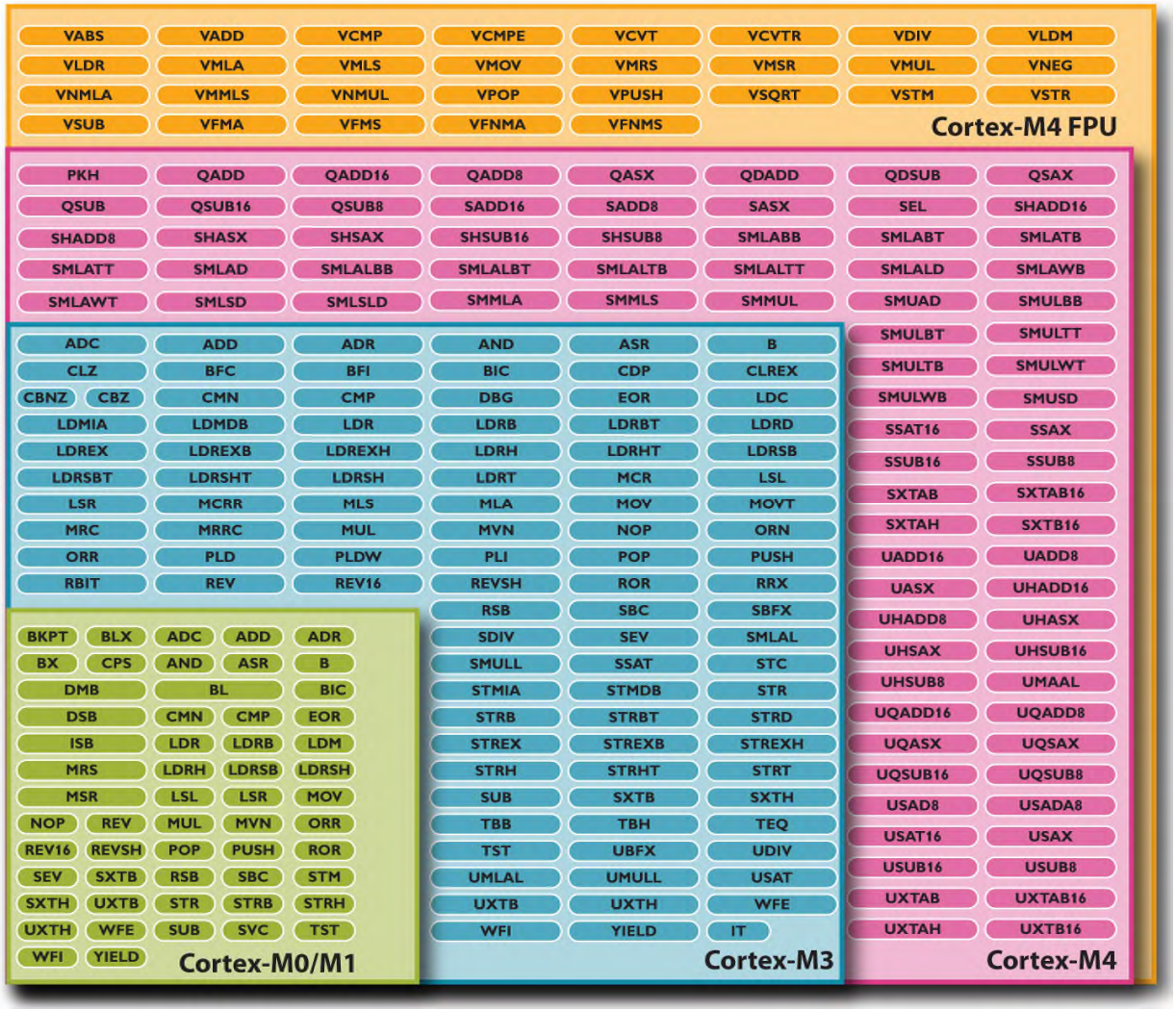
Cortex®-M architecture comparison

ARM Cortex®-M	Thumb	Thumb-2	ARM architecture	Core architecture
Cortex®-M0	Most	Subset	ARMv6-M	Von Neumann
Cortex®-M3	Entire	Entire	ARMv7-M	Harvard
Cortex®-M4	Entire	Entire	ARMv7E-M	Harvard

ARM Cortex®-M	Hardware multiply	Hardware divide	Saturated math	DSP extensions	Floating-point
Cortex®-M0	1 or 32 cycle	No	No	No	No
Cortex®-M3	1 cycle	Yes	Yes	No	No
Cortex®-M4	1 cycle	Yes	Yes	Yes	Optional

Source: Wikipedia

Architecture of Cortex®-M4 DSP instruction set (1/2)



Cortex®-M4F is an ARMv7E-M architecture.

It has a full Thumb and Thumb-2 instruction set.

Cortex®-M4F has a single precision floating point unit.

Source: http://www.arm.com

Architecture of Cortex®-M4
DSP instruction set (2/2)

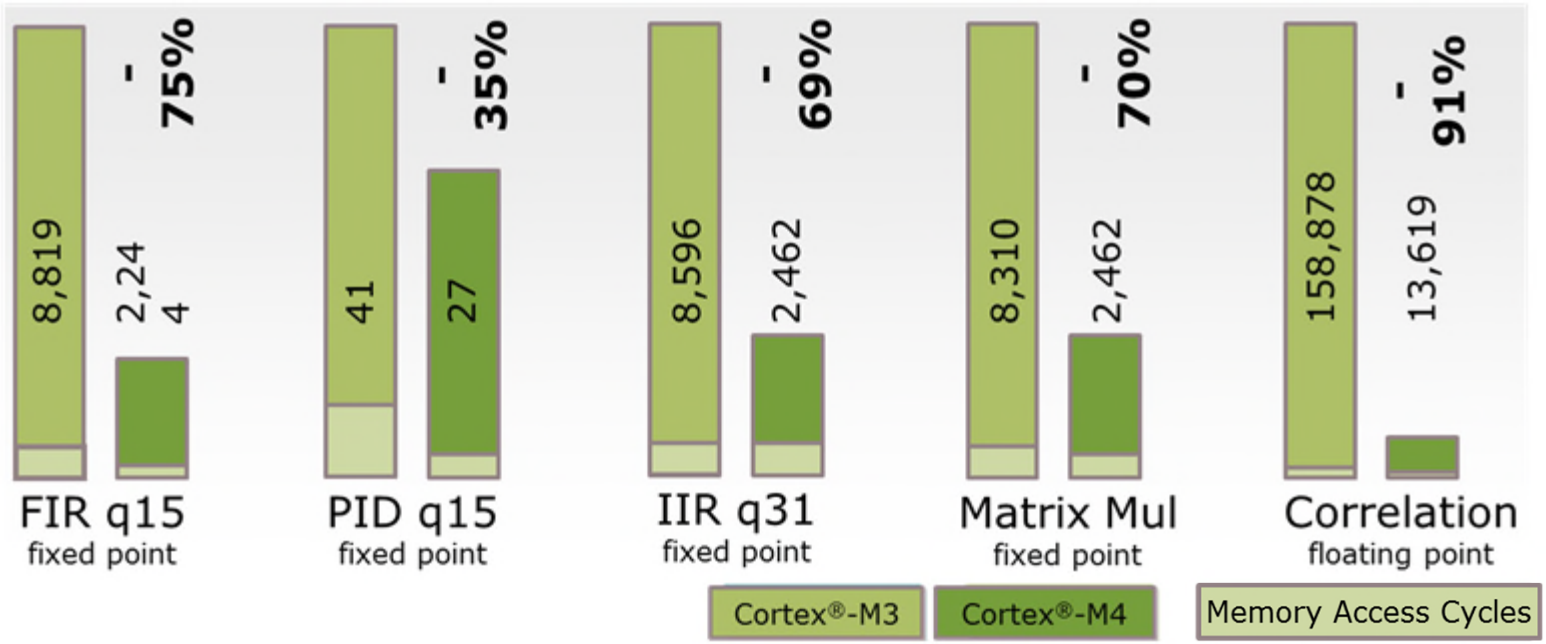
		Cycle counts		
CLASS	INSTRUCTION	ARM9E-S	CORTEX-M3	Cortex-M4
Arithmetic	ALU operation (not PC)	1 - 2	1	1
	ALU operation to PC	3 - 4	3	3
	CLZ	1	1	1
	QADD, QDADD, QSUB, QDSUB	1 - 2	n/a	1
	QADD8, QADD16, QSUB8, QSUB16	n/a	n/a	1
	QDADD, QDSUB	n/a	n/a	1
	QASX, QSAX, SASX, SSAX	n/a	n/a	1
	SHASX, SHSAX, UHASX, UHSAX	n/a	n/a	1
	SADD8, SADD16, SSUB8, SSUB16	n/a	n/a	1
	SHADD8, SHADD16, SHSUB8, SHSUB16	n/a	n/a	1
	UQADD8, UQADD16, UQSUB8, UQSUB16	n/a	n/a	1
	UHADD8, UHADD16, UHSUB8, UHSUB16	n/a	n/a	1
	UADD8, UADD16, USUB8, USUB16	n/a	n/a	1
	UQASX, UQSAX, USAX, UASX	n/a	n/a	1
	UXTAB, UXTAB16, UXTAH	n/a	n/a	1
	USAD8, USADA8	n/a	n/a	1
Multiplication	MUL, MLA	2 - 3	1 - 2	1
	MULS, MLAS	4	1 - 2	1
	SMULL, UMULL, SMLAL, UMLAL	3 - 4	5 - 7	1
	SMULBB, SMULBT, SMULTB, SMULTT	1 - 2	n/a	1
	SMLABB, SMLBT, SMLATB, SMLATT	1 - 2	n/a	1
	SMULWB, SMULWT, SMLAWB, SMLAWT	1 - 2	n/a	1
	SMLALBB, SMLALBT, SMLALTB, SMLALTT	2 - 3	n/a	1
	SMLAD, SMLADX, SMLALD, SMLALDX	n/a	n/a	1
	SMLSD, SMLSDX	n/a	n/a	1
	SMLSLD, SMLSLD	n/a	n/a	1
	SMMLA, SMMLAR, SMMLS, SMMLSR	n/a	n/a	1
	SMMUL, SMMULR	n/a	n/a	1
	SMUAD, SMUADX, SMUSD, SMUSDX	n/a	n/a	1
	UMAAL	n/a	n/a	1
Division	SDIV, UDIV	n/a	2 - 12	2 - 12

Single Cycle MAC Instructions

Source: http://www.arm.com

Architecture of Cortex®-M4
Single precision floating point unit

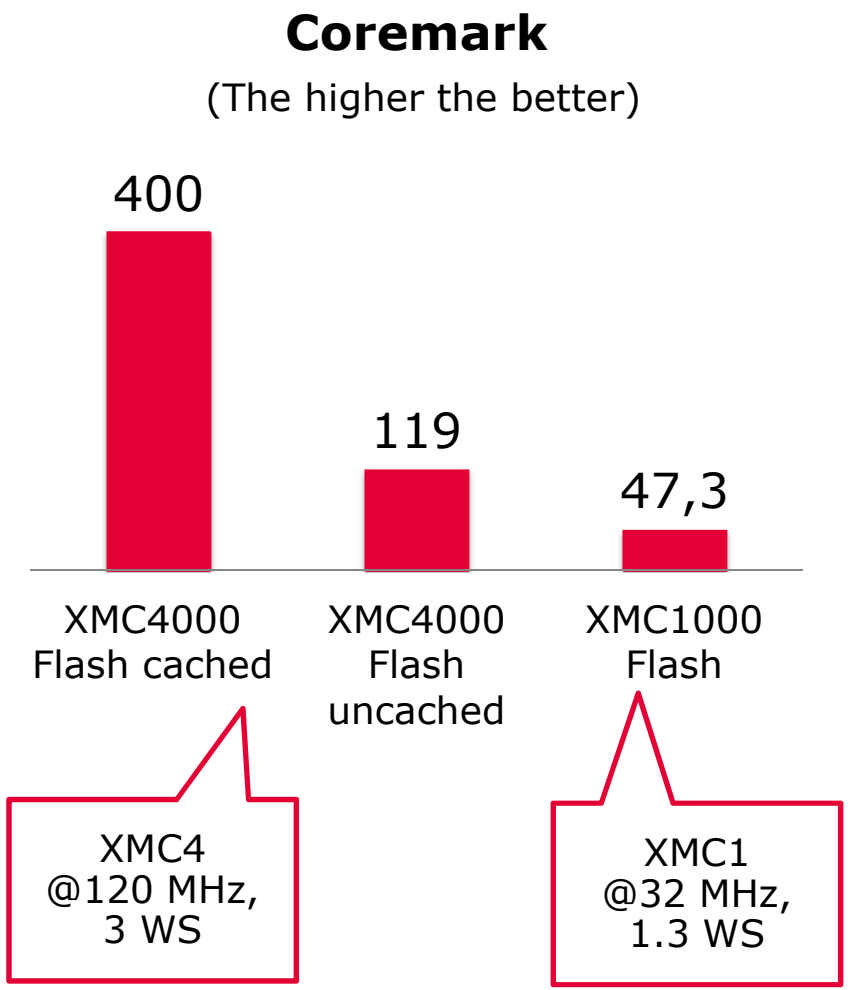
- Cortex®-M4 SIMD + FPU
  - Fix point: ~2x faster
  - Floating point: ~10x faster

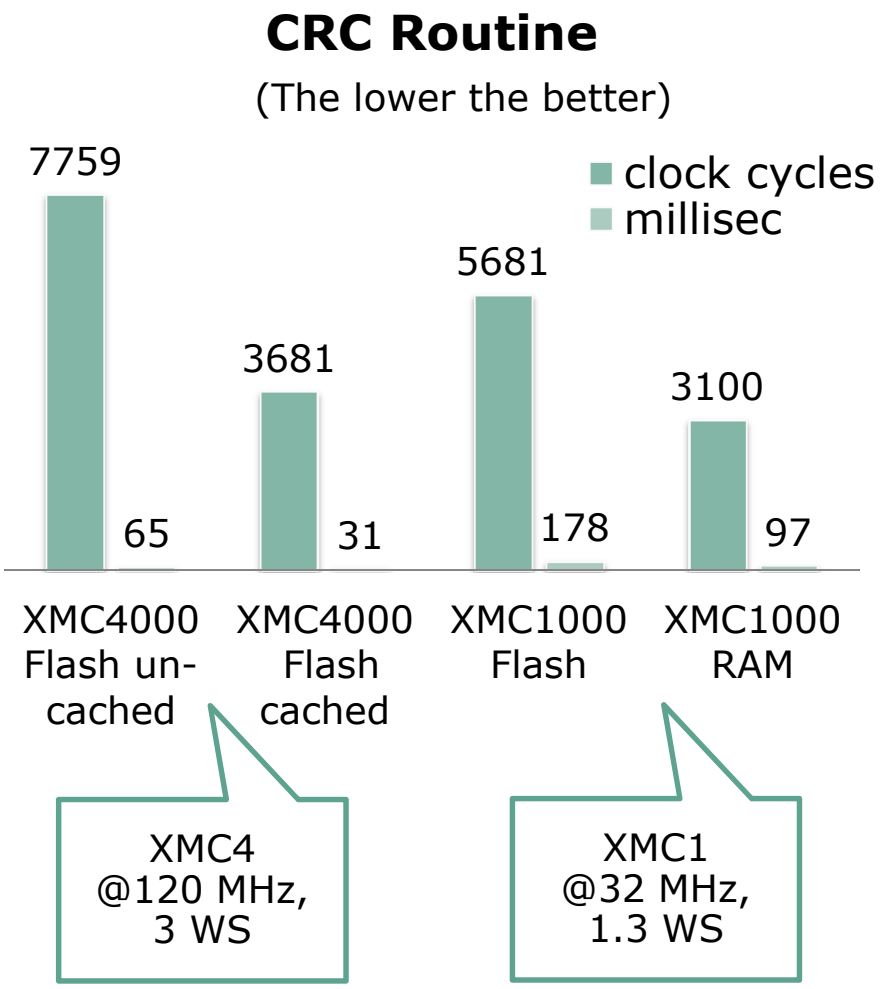


Cycles: smaller numbers are better

Source: http://www.arm.com

Architecture of Cortex®-M4
CPU performance benchmark

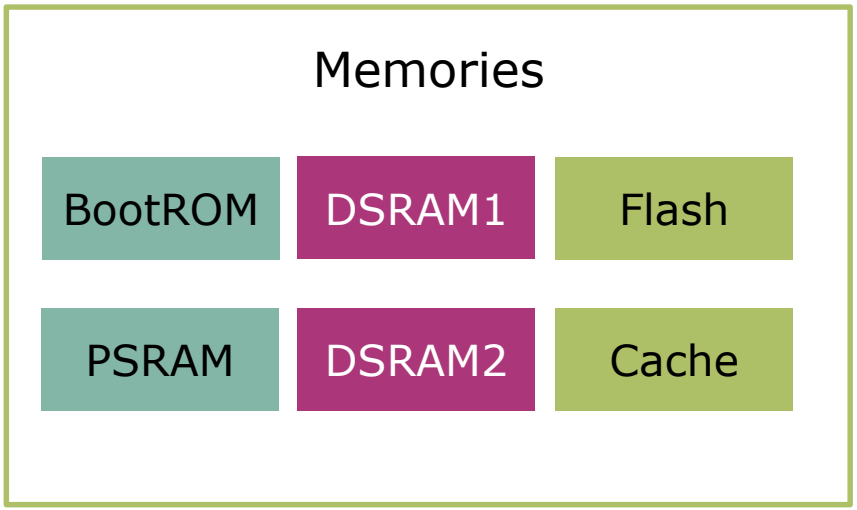




Agenda

1. XMC4000 and ARM® Cortex ®-M4
2. Memories
3. Bus system
4. Interrupt system
5. Clock
6. Reset
7. Power
8. Ports

Memories



Highlights

The memory map is based on standard ARM Cortex®-M4 system memory map.

The peripherals are memory mapped into the linear address space. The on-chip memories provide zero wait state accesses to code and data. Flash module has very high performance with low latency. With the external bus interface (EBU), memory can be extended to 65 MBytes.

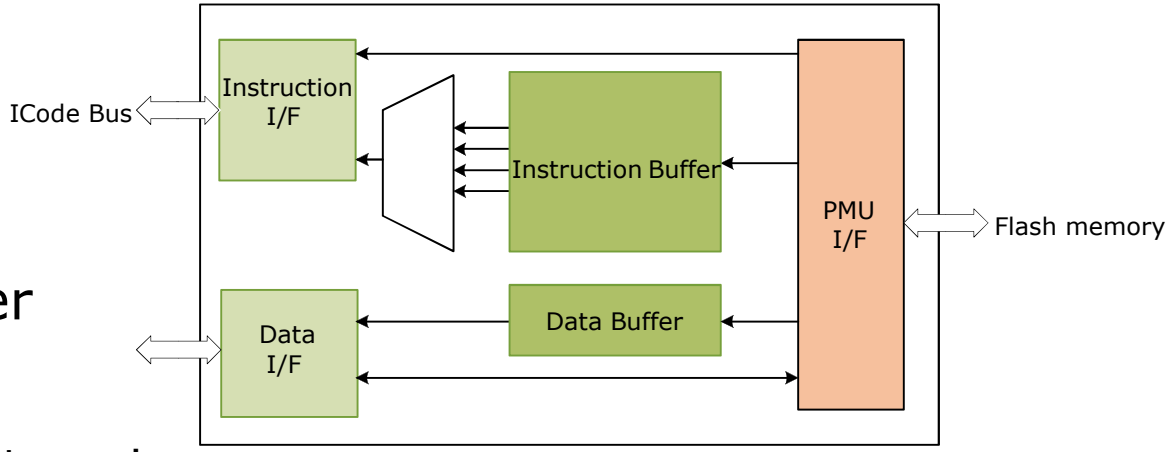
Key feature

- Flash acceleration with cache
- Parity protected RAM and ECC protected Flash
- Flash IP protection

Customer benefits

- Integrated cache reduces code execution time and saves power
- High code/data integrity due to hardware error detection and correction mechanism
- Protect against unauthorized read-out of memory content

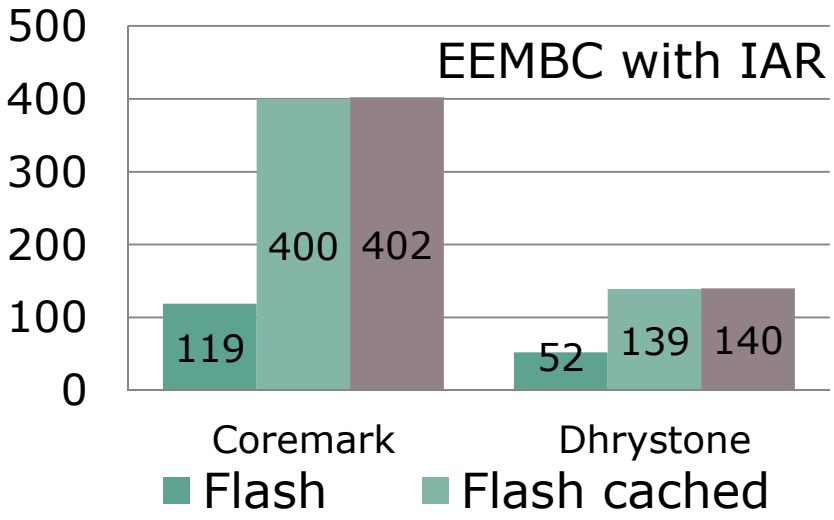
Memories
Flash acceleration with cache (1/2)

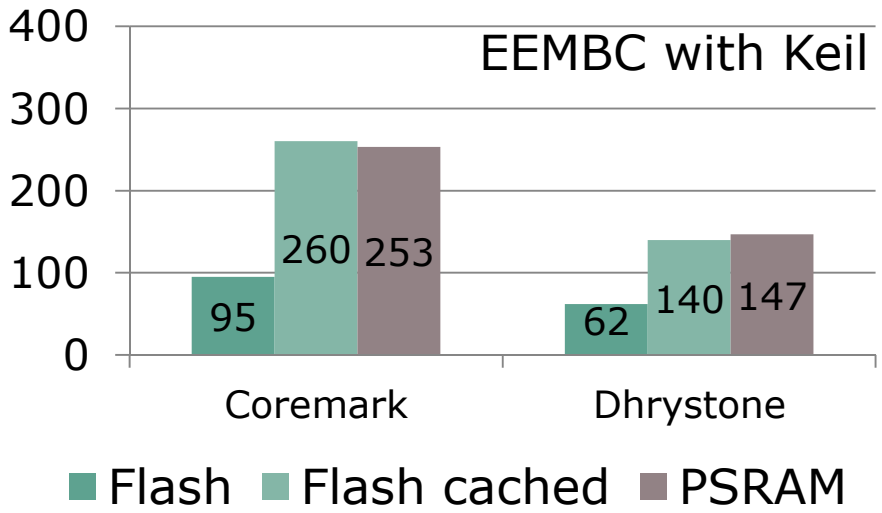


- 256-bit data buffer
  - Single line
  - Critical word first, streaming
- 4-8 Kbyte instruction buffer
  - 2 way set associative, 256-bit line
  - Bypass and invalidation supported
  - LRU replacement policy
  - Critical word first, streaming

Memories
Flash acceleration with cache (2/2)

- Performance for Cortex®-M4 @ 120 MHz, 3 wait states:
  - Execution with cache: ~2.5 – 3.4x faster
  - Note: compiler play a big role in benchmark results





Numbers in clock cycles: higher numbers are better

Memories
Parity protected RAM and ECC protected Flash

- RAM supports parity bit generation and error detection
  - Parity bit can be inverted to force an error for test purposes
  - Hardware support for ClassB Library saves CPU load for volatile memory test
- Flash supports ECC with:
  - Single bit error detection and correction
  - Double bit error detection
- Flash supports margin checks for preventive maintenance
- All memories with zero wait state:
  - 8-bit, 16-bit and 32-bit write access
  - 32-bit read access

Memories
Flash IP protection

- For **software IP protection** there are several modes available which allow both flexibility and safe protection against un-authorized read or manipulation
  - Hierarchical write protection control with 3 levels
  - Password based read protection for temporary disabling protection
  - Global read and sector-specific write granularity
- **Read protection** blocks debug accesses and is globally for the whole flash area.
- **Write protection** blocks flash write and erase accesses and is sector-specific
- **OTP protection**
  - one-time set, never change
  - Granularity: sector-specific, e. g. for a user-specific bootloader

Memories
External memory interface with EBU

- External Bus Interface (EBU) enables communication with external memories and on-chip peripherals
- Supports memory access types:
  - Synchronous
  - Asynchronous
  - Burst Protocol
- Multiple memory types can be accessed in time-multiplex

Memories
Flash memory characteristics

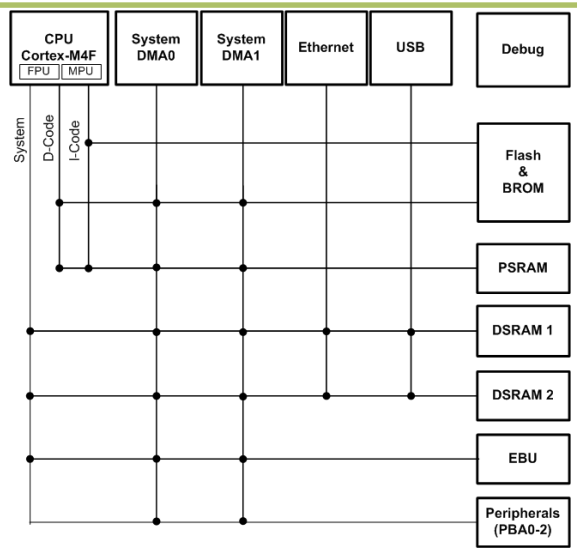
- When Flash is busy, any access to the Flash memory or write access to Flash SFRs will be stalled automatically

Parameter	Min.	Typical	Max.	Unit	Note
Voltage range	3.1	-	3.6	V	
Erase time per sector	-	0.3	0.4	s	Sector = 16 Kbytes
	-	1.2	1.4	s	Sector = 64 Kbytes
	-	5	5.5	s	Sector = 256 Kbytes
Erase all	-	-	22	s	At 1 Mbyte
Write time per page	-	5.5	11	ms	Page = 256 bytes
Data retention	20	-	-	years	Ta = 125°C, 1k cycles
Flash Wait States	2	-	-	cycles	f_{MCLK} = 80 MHz
	3	-	-	cycles	f_{MCLK} = 120 MHz
Erase cycles per sector	-	-	10k	cycles	Sector = 64 Kbytes
Total erase cycles	-	-	20k	cycles	Cycling @ 2 x 64 Kbytes

Agenda

1. XMC4000 and ARM® Cortex ®-M4
2. Memories
3. Bus system
4. Interrupt system
5. Clock
6. Reset
7. Power
8. Ports

Bus system



Highlights

Access parallelism through multilayer bus, multi-masters and multi-slaves.
Zero wait state data accesses between masters and slaves.
Simultaneous access can be handled by arbitration priority scheme.

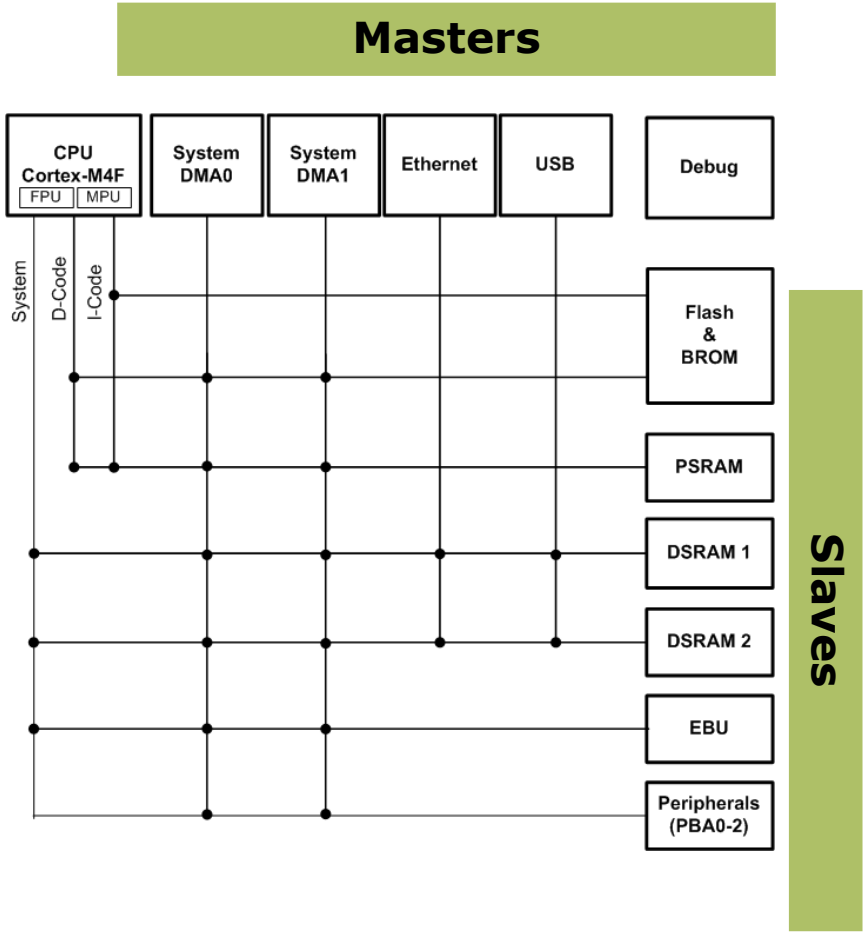
Key feature

- Access parallelism
- Zero wait state access to memories
- Arbitration priority scheme

Customer benefits

- High throughput and real-time access of the system
- Fast memory access
- Optimize system performance

Bus System Prioritization Scheme



Access Priorities per Slave

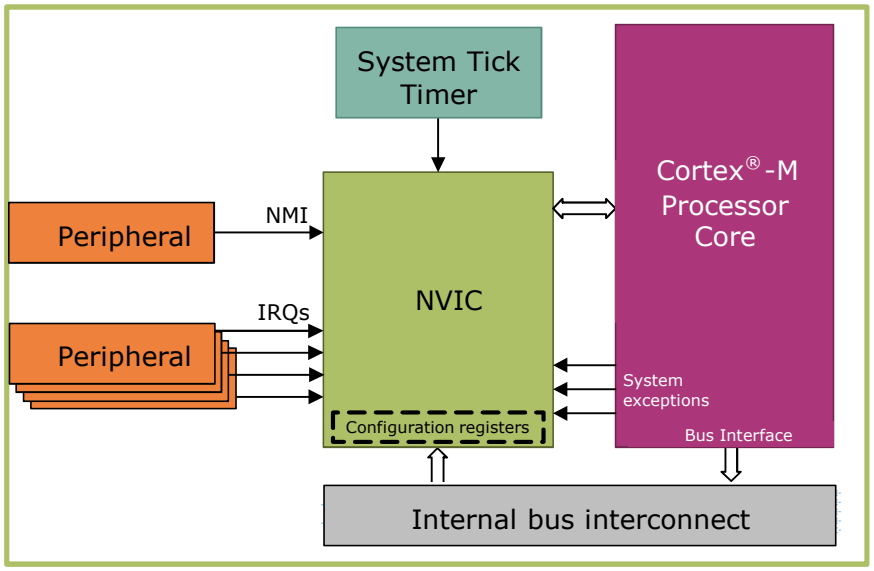
Examples of simultaneous accesses:
1. Ethernet-DMA to DSRAM2 while
2. CPU to PSRAM while
3. DMA0 to DSRAM1

	CPU	GPD MA0	GPD MA1	ETH	USB
FLASH	1	2	3	-	-
PSRAM	1	2	3	-	-
DSRAM 1	1	2	3	4	5
DSRAM 2	1	4	5	2	3
EBU	1	2	3	-	-
PBA0-2	1	2	3	-	-

Agenda

1. XMC4000 and ARM® Cortex ®-M4
2. Memories
3. Bus system
4. Interrupt system
5. Clock
6. Reset
7. Power
8. Ports

Interrupt system



Highlights

The interrupt system consists of the Nested Vectored Interrupt Controller (NVIC) and the interrupt generation blocks in the individual modules.

Events from peripherals can be routed via the connection matrix directly to other peripherals and can trigger interrupt requests.

Key feature

- 64 programmable priority level
- Interrupt Tail-chaining
- Automatic State Saving and Restoring

Customer benefits

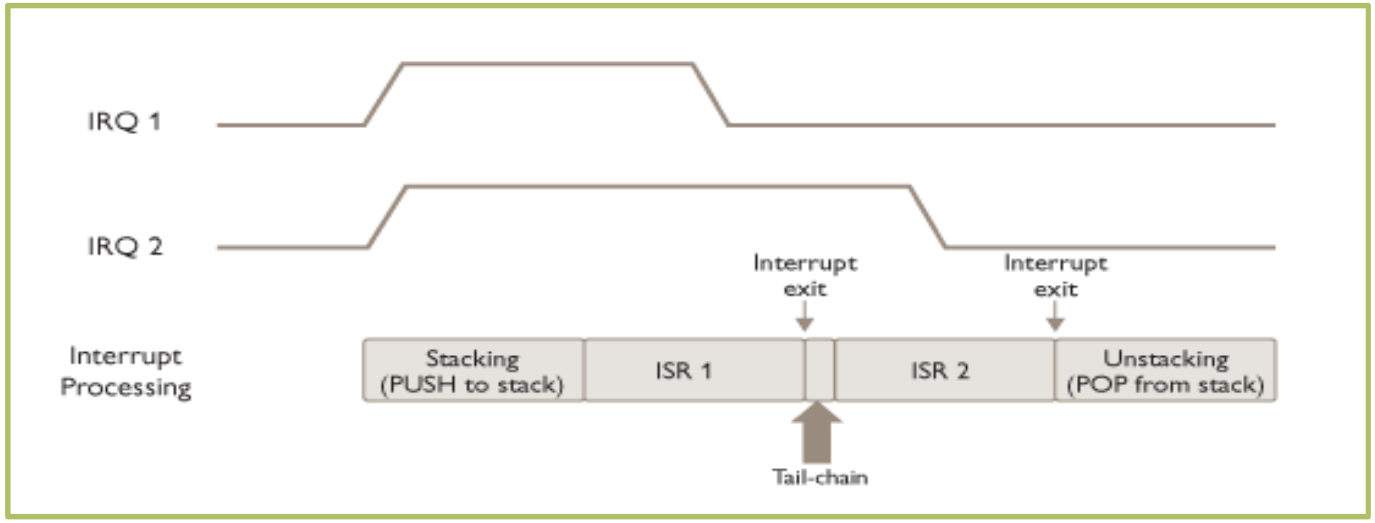
- Flexible priority control
- Speed up interrupt servicing
- Low latency exception handling

Interrupt system
64 Programmable priority level

- NVIC supports 112 interrupt nodes
- Each interrupt node has an 8-bit priority field in IPRn register
- One of 64 priority levels is assigned to an interrupt node by writing to its corresponding priority field
- The lower the value written to the priority field, the higher the priority
- A higher priority interrupt can interrupt an existing lower priority interrupt, resulting in nested interrupts
- When multiple interrupts have the same priority level, the pending interrupt with the lower node ID takes precedence

Interrupt system
Interrupt tail-chaining

- When a current ISR is completed and there is a pending interrupt, stack pop is skipped and control is transferred to the new ISR
- Allows interrupt servicing to speed up



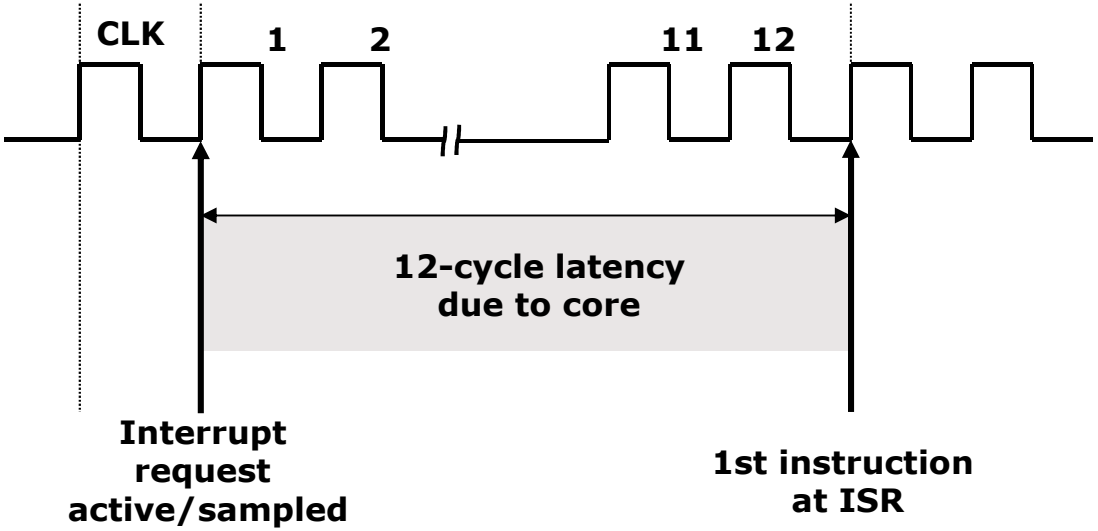
Tail-chaining
Reference: http://www.arm.com

Interrupt system
Automatic state saving and restoring

- Automatic state saving and restoring are done without any instruction overhead:
  - State registers are pushed onto the stack before entering the ISR
  - Reading of vector table entry is done after the state saving
  - State registers are popped after exiting the ISR

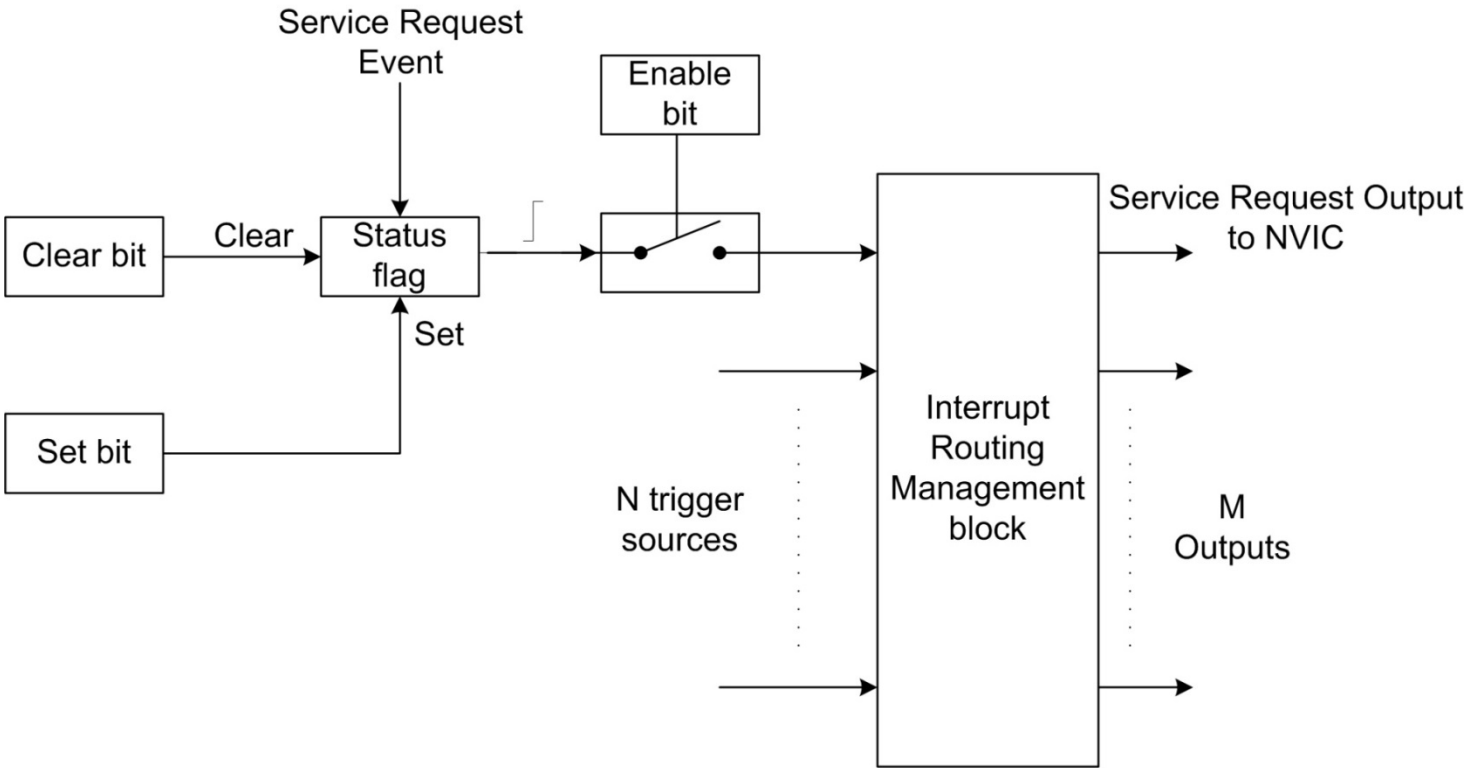
Interrupt system
Interrupt latency

- Defined as the time from detection of interrupt pulse and latching of interrupt by NVIC, to execution of first instruction in ISR
- Typically **12** Cycles



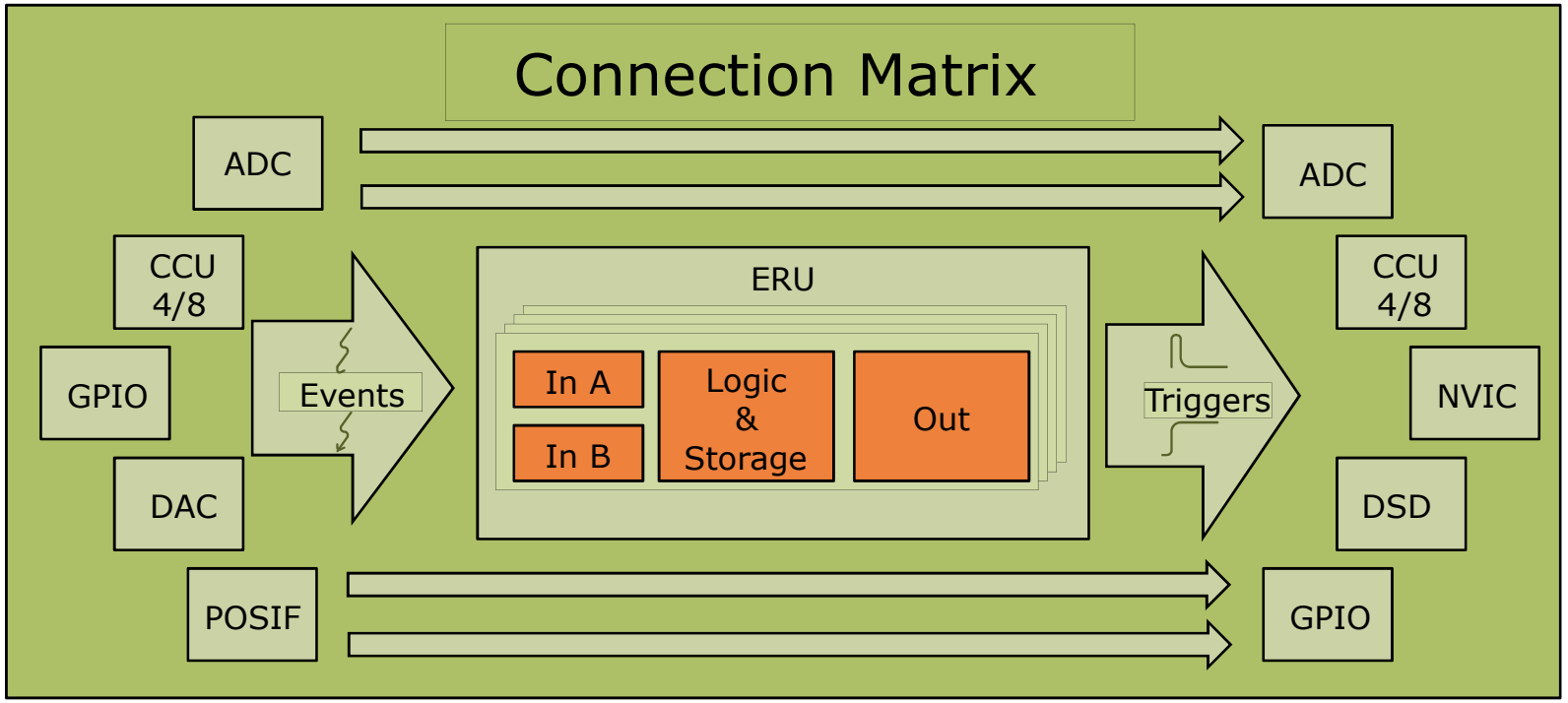
Note: Flash wait states are not considered

Interrupt system
General module interrupt structure



- Generates interrupt levels to NVIC
- Status flag is a sticky bit – has to be cleared by SW

Interrupt system connection matrix

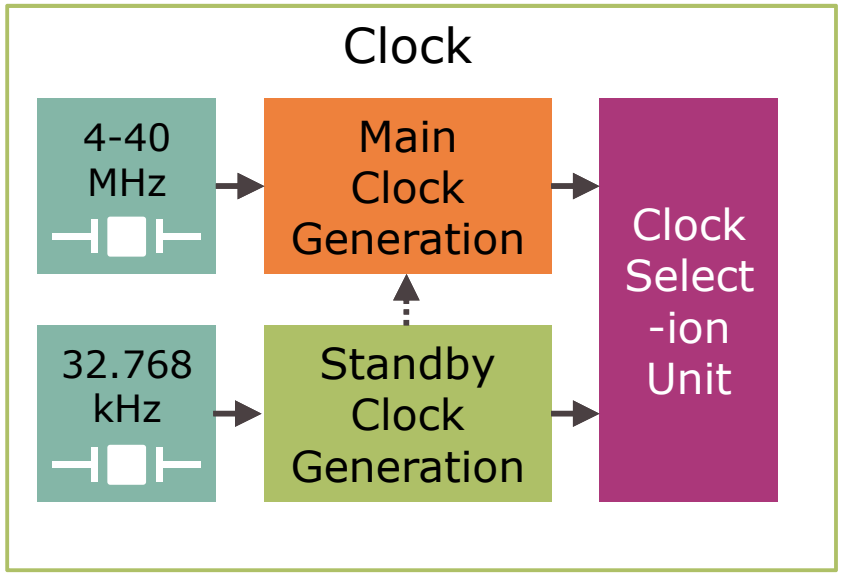


- Peripheral outputs can trigger events which are routed to peripheral inputs or to the NVIC
- Additional event routing is possible via the Event Request Unit ERU which can store events and logically combine two events
- This allows very flexible and powerful system design

Agenda

1. XMC4000 and ARM® Cortex ®-M4
2. Memories
3. Bus system
4. Interrupt system
5. Clock
6. Reset
7. Power
8. Ports

Clock



Highlights

The main function of the clock system is to generate clock for the CPU and on-chip peripherals. The clock can be generated internally, by an external crystal or via direct input. The clock selection is flexible, and clocking system prevents against invalid clock configuration. A clock supervisory ensures always a safe clock for operation. Peripheral clock can run at twice the speed of CPU clock.

Key feature

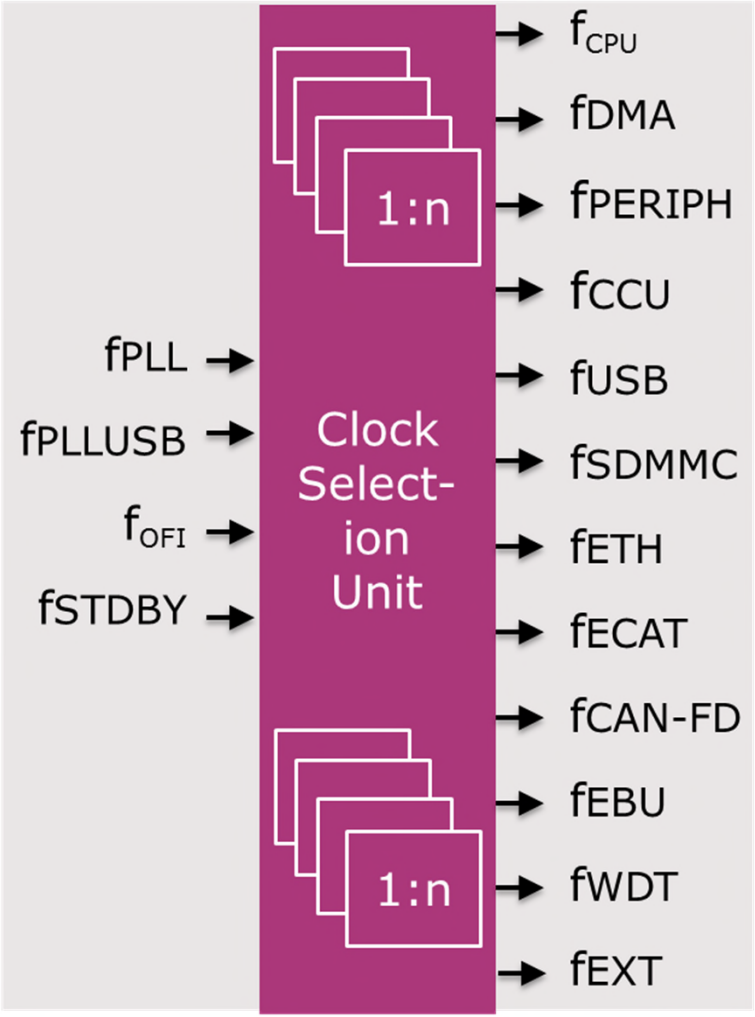
- Separate clock sources
- Internal backup clock
- Optional automatic clock calibration

Customer benefits

- Wide choice of clock frequency for peripherals and system
- Emergency clock if external clock fails (fail safe feature)
- Reduce bill of material

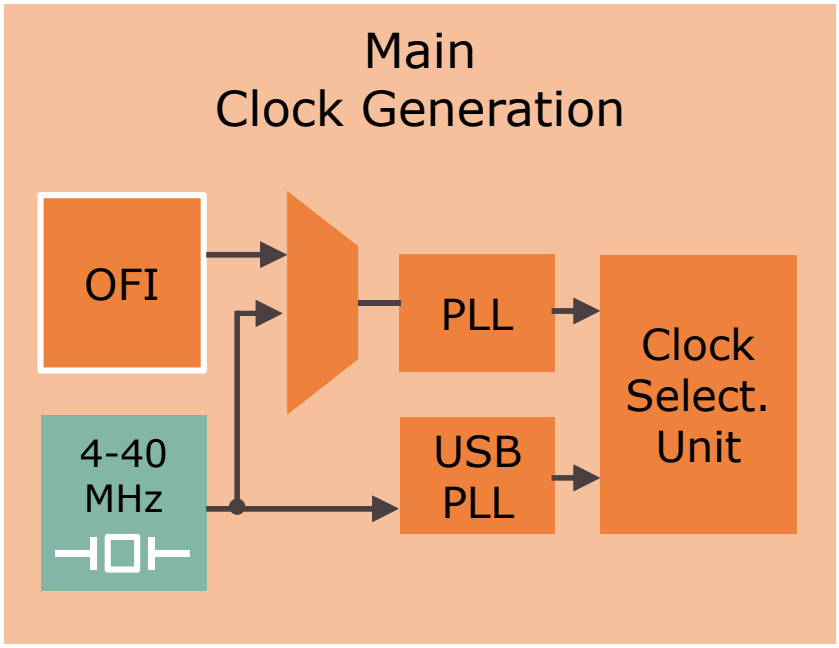
Clock
Separate clock sources

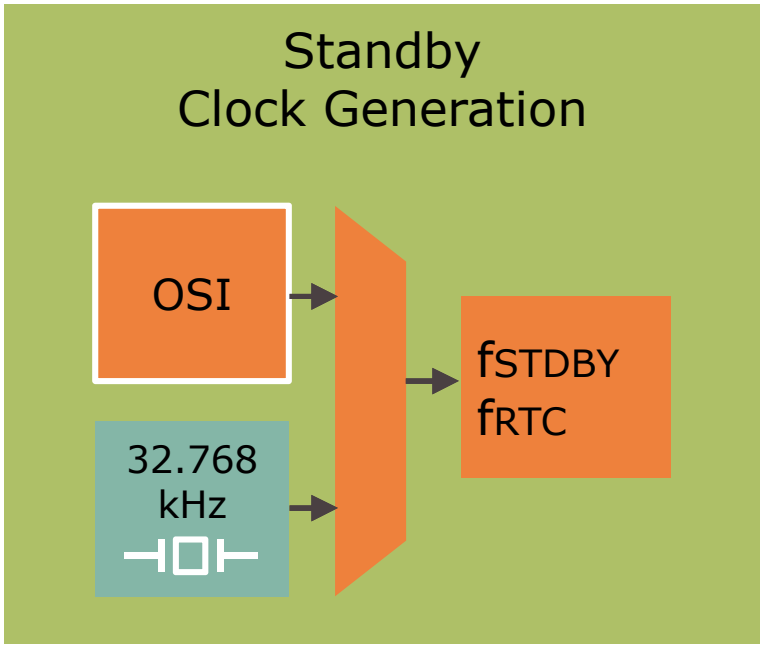
- Two separate PLLs available for the system clock, ETH and USB
  - System clock (32 kHz to 144 MHz)
  - ETH (>100 MHz)
  - EtherCAT (100 MHz)
  - USB/SDMMC (48 MHz)
  - CAN-FD (20;40;80 MHz)
- Due to individual prescaler a very flexible configuration for the CPU system and peripherals is possible
- Clock source (e.g. f_{OFI}) for WDT can be independent from CPU (f_{CPU}) clock



Clock
Internal backup clock

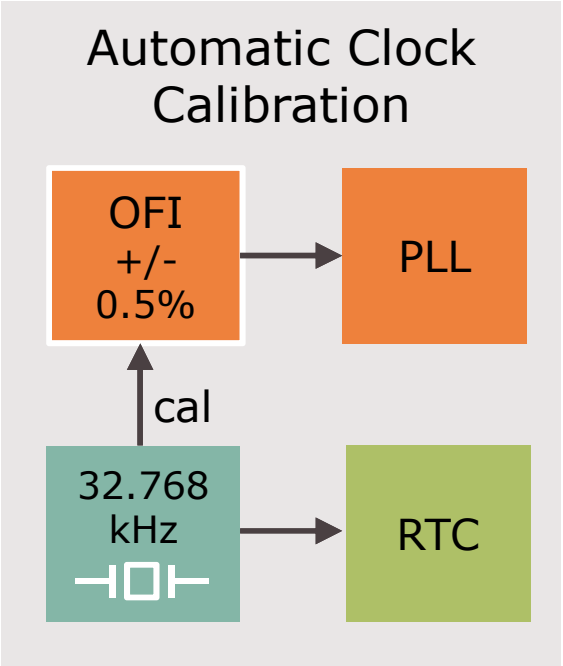
- A clock malfunction e.g. no clock, clock too slow, clock too fast and spikes in the clock is detected via a hardware oscillator watchdog
- A backup clock (OFI or OSI) is available in case of a clock malfunction; the backup clock is automatically selected in emergency case





Clock
Optional automatic clock calibration

- The two backup clocks can be used as clock source for cost sensitive applications
- OFI and OSI can be calibrated
  - Factory calibration via one-time setting from flash configuration sector
  - Automatic calibration – **on-the fly**
    - In case of using an accurate 32.678 kHz crystal for the RTC, one can use this clock to calibrate the OFI
    - OFI can be used as PLL input clock

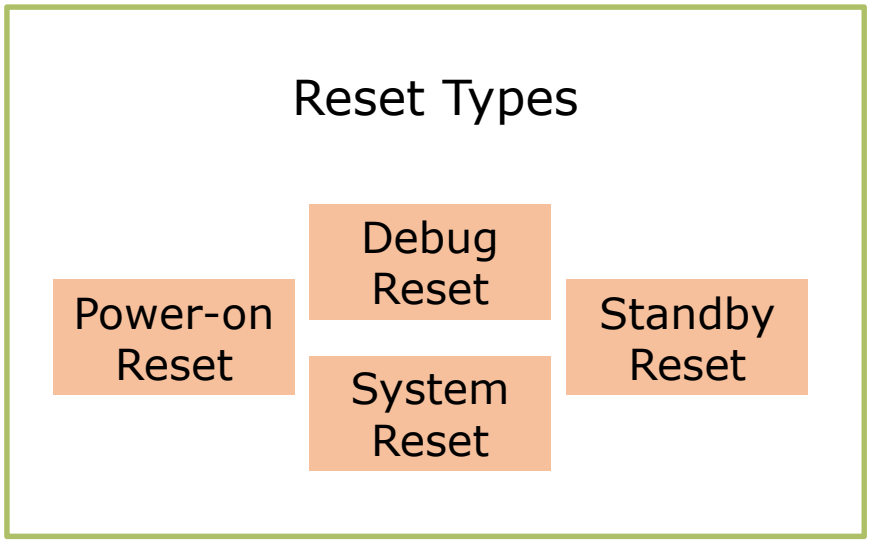


	OFI	OSI
Uncalibrated	36.5 MHz +/- 25%	32 kHz +/- 10%
Factory calibrated	24 MHz +/-15%	32 kHz +/- 4-10%
Automatic calibrated	24 MHz +/-0.5%	-

Agenda

1. XMC4000 and ARM® Cortex ®-M4
2. Memories
3. Bus system
4. Interrupt system
5. Clock
6. **Reset**
7. Power
8. Ports

Reset



Highlights

Always ensure safe application state via on-chip reset generation.

Full user control via reset event signalizing and software control.

Two domains core and hibernate with independent power supply and reset generation.

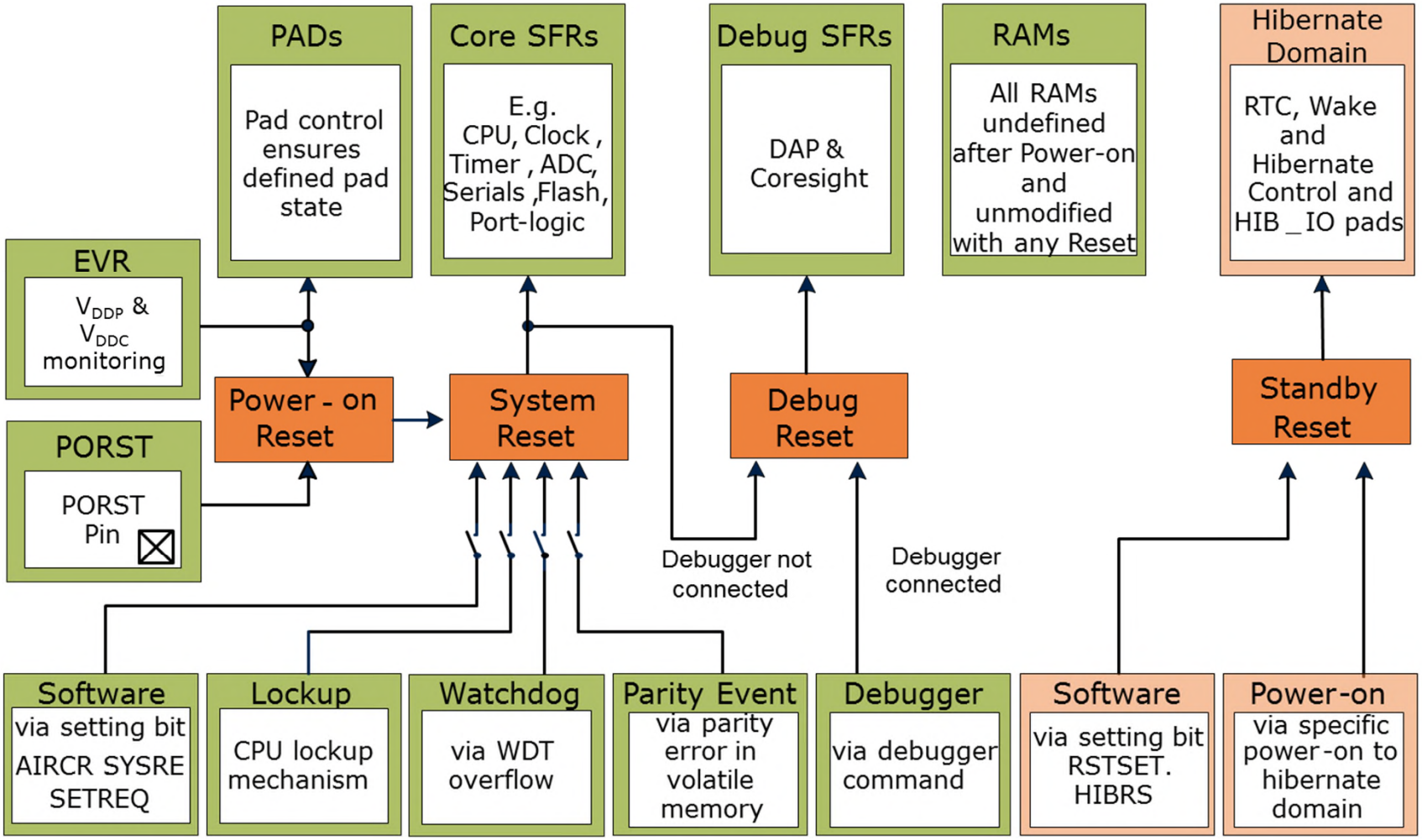
Key feature

- Power-on reset
- Fail safe reset mechanisms
- Hibernate domain with separate reset

Customer benefits

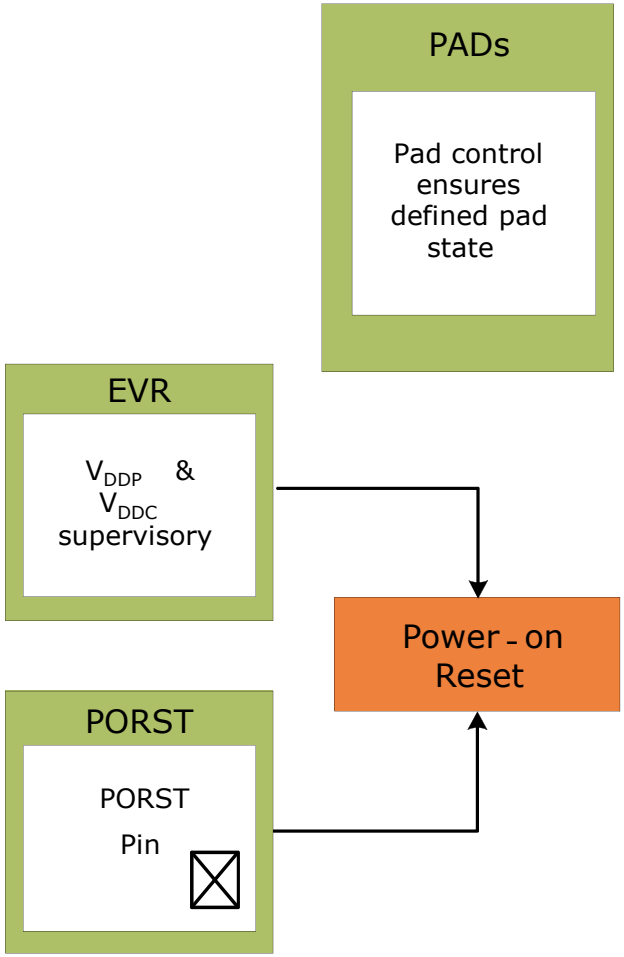
- Ensure safe operating conditions and defined system state
- Reset will bring the system into safe operation
- Hibernate domain can save information safely when using a backup supply

Reset
Reset types and reset sources



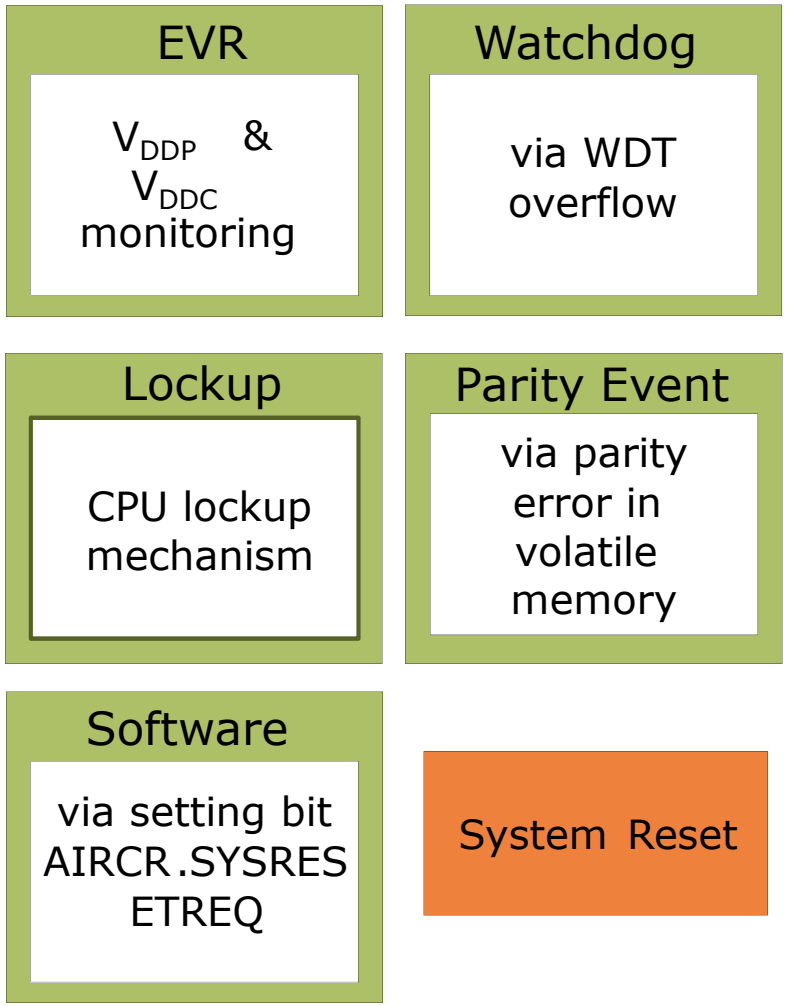
Reset
Power-on reset

- Power-on reset
  - Supply voltage (3.3 V) and core voltage (internally generated 1.3 V) monitoring inside EVR
  - Hysteresis on threshold of power-on reset for supply voltage
    - Immunity to noise (>250 mV) in the supply voltage
  - Pad control ensures defined pad state in undefined voltage range (V_{DDP} < V_{DDP}min)



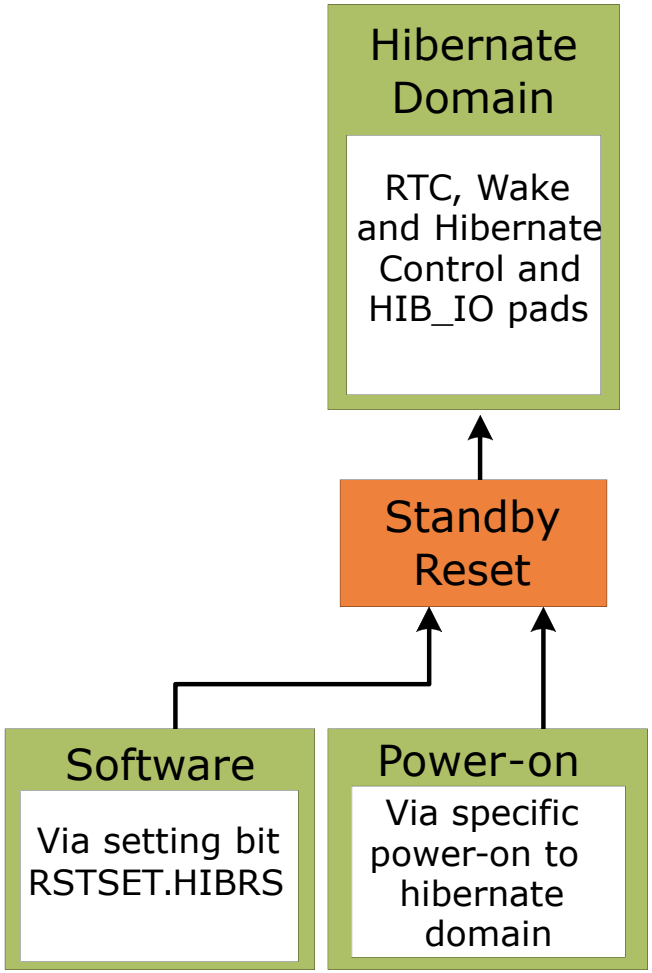
Reset
Fail safe reset mechanisms

- Brown-out detection pre-warning
  - Prepare system for reset, or, perform software corrective actions avoiding reset
- Parity error detection
  - Optional system reset in case of parity error
- Lockup reset
  - Standard ARM CPU feature
  - Optional reset in case of unrecoverable CPU states
- Watchdog Timer reset
  - Watchdog reset generated if CPU is not responsive
  - Watchdog reset can be avoided with optional pre-warning
- Software can trigger a System Reset



Reset
Hibernate domain with separate reset

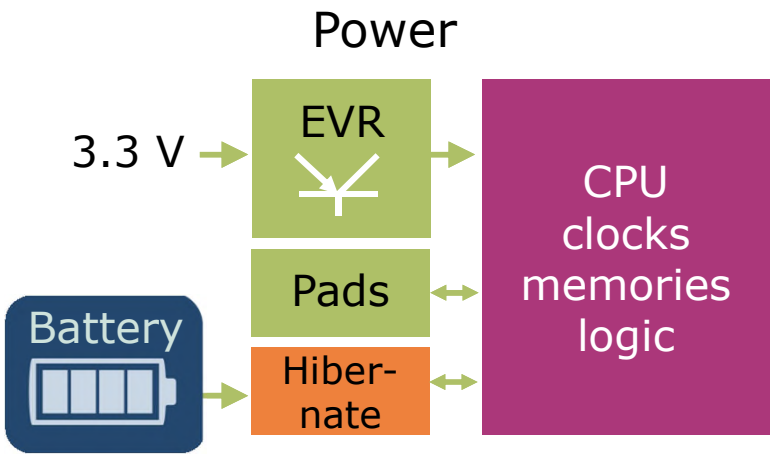
- Hibernate state will be preserved in case of system reset
  - System context memory retention
  - Hibernate configuration preserved
  - No impact on real-time clock
- Hibernate domain can be reset with a software reset
  - Explicit reset triggered in user's software
- Note: the Hibernate Domain has a separate Power-on mechanism (via V_{bat} supply) independent to the core system



Agenda

1. XMC4000 and ARM® Cortex ®-M4
2. Memories
3. Bus system
4. Interrupt system
5. Clock
6. Reset
7. Power
8. Ports

Power



Highlights

The single 3.3 V supply concept allows a wide range of use cases. There are several power saving modes to tailor the current consumption to the application requirements. The power system is optimized for robustness and fail safe behavior for usage in a rugged industrial environment.
A battery backed hibernate domain with real-time clock is good for current critical application.

Key feature

- Different power saving modes
- Hibernate domain with RTC
- Power monitoring and validation

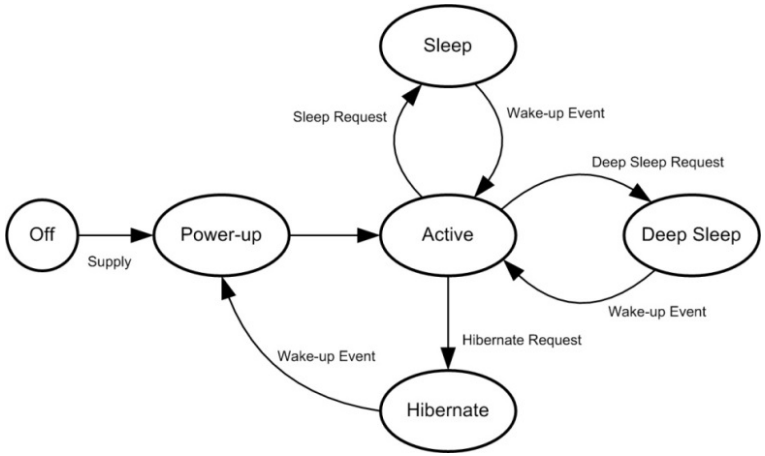
Customer benefits

- Scalable power consumption according to application use case
- Time and context keeping at low current consumption
- Fail safe functionality

Power
Different power saving modes (1/2)

- Simple transition scheme
  - All power saving states entered from Active state
    - Active – Sleep
    - Active – Deep Sleep
    - Active – Hibernate
  - Power saving states entered with a single request
    - Sleep and Deep Sleep entered via CPU instruction
    - Hibernate request with single register access
  - Flexible Wake-up trigger
    - One wake-up trigger source can serve different power saving modes

› Prevention of wrong clock configuration in hardware



Mode	Power Saving	Recovery
Active	- Clock scaling - Disable selected peripherals - Flash on/off	- Fully SW control
Sleep	- Stop CPU - Disable ETH, USB, CCU, WDT	- Wake via interrupt - Automatic restore
Deep Sleep	- Stop CPU - Disable ETH, USB, CCU, WDT - Disable PLLs, flash	- Wake via interrupt - Automatic restore
Hibernate	- Core power supply off	- Wake via event

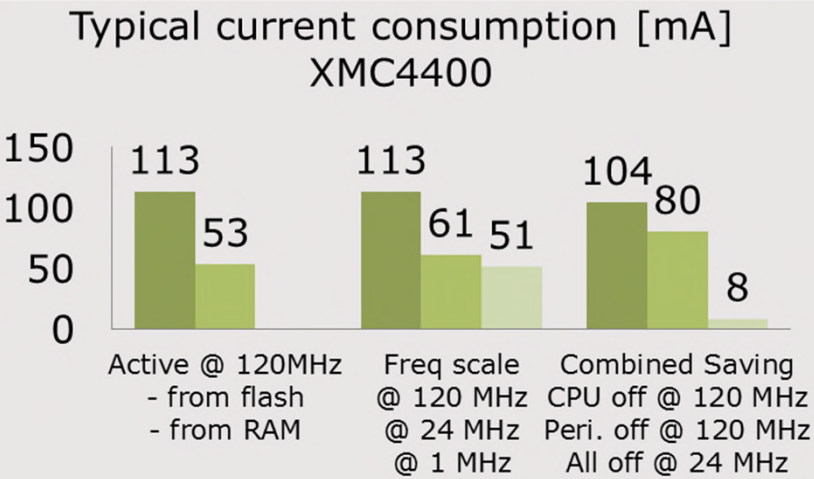
Power
Different power saving modes (2/2)

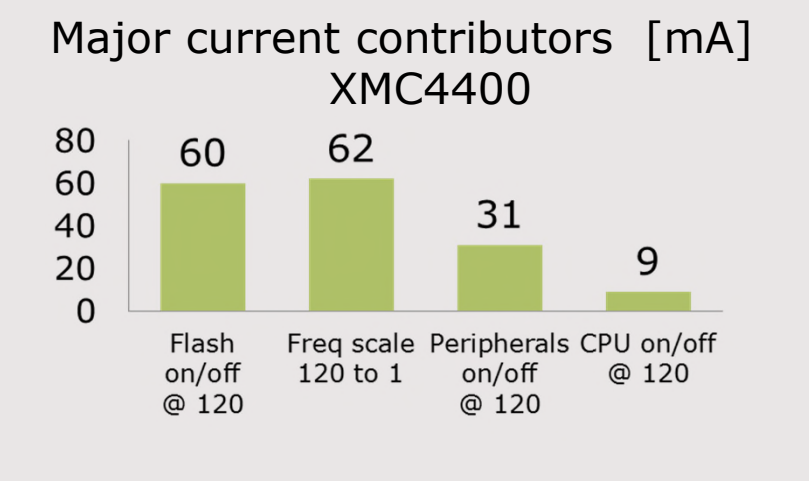
- Robust supply concept can be adapted to a wide range of power consumption to the user`s need
- Supports enough headroom for worst case current consumption

XMC Family	Typical [mA]	Worst case [mA]
XMC4700	~146	~190
XMC4500	~122	~180
XMC4400	~113	~170
XMC4200	~80	~140

- Load step due to frequency change*)
  - Max negative load step is ~150mA
  - Max positve load step is ~50mA

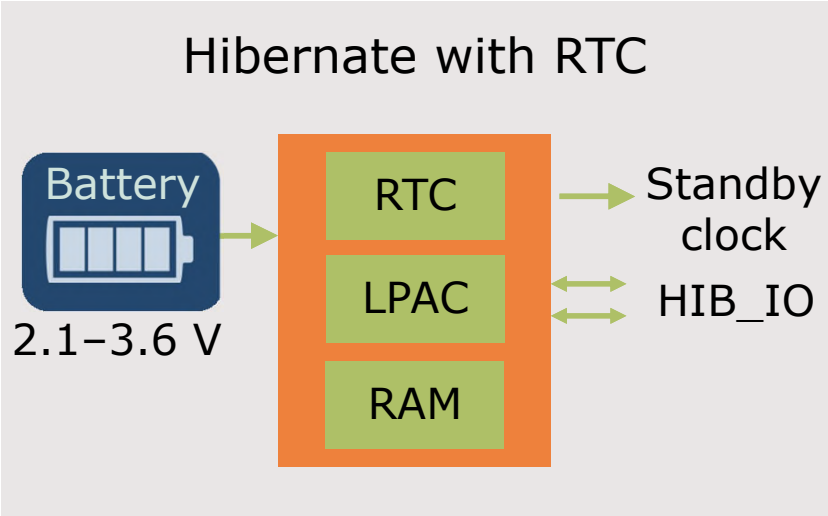
*) Increasing the frequency has to be done in steps, e. g. 24 MHz - 68 MHz - 120 MHz – 144 MHz.

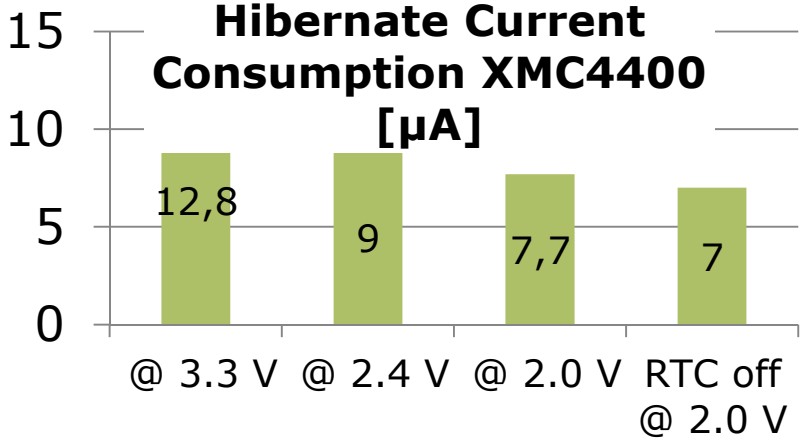




Power
Hibernate with RTC

- Hibernate state will be preserved in case of system power off
  - System context memory retention (16 x 32-bit register)
  - Hibernate configuration preserved
  - No impact on real-time clock
- Very low current consumption in Hibernate Domain
  - Battery supply

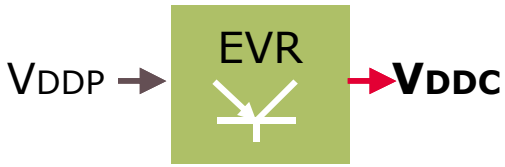


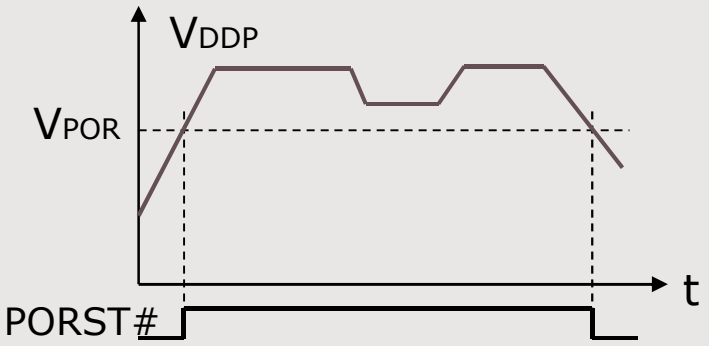


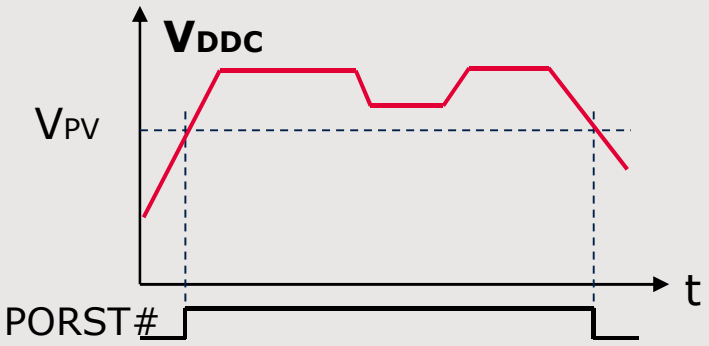
Power
Power monitoring and validation

- Pad domain power monitoring
  - Supply voltage V_{DDP}
  - Threshold level V_{POR}
    - Upper level of hysteresis is 3.1 V
    - Lower level of hysteresis is 2.85 V
  - PORST activated when $V_{DDP} < V_{PV}$
- Core domain power validation
  - Core voltage V_{DDC}
  - Threshold level V_{PV} (1.3 V)
  - PORST activated when $V_{DDC} < V_{PV}$

Safe power up and down with automatic internal reset handling (release & issue) and external signalization via bi-directional PORST signal.



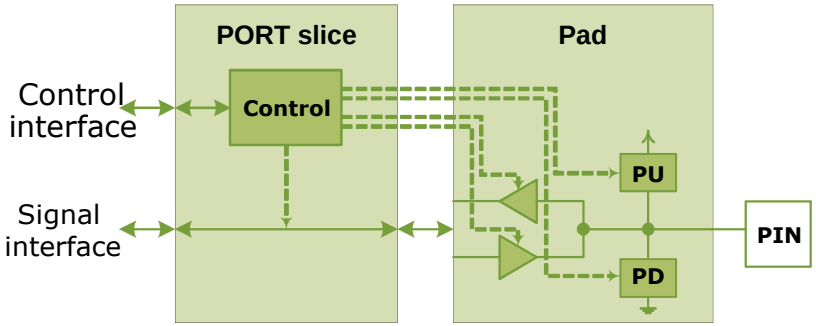




Agenda

1. XMC4000 and ARM® Cortex ®-M4
2. Memories
3. Bus system
4. Interrupt system
5. Clock
6. Reset
7. Power
8. Ports

Ports



Highlights

The Ports module provides a generic and flexible interface to the microcontroller.

Key feature

- Flexible function mapping
- Programmable driver strength and slew rate
- Individual port pin manipulation via special control register

Customer benefits

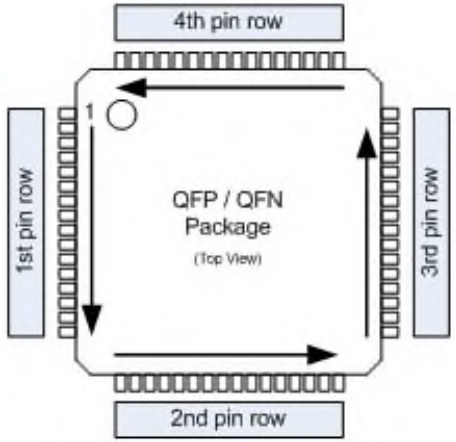
- Up to 11 input / 7 output functions per pin allowing flexible board layout
- Application specific adaption for optimal EMI performance
- Bitwise access without read-modify-write instruction

Ports
Flexible function mapping 1-2

Example Motor Control + PFC in 48 pins

1x SPI with CS
1x UART/CAN

2x Ext. Interrupt
1x WDT Service
1x RTC
1x USB
3x Debug



3x HB-PWM
2x Comparator
2x Iphasecurrent
1x Udc
1x Motor-Trap
1x QuadEncoder

1x PFC-PWM
1x PFC-Trap
2x Ipfc/Uac
1x Comparator
1x DAC

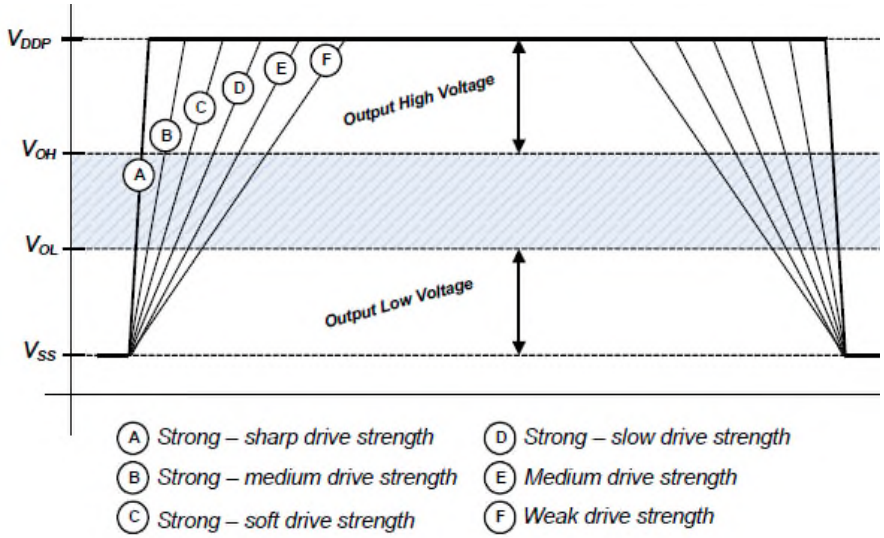
1st Pin Row	2nd Pin Row	3rd Pin Row	4th Pin Row
1 P0.1 CCU80.OUT11	13 P14.5 VADC.G0CH5	25 P2.1 DB.TDO/TRACESV	37 P1.3 HRPWM0.C0INB
2 P0.0 CCU80.OUT21	14 P14.4 VADC.G0CH4	26 P2.0 CCU40.IN1C	38 P1.2 HRPWM0.C2INA
3 USB_DM USB.USB_DM	15 P14.3 VADC.G1CH3	27 VSS	39 P1.1 HRPWM0.C1INA
4 USB_DP USB.USB_DP	16 P14.0 VADC.G0CH0	28 VDDP	40 P1.0 CCU40.OUT3
5 VDDP	17 VSSA/VAGND	29 XTAL1 HP Oscillator.XT	41 VDDP
6 VDDC	18 VDDA/VAREF	30 XTAL2 HP Oscillator.XT	42 P0.8 ERU0.2A1
7 HIB_IO_0 WWDT.SERV	19 P14.9 DAC.OUT_1	31 VDDC	43 P0.7 CCU80.IN0A
8 RTC_XTAL1 ULP Oscillat	20 P14.8 VADC.G1CH0	32 PORST#	44 P0.6 ERU0.3B2
9 RTC_XTAL2 ULP Oscillat	21 P2.5 U0C1.DOUT0	33 TMS DB.TMS	45 P0.5 CCU80.OUT00
10 VBAT	22 P2.4 U0C1.SCLKOUT	34 TCK DB.TCK	46 P0.4 CCU80.OUT10
11 P14.7 POSIF0.IN0B	23 P2.3 U0C1.SELO0	35 P1.5 U0C0.DOUT0	47 P0.3 CCU80.OUT20
12 P14.6 POSIF0.IN1B	24 P2.2 U0C1.DX0A	36 P1.4 U0C0.DX0B	48 P0.2 CCU80.OUT01
Exp. Pad VSS			

Ports
Flexible function mapping 2-2

- Up to 11 input / 7 output functions per pin
  - Allowing high pin utilization for efficient board design at small footprint
- Adjustable pad properties
  - Input / output
  - Push-pull / open drain
  - Pull-up / pull-down / tristate
  - Power save
- Direct control by peripherals
  - Analog reference generation by pull-up/pull-down control using a switching signal
  - Quad SPI functionality without software overhead
- Separate input and output path
  - Allows to evaluate the input while the output is active (feedback, plausability check)

Ports
Programmable driver strength and slew rate

- Pad Driver Mode
  - 3 pad classes with adjustable driver strength and slew rate for 3.3 V LVTTL outputs
    - low speed for e. g. GPIO
    - medium speed for e. g. SPI
    - high speed for e. g. SDRAM
  - Allowing application specific adaption for optimal EMI performance

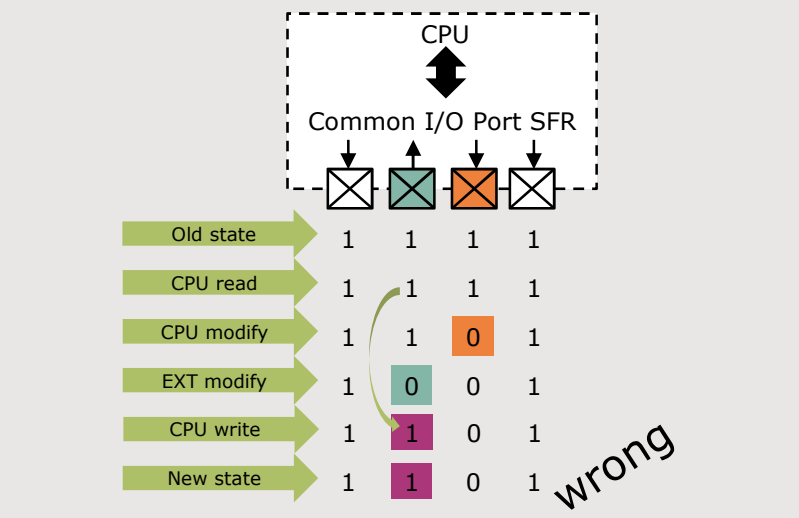


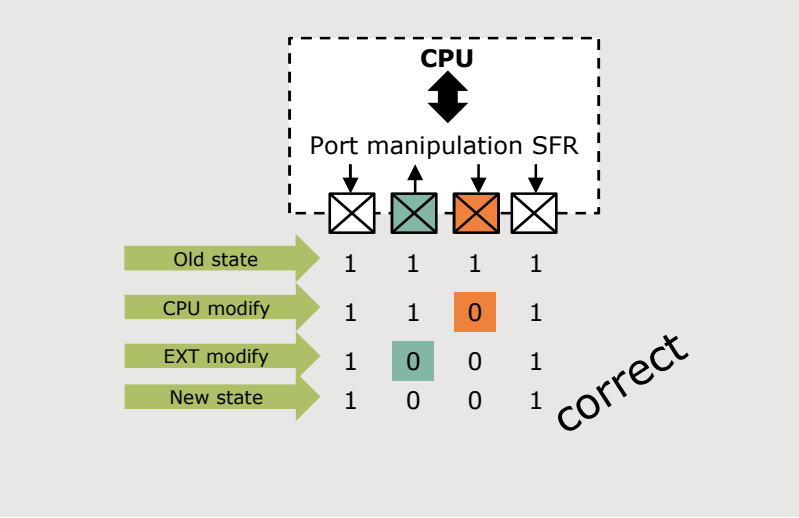
	Class A1	Class A1+	Class A2
Driver Strength	weak	weak	weak
	medium	medium	medium
	-	strong	strong
Slew Rate		slow	soft
		medium	medium
		-	sharp

Ports
Individual port pin manipulation

- The port output modification register (Px OMR) allows individually manipulation of a port pin:
  - Set
  - Reset
  - Toggle
- This allows a concurrent hardware input event and change of a output value on two port pins
- Avoids read-modify-write instruction sequences to change the output value of a single port pin simultaneous to an input hardware event

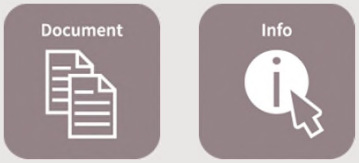
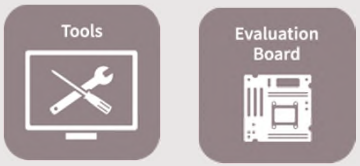
Ports
Individual port pin manipulation





- Traditional port structure
  - Read-modify-write may lead to a wrong input state if input changes simultaneously with the output modification
  - Data may be lost
- XMCs port structure
  - The individual port manipulation registers allows simultaneous HW and SW events without any data loss

Support material

Collaterals and brochures 	– Product briefs – Selection guides – Application brochures – Presentations – Press releases, ads	– www.infineon.com/XMC
Technical material 	– Application notes – Technical articles – Simulation models – Datasheets, MCDS files – PCB design data	– www.infineon.com/XMC – Kits and Boards – DAVE™ – Software and Tool Ecosystem
Videos 	– Technical videos – Product information Videos	– Infineon Media Center – XMC Mediathek
Contact 	– Forums – Product support	– Infineon Forums – Technical Assistance Center (TAC)

Disclaimer

The information given in this training materials is given as a hint for the implementation of the Infineon Technologies component only and shall not be regarded as any description or warranty of a certain functionality, condition or quality of the Infineon Technologies component.

Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind (including without limitation warranties of non-infringement of intellectual property rights of any third party) with respect to any and all information given in this training material.

All the images used in the trainings are free for commercial use or free for use with attribution and were designed by Freepik.

Part of your life. Part of tomorrow.

Infineon