User Manual

for S32K3 MCU Driver

Document Number: UM34MCUASRR21-11 Rev0000R3.0.0 Rev. 1.0

Chapter 1

Revision History

Revision	Date	Author	Description
1.0	31.03.2023	NXP RTD Team	S32K3 Real-Time Drivers AUTOSAR 4.4 & R21-11 Version 3.0.0

Chapter 2

Introduction

This User Manual describes NXP Semiconductors' AUTOSAR Mcu Driver for S32K3XX.

AUTOSAR Mcu Driver configuration parameters description can be found in the Tresos Configuration Plugin section. Deviations from the specification are described in the Deviations from Requirements section.

AUTOSAR Mcu driver requirements and APIs are described in the Mcu Driver Software Specification Document (version R21-11).

2.1 Supported Derivatives

The software described in this document is intended to be used with the following microcontroller devices of NXP Semiconductors:

- s32k310_mqfp100
- s32k310_lqfp48
- s32k311_mqfp100 / MWCT2015S_mqfp100
- s32k311_lqfp48
- s32k312_mqfp100 / MWCT2016S_mqfp100
- s32k312_mqfp172 / MWCT2016S_mqfp172
- s32k314_mqfp172
- s32k314_mapbga257

- s32k322_mqfp100 / MWCT2D16S_mqfp100
- s32k322_mqfp172 / MWCT2D16S_mqfp172
- s32k324_mqfp172 / MWCT2D17S_mqfp172
- s32k324_mapbga257
- s32k341_mqfp100
- s32k341_mqfp172
- s32k342_mqfp100
- s32k342_mqfp172
- s32k344_mqfp172
- s32k344_mapbga257
- s32k394_mapbga289
- s32k396_mapbga289
- s32k358_mqfp172
- s32k358_mapbga289
- s32k328_mqfp172
- s32k328_mapbga289
- s32k338_mqfp172
- s32k338_mapbga289
- s32k348_mqfp172
- s32k348_mapbga289
- s32m274_lqfp64
- s32m276_lqfp64

All of the above microcontroller devices are collectively named as S32K3.

Note: MWCT part numbers contain NXP confidential IP for Qi Wireless Power.

2.2 Overview

AUTOSAR (AUTomotive Open System ARchitecture) is an industry partnership working to establish standards for software interfaces and software modules for automobile electronic control systems.

AUTOSAR:

- paves the way for innovative electronic systems that further improve performance, safety and environmental friendliness.
- is a strong global partnership that creates one common standard: "Cooperate on standards, compete on implementation".
- is a key enabling technology to manage the growing electrics/electronics complexity. It aims to be prepared for the upcoming technologies and to improve cost-efficiency without making any compromise with respect to quality.
- facilitates the exchange and update of software and hardware over the service life of the vehicle.

2.3 About This Manual

This Technical Reference employs the following typographical conventions:

- **Boldface** style: Used for important terms, notes and warnings.
- *Italic* style: Used for code snippets in the text. Note that C language modifiers such "const" or "volatile" are sometimes omitted to improve readability of the presented code.

Notes and warnings are shown as below:

Note

This is a note.

Warning

This is a warning

2.4 Acronyms and Definitions

Term	Definition
API	Application Programming Interface
ASM	Assembler
BSMI	Basic Software Make file Interface
CAN	Controller Area Network
C/CPP	C and C++ Source Code
CS	Chip Select
CTU	Cross Trigger Unit
DEM	Diagnostic Event Manager
DET	Development Error Tracer
DMA	Direct Memory Access
ECU	Electronic Control Unit
FIFO	First In First Out
LSB	Least Signifigant Bit
MCU	Micro Controller Unit
MIDE	Multi Integrated Development Environment
MSB	Most Significant Bit
N/A	Not Applicable
RAM	Random Access Memory
SIU	Systems Integration Unit
SWS	Software Specification
VLE	Variable Length Encoding
XML	Extensible Markup Language

2.5 Reference List

#	Title	Version
1	Specification of MCU Driver	AUTOSAR R21-11
2	S32K3XX Reference Manual	Rev.6, Draft B, 01/2023
3	S32K3xx Data Sheet	Rev.6, 01/2023
4	S32K396 Reference Manual	Rev. 2 Draft A, 11/2022
5	S32K396 Data Sheet	Rev. 1.1 — 08/2022
6	S32M27x Reference Manual	Rev.2, Draft A, — 02/2023
7	S32M2xx Data Sheet	Rev. 2 RC — 12/2022
8	S32K358_0P14E Mask Set Errata	Rev. 28, 9/2022
9	S32K396_0P40E Mask Set Errata	Rev. DEC2022, 12/2022
10	S32K311_0P98C Mask Set Errata	Rev. 6/March/2023, 3/2023
11	S32K312: Mask Set Errata for Mask 0P09C	Rev. 25/April/2022
12	S32K342: Mask Set Errata for Mask 0P97C	Rev. 10, 11/2022
13	S32K3x4: Mask Set Errata for Mask 0P55A/1P55A	Rev. 14/Oct/2022

Chapter 3

Driver

3.1 Requirements

Requirements for this driver are detailed in the AUTOSAR R21-11 Mcu Driver Software Specification document (See Table Reference List)

3.2 Driver Design Summary

The Mcu Driver controls the CLOCK, POWER and RAM modules of the S32K3XX device. It provides the following features:

- Configuration and initialization of the CLOCK.
- Configuration and initialization of the POWER.
- Configuration and initialization of the RAM.

3.3 Hardware Resources

The Mcu Driver consists of:

1. Clock IPs (FIRC,SIRC,FXOSC,SXOSC,PLLDIG,MC_CGM,MC_ME,MC_RGM,CMU,C40ASF)
2. Power IPs (MC_ME,PMC,MC_RGM,MC_PCU,PMC_AE,C40ASF)
3. Ram IPs (PRAMC,STCU2)

3.4 Deviations from Requirements

The driver deviates from the AUTOSAR MCU Driver software specification in some places. The table below identifies the AUTOSAR requirements that are not implemented or out of scope for the MCU Driver.

Term	Definition
N/S	Out of scope
N/I	Not implemented
N/F	Not fully implemented

Below table identifies the AUTOSAR requirements that are not fully implemented, implemented differently or out of scope for the MCU driver.

Requirement	Status	Description	Notes
SWS_Mcu_00053	N/S	If clock failure notification is enabled in the configuration set and a clock source failure error occurs, the error code MCU_↩ E_CLOCK_FAILURE shall be reported. (See also SWS_↩ Mcu_00051).	DEMs cannot be reported in ISR contexts. For the clock failure case the error MCU_E↩ _ISR_CLOCK_FAILURE was defined and if the error notification is configured by the application the CMU interrupt will report it.
SWS_Mcu_00257	N/S	Fail criteria for MCU_E_↩ CLOCK_FAILURE: a clock source failure occurs	For the clock failure case the error MCU_E_ISR_CLOCK_↩ FAILURE was defined and if the error notification is configured by the application the CMU interrupt will report it.
SWS_Mcu_00258	N/S	Pass criteria for MCU_E_↩ CLOCK_FAILURE: no clock source failure occurs	For the clock failure case the error MCU_E_ISR_CLOCK_↩ FAILURE was defined and if the error notification is configured by the application the CMU interrupt will report it.
SWS_Mcu_00245	N/S	If the register can affect several hardware modules and if it is not an I/O register, it shall be initialised by this MCU driver.	There is a separate plug-in that will cover shared ip's

Requirement	Status	Description	Notes
SWS_Mcu_00056	N/S	The function Mcu_Distribute↩ PllClock shall return without affecting the MCU hardware if the PLL clock has been automatically activated by the MCU hardware.	The function Mcu_Distribute↩ PllClock will change the Mcu hardware. The clock switching to PLL is not completed by Mcu_InitClock
SWS_Mcu_00259	N/S	: DRAFT: The MCU Driver module shall reject configurations with partition mappings which are not supported by the implementation.	Based on ticket AAI-462, this requirement is not applicable.
SWS_Mcu_CONSTR_00001	N/S	: DRAFT: The module will operate as an independent instance in each of the partitions, means the called API will only target the partition it is called in.	Based on ticket AAI-462, this requirement is not applicable.
CPR_RTD_00544.mcu	N/F	Driver shall support Autosar standard configuration format for the IP layer Note: EPD file for the IP shall be provided.	The Clock IP can't support this requirement.

3.5 Driver Limitations

The RTD S32K3 Mcu driver software have the following limitations:

- The CMU_FC_5 is only accessible by the HSE_B Cortex-M0+ core, as it is used for monitoring HSE_CLK.
- MCU_SWT1_RST_RESET reason is not available on S32K310, S32K311, S32K312, S32K314, S32K341, S32K342, S32K344, S32M274 and S32M276 derivatives, but will appear in McuResetReasonConf because a single PreConfiguration file is used for all derivatives.
- MCU_SWT2_RST_RESET reason is not available on S32K310, S32K311, S32K312, S32K314, S32K341, S32K342, S32K344, S32M274, S32M276, S32K322 and S32K324 derivatives, but will appear in McuReset↩ ReasonConf because a single PreConfiguration file is used for all derivatives.
- MCU_SWT3_RST_RESET reason is not available on S32K3XX and S32M27X derivatives(except S32K388 derivative), but will appear in McuResetReasonConf because a single PreConfiguration file is used for all derivatives.
- MCU_PLL_AUX_RESET reason is not available on S32K3XX and S32M27X derivatives(except S32K328, S32K338, S32K348, S32K358 and S32K388 derivatives), but will appear in McuResetReasonConf because a single PreConfiguration file is used for all derivatives.
- MCU_CM7_CORE_CLK_FAIL_RESET reason is not available on S32K3XX and S32M27X derivatives(except S32K394, S32K396 and S32K388 derivative), but will appear in McuResetReasonConf because a single PreConfiguration file is used for all derivatives.
- ECPD for Clock Ip is not supported
- [ERR051494] Standby mode exit is not possible when "V15 domain enable during LPM" is enabled. Do not enable LPM trickle LDO (PMC.SMPSCONFIG[LPM15EN] with node "McuModuleConfiguration/McuPower↩ Control/McuSMPSLPM15EN" in configuration tool) before entering standby mode for S32K396.
- Low power mode feature hasn't tested for S32K358 and S32K396.

3.6 Driver usage and configuration tips

3.6.1 MCU Clock Management

- To generate a clock configuration in Clock Tool from Design Studio, a new project must be created. Project is created with an example clock configuration ClockConfig0. To create a new configuration from scratch (reset configuration), "add a new functional group" button must be pressed. To create a new configuration from an existing one, "copy functional group" button must be pressed. Clock configuration can be removed by removing functional group.

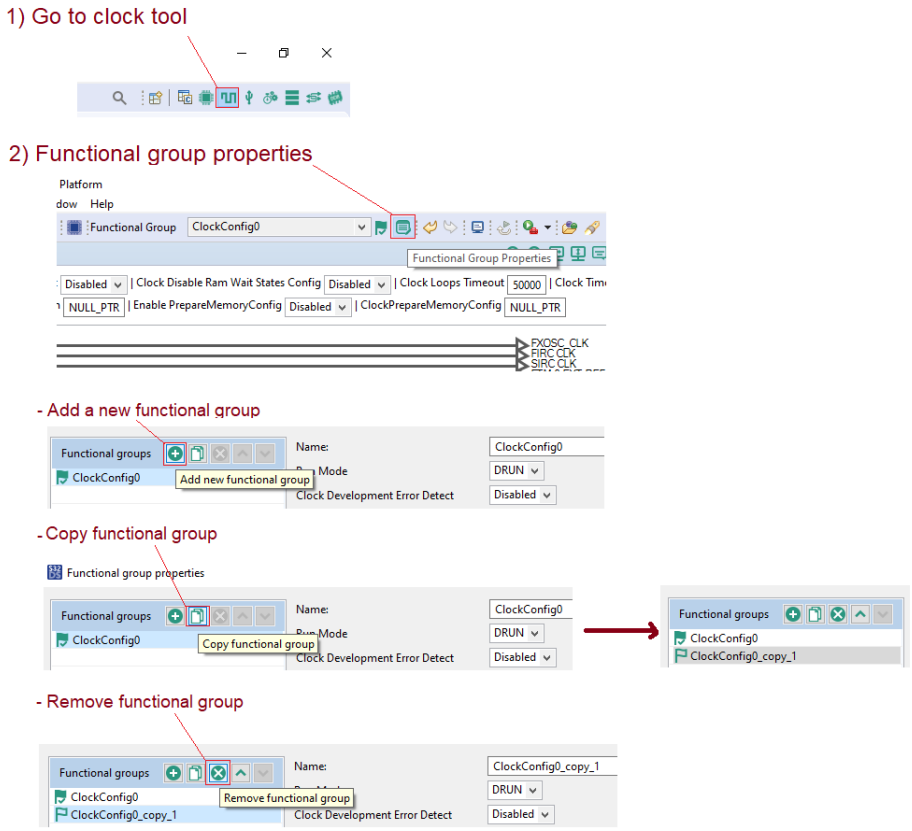


Figure 3.1 Clock tool snapshot for Functional Group Properties

- For bypassing the configuration of a clock source during Mcu_InitClock, the "[source] under MCU control" checkbox should be unchecked. This will generate smaller configurations that will be updated faster and more efficiently. In addition, if the application is configuring some clocks in advance, the UnderMcuControl should be unchecked so the MCU driver does not overwrite the initial settings.
- When the clock tree is initialized before the MCU driver is used. e.g The bootloader or user code initializes the clock tree. After that the control is passed to AUTOSAR software, the MCU is used to configured the ECU and clock again. The following sequence is required to successfully and safely re-configure the clock tree.
- System clock frequency selected must adhere to the same clock divider ratios shown in Clocking use case examples of Reference Manual.

- Because system clock frequency selected must adhere to the same clock divider ratios, so the trigger divider type selection should be 'COMMON_TRIGGER_DIVIDER_UPDATE'.

1. Mcu_Init
2. Mcu_InitClock (to reinitialize the clock tree)
3. If the PLL is used as a clock source, call Mcu_GetPllStatus until it returns MCU_PLL_LOCKED and call Mcu_DistributePllClock.
4. Mcu_SetMode (to gate the peripheral clocks)

3.6.2 MCU Mode Management

- For bypassing the configuration of a Partition, COFB set, or Core during Mcu_SetMode, the corresponding "[Block] Under MCU Control" checkbox should be unchecked. This will generate smaller configurations that will be updated faster and more efficiently. In addition, if the application is configuring a certain mode in advance, the UnderMcuControl should be unchecked so the MCU driver does not overwrite the initial settings.
- On S32K3, Bit field PMC_CONFIG[LMBCTLEN] (Last Mile Regulator Base Control Enable) must be set to 1 if external BJT between VDD_HV_A and V15 is used on the PCB. The base of this BJT must be connected to the VRC_CTRL pin and is controlled by the PMC to regulate a voltage of 1.5V on V15 pin to supply the Last Mile Regulator .
- The clock sources that aren't used can be disabled in different run-modes to reduce power consumption.

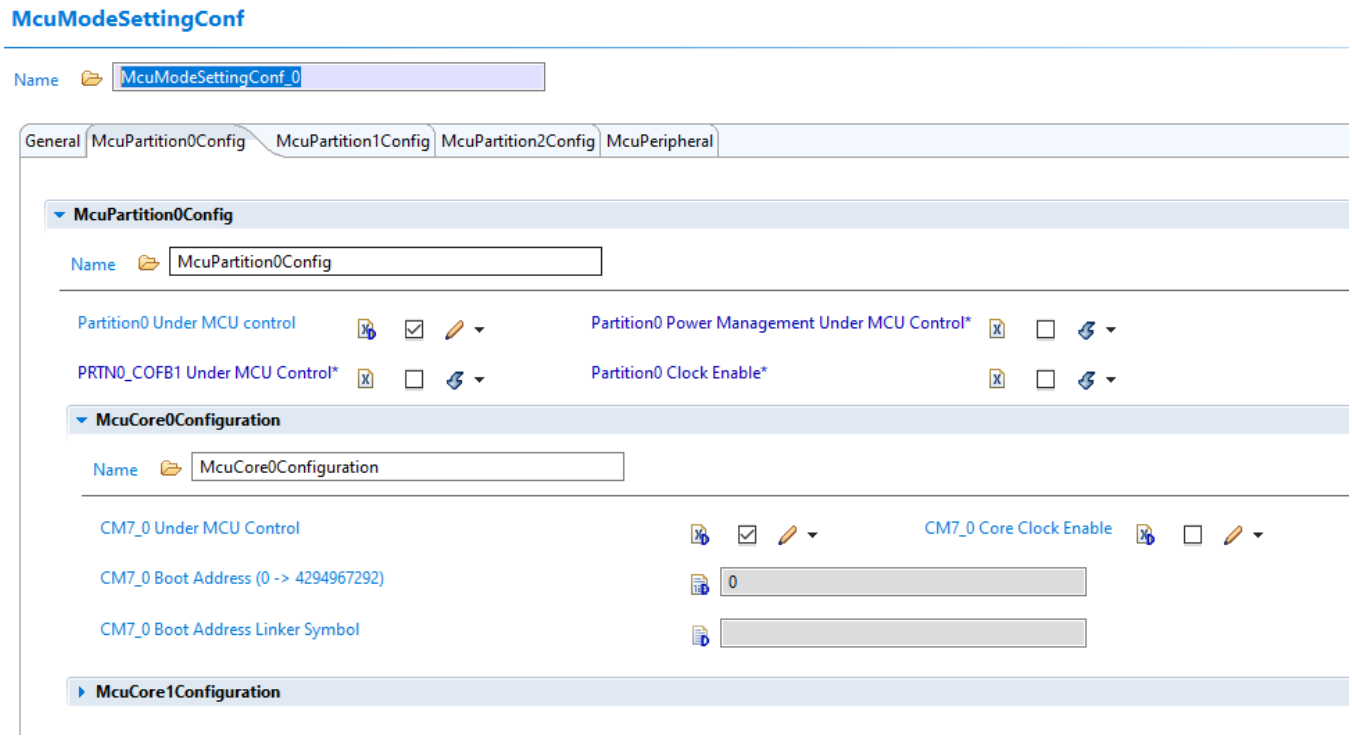
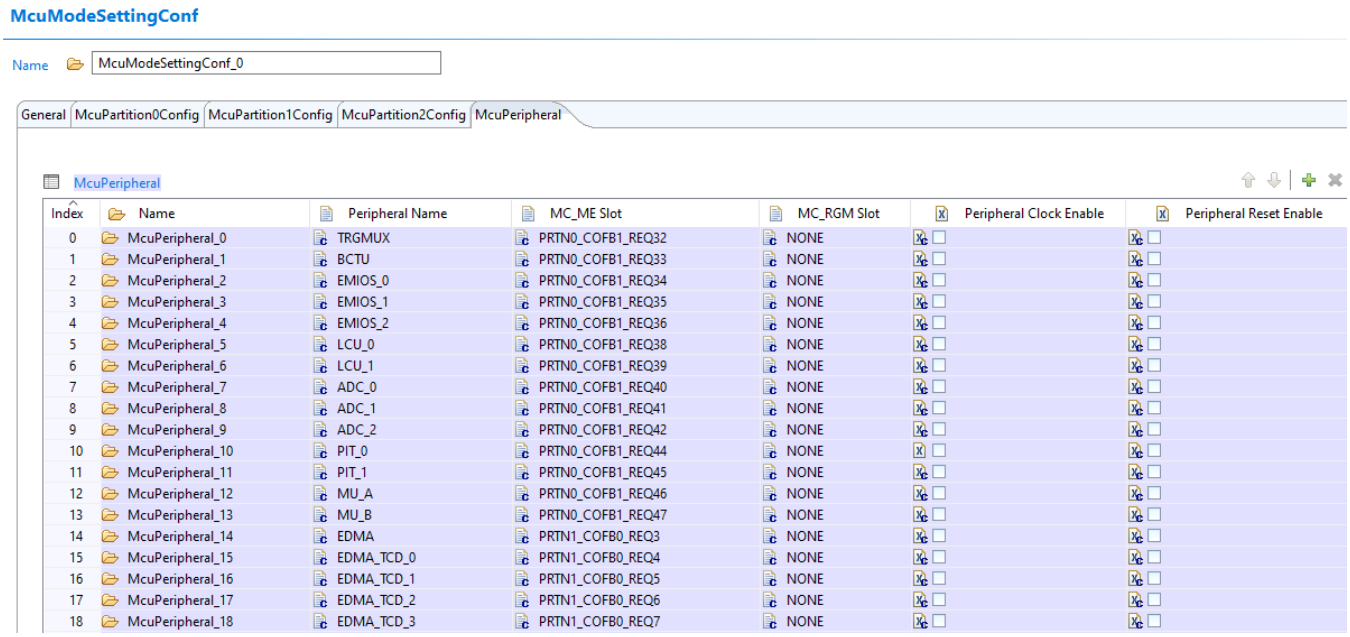


Figure 3.2 Tresos Plugin snapshot for McuPartition0Config form



Index	Name	Peripheral Name	MC_ME Slot	MC_RGM Slot	Peripheral Clock Enable	Peripheral Reset Enable
0	McuPeripheral_0	TRGMUX	PRTN0_COFB1_REQ32	NONE		
1	McuPeripheral_1	BCTU	PRTN0_COFB1_REQ33	NONE		
2	McuPeripheral_2	EMIOS_0	PRTN0_COFB1_REQ34	NONE		
3	McuPeripheral_3	EMIOS_1	PRTN0_COFB1_REQ35	NONE		
4	McuPeripheral_4	EMIOS_2	PRTN0_COFB1_REQ36	NONE		
5	McuPeripheral_5	LCU_0	PRTN0_COFB1_REQ38	NONE		
6	McuPeripheral_6	LCU_1	PRTN0_COFB1_REQ39	NONE		
7	McuPeripheral_7	ADC_0	PRTN0_COFB1_REQ40	NONE		
8	McuPeripheral_8	ADC_1	PRTN0_COFB1_REQ41	NONE		
9	McuPeripheral_9	ADC_2	PRTN0_COFB1_REQ42	NONE		
10	McuPeripheral_10	PIT_0	PRTN0_COFB1_REQ44	NONE		
11	McuPeripheral_11	PIT_1	PRTN0_COFB1_REQ45	NONE		
12	McuPeripheral_12	MU_A	PRTN0_COFB1_REQ46	NONE		
13	McuPeripheral_13	MU_B	PRTN0_COFB1_REQ47	NONE		
14	McuPeripheral_14	EDMA	PRTN1_COFB0_REQ3	NONE		
15	McuPeripheral_15	EDMA_TCD_0	PRTN1_COFB0_REQ4	NONE		
16	McuPeripheral_16	EDMA_TCD_1	PRTN1_COFB0_REQ5	NONE		
17	McuPeripheral_17	EDMA_TCD_2	PRTN1_COFB0_REQ6	NONE		
18	McuPeripheral_18	EDMA_TCD_3	PRTN1_COFB0_REQ7	NONE		

Figure 3.3 Tresos Plugin snapshot for McuPeripheral form

Note: Clock for main Core must be always enabled by McuCoreUnderMcuControl should be unchecked if main core configured by application or McuCoreUnderMcuControl and McuCoreClockEnable must be checked together.

3.6.3 MCU Flash Configuration

- When the "McuDisableFlashWaitStatesConfig" checkbox is unchecked in the MCU configuration, the "Clock↩ _Ip_CodeInRamSetFlashWaitStates" and "Clock_Ip_FLASH_SetFlashIWS" functions shall be placed into PRAM.
- This rises from the fact that updates to the flash programming model while the flash memory controller is in the midst of an operation (including code execution) will result in non-deterministic behavior.
- The recommended flow for multi-core applications is to start only one core and execute initialization code to completion before starting the remaining cores.
- To do so, please refer to the memory region delimited by the "MCU_START_SEC_RAMCODE" and "MCU↩ _STOP_SEC_RAMCODE" macros.

3.6.4 MCU RAM Configuration

- When the "McuDisableRamWaitStatesConfig" checkbox is unchecked in the MCU configuration, the "Mcu_↩ PRAMC_SetRamWS" function shall be placed outside the PRAM, e.g. to the core local memory space.
- This rises from the fact that updates to the PRAMC programming model while a PRAMC operation is in progress will result in non-deterministic behavior.

- Software must be architected to avoid this scenario by ensuring that PRAMC configuration changes are made only during system boot or when only one master is enabled. In multi-core applications, updates to the PRAMC configuration shall be made only when one core is active and no other masters (e.g. DMA or communications modules) are enabled.

- To do so, please refer to the memory region delimited by the "MCU_START_SEC_CODE_AC" and "MCU↩ _STOP_SEC_CODE_AC" macros.

3.6.5 How Mcu_SetMode works when entering Low Power mode

Before entering the software Standby mode sequence, the system clock source must be changed to FIRC because PLLDIG is not available in Standby mode. In this mode, all clock sources can be optionally disabled (including FIRC, which results in a no-clock, low-power consumption mode). You could use FXOSC, if enabled, when the 2.5 V supply is available by appropriate configuration of PMC's CONFIG[LPM25EN].
For Standby mode, the standby entry sequence consists of the following software configuration phases:

1. SW1: Peripheral Shutdown

2. SW2: Application Core Shutdown

3. SW3: Flash Low-Power Handshake and PMC_LASTMILE Regulator Disable

4. SW4: Main Core ShutDown

3.6.5.1 SW1: Peripheral Shutdown

I/O and module configuration for Standby mode follow to this diagram below:

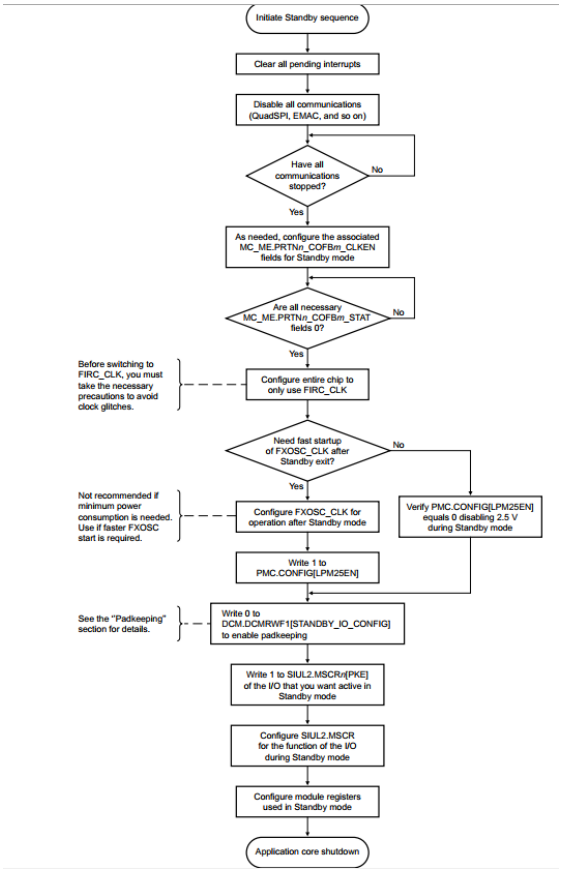


Figure 3.4 I/O and module configuration for Standby

3.6.5.2 SW2: Application Core Shutdown

The application will make sure that core(s) to be shutdown is/are prepared for standby entry.

1. From the main core, request the application core(s) to stop.

2. Transition the application core(s) into standby (i.e. the application core(s) must call Mcu_SetMode with a mode configuration having "McuPowerMode" = CORE_STANDBY)

3. From the main core, shutdown the application core(s) (i.e. the main core must call Mcu_SetMode with a mode configuration having the "McuCoreUnderMcuControl") checkbox(es) checked and "McuCoreClockEnable" checkbox(es) unchecked for the application core(s) to be shutdown).

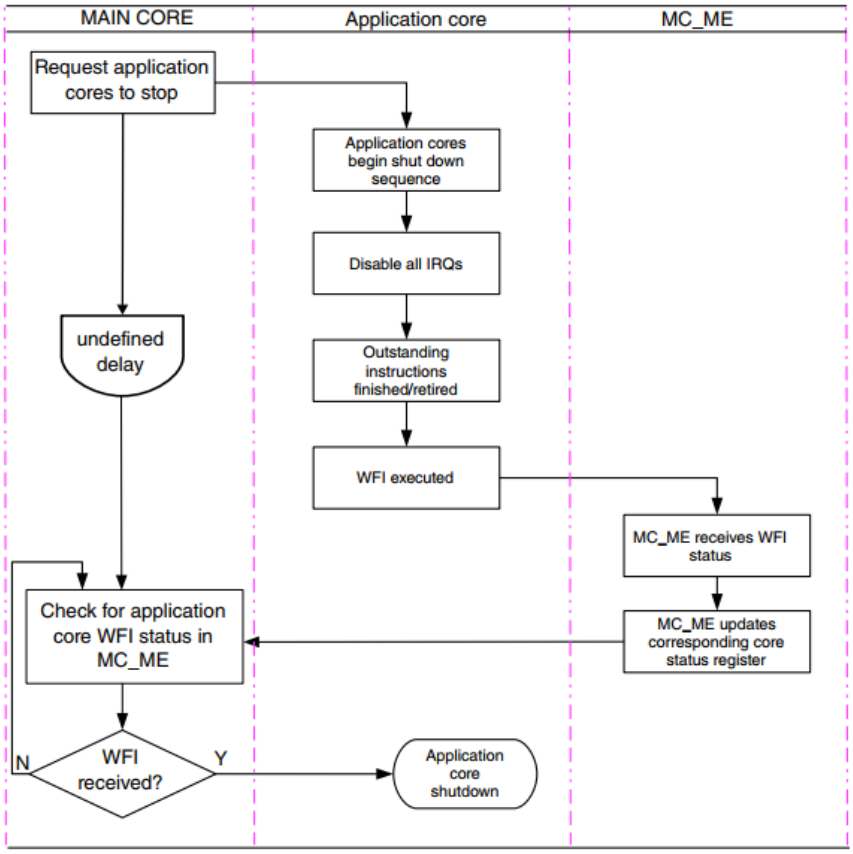


Figure 3.5 Application Core Shutdown.

3.6.5.3 SW3: Flash Low-Power Handshake and PMC_LASTMILE Regulator Disable

The application will disable the PLL (and optionally disable FIRC/SIRC/SXOSC/FXOSC as per power-consumption requirement in order to reduce power consumption in STANBY mode) and ensure that no falsh high voltage operations are ongoing.

1. Suspend or wait for any ongoing flash high voltage operations.

2. Configure FIRC as the system clock and disable the PLL (i.e. call Mcu_InitClock with such a clock configuration)

3. Prepare the SoC for standby mode(i.e. call Mcu_SetMode with a mode configuration having "McuPowerMode" = SOC_PREPARE_STANDBY).

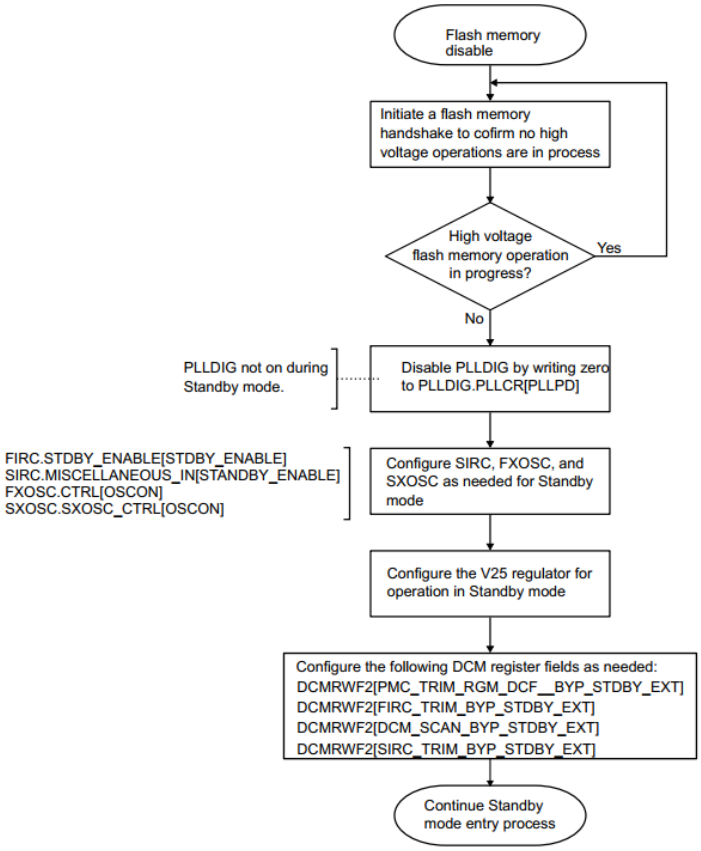


Figure 3.6 Flash Memory Standby Mode Configuration.

3.6.5.4 SW4: Main Core Shutdown

The application will program the necessary wakeup IP(s).

1. Program the necessary wakeup IP(s).

2. Transition the SoC into standby mode (i.e. call Mcu_SetMode with a mode configuration having "McuPower↩ Mode" = SOC_STANDBY and "McuMainCoreSelect" set to the core performing this transition).

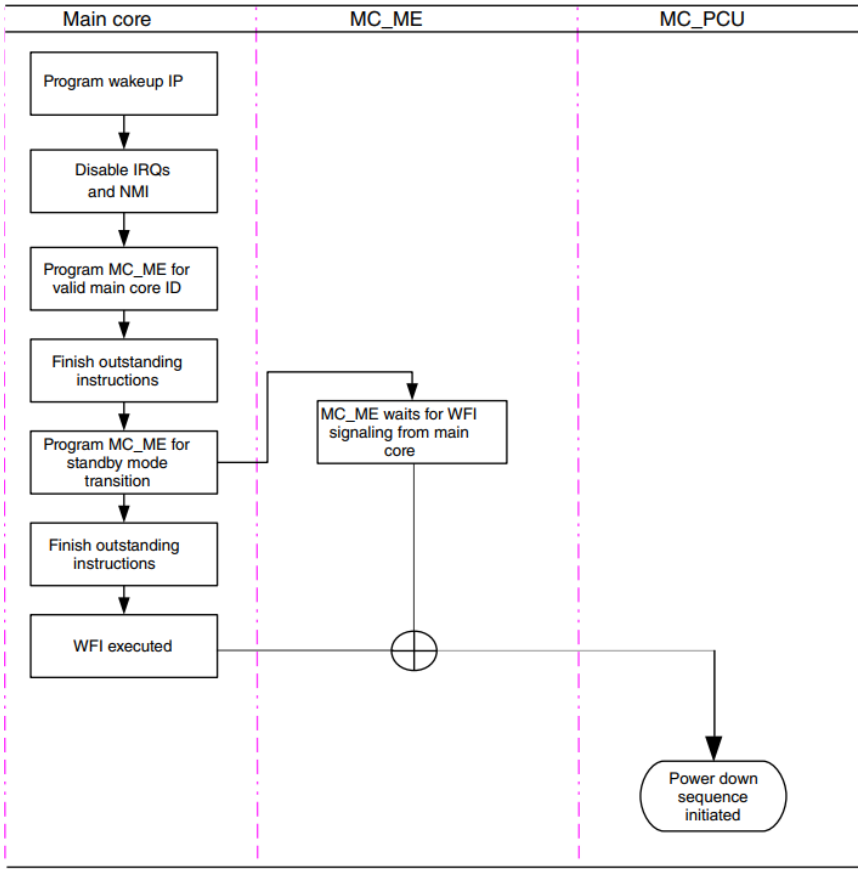


Figure 3.7 Standby entry sequence along with main core shutdown.

Note: The EcuM shall ensure that no wakeup interrupt event is lost. The MCU driver cannot handle wakeup interrupt as it doesn't have any information related to those. It is responsibility of the application to ensure that no wakeup interrupt occurs before the SoC completes the stanby mode transition.

Example about configuration and driver used:

1. Mcu_InitClock (the system clock source must be changed to FIRC).

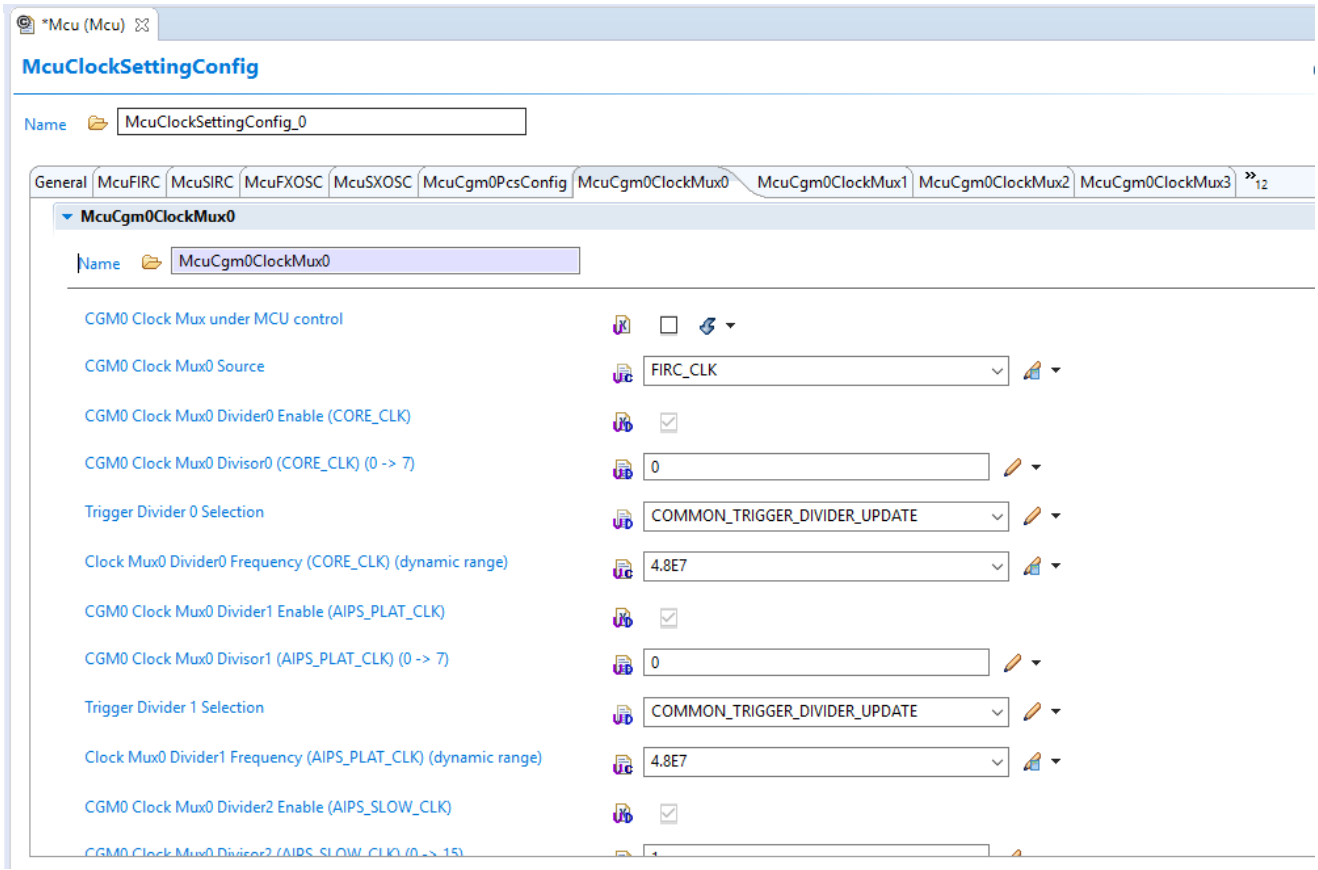


Figure 3.8 Tresos Plugin snapshot for system clock form

2. Mcu_SetMode (Disable clock of all peripherals and set Operating Mode = STANDBY).

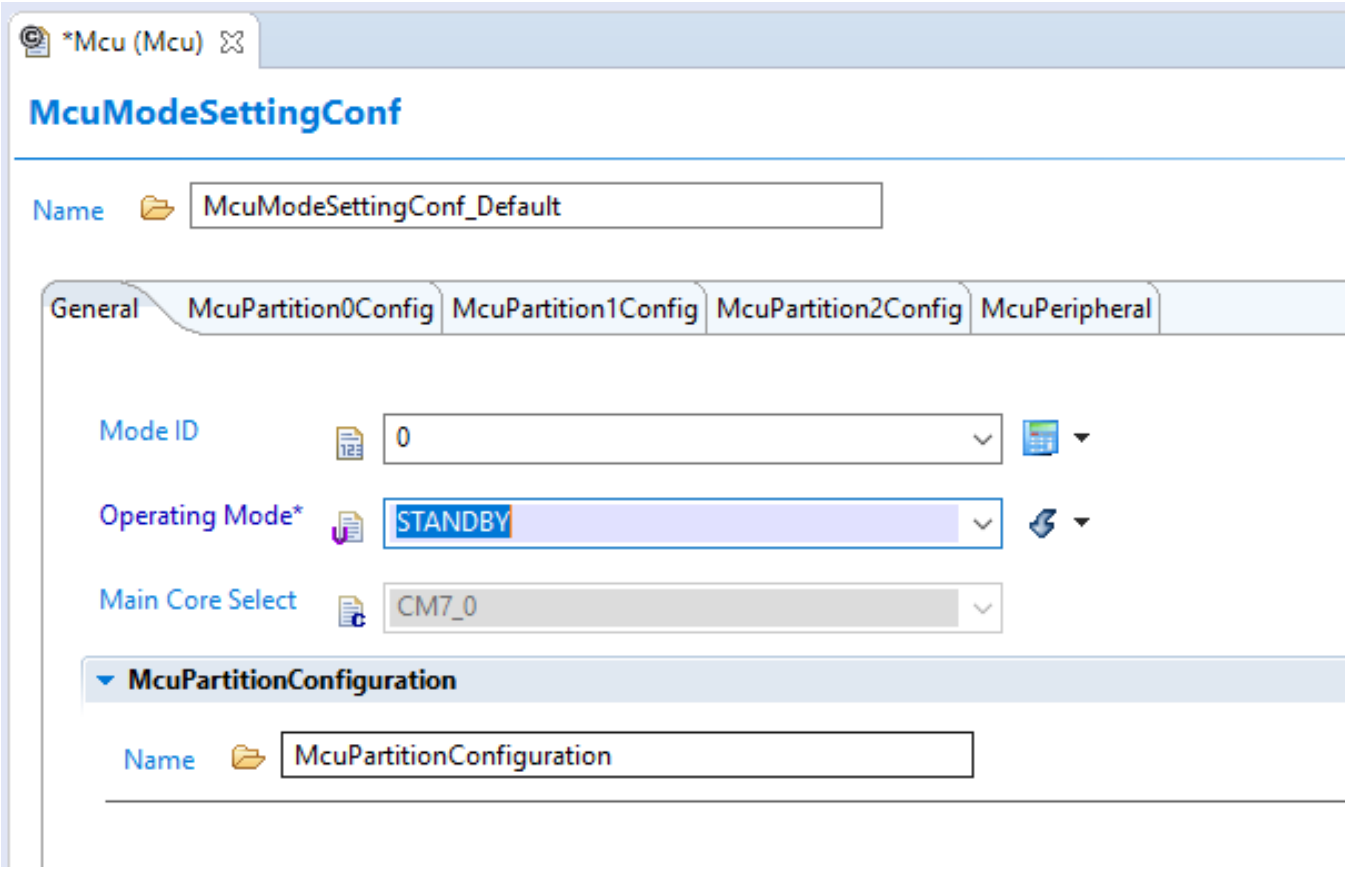
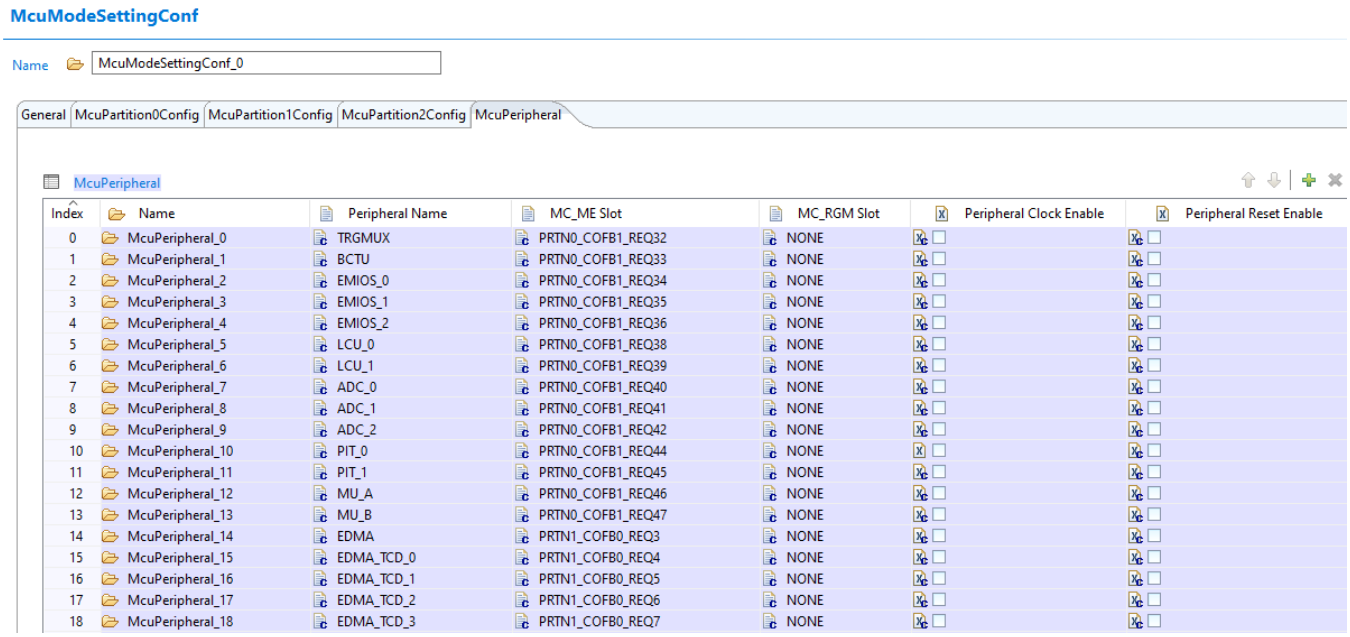


Figure 3.9 Tresos Plugin snapshot for Operating Mode = STANDBY mode form.



Index	Name	Peripheral Name	MC_ME Slot	MC_RGM Slot	Peripheral Clock Enable	Peripheral Reset Enable
0	McuPeripheral_0	TRGMUX	PRTN0_COFB1_REQ32	NONE		
1	McuPeripheral_1	BCTU	PRTN0_COFB1_REQ33	NONE		
2	McuPeripheral_2	EMIOS_0	PRTN0_COFB1_REQ34	NONE		
3	McuPeripheral_3	EMIOS_1	PRTN0_COFB1_REQ35	NONE		
4	McuPeripheral_4	EMIOS_2	PRTN0_COFB1_REQ36	NONE		
5	McuPeripheral_5	LCU_0	PRTN0_COFB1_REQ38	NONE		
6	McuPeripheral_6	LCU_1	PRTN0_COFB1_REQ39	NONE		
7	McuPeripheral_7	ADC_0	PRTN0_COFB1_REQ40	NONE		
8	McuPeripheral_8	ADC_1	PRTN0_COFB1_REQ41	NONE		
9	McuPeripheral_9	ADC_2	PRTN0_COFB1_REQ42	NONE		
10	McuPeripheral_10	PIT_0	PRTN0_COFB1_REQ44	NONE		
11	McuPeripheral_11	PIT_1	PRTN0_COFB1_REQ45	NONE		
12	McuPeripheral_12	MU_A	PRTN0_COFB1_REQ46	NONE		
13	McuPeripheral_13	MU_B	PRTN0_COFB1_REQ47	NONE		
14	McuPeripheral_14	EDMA	PRTN1_COFB0_REQ3	NONE		
15	McuPeripheral_15	EDMA_TCD_0	PRTN1_COFB0_REQ4	NONE		
16	McuPeripheral_16	EDMA_TCD_1	PRTN1_COFB0_REQ5	NONE		
17	McuPeripheral_17	EDMA_TCD_2	PRTN1_COFB0_REQ6	NONE		
18	McuPeripheral_18	EDMA_TCD_3	PRTN1_COFB0_REQ7	NONE		

Figure 3.10 Tresos Plugin snapshot for disable clock of all peripherals form

The sequence can also be split, giving the posibility to run application code in between the steps:

1. Mcu_InitClock (the system clock source must be changed to FIRC).

2. Mcu_SetMode (Disable clock of all peripherals and Operating Mode = SOC_PREPARE_STANDBY).

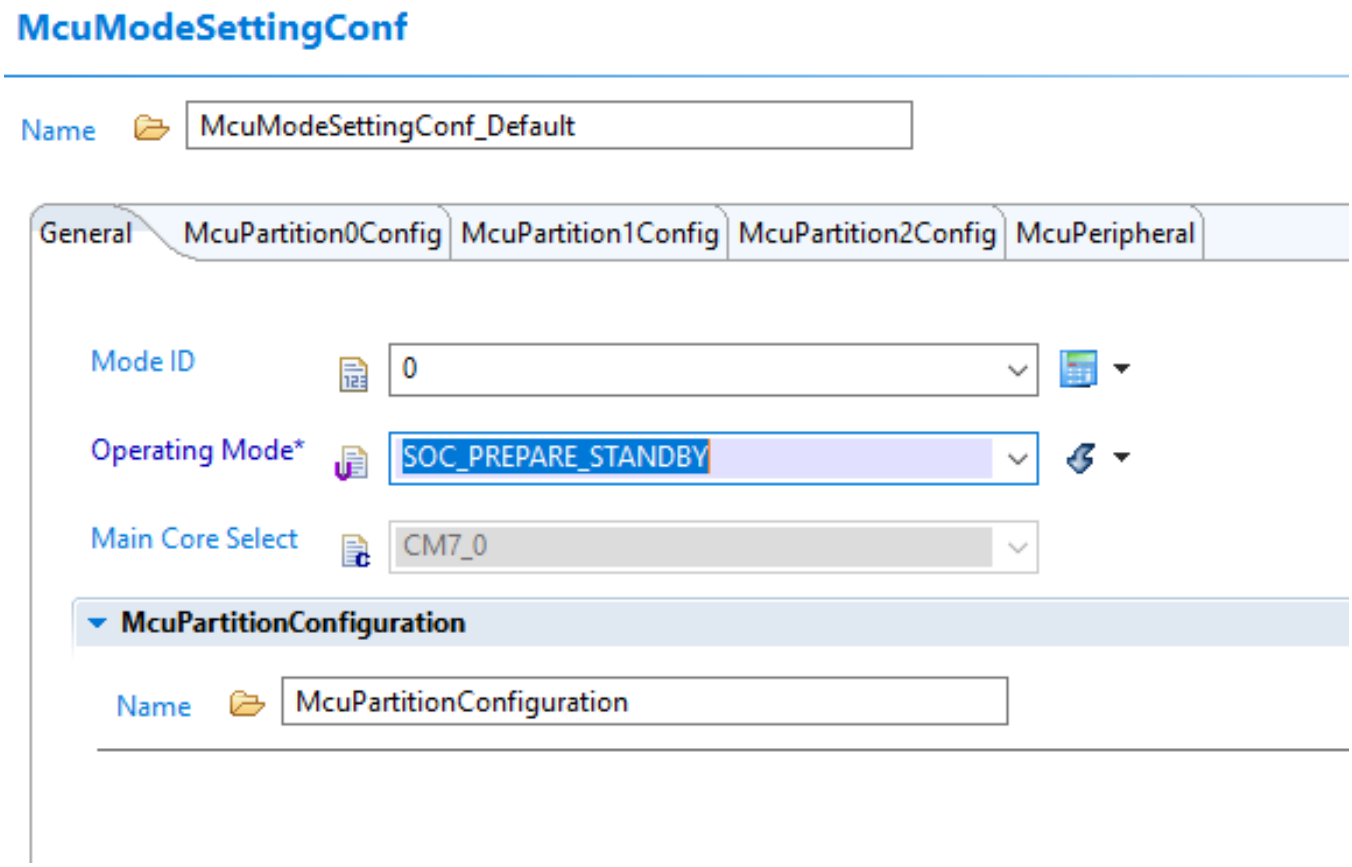


Figure 3.11 Tresos Plugin snapshot for Operating Mode = SOC_PREPARE_STANDBY mode form.

3. Mcu_SetMode (Operating Mode = SOC_STANDBY).

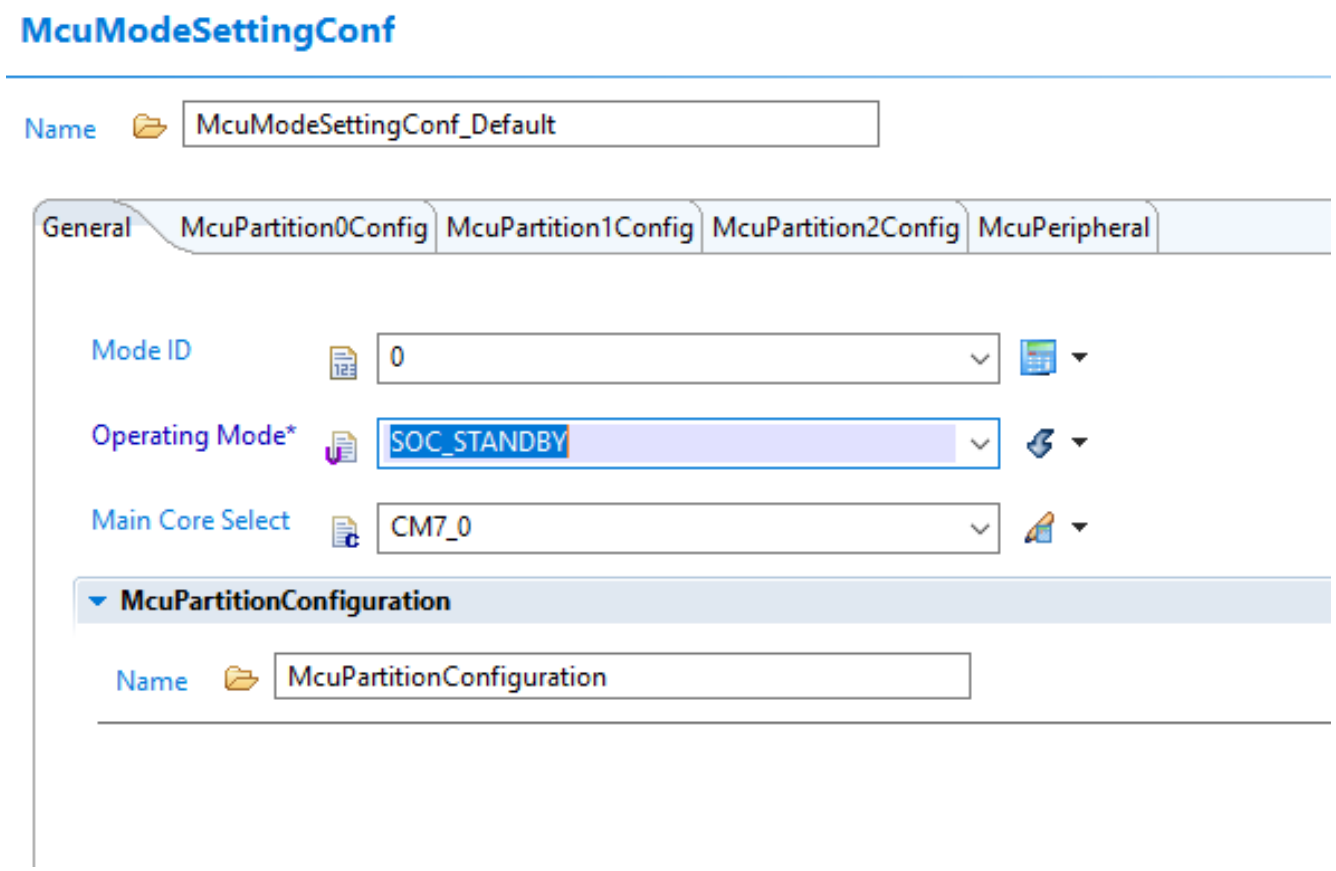


Figure 3.12 Tresos Plugin snapshot for Operating Mode = SOC_STANDBY mode form.

3.6.6 How to configure the system clock frequency

- The chip supports the clocking modes, as follows:

1. Option A - High Performance mode (CORE_CLK 160 MHz) (only available in Run mode)
2. Option B - Reduced Speed mode (CORE_CLK 120 MHz) (only available in Run mode)
3. Option C - Boot Standby mode (CORE_CLK 24 MHz)
4. Option D - Low-Speed RUN mode (CORE_CLK 48 MHz)
5. Option E - Low-Speed Run mode (CORE_CLK 3 MHz)
6. Option E2 - Very-Low-Speed Run mode (CORE_CLK 750 kHz)
7. Option F - Operation in 1:1 mode with CORE_CLK and AXBS_CLK at same speed (only available in Run mode)

Any clock frequency selected must adhere to the same clock divider ratios shown in those modes above, if not the MCU may be loss of clock.

The options A/B/F are only available in Run mode:

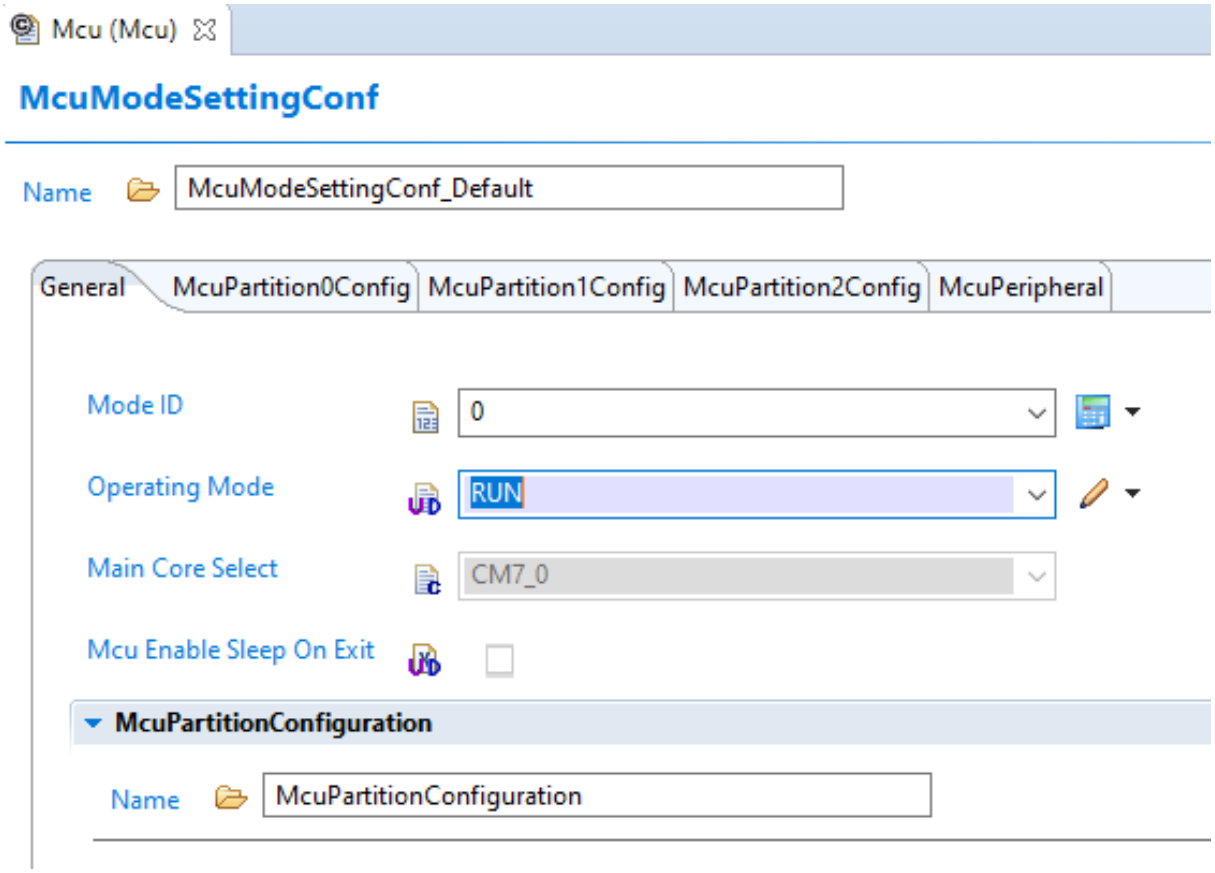


Figure 3.13 Tresos Plugin snapshot for Operating Mode = RUN mode form.

The system clock need to be configured input source(FIRC_CLK or PLL_PHI0_CLK) and divisor in the Mcu↩ Cgm0ClockMux0 tab so that any clock frequency selected must adhere to the same clock divider ratios shown in Clocking use case examples.

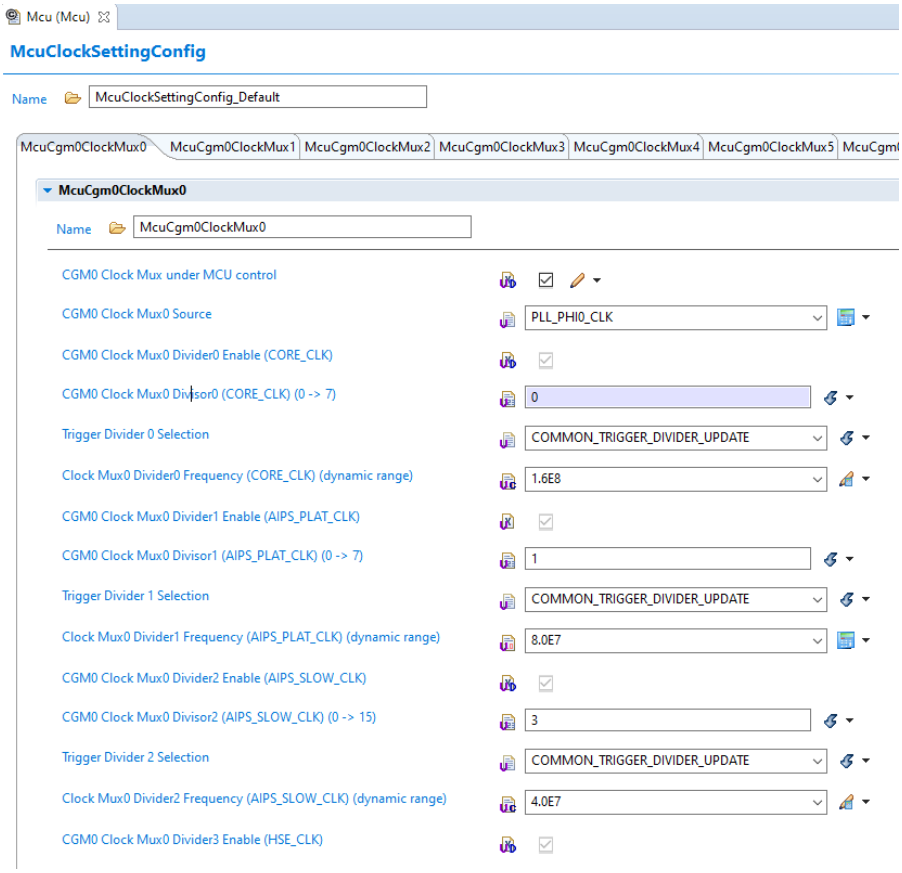


Figure 3.14 Tresos Plugin snapshot for SystemClock = PLL_PHI0_CLK form.

3.6.7 Clock ratio for specific case

- When using Power Conversion and Motor Control (PCMC) subsystem ,must set ratio AIPS_SLOW_CLK to one-fourth or one-half the frequency of CORE_CLK.
- Ratio between HSE IPS interface clock (AIPS_SLOW_CLK) and HSE module clock (HSE_CLK) base on status of bit field HSE_CLK_MODE_OPTION of register DCM_GPR:DCMROF21 :
  - 00b : Applicable for clocking option A. Ratio of 1:2 in between AIPS_SLOW_CLK and HSE_CLK, HSE_IAHB gasket enabled.
  - 01b : Applicable for clocking option C, D, E, E2, and F. Ratio of 1:2 in between AIPS_SLOW_CLK and HSE_CLK, HSE_IAHB gasket bypass.
  - 10b and 11b : both are applicable for clocking option B in same way. Ratio of 1:4 in between AIPS_↩ SLOW_CLK and HSE_CLK, HSE_IAHB gasket enabled.

3.6.8 CLKOUT_STANDBY measure via pin

- CLKOUT_STANDBY is available on two pads GPIO[12] and GPIO[138] but CLKOUT across functional reset and standby is supported only on GPIO[12] and OBE(output buffer enable) is controlled by DCM GPR bit. Please refer to DCMRWP1[3] bit for detail.

3.6.9 CMU_FC feature with FIRC_CLK frequency modes less than 24Mhz

- For FIRC_CLK frequency modes less than 24 MHz, safety modules like the CMU_Fx_n must be disabled for safety applications, because safety applications are to run on the PLL clocks. The CMU_Fx_n will cause erroneous FHH events if not disabled.

3.6.10 SMPS (DCDC) mode is used and the PMC is in low power mode

- If SMPS (DCDC) mode is used and the PMC is in low power mode, it must be ensured that SMPS supply input VDD_DCDC is fully operational and loadable before wake-up from low power mode. Otherwise a low voltage reset might occur on wakeup.

3.6.11 Note

The function Mcu_GetPowerModeState is not available on this platform.
Rationale: there are only two modes supported by the hardware (run and standby); The only mode which executes code is Run, thus mode status is implicitly known.

3.7 Runtime errors

The driver generates the following DET errors at runtime.

Table 3.3 Default Errors (reported by DET)

Function	Error Code	Condition triggering the error
Mcu_Init	MCU_E_INIT_FAILED	Invalid configuration pointer.
Mcu_InitClock	MCU_E_PARAM_CLOCK	Invalid input parameter.
Mcu_SetMode	MCU_E_PARAM_MODE	Invalid input parameter.
Mcu_InitRamSection	MCU_E_PARAM_↩ RAMSECTION	Invalid input parameter or invalid memory configuration.
Mcu_DistributePllClock	MCU_E_PLL_NOT_LOCKED	One of the used PLL's failed to achieve lock
All functions, except Mcu_Init and Mcu_GetVersionInfo	MCU_E_UNINIT	The driver is in an uninitialized state.
Mcu_GetMidrStructure	MCU_E_PARAM_POINTER	Invalid input parameter.
Mcu_GetVersionInfo	MCU_E_PARAM_POINTER	Invalid input parameter.
Mcu_Init	MCU_E_ALREADY_↩ INITIALIZED	The driver is already initialized.
Mcu_DisableCmu	MCU_E_CMU_INDEX_OUT↩ _OF_RANGE	Invalid input parameter.

The driver generates the following DEM errors at runtime.

Table 3.5 Default Errors (reported by DEM)

Function	Error Code	Condition triggering the error
Mcu_GetResetReason	Mcu_E_TimeoutFailure	Reset flags could not be cleared.
Mcu_GetResetRawValue	Mcu_E_TimeoutFailure	Reset flags could not be cleared.
Mcu_SetMode	Mcu_E_TimeoutFailure	The MC_ME or LPU mode transition failed.
Mcu_Init	Mcu_E_TimeoutFailure	The MC_ME mode transition failed.
Mcu_InitClock	Mcu_E_TimeoutFailure	The MC_ME mode transition failed.
Mcu_DisableCmu	Mcu_E_TimeoutFailure	Disable CMU failed.
Mcu_GetRamState	Mcu_E_TimeoutFailure	Get RAM state failed.

3.8 Symbolic Names Disclaimer

All containers having symbolicNameValue set to TRUE in the AUTOSAR schema will generate defines like:

#define <Mip>Conf_<Container_ShortName>_<Container_ID>

For this reason it is forbidden to duplicate the names of such containers across the RTD configurations or to use names that may trigger other compile issues (e.g. match existing `#ifdefs` arguments).

Chapter 4

Tresos Configuration Plug-in

This chapter describes the Tresos configuration plug-in for the driver. All the parameters are described below.

- Module Mcu
  - Container McuGeneralConfiguration
    - Parameter McuDevErrorDetect
    - Parameter McuVersionInfoApi
    - Parameter McuGetRamStateApi
    - Parameter McuInitClock
    - Parameter McuNoPll
    - Parameter McuEnterLowPowerMode
    - Parameter McuTimeout
    - Parameter McuEnableUserModeSupport
    - Parameter McuPerformResetApi
    - Parameter McuCalloutBeforePerformReset
    - Parameter McuPerformResetCallout
    - Parameter McuPmcNotification
    - Parameter McuCmuNotification
    - Parameter McuAlternateResetIsrUsed
    - Parameter McuCmuErrorIsrUsed
    - Parameter McuVoltageErrorIsrUsed
    - Parameter McuErrorIsrNotification
    - Parameter McuDisableRgmInit
    - Parameter McuDisablePmcInit
    - Parameter McuDisableRamWaitStatesConfig
    - Parameter McuDisableFlashWaitStatesConfig
    - Parameter McuPrepareMemoryConfig
    - Parameter McuTimeoutMethod
    - Parameter McuRegisterValuesOptimization
    - Reference McuEcucPartitionRef
    - Container McuControlledClocksConfiguration
      - Parameter McuFxoscUnderMcuControl
      - Parameter McuSxoscUnderMcuControl

- Parameter McuFircUnderMcuControl
- Parameter McuSircUnderMcuControl
- Parameter McuPll0UnderMcuControl
- Parameter McuPll1UnderMcuControl

– Container McuDebugConfiguration
  * Parameter McuDisableDemReportErrorStatus
  * Parameter McuGetSystemStateApi
  * Parameter McuGetPowerModeStateApi
  * Parameter McuGetPowerDomainApi
  * Parameter McuSscmGetMemConfigApi
  * Parameter McuSscmGetStatusApi
  * Parameter McuSscmGetUoptApi
  * Parameter McuGetMidrStructureApi
  * Parameter McuDisableCmuApi
  * Parameter McuEmiosConfigureGprenApi
  * Parameter McuGetClockFrequencyApi
  * Parameter McuPmcAeConfigApi
  * Parameter McuAecResetConfigApi
– Container McuCoreControlConfiguration
  * Parameter McuCoreBootAddressControl
– Container McuPublishedInformation
  * Container McuResetReasonConf
    - Parameter McuResetReason
– Container CommonPublishedInformation
  * Parameter ArReleaseMajorVersion
  * Parameter ArReleaseMinorVersion
  * Parameter ArReleaseRevisionVersion
  * Parameter ModuleId
  * Parameter SwMajorVersion
  * Parameter SwMinorVersion
  * Parameter SwPatchVersion
  * Parameter VendorApiInfix
  * Parameter VendorId
– Container McuModuleConfiguration
  * Parameter McuNumberOfMcuModes
  * Parameter McuRamSectors
  * Parameter McuResetSetting
  * Parameter McuCrystalFrequencyHz
  * Parameter McuSlowCrystalFrequencyHz
  * Parameter McuEMAC_MII_RMII_TX_CLKFrequencyHz
  * Parameter McuEMAC_MII_RX_CLKFrequencyHz
  * Parameter McuGMAC0_MII_RMII_TX_CLKFrequencyHz
  * Parameter McuGMAC0_MII_RX_CLKFrequencyHz
  * Parameter McuLFAST_REF_EXT_CLKFrequencyHz
  * Parameter McuSWG_PAD_CLKFrequencyHz

* Parameter McuClockSrcFailureNotification
* Container McuClockSettingConfig
  - Parameter McuClockSettingId
  - Container McuFIRC
  - Parameter McuFircUnderMcuControl
  - Parameter McuFircDivSel
  - Parameter McuFIRC_Frequency
  - Parameter McuFircStandbyEnable
  - Container McuSIRC
  - Parameter McuSircUnderMcuControl
  - Parameter McuSIRC_Frequency
  - Parameter McuSircStandbyEnable
  - Container McuFXOSC
  - Parameter McuFxoscUnderMcuControl
  - Parameter McuFxoscPowerDownCtr
  - Parameter McuFxoscBypass
  - Parameter McuFxoscMainComparator
  - Parameter McuFxoscCounter
  - Parameter McuFxoscOverdriveProtection
  - Parameter McuFXOSC_Frequency
  - Container McuSXOSC
  - Parameter McuSxoscUnderMcuControl
  - Parameter McuSxoscCounter
  - Parameter McuSxoscPowerDownCtr
  - Parameter McuSXOSC_Frequency
  - Container McuCgm0SettingConfig
  - Parameter McuPCSStepDuration
  - Parameter McuPCSSwitchDuration
  - Container McuCgm0PcsConfig
  - Parameter McuClockPcfsUnderMcuControl
  - Parameter McuPCS_Name
  - Parameter McuPCS_SourceFrequency
  - Parameter McuPCS_MaxAllowableDynamicIDD
  - Container McuCgm0ClockMux0
  - Parameter McuClockMuxUnderMcuControl
  - Parameter McuClkMux0_Source
  - Parameter McuClkMux0Div0_En
  - Parameter McuClkMux0Div0_Divisor
  - Parameter McuClkMux0Div0Trigger
  - Parameter McuClockMux0Divider0_Frequency
  - Parameter McuClkMux0Div1_En
  - Parameter McuClkMux0Div1_Divisor
  - Parameter McuClkMux0Div1Trigger
  - Parameter McuClockMux0Divider1_Frequency
  - Parameter McuClkMux0Div2_En
  - Parameter McuClkMux0Div2_Divisor
  - Parameter McuClkMux0Div2Trigger

- Parameter McuClockMux0Divider2_Frequency
- Parameter McuClkMux0Div3_En
- Parameter McuClkMux0Div3_Divisor
- Parameter McuClkMux0Div3Trigger
- Parameter McuClockMux0Divider3_Frequency
- Parameter McuClkMux0Div4_En
- Parameter McuClkMux0Div4_Divisor
- Parameter McuClkMux0Div4Trigger
- Parameter McuClockMux0Divider4_Frequency
- Parameter McuClkMux0Div5_En
- Parameter McuClkMux0Div5_Divisor
- Parameter McuClkMux0Div5Trigger
- Parameter McuClockMux0Divider5_Frequency
- Parameter McuClkMux0Div6_En
- Parameter McuClkMux0Div6_Divisor
- Parameter McuClkMux0Div6Trigger
- Parameter McuClockMux0Divider6_Frequency
- Parameter McuClkMux0Div7_En
- Parameter McuClkMux0Div7_Divisor
- Parameter McuClkMux0Div7Trigger
- Parameter McuClockMux0Divider7_Frequency
- Container McuCgm0ClockMux1
- Parameter McuClockMuxUnderMcuControl
- Parameter McuClkMux1_Source
- Parameter McuClkMux1Div0_En
- Parameter McuClkMux1Div0_Divisor
- Parameter McuClockMux1Divider0_Frequency
- Container McuCgm0ClockMux2
- Parameter McuClockMuxUnderMcuControl
- Parameter McuClkMux2_Source
- Parameter McuClkMux2Div0_En
- Parameter McuClkMux2Div0_Divisor
- Parameter McuClockMux2Divider0_Frequency
- Container McuCgm0ClockMux3
- Parameter McuClockMuxUnderMcuControl
- Parameter McuClkMux3_Source
- Parameter McuClkMux3Div0_En
- Parameter McuClkMux3Div0_Divisor
- Parameter McuClockMux3Divider0_Frequency
- Container McuCgm0ClockMux4
- Parameter McuClockMuxUnderMcuControl
- Parameter McuClkMux4_Source
- Parameter McuClkMux4Div0_En
- Parameter McuClkMux4Div0_Divisor
- Parameter McuClockMux4Divider0_Frequency
- Container McuCgm0ClockMux5
- Parameter McuClockMuxUnderMcuControl

- Parameter McuClkMux5_Source
- Parameter McuClkMux5Div0_En
- Parameter McuClkMux5Div0_Divisor
- Parameter McuClockMux5Divider0_Frequency
- Container McuCgm0ClockMux6
- Parameter McuClockMuxUnderMcuControl
- Parameter McuClkMux6_Source
- Parameter McuClkMux6Div0_En
- Parameter McuClkMux6Div0_Divisor
- Parameter McuClockMux6Divider0_Frequency
- Container McuCgm0ClockMux7
- Parameter McuClockMuxUnderMcuControl
- Parameter McuClkMux7_Source
- Parameter McuClkMux7Div0_En
- Parameter McuClkMux7Div0_Divisor
- Parameter McuClockMux7Divider0_Frequency
- Container McuCgm0ClockMux8
- Parameter McuClockMuxUnderMcuControl
- Parameter McuClkMux8_Source
- Parameter McuClkMux8Div0_En
- Parameter McuClkMux8Div0_Divisor
- Parameter McuClockMux8Divider0_Frequency
- Container McuCgm0ClockMux9
- Parameter McuClockMuxUnderMcuControl
- Parameter McuClkMux9_Source
- Parameter McuClkMux9Div0_En
- Parameter McuClkMux9Div0_Divisor
- Parameter McuClockMux9Divider0_Frequency
- Container McuCgm0ClockMux10
- Parameter McuClockMuxUnderMcuControl
- Parameter McuClkMux10_Source
- Parameter McuClkMux10Div0_En
- Parameter McuClkMux10Div0_Divisor
- Parameter McuClockMux10Divider0_Frequency
- Container McuCgm0ClockMux11
- Parameter McuClockMuxUnderMcuControl
- Parameter McuClkMux11_Source
- Parameter McuClkMux11Div0_En
- Parameter McuClkMux11Div0_Divisor
- Parameter McuClockMux11Divider0_Frequency
- Container McuCgm0ClockMux12
- Parameter McuClockMuxUnderMcuControl
- Parameter McuClkMux12_Source
- Parameter McuClkMux12Div0_En
- Parameter McuClkMux12Div0_Divisor
- Parameter McuClockMux12Divider0_Frequency
- Container McuCgm0ClockMux13

- Parameter McuClockMuxUnderMcuControl
- Parameter McuClkMux13_Source
- Parameter McuClkMux13Div0_En
- Parameter McuClkMux13Div0_Divisor
- Parameter McuClockMux13Divider0_Frequency
- Container McuCgm0ClockMux14
- Parameter McuClockMuxUnderMcuControl
- Parameter McuClkMux14_Source
- Parameter McuClkMux14Div0_En
- Parameter McuClkMux14Div0_Divisor
- Parameter McuClockMux14Divider0_Frequency
- Container McuCgm0ClockMux15
- Parameter McuClockMuxUnderMcuControl
- Parameter McuClkMux15_Source
- Parameter McuClkMux15Div0_En
- Parameter McuClkMux15Div0_Divisor
- Parameter McuClockMux15Divider0_Frequency
- Container McuCgm0ClockMux16
- Parameter McuClockMuxUnderMcuControl
- Parameter McuClkMux16_Source
- Parameter McuClkMux16Div0_En
- Parameter McuClkMux16Div0_Divisor
- Parameter McuClockMux16Divider0_Frequency
- Container McuCgm0ClockMux17
- Parameter McuClockMuxUnderMcuControl
- Parameter McuClkMux17_Source
- Parameter McuClkMux17Div0_En
- Parameter McuClkMux17Div0_Divisor
- Parameter McuClockMux17Divider0_Frequency
- Container McuCgm0ClockMux18
- Parameter McuClockMuxUnderMcuControl
- Parameter McuClkMux18_Source
- Parameter McuClkMux18Div0_En
- Parameter McuClkMux18Div0_Divisor
- Parameter McuClockMux18Divider0_Frequency
- Container McuCgm0ClockMux19
- Parameter McuClockMuxUnderMcuControl
- Parameter McuClkMux19_Source
- Parameter McuClkMux19Div0_En
- Parameter McuClkMux19Div0_Divisor
- Parameter McuClockMux19Divider0_Frequency
- Container McuRtcClockSelect
- Parameter McuClockMuxUnderMcuControl
- Parameter McuRtc_Source
- Parameter McuRtc_Frequency
- Container McuPll_0
- Parameter McuPLLUnderMcuControl

- Parameter McuPLLEnabled
- Container McuPll_Configuration
- Parameter McuPllDvRdiv
- Parameter McuPllDvMfi
- Parameter McuPllDvOdiv2
- Parameter McuPllFmSscgbyp
- Parameter McuPllFmSpreadctl
- Parameter McuPllFmStepSize
- Parameter McuPllFmStepNo
- Parameter McuPllFdFmod
- Parameter McuPllFdMdp
- Parameter McuPllFdMfn
- Parameter McuPllFdSdmen
- Parameter McuPllFdSdm2
- Parameter McuPllFdSdm3
- Parameter McuPllOdiv0_En
- Parameter McuPllOdiv0_Div
- Parameter McuPllOdiv1_En
- Parameter McuPllOdiv1_Div
- Container McuPll_Parameter
- Parameter PLL_PHI0_Frequency
- Parameter PLL_PHI1_Frequency
- Parameter PLL_VCO_Frequency
- Container McuPll_1
- Parameter McuPLLUnderMcuControl
- Parameter McuPLLEnabled
- Container McuPll_Configuration
- Parameter McuPllDvRdiv
- Parameter McuPllDvMfi
- Parameter McuPllDvOdiv2
- Parameter McuPllOdiv0_En
- Parameter McuPllOdiv0_Div
- Parameter McuPllOdiv1_En
- Parameter McuPllOdiv1_Div
- Parameter McuPllOdiv2_En
- Parameter McuPllOdiv2_Div
- Container McuPll_Parameter
- Parameter PLL_PHI0_Frequency
- Parameter PLL_PHI1_Frequency
- Parameter PLL_PHI2_Frequency
- Parameter PLL_VCO_Frequency
- Container McuClkMonitor
- Parameter McuClockMonitorUnderMcuControl
- Parameter McuClkMonitorEn
- Parameter McuCmuName
- Parameter McuFHHAsyncEventEn
- Parameter McuFLLAsyncEventEn

- Parameter McuFHHInterruptEn
- Parameter McuFLLInterruptEn
- Container McuClockReferencePoint
- Parameter McuClockReferencePointFrequency
- Parameter McuClockFrequencySelect

* Container McuDemEventParameterRefs
  - Reference MCU_E_TIMEOUT_FAILURE
  - Reference MCU_E_INVALIDFXOSC_CONFIG
  - Reference MCU_E_CLOCKMUXSWITCH_FAILURE
  - Reference MCU_E_CLOCK_FAILURE
* Container McuModeSettingConf
  - Parameter McuMode
  - Parameter McuPowerMode
  - Parameter McuMainCoreSelect
  - Parameter McuCoreLockStepEnable
  - Parameter McuEnableSleepOnExit
  - Container McuPartitionConfiguration
  - Container McuPartition0Config
  - Parameter McuPartitionUnderMcuControl
  - Parameter McuPartitionPowerUnderMcuControl
  - Parameter McuPrtnCofb0UnderMcuControl
  - Parameter McuPrtnCofb1UnderMcuControl
  - Parameter McuPartitionClockEnable
  - Container McuCore0Configuration
  - Parameter McuCoreUnderMcuControl
  - Parameter McuCoreClockEnable
  - Parameter McuCoreBootAddress
  - Parameter McuCoreBootAddressLinkerSym
  - Container McuCore1Configuration
  - Parameter McuCoreUnderMcuControl
  - Parameter McuCoreClockEnable
  - Parameter McuCoreBootAddress
  - Parameter McuCoreBootAddressLinkerSym
  - Container McuCore4Configuration
  - Parameter McuCoreUnderMcuControl
  - Parameter McuCoreClockEnable
  - Parameter McuCoreBootAddress
  - Parameter McuCoreBootAddressLinkerSym
  - Container McuCore3Configuration
  - Parameter McuCoreUnderMcuControl
  - Parameter McuCoreClockEnable
  - Parameter McuCoreBootAddress
  - Parameter McuCoreBootAddressLinkerSym
  - Container McuPartition1Config
  - Parameter McuPartitionUnderMcuControl
  - Parameter McuPartitionPowerUnderMcuControl
  - Parameter McuPrtnCofb0UnderMcuControl

- Parameter McuPrtnCofb1UnderMcuControl
- Parameter McuPrtnCofb2UnderMcuControl
- Parameter McuPrtnCofb3UnderMcuControl
- Parameter McuPartitionClockEnable
- Container McuPartition2Config
- Parameter McuPartitionUnderMcuControl
- Parameter McuPartitionPowerUnderMcuControl
- Parameter McuPrtnCofb0UnderMcuControl
- Parameter McuPrtnCofb1UnderMcuControl
- Parameter McuPrtnCofb2UnderMcuControl
- Parameter McuPartitionClockEnable
- Container McuPartition3Config
- Parameter McuPartitionUnderMcuControl
- Parameter McuPartitionPowerUnderMcuControl
- Parameter McuPrtnCofb0UnderMcuControl
- Parameter McuPrtnCofb1UnderMcuControl
- Parameter McuPrtnCofb2UnderMcuControl
- Parameter McuPartitionClockEnable
- Container McuPeripheral
- Parameter McuPeripheralName
- Parameter McuModeEntrySlot
- Parameter McuPeripheralClockEnable
- Container McuDcmGprConfiguration
- Parameter McuDcmGprUnderMcuControl
- Parameter McuBootBaseAddress
- Parameter McuSIRC_TRIM_BYP_STDBY_EXTControl
- Parameter McuPMC_TRIM_RGM_DCF_BYP_STDBY_EXTControl
- Parameter McuFIRC_TRIM_BYP_STDBY_EXTControl
- Parameter McuDCM_SCAN_BYP_STDBY_EXTControl
- Parameter McuGlobalPadkeepingEnable

* Container McuRamSectorSettingConf
  - Parameter McuRamSectorId
  - Parameter McuRamDefaultValue
  - Parameter McuRamSectionBaseAddress
  - Parameter McuRamSectionSize
  - Parameter McuRamSectionWriteSize
  - Parameter McuRamSectionBaseAddrLinkerSym
  - Parameter McuRamSectionSizeLinkerSym
* Container McuResetConfig
  - Parameter McuResetType
  - Parameter McuFuncResetEscThreshold
  - Parameter McuDestResetEscThreshold
  - Container McuResetSourcesConfig
  - Container McuFCCU_RST_ResetSource
  - Parameter McuDisableReset
  - Container McuSWT0_RST_ResetSource
  - Parameter McuDisableReset

- Container McuSWT1_RST_ResetSource
- Parameter McuDisableReset
- Container McuSWT2_RST_ResetSource
- Parameter McuDisableReset
- Container McuSWT3_RST_ResetSource
- Parameter McuDisableReset
- Container McuJTAG_RST_ResetSource
- Parameter McuDisableReset
- Container McuDEBUG_FUNC_ResetSource
- Parameter McuDisableReset
- Container McuResetGeneratorConfiguration
- Parameter McuRegsOtpReset
- Parameter McuCanPhyReset
- Parameter McuLinPhyHpReset
- Parameter McuLinPhyLpReset
- Parameter McuGduReset
- Parameter McuHviReset
- Parameter McuDpgaReset
- Parameter McuTempsensorReset
- Parameter McuCxpiReset

* Container McuPowerControl
  - Container McuPMC_Config
  - Parameter McuLMAUTOENEnable
  - Parameter McuLVDIEEnable
  - Parameter McuHVDIEEnable
  - Parameter McuLMSMPSENEnable
  - Parameter McuLVRBLPENEnable
  - Parameter McuLPM25ENEnable
  - Parameter McuFASTRECEnable
  - Parameter McuLMBCTLENEnable
  - Parameter McuLMENEnable
  - Parameter McuSMPSConfSelect
  - Parameter McuSMPSPeriod
  - Parameter McuSMPSOnTime3V
  - Parameter McuSMPSOnTime5V
  - Parameter McuSMPSLPM15EN
  - Parameter McuSMPSDitherEn
  - Parameter McuSMPSDitherConf
  - Container McuPMC_AE_Config
  - Parameter McuLowVoltageDetectInterruptsOnVLSEnable
  - Parameter McuLowVoltageDetectInterruptsOnVDDCEnable
  - Parameter McuHighVoltageDetectInterruptOnVDDEnable
  - Parameter McuHighVoltageDetectInterruptOnVDDINTAndVDD15Enable
  - Parameter McuLinphySupplyEnable
  - Parameter McuVDDCEnable
  - Parameter McuLvdVlsSelect
  - Parameter McuLinphySupplySelect
  - Parameter McuVddVoltageLevelSelect

4.1 Module Mcu

Configuration of the MicroController Unit (MCU) module.

Included containers:

- McuGeneralConfiguration
- McuDebugConfiguration
- McuCoreControlConfiguration
- McuPublishedInformation
- CommonPublishedInformation
- McuModuleConfiguration

Property	Value
type	ECUC-MODULE-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantSupport	true
supportedConfigVariants	VARIANT-POST-BUILD, VARIANT-PRE-COMPILE

4.2 Container McuGeneralConfiguration

This container contains the general configuration for the MCU driver.

Included subcontainers:

- McuControlledClocksConfiguration

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.3 Parameter McuDevErrorDetect

Pre-processor switch for enabling the default error detection and reporting to the DET.

The switch McuDevErrorDetect shall switch the Default Error Tracer (Det) detection and notification ON or OFF.

The detection of default errors is configurable (ON/OFF) at precompile time.

#define MCU_DEV_ERROR_DETECT (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.4 Parameter McuVersionInfoApi

Pre-processor switch to enable/disable the API to read out the modules version information.

#define MCU_VERSION_INFO_API (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.5 Parameter McuGetRamStateApi

Pre-processor switch to enable/disable the API Mcu_GetRamState.

#define MCU_GET_RAM_STATE_API (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.6 Parameter McuInitClock

If this parameter is set to FALSE, the clock initialization has to be disabled from the MCU driver. This concept applies when there are some write once clock registers and a bootloader is present. If this parameter is set to TRUE, the MCU driver is responsible with the clock initialization.

#define MCU_INIT_CLOCK (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.7 Parameter McuNoPll

This parameter shall be set True, if the H/W does not have a PLL or the PLL circuitry is enabled after the power on without S/W intervention. In this case MCU_DistributePllClock has to be disabled and MCU_GetPllStatus has to return MCU_PLL_STATUS_UNDEFINED. Otherwise this parameters has to be set False.

#define MCU_NO_PLL (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.8 Parameter McuEnterLowPowerMode

If this parameter has been configured to 'TRUE', the function 'Mcu_SetMode()' shall not be impacted and behave as specified.

If this parameter has been configured to 'FALSE', the function 'Mcu_SetMode()' shall not perform the transition to any low power modes as are 'STOP' or 'HALT' or any other mode, where the core stops execution.

#define MCU_ENTER_LOW_POWER_MODE (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.9 Parameter McuTimeout

This parameter represents the maximum number of loops for blocking functionality.

The maximum time needed for a MC_ME transition from DRUN to DRUN with keeping PLL running is 3 ms.

Please take this into consideration when choosing the value for this parameter.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	50000
max	4294967295
min	1

4.10 Parameter McuEnableUserModeSupport

When this parameter is enabled, the MDL module will adapt to run from User Mode, with the following measures:

a) configuring REG_PROT for ABC1, ABC2 IPs so that the registers under protection can be accessed from user mode by setting UAA bit in REG_PROT_GCR to 1

b) using 'call trusted function' stubs for all internal function calls that access registers requiring supervisor mode.

c) other module specific measures

for more information, please see chapter 5.7 User Mode Support in IM

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.11 Parameter McuPerformResetApi

Pre-processor switch to enable/disable the use the Mcu_PerformReset() API.

OFF - Mcu_PerformReset() API is not used.

ON - Mcu_PerformReset() API is used.

#define MCU_PERFORM_RESET_API (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.12 Parameter McuCalloutBeforePerformReset

Check this if you want a callout function, called by MCU right before Mcu_PerformReset().

This parameter is available for configuration only if "McuPerformResetApi" is ON.

#define MCU_RESET_CALLOUT_USED (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.13 Parameter McuPerformResetCallout

Function name of callout.

The field is editable only if "McuCalloutBeforePerformReset" is ON.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FUNCTION-NAME-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	NULL_PTR

4.14 Parameter McuPmcNotification

Function pointer to callback function.

This notification will be called by the driver before entering in Standby Mode,

after the Lastmile Regulator is disabled.

This node is not supported on S32K312 derivatives

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FUNCTION-NAME-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	NULL_PTR

4.15 Parameter McuCmuNotification

Function pointer to callback function.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FUNCTION-NAME-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	NULL_PTR

4.16 Parameter McuAlternateResetIsrUsed

Check this if you have any reset source demoted to IRQ (i.e. at least one McuModuleConfiguration/McuResetConfig/*/McuDisableReset = 'true').

#define MCU_RESET_ALTERNATE_ISR_USED (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.17 Parameter McuCmuErrorIsrUsed

Check this if clock source failure notifications are enabled (i.e. McuModuleConfiguration/McuClockSrcFailureNotification = 'ENABLED').

#define MCU_CMU_ERROR_ISR_USED (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.18 Parameter McuVoltageErrorIsrUsed

Check this if under-voltage or over-voltage monitoring IRQs are enabled in the McuPowerControl container.

#define POWER_IP_VOLTAGE_ERROR_ISR_USED (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.19 Parameter McuErrorIsrNotification

Function name of callout. This function will be called by the error ISR.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FUNCTION-NAME-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	NULL_PTR

4.20 Parameter McuDisableRgmInit

If this parameter is set to TRUE, the Reset Generation Module (MC_RGM) initialization will be disabled in the MCU driver.

If this parameter is set to FALSE, the MCU driver is responsible for the Reset Generation Module (MC_RGM) initialization.

#define POWER_IP_DISABLE_RGM_INIT (STD_ON)/(STD_OFF) will be generated in Power_Ip_Cfg_Defines.h file.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.21 Parameter McuDisablePmcInit

If this parameter is set to TRUE, the Power Management Controller (PMC) initialization will be disabled in the MCU driver.

If this parameter is set to FALSE, the MCU driver is responsible for the Power Management Controller (PMC) initialization.

#define POWER_IP_DISABLE_PMC_INIT (STD_ON)/(STD_OFF) will be generated in Power_Ip_Cfg_Defines.h file.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.22 Parameter McuDisableRamWaitStatesConfig

Check this if you want the PRAMC configuration to be bypassed.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.23 Parameter McuDisableFlashWaitStatesConfig

Check this if you want the Flash configuration from RAM to be bypassed.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.24 Parameter McuPrepareMemoryConfig

Function name of a callout that will be called before and after configuring

the PRAM controller. It will have a parameter that will specify if it is

the entry or the exit point of the controllers configuration.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FUNCTION-NAME-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	NULL_PTR

4.25 Parameter McuTimeoutMethod

McuTimeoutMethod

Configures the timeout method.

Based on this selection a certain timeout method from OsIf will be used in the driver.

Note: If OSIF_COUNTER_SYSTEM or OSIF_COUNTER_CUSTOM are selected make sure the corresponding timer is enabled in OsIf General configuration.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	OSIF_COUNTER_DUMMY
literals	['OSIF_COUNTER_DUMMY', 'OSIF_COUNTER_SYSTEM', 'OSIF_↩ COUNTER_CUSTOM']

4.26 Parameter McuRegisterValuesOptimization

Check this if register values of CMUs, PCFS will be generated by configuration tool.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE

Property	Value
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	DISABLED
literals	['ENABLED', 'DISABLED']

4.27 Reference McuEcucPartitionRef

Maps the MCU driver to zero or multiple ECUC partitions to make the

modules API available in this partition.

Tags: atp.Status=draft

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	Infinite
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/EcuC/EcucPartitionCollection/EcucPartition

4.28 Container McuControlledClocksConfiguration

This container contains pre-compile options for all the clock sources under MCU control.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF

Property	Value
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.29 Parameter McuFxoscUnderMcuControl

Check this if FXOSC is under MCU control in any of the Mcu Clock Setting Configurations

(i.e. at least one McuClockSettingConfig/*/McuFXOSC/McuFxoscUnderMcuControl = 'true').

#define MCU_FXOSC_UNDER_MCU_CONTROL (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.30 Parameter McuSxoscUnderMcuControl

Check this if SXOSC is under MCU control in any of the Mcu Clock Setting Configurations

(i.e. at least one McuClockSettingConfig/*/McuSXOSC/McuSxoscUnderMcuControl = 'true').

#define MCU_SXOSC_UNDER_MCU_CONTROL (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF

Property	Value
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.31 Parameter McuFircUnderMcuControl

Check this if FIRC is under MCU control in any of the Mcu Clock Setting Configurations

(i.e. at least one McuClockSettingConfig/*/McuFIRC/McuFircUnderMcuControl = 'true').

#define MCU_FIRC_UNDER_MCU_CONTROL (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.32 Parameter McuSircUnderMcuControl

Check this if SIRC is under MCU control in any of the Mcu Clock Setting Configurations
(i.e. at least one McuClockSettingConfig/*/McuSIRC/McuSircUnderMcuControl = 'true').
#define MCU_SIRC_UNDER_MCU_CONTROL (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.
Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.33 Parameter McuPll0UnderMcuControl

Check this if PLL is under MCU control in any of the Mcu Clock Setting Configurations

(i.e. at least one McuClockSettingConfig/*/McuPll_0/McuPLLUnderMcuControl = 'true').

#define MCU_PLL0_UNDER_MCU_CONTROL (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.34 Parameter McuPll1UnderMcuControl

Check this if PLLAUX is under MCU control in any of the Mcu Clock Setting Configurations

(i.e. at least one McuClockSettingConfig/*/McuPll_1/McuPLLUnderMcuControl = 'true').

#define MCU_PLL1_UNDER_MCU_CONTROL (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.35 Container McuDebugConfiguration

This container contains option for non-ASR APIs used for debug or extra-implementation.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.36 Parameter McuDisableDemReportErrorStatus

Enable/Disable the API for reporting the Dem Error.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.37 Parameter McuGetSystemStateApi

Enable/Disable the API for System state information: Mcu_GetSystem_State().

Information extracted from SSCM hw IP.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.38 Parameter McuGetPowerModeStateApi

Enable/Disable the API for MC_ME state: Mcu_GetPowerMode_State().

Get information regarding current power mode, enabled clocks, etc (content of ME_GS register).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.39 Parameter McuGetPowerDomainApi

Enable/Disable the API for MC_PCU state: Mcu_GetPowerDomain_Status().

Get information from PCU_STAT register.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.40 Parameter McuSscmGetMemConfigApi

Enable/Disable the API for Mcu_SscmGetMemConfig().

Get information from SSCM_MEMCONFIG register.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.41 Parameter McuSscmGetStatusApi

Enable/Disable the API for Mcu_SscmGetStatus().

Get information from SSCM_STATUS register.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.42 Parameter McuSscmGetUoptApi

Enable/Disable the API for Mcu_SscmGetUopt().

Get information from SSCM_UOPT register.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.43 Parameter McuGetMidrStructureApi

Enable/Disable the API for Mcu_GetMidrStructure().

Get information from SIUL2 MIDRn registers.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.44 Parameter McuDisableCmuApi

Enable/Disable the API for disabling the clock monitoring unit.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.45 Parameter McuEmiosConfigureGprenApi

Enable/Disable the API for Mcu_EmiosConfigureGpren().

Changes the GPREN bit of the EMIOS_MCR register of an addressed eMIOS instance.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.46 Parameter McuGetClockFrequencyApi

Enable/Disable the API for Mcu_GetClockFrequency().

Return the frequency of a given clock.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.47 Parameter McuPmcAeConfigApi

Enable/Disable the API for Mcu_PmcAeConfig().

Configure the Power Management Controller AE.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.48 Parameter McuAecResetConfigApi

Enable/Disable the API for Mcu_AecResetConfig().

Configure the reset generator AEC.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.49 Container McuCoreControlConfiguration

This configuration holds global control over the partition specific core control features.

This container is implementation specific.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.50 Parameter McuCoreBootAddressControl

Global ENABLE / DISABLE of the code that writes the PRTNm_COREn_ADDR registers.

These registers give the boot addresses for the cores in their corresponding partitions.

If this check box is ON, the registers will be written during each Mcu_SetMode() call.

#define MCU_CONFIGURE_CADDRN (STD_ON)/(STD_OFF) will be generated in Mcu_Cfg.h file.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.51 Container McuPublishedInformation

Container holding all MCU specific published information parameters.

Included subcontainers:

- McuResetReasonConf

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.52 Container McuResetReasonConf

This container contains the configuration for the different type of reset reason that can be retrieved from Mcu_GetResetReason Api.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF

Property	Value
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.53 Parameter McuResetReason

The parameter represents the different type of reset that a Micro supports. This parameter is referenced by the parameter EcuMResetReason in the ECU State manager module.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	true
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	255
min	0

4.54 Container CommonPublishedInformation

Common container, aggregated by all modules.

It contains published information about vendor and versions.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.55 Parameter ArReleaseMajorVersion

Major version number of AUTOSAR specification on which the appropriate implementation is based on.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	4
max	4
min	4

4.56 Parameter ArReleaseMinorVersion

Minor version number of AUTOSAR specification on which the appropriate implementation is based on.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	7
max	7
min	7

4.57 Parameter ArReleaseRevisionVersion

Revision version number of AUTOSAR specification on which the appropriate implementation is based on.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	0
max	0
min	0

4.58 Parameter ModuleId

Module ID of this module from Module List.

Note: Implementation Specific Parameter

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	101
max	101
min	101

4.59 Parameter SwMajorVersion

Major version number of the vendor specific implementation of the module. The numbering is vendor specific.

Note: Implementation Specific Parameter

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	3
max	3
min	3

4.60 Parameter SwMinorVersion

Minor version number of the vendor specific implementation of the module. The numbering is vendor specific.

Note: Implementation Specific Parameter

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	0
max	0
min	0

4.61 Parameter SwPatchVersion

Patch level version number of the vendor specific implementation of the module. The numbering is vendor specific.

Note: Implementation Specific Parameter

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	0
max	0
min	0

4.62 Parameter VendorApiInfix

In driver modules which can be instantiated several times on a single ECU, BSW00347 requires that the name of APIs is extended by the VendorId and a vendor specific name.

This parameter is used to specify the vendor specific name. In total, the Implementation specific name is generated as follows:

<ModuleName>_>VendorId>_<VendorApiInfix>.

E.g. assuming that the VendorId of the implementor is 123 and the implementer chose a VendorApiInfix of "v11r456" a api name

Can_Write defined in the SWS will translate to Can_123_v11r456Write.

This parameter is mandatory for all modules with upper multiplicity >

1. It shall not be used for modules with upper multiplicity =1.

Note: Implementation Specific Parameter

Property	Value
type	ECUC-STRING-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	

4.63 Parameter VendorId

Vendor ID of the dedicated implementation of this module according to the AUTOSAR vendor list.

Note: Implementation Specific Parameter

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	43
max	43
min	43

4.64 Container McuModuleConfiguration

This container contains the configuration for the MCU driver.

Included subcontainers:

- McuClockSettingConfig
- McuDemEventParameterRefs
- McuModeSettingConf
- McuRamSectorSettingConf
- McuResetConfig
- McuPowerControl

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.65 Parameter McuNumberOfMcuModes

This parameter shall represent the number of Modes available for the MCU (from "McuModeSettingConf" list).

CalculationFormula = Number of configured "McuModeSettingConf".

This parameter is not used.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	1
max	255
min	1

4.66 Parameter McuRamSectors

This parameter shall represent the number of RAM sectors available for the MCU (from "McuRamSectorSettingConf" list).

CalculationFormula = Number of configured "McuRamSectorSettingConf".

This parameter is not used.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	1
max	4294967295
min	0

4.67 Parameter McuResetSetting

This parameters applies to the function Mcu_PerformReset(), which performs a microcontroller reset using the hardware feature of the microcontroller.

Note: This parameter is not used by the current Implementation.

Software Reset occurs when Mcu_PerformReset() function is called.

This parameter is not used.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	1
max	255
min	1

4.68 Parameter McuCrystalFrequencyHz

Crystal Frequency or External Reference Frequency [Hz].

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE

Property	Value
defaultValue	1.6E7
max	4.0E7
min	8000000.0

4.69 Parameter McuSlowCrystalFrequencyHz

Slow Crystal Frequency or External Reference Frequency [Hz].

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	32768.0
max	32768.0
min	0.0

4.70 Parameter McuEMAC_MII_RMII_TX_CLKFrequencyHz

EMAC_MII_RMII_TX_CLK External Reference Frequency [Hz].

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A

Property	Value
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	5.0E7
max	3.2E9
min	0.0

4.71 Parameter McuEMAC_MII_RX_CLKFrequencyHz

EMAC_MII_RX_CLK External Reference Frequency [Hz].

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	2.5E7
max	3.2E9
min	0.0

4.72 Parameter McuGMAC0_MII_RMII_TX_CLKFrequencyHz

GMAC0_MII_RMII_TX_CLK External Reference Frequency [Hz].

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF

Property	Value
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	5.0E7
max	3.2E9
min	0.0

4.73 Parameter McuGMAC0_MII_RX_CLKFrequencyHz

GMAC0_MII_RX_CLK External Reference Frequency [Hz].

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	5.0E7
max	3.2E9
min	0.0

4.74 Parameter McuLFAST_REF_EXT_CLKFrequencyHz

LFAST_REF_EXT_CLK External Reference Frequency [Hz].

This node is supported on S32K39x derivative only.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	2.6E7
max	2.08E8
min	0.0

4.75 Parameter McuSWG_PAD_CLKFrequencyHz

SWG_PAD_CLK External Reference Frequency [Hz].

This node is supported on S32K39x derivative only.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	1.6E8
max	1.6E8
min	0.0

4.76 Parameter McuClockSrcFailureNotification

Enables/Disables clock failure notification.

In case this feature is not supported by HW the setting should be disabled.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	DISABLED
literals	['ENABLED', 'DISABLED']

4.77 Container McuClockSettingConfig

This container contains the configuration for the Clock settings of the MCU.

Included subcontainers:

- McuFIRC
- McuSIRC
- McuFXOSC
- McuSXOSC
- McuCgm0SettingConfig
- McuRtcClockSelect
- McuPll_0
- McuPll_1
- McuClkMonitor
- McuClockReferencePoint

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.78 Parameter McuClockSettingId

The Id of this McuClockSettingConfig to be used as argument for the API call Mcu_InitClock().

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	true
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	0
max	255
min	0

4.79 Container McuFIRC

This container contains the specific configuration of the MCU FIRC (Fast Internal RC Oscillator) configuration.

Note: Implementation Specific Container.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.80 Parameter McuFircUnderMcuControl

Set this to TRUE if FIRC is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.81 Parameter McuFircDivSel

Indicates this chip's FIRC clock division factor.

CONFIG_REG_GPR[FIRC_DIV_SEL] - This field provides the division value for the clock divider.

This field is not support for S32K39x derivative.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	Div_by_1
literals	['Div_by_1', 'Div_by_2', 'Div_by_16']

4.82 Parameter McuFIRC_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Warning: This frequency should be chosen based on the value of CONFIG_REG_GPR[FIRC_DIV_SEL]

set at boot time.

48MHz - CONFIG_REG_GPR[FIRC_DIV_SEL] = 3

24MHz - CONFIG_REG_GPR[FIRC_DIV_SEL] = 1 or CONFIG_REG_GPR[FIRC_DIV_SEL] = 0

3MHz - CONFIG_REG_GPR[FIRC_DIV_SEL] = 2

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.8E7
max	4.8E7
min	3000000.0

4.83 Parameter McuFircStandbyEnable

FIRC Standby Enable.

This bit specifies whether FIRC is enabled or disabled in STANDBY mode.

0 - FIRC is disabled in STANDBY.

1 - FIRC is enabled in STANDBY.

Configures the FIRC_STDBY_ENABLE[STDBY_EN] register field.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.84 Container McuSIRC

This container contains the specific configuration of the MCU SIRC (Slow Internal RC Oscillator) configuration.

Note: Implementation Specific Container.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.85 Parameter McuSircUnderMcuControl

Set this to TRUE if SIRC is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.86 Parameter McuSIRC_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	32000.0
max	32000.0
min	0.0

4.87 Parameter McuSircStandbyEnable

SIRC Standby Enable.

This bit specifies whether SIRC is enabled or disabled in STANDBY mode.

0 - SIRC is disabled in STANDBY.

1 - SIRC is enabled in STANDBY.

Configures the SIRC_MISCELLANEOUS_IN[STANDBY_ENABLE] register field.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.88 Container McuFXOSC

This container contains the specific configuration of the MCU FXOSC (Fast External Crystal Oscillator) configuration.

Note: Implementation Specific Container.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.89 Parameter McuFxoscUnderMcuControl

Set this to TRUE if FXOSC is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.90 Parameter McuFxoscPowerDownCtr

Crystal oscillator power-down control:

Checked - Crystal oscillator is switched ON.

Unchecked - Crystal oscillator is switched OFF.

Configure the FXOSC_CTRL[OSCON] field.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.91 Parameter McuFxoscBypass

Crystal Oscillator Bypass.

This bit specifies whether the oscillator should be bypassed or not.

0 - Internal oscillator not bypassed.

1 - Internal oscillator bypassed.

Configure the FXOSC_CTRL[OSC_BYP] field.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.92 Parameter McuFxoscMainComparator

Power down signal for main comparator.

This field should be 1 when external crystal is used, and 0 when FXOSC is in Single-Input Bypass Mode (i.e. FXOSC_CTRL[OSC_BYP] = 1).

0 - Comparator disabled.

1 - Comparator enabled.

Configure the FXOSC_CTRL[COMP_EN] field.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false

Property	Value
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.93 Parameter McuFxoscCounter

When the internal counter reaches this value, the oscillator is stable.

These bits specify the "end of count value" to be used for comparison by the

oscillator stabilization counter after reset or whenever it is switched on.

The counter is kept under reset if operating in Single-Input Bypass Mode (i.e. FXOSC_CTRL[OSC_BYP] = 1).

EOCV value is always 1ms in Differential Bypass mode.

Note: Please ensure that the internal counter is running for at least the stabilization

time of the crystal as given in the Data Sheet.

To calculate EOCV from startup time (of crystal), use the following formula:

EOCV (in decimal) = (Stabilization time in ns) / (4 * 128 * Time period of clock in ns)

Configure the FXOSC_CTRL[EOCV] field.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	49
max	255
min	0

4.94 Parameter McuFxoscOverdriveProtection

Crystal overdrive protection.

This value decides the trans-conductance applied by the FXOSC amplifier,

and it will depend on crystal specification.

FXOSC will not function when this field is 0 (0 trans-conductance).

In Differential Bypass Mode, this field must be set to 1.

Configure the FXOSC_CTRL[GM_SEL] field.

Note: FXOSC will not function when this field is set to 0 (0 trans-conductance).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	12
max	15
min	0

4.95 Parameter McuFXOSC_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false

Property	Value
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.0E7
max	4.0E7
min	0.0

4.96 Container McuSXOSC

This container contains the specific configuration of the MCU SXOSC (External Slow Oscillator) configuration.

This node is not supported on S32K311/S32K310/S32M27x/S32K39x derivatives.

Note: Implementation Specific Container.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.97 Parameter McuSxoscUnderMcuControl

Set this to TRUE if SXOSC is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points.

This node is not supported on S32K311/S32K310/S32M27x/S32K39x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.98 Parameter McuSxoscCounter

Configures the SXOSC_CTRL[EOCV] register field.

When the internal counter reaches this value, the oscillator is stable.

These bits specify the "end of count value" to be used for comparison by the

oscillator stabilization counter after reset or whenever it is switched on.

Note: Please ensure that the internal counter is running for at least the stabilization

time of the crystal as given in the Data Sheet.

To calculate EOCV from startup time (of crystal), use the following formula:

EOCV (in decimal) = (Stabilization time in ns) / (4 * 128 * Time period of clock in ns)

This node is not supported on S32K311/S32K310/S32M27x/S32K39x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	125
max	255
min	0

4.99 Parameter McuSxoscPowerDownCtr

Configure the SXOSC_CTRL[OSCON] register field.

Crystal oscillator power-down control:

Checked - Crystal oscillator is switched ON.

Unchecked - Crystal oscillator is switched OFF.

This node is not supported on S32K311/S32K310/S32M27x/S32K39x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.100 Parameter McuSXOSC_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

This node is not supported on S32K311/S32K310/S32M27x/S32K39x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0.0
max	32768.0
min	0.0

4.101 Container McuCgm0SettingConfig

This container contains the configuration for the CGM_0 settings of the MCU.

Included subcontainers:

- McuCgm0PcsConfig
- McuCgm0ClockMux0
- McuCgm0ClockMux1
- McuCgm0ClockMux2
- McuCgm0ClockMux3
- McuCgm0ClockMux4
- McuCgm0ClockMux5
- McuCgm0ClockMux6
- McuCgm0ClockMux7
- McuCgm0ClockMux8
- McuCgm0ClockMux9
- McuCgm0ClockMux10
- McuCgm0ClockMux11
- McuCgm0ClockMux12
- McuCgm0ClockMux13
- McuCgm0ClockMux14
- McuCgm0ClockMux15
- McuCgm0ClockMux16
- McuCgm0ClockMux17
- McuCgm0ClockMux18
- McuCgm0ClockMux19

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.102 Parameter McuPCSStepDuration

The value provided specifies the number of microseconds per step (i.e. the duration of a step, given in microseconds).

If more time is needed for the power supply to come to full load, this value should be increased.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1
max	100
min	1

4.103 Parameter McuPCSSwitchDuration

MC_CGM_PCFS_SDUR register configuration.

The value provided defines the duration of one PCS clock switch step in terms of FIRC cycles.

Note: This field must not be manually modified.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF

Property	Value
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	48
max	65535
min	0

4.104 Container McuCgm0PcsConfig

This register defines the rate of frequency change and initial change value for the

progressive system clock switching when switching the system clock source to or from

the PLL_PHI0_CLK on ramp-up and ramp-down, respectively.

Note: Implementation Specific Container.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE

4.105 Parameter McuClockPcfsUnderMcuControl

Set this to TRUE if this clock PCFS is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.106 Parameter McuPCS_Name

This is the name of the PCFS module.

PCFS_x corresponds to clock_src_x.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	PCFS_8
literals	['PCFS_8']

4.107 Parameter McuPCS_SourceFrequency

This is the frequency of the input clock source (i.e. the frequency of clk_src_x).

Note: This field must not be manually modified.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0.0
max	2.0E9
min	0.0

4.108 Parameter McuPCS_MaxAllowableDynamicIDD

This value defines the maximum allowable change in current (IDD) per microsecond.

It depends on the application and on the power supply (how much current can it deliver rapidly).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0.0
max	150.0
min	0.0

4.109 Container McuCgm0ClockMux0

This container enables and selects the configuration clocks for

CORE_CLK, AIPS_PLAT_CLK, AIPS_SLOW_CLK, HSE_CLK, DCM_CLK, LBIST_CLK and QSPI_MEM_CLK.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.110 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.111 Parameter McuClkMux0_Source

Clock Mux 0 Source Selection.

Sets the MC_CGM_MUX_0_CSC[SELCTL] field register.

MC_CGM_MUX_0_CSC[SELCTL] - This field selects the source clock for Clock Mux 0.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'PLL_PHI0_CLK']

4.112 Parameter McuClkMux0Div0_En

Clock Mux 0 Divider enable.

This field enables the Clock Divider for CORE_CLK.

Sets the MC_CGM_MUX_0_DC_0[DE] field register.

0 - Reserved.

1 - Divider is enabled.

Divider 0 is always enabled, the CORE_CLK clock is enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.113 Parameter McuClkMux0Div0_Divisor

Clock Mux 0 Division value.

Sets the MC_CGM_MUX_0_DC_0[DIV] field register.

MC_CGM_MUX_0_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux0Div0_En is true.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	7
min	0

4.114 Parameter McuClkMux0Div0Trigger

Trigger Divider type selection:

- Immediate divider update.

- Common trigger divider update.

Configure the MUX_0_DIV_TRIG_CTRL[TCTL] field register.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	COMMON_TRIGGER_DIVIDER_UPDATE
literals	['IMMEDIATE_DIVIDER_UPDATE', 'COMMON_TRIGGER_DIVIDER_↩ UPDATE']

4.115 Parameter McuClockMux0Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

System clock frequency selected must adhere to the same clock divider ratios shown in Clocking use case examples.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.8E7
max	1.2E8
min	0.0

4.116 Parameter McuClkMux0Div1_En

Clock Mux 0 Divider enable.

This field enables the Clock Divider for AIPS_PLAT_CLK.

Sets the MC_CGM_MUX_0_DC_1[DE] field register.

0 - Reserved

1 - Divider is enabled

Divider 1 is always enabled, the AIPS_PLAT_CLK clock is enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.117 Parameter McuClkMux0Div1_Divisor

Clock Mux 0 Division value.

Sets the MC_CGM_MUX_0_DC_1[DIV] field register.

MC_CGM_MUX_0_DC_1[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux0Div1_En is true.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF

Property	Value
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	7
min	0

4.118 Parameter McuClkMux0Div1Trigger

Trigger Divider type selection:

- Immediate divider update.

- Common trigger divider update.

Configure the MUX_0_DIV_TRIG_CTRL[TCTL] field register.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	COMMON_TRIGGER_DIVIDER_UPDATE
literals	['IMMEDIATE_DIVIDER_UPDATE', 'COMMON_TRIGGER_DIVIDER_↩ UPDATE']

4.119 Parameter McuClockMux0Divider1_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.8E7
max	8.0E7
min	0.0

4.120 Parameter McuClkMux0Div2_En

Clock Mux 0 Divider enable.

This field enables the Clock Divider for AIPS_SLOW_CLK.

Sets the MC_CGM_MUX_0_DC_2[DE] field register.

0 - Reserved

1 - Divider is enabled

Divider 2 is always enabled, the AIPS_SLOW_CLK clock is enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.121 Parameter McuClkMux0Div2_Divisor

Clock Mux 0 Division value.

Sets the MC_CGM_MUX_0_DC_2[DIV] field register.

MC_CGM_MUX_0_DC_2[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux0Div2_En is true.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1
max	15
min	0

4.122 Parameter McuClkMux0Div2Trigger

Trigger Divider type selection:

- Immediate divider update.

- Common trigger divider update.

Configure the MUX_0_DIV_TRIG_CTRL[TCTL] field register.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false

Property	Value
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	COMMON_TRIGGER_DIVIDER_UPDATE
literals	['IMMEDIATE_DIVIDER_UPDATE', 'COMMON_TRIGGER_DIVIDER_↩ UPDATE']

4.123 Parameter McuClockMux0Divider2_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	2.4E7
max	4.0E7
min	0.0

4.124 Parameter McuClkMux0Div3_En

Clock Mux 0 Divider enable.

This field enables the Clock Divider for HSE_CLK.

Sets the MC_CGM_MUX_0_DC_3[DE] field register.

0 - Reserved

1 - Divider is enabled

Divider 3 is always enabled, the HSE_CLK clock is enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.125 Parameter McuClkMux0Div3_Divisor

Clock Mux 0 Division value.

Sets the MC_CGM_MUX_0_DC_3[DIV] field register.

MC_CGM_MUX_0_DC_3[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux0Div3_En is true.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	7
min	0

4.126 Parameter McuClkMux0Div3Trigger

Trigger Divider type selection:

- Immediate divider update.

- Common trigger divider update.

Configure the MUX_0_DIV_TRIG_CTRL[TCTL] field register.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	COMMON_TRIGGER_DIVIDER_UPDATE
literals	['IMMEDIATE_DIVIDER_UPDATE', 'COMMON_TRIGGER_DIVIDER_↩ UPDATE']

4.127 Parameter McuClockMux0Divider3_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD

Property	Value
defaultValue	4.8E7
max	1.2E8
min	0.0

4.128 Parameter McuClkMux0Div4_En

Clock Mux 0 Divider enable.

This field enables the Clock Divider for DCM_CLK.

Sets the MC_CGM_MUX_0_DC_4[DE] field register.

0 - Reserved

1 - Divider is enabled

Divider 4 is always enabled, the DCM_CLK clock is enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.129 Parameter McuClkMux0Div4_Divisor

Clock Mux 0 Division value.

Sets the MC_CGM_MUX_0_DC_4[DIV] field register.

MC_CGM_MUX_0_DC_4[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux0Div4_En is true.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	7
min	0

4.130 Parameter McuClkMux0Div4Trigger

Trigger Divider type selection:

- Immediate divider update.

- Common trigger divider update.

Configure the MUX_0_DIV_TRIG_CTRL[TCTL] field register.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	COMMON_TRIGGER_DIVIDER_UPDATE
literals	['IMMEDIATE_DIVIDER_UPDATE', 'COMMON_TRIGGER_DIVIDER_↩ UPDATE']

4.131 Parameter McuClockMux0Divider4_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.8E7
max	4.8E7
min	0.0

4.132 Parameter McuClkMux0Div5_En

Clock Mux 0 Divider enable.

This field enables the Clock Divider for LBIST_CLK.

Sets the MC_CGM_MUX_0_DC_5[DE] field register.

0 - Reserved.

1 - Divider is enabled.

Divider 5 is always enabled, the LBIST_CLK clock is enabled.

This node is not supported on S32K312 and S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1

Property	Value
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.133 Parameter McuClkMux0Div5_Divisor

Clock Mux 0 Division value.

Sets the MC_CGM_MUX_0_DC_5[DIV] field register.

MC_CGM_MUX_0_DC_5[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux0Div5_En is true.

This node is not supported on S32K312 and S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	3
max	7
min	0

4.134 Parameter McuClkMux0Div5Trigger

Trigger Divider type selection:

- Immediate divider update.

- Common trigger divider update.

Configure the MUX_0_DIV_TRIG_CTRL[TCTL] field register.

This node is not supported on S32K312 and S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	COMMON_TRIGGER_DIVIDER_UPDATE
literals	['IMMEDIATE_DIVIDER_UPDATE', 'COMMON_TRIGGER_DIVIDER_↩ UPDATE']

4.135 Parameter McuClockMux0Divider5_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

This node is not supported on S32K312 and S32K311/S32K310/S32M27x derivatives.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.8E7
max	4.8E7
min	0.0

4.136 Parameter McuClkMux0Div6_En

Clock Mux 0 Divider enable.

This field enables the Clock Divider for QSPI_MEM_CLK.

Sets the MC_CGM_MUX_0_DC_6[DE] field register.

0 - Reserved

1 - Divider is enabled

Divider 6 is always enabled, the QSPI_MEM_CLK clock is enabled.

This node is not supported on S32K312 and S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.137 Parameter McuClkMux0Div6_Divisor

Clock Mux 0 Division value.

Sets the MC_CGM_MUX_0_DC_6[DIV] field register.

MC_CGM_MUX_0_DC_6[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux0Div6_En is true.

This node is not supported on S32K312 and S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	7
min	0

4.138 Parameter McuClkMux0Div6Trigger

Trigger Divider type selection:

- Immediate divider update.

- Common trigger divider update.

Configure the MUX_0_DIV_TRIG_CTRL[TCTL] field register.

This node is not supported on S32K312 and S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	COMMON_TRIGGER_DIVIDER_UPDATE
literals	['IMMEDIATE_DIVIDER_UPDATE', 'COMMON_TRIGGER_DIVIDER_↩ UPDATE']

4.139 Parameter McuClockMux0Divider6_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

This node is not supported on S32K312 and S32K311/S32K310/S32M27x derivatives.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.8E7
max	1.6E8
min	0.0

4.140 Parameter McuClkMux0Div7_En

Clock Mux 0 Divider enable.

This field enables the Clock Divider for CM7_CORE_CLK.

Sets the MC_CGM_MUX_0_DC_7[DE] field register.

0 - Reserved

1 - Divider is enabled

Divider 7 is always enabled, the CM7_CORE_CLK clock is enabled.

This node is supported on S32K39x and S32K388 derivative only.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.141 Parameter McuClkMux0Div7_Divisor

Clock Mux 0 Division value.

Sets the MC_CGM_MUX_0_DC_7[DIV] field register.

MC_CGM_MUX_0_DC_7[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux0Div7_En is true.

This node is supported on S32K39x and S32K388 derivative only.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	3
min	0

4.142 Parameter McuClkMux0Div7Trigger

Trigger Divider type selection:

- Immediate divider update.

- Common trigger divider update.

Configure the MUX_0_DIV_TRIG_CTRL[TCTL] field register.

This node is supported on S32K39x and S32K388 derivative only.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	COMMON_TRIGGER_DIVIDER_UPDATE
literals	['IMMEDIATE_DIVIDER_UPDATE', 'COMMON_TRIGGER_DIVIDER_↩ UPDATE']

4.143 Parameter McuClockMux0Divider7_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

This node is supported on S32K39x and S32K388 derivative only.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A

Property	Value
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.8E7
max	3.2E8
min	0.0

4.144 Container McuCgm0ClockMux1

This container enables and selects the configuration clocks

for STM0_CLK.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.145 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP

Property	Value
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.146 Parameter McuClkMux1_Source

Clock Mux 1 Source Selection.

Sets the MC_CGM_MUX_1_CSC[SELCTL] field register.

MC_CGM_MUX_1_CSC[SELCTL] - This field selects the source clock for Clock Mux 1.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'FXOSC_CLK', 'AIPS_PLAT_CLK']

4.147 Parameter McuClkMux1Div0_En

Clock Mux 1 Divider enable.

This field enables the Clock Divider for STM0_CLK.

Sets the MC_CGM_MUX_1_DC_0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

If it is set to 0 (Divider is disabled), any write access to the MC_CGM_MUX_1_DC_0[DIV] field is ignored and the STM0_CLK clock remains disabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.148 Parameter McuClkMux1Div0_Divisor

Clock Mux 1 Division value.

Sets the MC_CGM_MUX_1_DC_0[DIV] field register.

MC_CGM_MUX_1_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux1Div0_En is true.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true

Property	Value
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	1
min	0

4.149 Parameter McuClockMux1Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0.0
max	8.0E7
min	0.0

4.150 Container McuCgm0ClockMux2

This container enables and selects the configuration clocks

for STM1_CLK.

Note: This node is not supported on S32K314/S32K312/S32K311/S32K310/S32M27x derivatives.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.151 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

This node is not supported on S32K314/S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.152 Parameter McuClkMux2_Source

Clock Mux 2 Source Selection.

Sets the MC_CGM_MUX_2_CSC[SELCTL] field register.

MC_CGM_MUX_2_CSC[SELCTL] - This field selects the source clock for Clock Mux 2.

This node is not supported on S32K314/S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'FXOSC_CLK', 'AIPS_PLAT_CLK']

4.153 Parameter McuClkMux2Div0_En

Clock Mux 2 Divider enable.

This field enables the Clock Divider for STM1_CLK.

Sets the MC_CGM_MUX_2_DC_0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

If it is set to 0 (Divider is disabled), any write access to the MC_CGM_MUX_2_DC_0[DIV] field is ignored and the STM1_CLK clock remains disabled.

This node is not supported on S32K314/S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.154 Parameter McuClkMux2Div0_Divisor

Clock Mux 2 Division value.

Sets the MC_CGM_MUX_2_DC_0[DIV] field register.

MC_CGM_MUX_2_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux2Div0_En is true.

This node is not supported on S32K314/S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	1
min	0

4.155 Parameter McuClockMux2Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

This node is not supported on S32K314/S32K312/S32K311/S32K310/S32M27x derivatives.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0.0
max	8.0E7
min	0.0

4.156 Container McuCgm0ClockMux3

This container enables and selects the configuration clocks

for FLEXCAN_PE_CLK0, FLEXCAN_PE_CLK1 and FLEXCAN_PE_CLK2.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.157 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.158 Parameter McuClkMux3_Source

Clock Mux 3 Source Selection.

Sets the MC_CGM_MUX_3_CSC[SELCTL] field register.

MC_CGM_MUX_3_CSC[SELCTL] - This field selects the source clock for Clock Mux 3.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'FXOSC_CLK', 'AIPS_PLAT_CLK']

4.159 Parameter McuClkMux3Div0_En

Clock Mux 3 Divider enable.

This field enables the Clock Divider for FLEXCAN_PE_CLK0_2.

Sets the MC_CGM_MUX_3_DC_0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

If it is set to 0 (Divider is disabled), any write access to the MC_CGM_MUX_3_DC_0[DIV] field is ignored and the FLEXCAN_PE_CLK0_2 clock remains disabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.160 Parameter McuClkMux3Div0_Divisor

Clock Mux 3 Division value.

Sets the MC_CGM_MUX_3_DC_0[DIV] field register.

MC_CGM_MUX_3_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux3Div0_En is true.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	3
min	0

4.161 Parameter McuClockMux3Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0.0
max	8.0E7
min	0.0

4.162 Container McuCgm0ClockMux4

This container enables and selects the configuration clocks

for FLEXCAN_PE_CLK3_5/FLEXCAN_PE_CLK3_7.

This node is not supported on S32K311/S32K310/S32M27x derivative.

This node supports the following clocks :

FLEXCAN_PE_CLK3_7 : S32K358

FLEXCAN_PE_CLK3_5 : S32K3XX

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.163 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

This node is not supported on S32K311/S32K310/S32M27x derivative.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.164 Parameter McuClkMux4_Source

Clock Mux 4 Source Selection.

Sets the MC_CGM_MUX_4_CSC[SELCTL] field register.

MC_CGM_MUX_4_CSC[SELCTL] - This field selects the source clock for Clock Mux 4.

This node is not supported on S32K311/S32K310/S32M27x derivative.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'FXOSC_CLK', 'AIPS_PLAT_CLK']

4.165 Parameter McuClkMux4Div0_En

Clock Mux 4 Divider enable.

Sets the MC_CGM_MUX_4_DC_0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

If it is set to 0 (Divider is disabled), any write access to the MC_CGM_MUX_4_DC_0[DIV] field is ignored and the FLEXCAN_PE_CLK3_5/FLEXCAN_PE_CLK3_7 clock remains disabled.

This node enables the Clock Divider for FLEXCAN_PE_CLK3_5/FLEXCAN_PE_CLK3_7.

This node is not supported on S32K311/S32K310/S32M27x derivative.

This node supports the following clocks :

FLEXCAN_PE_CLK3_7 : S32K358

FLEXCAN_PE_CLK3_5 : S32K3XX

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.166 Parameter McuClkMux4Div0_Divisor

Clock Mux 4 Division value.

Sets the MC_CGM_MUX_4_DC_0[DIV] field register.

MC_CGM_MUX_4_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux4Div0_En is true.

This node is not supported on S32K311/S32K310/S32M27x derivative.

This node supports the following clocks :

FLEXCAN_PE_CLK3_7 : S32K358

FLEXCAN_PE_CLK3_5 : S32K3XX

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	3

4.167 Parameter McuClockMux4Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

This node is not supported on S32K311/S32K310/S32M27x derivative.

This node supports the following clocks :

FLEXCAN_PE_CLK3_7 : S32K358

FLEXCAN_PE_CLK3_5 : S32K3XX

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	2.4E7
max	8.0E7
min	0.0

4.168 Container McuCgm0ClockMux5

This container enables and selects the configuration clocks

for CLKOUT_STANDBY.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.169 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.170 Parameter McuClkMux5_Source

Clock Mux 5 Source Selection.

Sets the MC_CGM_MUX_5_CSC[SELCTL] field register.

MC_CGM_MUX_5_CSC[SELCTL] - This field selects the source clock for Clock Mux 5.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'SIRC_CLK', 'FXOSC_CLK', 'AIPS_SLOW_CLK']

4.171 Parameter McuClkMux5Div0_En

Clock Mux 5 Divider enable.

This field enables the Clock Divider for CLKOUT_STANDBY.

Sets the MC_CGM_MUX_5_DC_0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

If it is set to 0 (Divider is disabled), any write access to the MC_CGM_MUX_5_DC_0[DIV] field is ignored and the CLKOUT_STANDBY clock remains disabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.172 Parameter McuClkMux5Div0_Divisor

Clock Mux 5 Division value.

Sets the MC_CGM_MUX_5_DC_0[DIV] field register.

MC_CGM_MUX_5_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux5Div0_En is true.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1
max	7
min	0

4.173 Parameter McuClockMux5Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	2.4E7
max	4.8E7
min	0.0

4.174 Container McuCgm0ClockMux6

This container enables and selects the configuration clocks

for CLKOUT_RUN.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.175 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.176 Parameter McuClkMux6_Source

Clock Mux 6 Source Selection.

Sets the MC_CGM_MUX_6_CSC[SELCTL] field register.

MC_CGM_MUX_6_CSC[SELCTL] - This field selects the source clock for Clock Mux 6.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'SIRC_CLK', 'FXOSC_CLK', 'PLL_PHI0_CLK', 'PLL_PHI1_CLK', 'CORE_CLK', 'HSE_CLK', 'AIPS_PLAT_CLK', 'AIPS_SLOW_CLK']

4.177 Parameter McuClkMux6Div0_En

Clock Mux 6 Divider enable.

This field enables the Clock Divider for CLKOUT_RUN.

Sets the MC_CGM_MUX_6_DC_0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

If it is set to 0 (Divider is disabled), any write access to the MC_CGM_MUX_6_DC_0[DIV] field is ignored and the CLKOUT_RUN clock remains disabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.178 Parameter McuClkMux6Div0_Divisor

Clock Mux 6 Division value.

Sets the MC_CGM_MUX_6_DC_0[DIV] field register.

MC_CGM_MUX_6_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux6Div0_En is true.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1
max	63
min	0

4.179 Parameter McuClockMux6Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.8E7
max	3.2E8
min	0.0

4.180 Container McuCgm0ClockMux7

This container enables and selects the configuration clocks

for EMAC_CLK_RX / GMAC0_CLK_RX.

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivative.

This node supports the following clocks :

GMAC0_CLK_RX : S32K388

EMAC_CLK_RX : S32K3XX

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.181 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.182 Parameter McuClkMux7_Source

Clock Mux 7 Source Selection.

Sets the MC_CGM_MUX_7_CSC[SELCTL] field register.

MC_CGM_MUX_7_CSC[SELCTL] - This field selects the source clock for Clock Mux 7.

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'EMAC_MII_RMII_TX_CLK', 'EMAC_MII_RX_CLK']

4.183 Parameter McuClkMux7Div0_En

Clock Mux 7 Divider enable.

This field enables the Clock Divider for EMAC_CLK_RX / GMAC0_CLK_RX.

Sets the MC_CGM_MUX_7_DC_0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

If it is set to 0 (Divider is disabled), any write access to the MC_CGM_MUX_7_DC_0[DIV] field is ignored and the EMAC_CLK_RX / GMAC0_CLK_RX clock remains disabled.

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

This node supports the following clocks :

GMAC0_CLK_RX : S32K388

EMAC_CLK_RX : S32K3XX

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.184 Parameter McuClkMux7Div0_Divisor

Clock Mux 7 Division value.

Sets the MC_CGM_MUX_7_DC_0[DIV] field register.

MC_CGM_MUX_7_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux7Div0_En is true.

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

This node supports the following clocks :

GMAC0_CLK_RX : S32K388

EMAC_CLK_RX : S32K3XX

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	63
min	0

4.185 Parameter McuClockMux7Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

This node supports the following clocks :

GMAC0_CLK_RX : S32K388

EMAC_CLK_RX : S32K3XX

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A

Property	Value
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.8E7
max	5.0E7
min	0.0

4.186 Container McuCgm0ClockMux8

This container enables and selects the configuration clocks

for EMAC_CLK_TX /GMAC0_TX_CLK.

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivative.

This node supports the following clocks :

GMAC0_TX_CLK : S32K388

EMAC_CLK_TX : S32K3XX

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.187 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.188 Parameter McuClkMux8_Source

Clock Mux 8 Source Selection.

Sets the MC_CGM_MUX_8_CSC[SELCTL] field register.

MC_CGM_MUX_8_CSC[SELCTL] - This field selects the source clock for Clock Mux 8.

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'EMAC_MII_RMII_TX_CLK']

4.189 Parameter McuClkMux8Div0_En

Clock Mux 8 Divider enable.

This field enables the Clock Divider for EMAC_CLK_TX / GMAC0_TX_CLK.

Sets the MC_CGM_MUX_8_DC_0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

If it is set to 0 (Divider is disabled), any write access to the MC_CGM_MUX_7_DC_0[DIV] field is ignored and the EMAC_CLK_RX clock remains disabled.

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

This node supports the following clocks :

GMAC0_TX_CLK : S32K388

EMAC_CLK_TX : S32K3XX

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.190 Parameter McuClkMux8Div0_Divisor

Clock Mux 8 Division value.

Sets the MC_CGM_MUX_8_DC_0[DIV] field register.

MC_CGM_MUX_8_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux8Div0_En is true.

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

This node supports the following clocks :

GMAC0_TX_CLK : S32K388

EMAC_CLK_TX : S32K3XX

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	63
min	0

4.191 Parameter McuClockMux8Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

This node supports the following clocks :

GMAC0_TX_CLK : S32K388

EMAC_CLK_TX : S32K3XX

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP

Property	Value
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.8E7
max	5.0E7
min	0.0

4.192 Container McuCgm0ClockMux9

This container enables and selects the configuration clocks

for EMAC_CLK_TS / GMAC_CLK_TS.

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivative.

This node supports the following clocks :

GMAC_CLK_TS : S32K388

EMAC_CLK_TS : S32K3XX

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.193 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.194 Parameter McuClkMux9_Source

Clock Mux 9 Source Selection.

Sets the MC_CGM_MUX_9_CSC[SELCTL] field register.

MC_CGM_MUX_9_CSC[SELCTL] - This field selects the source clock for Clock Mux 9.

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'FXOSC_CLK', 'PLL_PHI0_CLK', 'EMAC_MII_RMII_TX↩ _CLK', 'EMAC_MII_RX_CLK']

4.195 Parameter McuClkMux9Div0_En

Clock Mux 9 Divider enable.

This field enables the Clock Divider for EMAC_CLK_TS / GMAC_CLK_TS.

Sets the MC_CGM_0_MUX_9_DC_0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

If it is set to 0 (Divider 0 is disabled), any write access to the MC_CGM_0_MUX_9_DC_0[DIV] field is ignored and the EMAC_CLK_TS / GMAC_CLK_TS clock remains disabled.

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

This node supports the following clocks :

GMAC_CLK_TS : S32K388

EMAC_CLK_TS : S32K3XX

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.196 Parameter McuClkMux9Div0_Divisor

Clock Mux 9 Division value.

Sets the MC_CGM_MUX_9_DC_0[DIV] field register.

MC_CGM_MUX_9_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux9Div0_En is true.

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

This node supports the following clocks :

GMAC_CLK_TS : S32K388

EMAC_CLK_TS : S32K3XX

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	63
min	0

4.197 Parameter McuClockMux9Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

This node supports the following clocks :

GMAC_CLK_TS : S32K388

EMAC_CLK_TS : S32K3XX

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.8E7
max	1.2E8
min	0.0

4.198 Container McuCgm0ClockMux10

This container enables and selects the configuration clocks

for QuadSPI_SFCK / QSPI_2XSFIF(QSPI_SFCK is always QSPI_2XSFIF div 2).

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivative.

This node supports the following clocks :

QSPI_2XSFIF : S32K39x and S32K358(and its subderivatives)

QuadSPI_SFCK : S32K3XX

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.199 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.200 Parameter McuClkMux10_Source

Clock Mux 10 Source Selection.

Sets the MC_CGM_MUX_10_CSC[SELCTL] field register.

MC_CGM_MUX_10_CSC[SELCTL] - This field selects the source clock for Clock Mux 10.

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'FXOSC_CLK', 'PLL_PHI1_CLK']

4.201 Parameter McuClkMux10Div0_En

Clock Mux 10 Divider enable.

This field enables the Clock Divider for QuadSPI_SFCK/QSPI_2XSFIF.

Sets the MC_CGM_MUX_10_DC_0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

If it is set to 0 (Divider is disabled), any write access to the MC_CGM_MUX_10_DC_0[DIV] field is ignored and the QuadSPI_SFCK/QSPI_2XSFIF clock remains disabled.

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

This node supports the following clocks :

QSPI_2XSFIF : S32K39x and S32K358(and its subderivatives)

QuadSPI_SFCK : S32K3XX

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.202 Parameter McuClkMux10Div0_Divisor

Clock Mux 10 Division value.

Sets the MC_CGM_MUX_10_DC_0[DIV] field register.

MC_CGM_MUX_10_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux10Div0_En is true.

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

This node supports the following clocks :

QSPI_2XSFIF : S32K39x and S32K358(and its subderivatives)

QuadSPI_SFCK : S32K3XX

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	7
min	0

4.203 Parameter McuClockMux10Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

This node is not supported on S32K312/S32K311/S32K310/S32M27x derivatives.

This node supports the following clocks :

QSPI_2XSFIF : S32K39x and S32K358(and its subderivatives)

QuadSPI_SFCK : S32K3XX

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.8E7
max	1.2E8
min	0.0

4.204 Container McuCgm0ClockMux11

This container enables and selects the configuration clocks

for TRACE_CLK.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.205 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.206 Parameter McuClkMux11_Source

Clock Mux 11 Source Selection.

Sets the MC_CGM_MUX_11_CSC[SELCTL] field register.

MC_CGM_MUX_11_CSC[SELCTL] - This field selects the source clock for Clock Mux 11.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'FXOSC_CLK', 'PLL_PHI0_CLK', 'PLL_PHI1_CLK']

4.207 Parameter McuClkMux11Div0_En

Clock Mux 11 Divider enable.

This field enables the Clock Divider for TRACE_CLK.

Sets the MC_CGM_MUX_11_DC_0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

If it is set to 0 (Divider is disabled), any write access to the MC_CGM_MUX_11_DC_0[DIV] field is ignored and the TRACE_CLK clock remains disabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.208 Parameter McuClkMux11Div0_Divisor

Clock Mux 11 Division value.

Sets the MC_CGM_MUX_11_DC_0[DIV] field register.

MC_CGM_MUX_11_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux11Div0_En is true.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	7
min	0

4.209 Parameter McuClockMux11Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.8E7
max	1.2E8
min	0.0

4.210 Container McuCgm0ClockMux12

This container enables and selects the configuration clocks

for EMAC_TX_RMII_CLK/GMAC0_TX_RMII_CLK.

This node is supported on S32K39x and S32K3x8 derivative only.

This node supports the following clocks :

GMAC0_TX_RMII_CLK : S32K388

EMAC_TX_RMII_CLK : S32K3xx

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.211 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.212 Parameter McuClkMux12_Source

Clock Mux 12 Source Selection.

Sets the MC_CGM_MUX_12_CSC[SELCTL] field register.

MC_CGM_MUX_12_CSC[SELCTL] - This field selects the source clock for Clock Mux 12.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'FXOSC_CLK', 'PLL_PHI0_CLK']

4.213 Parameter McuClkMux12Div0_En

Clock Mux 12 Divider enable.

This field enables the Clock Divider for EMAC_TX_RMII_CLK/GMAC0_TX_RMII_CLK.

Sets the MC_CGM_MUX_12_DC_0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

If it is set to 0 (Divider is disabled), any write access to the MC_CGM_MUX_12_DC_0[DIV] field is ignored and the EMAC_TX_RMII_CLK/GMAC0_TX_RMII_CLK clock remains disabled.

This node is supported on S32K39x and S32K3x8 derivative only.

This node supports the following clocks :

GMAC0_TX_RMII_CLK : S32K388

EMAC_TX_RMII_CLK : S32K3xx

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.214 Parameter McuClkMux12Div0_Divisor

Clock Mux 12 Division value.

Sets the MC_CGM_MUX_12_DC_0[DIV] field register.

MC_CGM_MUX_12_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux12Div0_En is true.

This node is supported on S32K39x and S32K3x8 derivative only.

This node supports the following clocks :

GMAC0_TX_RMII_CLK : S32K388

EMAC_TX_RMII_CLK : S32K3xx

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	63

4.215 Parameter McuClockMux12Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

This node is supported on S32K39x and S32K3x8 derivative only.

This node supports the following clocks :

GMAC0_TX_RMII_CLK : S32K388

EMAC_TX_RMII_CLK : S32K3xx

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.8E7
max	5.0E7
min	0.0

4.216 Container McuCgm0ClockMux13

This container enables and selects the configuration clocks

for STM2_CLK.

This node is supported on S32K39x and S32K358 derivative only.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A

4.217 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.218 Parameter McuClkMux13_Source

Clock Mux 13 Source Selection.

Sets the MC_CGM_MUX_13_CSC[SELCTL] field register.

MC_CGM_MUX_13_CSC[SELCTL] - This field selects the source clock for Clock Mux 13.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'FXOSC_CLK', 'AIPS_PLAT_CLK']

4.219 Parameter McuClkMux13Div0_En

Clock Mux 13 Divider enable.

This field enables the Clock Divider for STM2_CLK.

Sets the MC_CGM_MUX_13_DC_0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

If it is set to 0 (Divider is disabled), any write access to the MC_CGM_MUX_13_DC_0[DIV] field is ignored and the STM2_CLK clock remains disabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.220 Parameter McuClkMux13Div0_Divisor

Clock Mux 13 Division value.

Sets the MC_CGM_MUX_13_DC_0[DIV] field register.

MC_CGM_MUX_13_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux13Div0_En is true.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	1
min	0

4.221 Parameter McuClockMux13Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.8E7
max	8.0E7
min	0.0

4.222 Container McuCgm0ClockMux14

This container enables and selects the configuration clocks

for uSDHC on S32K358 derivative.

Note: uSDHC is only available on: S32K358.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.223 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

This node is only available on: S32K358 derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.224 Parameter McuClkMux14_Source

Clock Mux 14 Source Selection.

Sets the MC_CGM_MUX_14_CSC[SELCTL] field register.

MC_CGM_MUX_14_CSC[SELCTL] - This field selects the source clock for Clock Mux 14.

This node is only available on: S32K358 derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'FXOSC_CLK', 'PLL_PHI1_CLK', 'PLLAUX_PHI2_CLK']

4.225 Parameter McuClkMux14Div0_En

Clock Mux 14 Divider enable.

This field enables the Clock Divider for uSDHC.

Sets the MC_CGM_MUX_14_DC_0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

If it is set to 0 (Divider is disabled), any write access to the MC_CGM_MUX_14_DC_0[DIV] field is ignored and the uSDHC clock remains disabled.

This node is only available on: S32K358 derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.226 Parameter McuClkMux14Div0_Divisor

Clock Mux 14 Division value.

Sets the MC_CGM_MUX_14_DC_0[DIV] field register.

MC_CGM_MUX_14_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux14Div0_En is true.

This node is only available on: S32K358 derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	1
min	0

4.227 Parameter McuClockMux14Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

This node is only available on: S32K358 derivatives.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.8E7
max	5.0E7
min	0.0

4.228 Container McuCgm0ClockMux15

This container enables and selects the configuration clocks

for LFAST_REF_CLK/GMAC1_RX_CLK.

This node is supported on S32K39x or S32K388 derivative only.

This node supports the following clocks :

GMAC1_RX_CLK : S32K388

LFAST_REF_CLK : S32K39x

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A

4.229 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.230 Parameter McuClkMux15_Source

Clock Mux 15 Source Selection.

Sets the MC_CGM_MUX_15_CSC[SELCTL] field register.

MC_CGM_MUX_15_CSC[SELCTL] - This field selects the source clock for Clock Mux 15.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'FXOSC_CLK', 'LFAST_REF_EXT_CLK', 'AIPS_SLOW_↩ CLK']

4.231 Parameter McuClkMux15Div0_En

Clock Mux 15 Divider enable.

This field enables the Clock Divider for LFAST_REF_CLK/GMAC1_RX_CLK.

Sets the MC_CGM_MUX_15_DC_0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

If it is set to 0 (Divider is disabled), any write access to the MC_CGM_MUX_15_DC_0[DIV] field is ignored and the clock remains disabled.

This node is supported on S32K39x or S32K388 derivative only.

This node supports the following clocks :

GMAC1_RX_CLK : S32K388

LFAST_REF_CLK : S32K39x

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.232 Parameter McuClkMux15Div0_Divisor

Clock Mux 15 Division value.

Sets the MC_CGM_MUX_15_DC_0[DIV] field register.

MC_CGM_MUX_15_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux15Div0_En is true.

This node is supported on S32K39x or S32K388 derivative only.

This node supports the following clocks :

GMAC1_RX_CLK : S32K388

LFAST_REF_CLK : S32K39x

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	2
max	63
min	0

4.233 Parameter McuClockMux15Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

This node is supported on S32K39x or S32K388 derivative only.

This node supports the following clocks :

GMAC1_RX_CLK : S32K388

LFAST_REF_CLK : S32K39x

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP

Property	Value
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1.6E7
max	2.6E7
min	0.0

4.234 Container McuCgm0ClockMux16

This container enables and selects the configuration clocks

for SWG_CLK/GMAC1_TX_CLK.

This node is supported on S32K39x or S32K388 derivative only.

This node supports the following clocks :

GMAC1_TX_CLK : S32K388

SWG_CLK : S32K39x

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.235 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.236 Parameter McuClkMux16_Source

Clock Mux 16 Source Selection.

Sets the MC_CGM_MUX_16_CSC[SELCTL] field register.

MC_CGM_MUX_16_CSC[SELCTL] - This field selects the source clock for Clock Mux 16.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'FXOSC_CLK', 'SWG_PAD_CLK']

4.237 Parameter McuClkMux16Div0_En

Clock Mux 16 Divider enable.

This field enables the Clock Divider for SWG_CLK/GMAC1_TX_CLK.

Sets the MC_CGM_MUX_16_DC_0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

If it is set to 0 (Divider is disabled), any write access to the MC_CGM_MUX_16_DC_0[DIV] field is ignored and the SWG_CLK / GMAC1_TX_CLK clock remains disabled.

This node is supported on S32K39x or S32K388 derivative only.

This node supports the following clocks :

GMAC1_TX_CLK : S32K388

SWG_CLK : S32K39x

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.238 Parameter McuClkMux16Div0_Divisor

Clock Mux 16 Division value.

Sets the MC_CGM_MUX_16_DC_0[DIV] field register.

MC_CGM_MUX_16_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux16Div0_En is true.

This node is supported on S32K39x or S32K388 derivative only.

This node supports the following clocks :

GMAC1_TX_CLK : S32K388

SWG_CLK : S32K39x

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	2
max	63
min	0

4.239 Parameter McuClockMux16Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

This node is supported on S32K39x or S32K388 derivative only.

This node supports the following clocks :

GMAC1_TX_CLK : S32K388

SWG_CLK : S32K39x

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP

Property	Value
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1.6E7
max	2.0E7
min	0.0

4.240 Container McuCgm0ClockMux17

This container enables and selects the configuration clocks

for GMAC1_RMII_CLK.

This node is supported on S32K388 derivative only.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.241 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.242 Parameter McuClkMux17_Source

Clock Mux 17 Source Selection.

Sets the MC_CGM_MUX_17_CSC[SELCTL] field register.

MC_CGM_MUX_17_CSC[SELCTL] - This field selects the source clock for Clock Mux 17.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'FXOSC_CLK', 'PLL_PHI0_CLK', 'PLLAUX_PHI0_CLK']

4.243 Parameter McuClkMux17Div0_En

Clock Mux 17 Divider enable.

This field enables the Clock Divider for GMAC1_RMII_CLK.

Sets the MC_CGM_MUX_17_DC_0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

If it is set to 0 (Divider is disabled), any write access to the MC_CGM_MUX_17_DC_0[DIV] field is ignored and the GMAC1_RMII_CLK clock remains disabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.244 Parameter McuClkMux17Div0_Divisor

Clock Mux 17 Division value.

Sets the MC_CGM_MUX_17_DC_0[DIV] field register.

MC_CGM_MUX_17_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux17Div0_En is true.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	2
max	63
min	0

4.245 Parameter McuClockMux17Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1.6E7
max	1.2E8
min	0.0

4.246 Container McuCgm0ClockMux18

This container enables and selects the configuration clocks

for STMD_CLK.

This node is supported on S32K388 derivative only.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.247 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.248 Parameter McuClkMux18_Source

Clock Mux 18 Source Selection.

Sets the MC_CGM_MUX_18_CSC[SELCTL] field register.

MC_CGM_MUX_18_CSC[SELCTL] - This field selects the source clock for Clock Mux 18.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF

Property	Value
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'FXOSC_CLK', 'AIPS_PLAT_CLK']

4.249 Parameter McuClkMux18Div0_En

Clock Mux 18 Divider enable.

This field enables the Clock Divider for STMD_CLK.

Sets the MC_CGM_MUX_18_DC_0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

If it is set to 0 (Divider is disabled), any write access to the MC_CGM_MUX_18_DC_0[DIV] field is ignored and the SWG_CLK clock remains disabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.250 Parameter McuClkMux18Div0_Divisor

Clock Mux 18 Division value.

Sets the MC_CGM_MUX_18_DC_0[DIV] field register.

MC_CGM_MUX_18_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux18Div0_En is true.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1
max	7
min	0

4.251 Parameter McuClockMux18Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1.6E7
max	1.2E8
min	0.0

4.252 Container McuCgm0ClockMux19

This container enables and selects the configuration clocks

for AES_CLK.

This node is supported on S32K388 derivative only.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.253 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP

Property	Value
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.254 Parameter McuClkMux19_Source

Clock Mux 19 Source Selection.

Sets the MC_CGM_MUX_19_CSC[SELCTL] field register.

MC_CGM_MUX_19_CSC[SELCTL] - This field selects the source clock for Clock Mux 19.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'PLLAUX_PHI1_CLK']

4.255 Parameter McuClkMux19Div0_En

Clock Mux 19 Divider enable.

This field enables the Clock Divider for AES_CLK.

Sets the MC_CGM_MUX_19_DC_0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

If it is set to 0 (Divider is disabled), any write access to the MC_CGM_MUX_19_DC_0[DIV] field is ignored and the SWG_CLK clock remains disabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.256 Parameter McuClkMux19Div0_Divisor

Clock Mux 19 Division value.

Sets the MC_CGM_MUX_19_DC_0[DIV] field register.

MC_CGM_MUX_19_DC_0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider.

This parameter is enabled only if McuClkMux19Div0_En is true.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true

Property	Value
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	2
max	63
min	0

4.257 Parameter McuClockMux19Divider0_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1.6E7
max	1.2E8
min	0.0

4.258 Container McuRtcClockSelect

This container selects the configuration clocks for RTC_CLK.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.259 Parameter McuClockMuxUnderMcuControl

Set this to TRUE if this clock mux is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.260 Parameter McuRtc_Source

RTC_CLK Source Selection.

Sets the RTCC[CLKSEL] field register.

Sets the RTCC[CLKSEL] - This field selects the source clock for RTC_CLK.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FIRC_CLK
literals	['FIRC_CLK', 'FXOSC_CLK', 'SIRC_CLK']

4.261 Parameter McuRtc_Frequency

This is the frequency for the specific instance of the "McuClockReferencePoint" container.

Value calculated for user info. It is given in Hz.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0.0
max	4.8E7
min	0.0

4.262 Container McuPll_0

This container provides the specific configuration for the PLL.

Note: Implementation Specific Container.

Included subcontainers:

- McuPll_Configuration
- McuPll_Parameter

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.263 Parameter McuPLLUnderMcuControl

Set this to TRUE if this PLL is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.264 Parameter McuPLLEnabled

0 - PLL is disabled.

1 - PLL is enabled (and can be used as a clock source).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.265 Container McuPll_Configuration

Configuration values for PLL.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.266 Parameter McuPllDvRdiv

Input clock predivider.

Sets the PLL: PLLDIG_PLLDV[RDIV] field register.

This field controls the value of the divider on the input clock. The output of the predivider circuit generates the reference clock to the PLL analog loop.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	2
max	7
min	1

4.267 Parameter McuPllDvMfi

Loop multiplication factor divider.

Sets the PLL: PLLDIG_PLLDV[MFI] field register.

This field controls the value of the divider in the feedback loop. The value specified by the MFI bits

establishes the multiplication factor applied to the reference frequency. Divider value = MFI, where MFI

should be choosen such that it does not violate VCO frequency specifications.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	120
max	255
min	1

4.268 Parameter McuPllDvOdiv2

ODIV2 Division value.

PLLDIG_PLLDV[ODIV2] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV' times the time period of the current input clock to the divider.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	2
max	63
min	1

4.269 Parameter McuPllFmSscgbyp

Modulation enable.

This bit enables spectrum modulation.

Set the PLL: PLLDIG_PLLFM[SSCGBYP] field.

0 - Spread spectrum modulation is not bypassed.

1 - Spread spectrum modulation is bypassed.

Note: PLLFM[SSCGBYP] must be cleared and PLLFD[SDMEN] must be set for the frequency modulation mechanism to be enabled.

Note: Frequency Modulation is only possible if PLLDIG_PLLFM[STEPSIZE] * PLLDIG_PLLFM[STEPNO] less than 18432.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.270 Parameter McuPllFmSpreadctl

Modulation type selection:

- Center around nominal frequency.

- Spread below nominal frequency.

Configure the PLLDIG_PLLFM[SPREADCTL] field register.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	Center_Spread
literals	['Down_Spread', 'Center_Spread']

4.271 Parameter McuPllFmStepSize

Modulation period.

Sets the PLL: PLLDIG_PLLFM[STEPSIZE] field register.

STEPSIZE is the binary equivalent of the modulation period variable.

It is calculated as: StepSize = [McuPllFdMdp * (McuPllDvMfi + McuPllFdMfn / 18432) * 18432] / (100 * McuPllFmStepNo).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1

Property	Value
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	1023
min	0

4.272 Parameter McuPllFmStepNo

Increment step.

Sets the PLL: PLLDIG_PLLFM[STEPNO] field register.

This field is the binary equivalent of the STEPNO variable.

It is calculated as: StepNo = McuClockReferencePointFrequency(McuPllClockSelection) / (2 * McuPllFdFmod * McuPllDvRdiv).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1
max	2047
min	1

4.273 Parameter McuPllFdFmod

The modulation frequency. This should be set to the highest frequency component present in the modulating signal.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0.0
max	32000.0
min	0.0

4.274 Parameter McuPllFdMdp

The modulation depth percentage. This value indicates by how much the modulated variable varies around its unmodulated level.

It is calculated as the FrequencyDeviation (deviation from the carrier/nominal frequency) divided by the ModulatingSignalFrequency(Fmod).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0.0
max	3.0
min	0.0

4.275 Parameter McuPllFdMfn

Numerator for fractional loop division factor - PLLDIG_PLLFD[MFN].

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	32767
min	0

4.276 Parameter McuPllFdSdmen

Sigma Delta Modulation Enable.

Set the PLL: PLLDIG_PLLFD[SDMEN] field register.

0 - Sigma delta modulation disabled.

1 - Sigma delta modulation enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.277 Parameter McuPllFdSdm2

Second Order Sigma Delta Modulation Select.

Set the PLL: PLLDIG_PLLFD[SDM2] field register.

0 - 2nd order sigma delta modulation disabled.

1 - 2nd order sigma delta modulation enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.278 Parameter McuPllFdSdm3

Third Order Sigma Delta Modulation Select.

Set the PLL: PLLDIG_PLLFD[SDM3] field register.

0 - 3nd order sigma delta modulation disabled.

1 - 3nd order sigma delta modulation enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.279 Parameter McuPllOdiv0_En

PHI0 Divider enable.

Set the PLL: PLLDIG_PLLODIV0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.280 Parameter McuPllOdiv0_Div

PHI0 Division value.

PLLDIG_PLLODIV0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	2
max	15
min	0

4.281 Parameter McuPllOdiv1_En

PHI1 Divider enable.

Set the PLL: PLLDIG_PLLODIV1[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.282 Parameter McuPllOdiv1_Div

PHI1 Division value.

PLLDIG_PLLODIV1[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	2
max	15
min	0

4.283 Container McuPll_Parameter

Calculated values for PLL.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.284 Parameter PLL_PHI0_Frequency

Output value for PLL_PHI0_CLK frequency.

The valid range is [48 ... 320] MHz for S32K3XX/S32M27x.

The valid range is [48 ... 640] MHz for S32K39x.

Note: This field must not be manually modified.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0.0
max	3.2E8
min	0.0

4.285 Parameter PLL_PHI1_Frequency

Output value for CORE_PLL_PHI1_CLK frequency.

The valid range is [48 ... 320] MHz for S32K3XX/S32M27x.

The valid range is [48 ... 640] MHz for S32K39x.

Note: This field must not be manually modified.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1

Property	Value
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0.0
max	3.2E8
min	0.0

4.286 Parameter PLL_VCO_Frequency

Output value for VCO frequency.

The valid range is [640 ... 1280] MHz.

Note: This field must not be manually modified.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0.0
max	1.28E9
min	0.0

4.287 Container McuPll_1

This container provides the specific configuration for the PLL.

Note: Implementation Specific Container.

Included subcontainers:

- McuPll_Configuration
- McuPll_Parameter

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.288 Parameter McuPLLUnderMcuControl

Set this to TRUE if this PLL is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.289 Parameter McuPLLEnabled

0 - PLL is disabled.

1 - PLL is enabled (and can be used as a clock source).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.290 Container McuPll_Configuration

Configuration values for PLL.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.291 Parameter McuPllDvRdiv

Input clock predivider.

Sets the PLL: PLLDIG_PLLDV[RDIV] field register.

This field controls the value of the divider on the input clock. The output of the predivider circuit generates the reference clock to the PLL analog loop.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	2
max	7
min	1

4.292 Parameter McuPllDvMfi

Loop multiplication factor divider.

Sets the PLL: PLLDIG_PLLDV[MFI] field register.

This field controls the value of the divider in the feedback loop. The value specified by the MFI bits establishes the multiplication factor applied to the reference frequency. Divider value = MFI, where MFI should be choosen such that it does not violate VCO frequency specifications.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	120
max	255
min	1

4.293 Parameter McuPllDvOdiv2

ODIV2 Division value.

PLLDIG_PLLDV[ODIV2] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV' times the time period of the current input clock to the divider.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1
max	63
min	1

4.294 Parameter McuPllOdiv0_En

PHI0 Divider enable.

Set the PLL: PLLDIG_PLLODIV0[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A

Property	Value
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.295 Parameter McuPllOdiv0_Div

PHI0 Division value.

PLLDIG_PLLODIV0[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	3
max	31
min	0

4.296 Parameter McuPllOdiv1_En

PHI1 Divider enable.

Set the PLL: PLLDIG_PLLODIV1[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.297 Parameter McuPllOdiv1_Div

PHI1 Division value.

PLLDIG_PLLODIV1[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	3
max	31
min	0

4.298 Parameter McuPllOdiv2_En

PHI2 Divider enable.

Set the PLL: PLLDIG_PLLODIV2[DE] field register.

0 - Divider is disabled

1 - Divider is enabled

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.299 Parameter McuPllOdiv2_Div

PHI2 Division value.

PLLDIG_PLLODIV2[DIV] - This field provides the division value for the clock divider. The clock period of the clock after division

would be 'DIV+1' times the time period of the current input clock to the divider

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	3
max	31
min	0

4.300 Container McuPll_Parameter

Calculated values for PLL.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.301 Parameter PLL_PHI0_Frequency

Output value for PLLAUX_PHI0_CLK frequency.

The valid range is [48 ... 320] MHz for S32K3XX.

The valid range is [48 ... 640] MHz for S32K39x.

Note: This field must not be manually modified.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0.0
max	3.2E8
min	0.0

4.302 Parameter PLL_PHI1_Frequency

Output value for CORE_PLL_PHI1_CLK frequency.

The valid range is [48 ... 320] MHz for S32K3XX.

The valid range is [48 ... 640] MHz for S32K39x.

Note: This field must not be manually modified.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0.0
max	3.2E8
min	0.0

4.303 Parameter PLL_PHI2_Frequency

Output value for CORE_PLL_PHI2_CLK frequency.

The valid range is [48 ... 320] MHz for S32K3XX.

The valid range is [48 ... 640] MHz for S32K39x.

Note: This field must not be manually modified.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1

Property	Value
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0.0
max	3.2E8
min	0.0

4.304 Parameter PLL_VCO_Frequency

Output value for VCO frequency.

The valid range is [640 ... 1280] MHz.

Note: This field must not be manually modified.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0.0
max	1.28E9
min	0.0

4.305 Container McuClkMonitor

This container contains the specific configuration (parameters) of the Clock Monitor Unit.

Each CMU is independently programmed. FIRC and FXOSC are used as the clock monitor references.

Detailed information on the CMUs can be found in the Clock Monitor Unit chapter.

This parameter is enabled only if "McuClockSrcFailureNotification" is enabled.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	4
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE

4.306 Parameter McuClockMonitorUnderMcuControl

Set this to TRUE if this clock monitor is under mcu control

If it is FALSE then the mcu driver will not write the corresponding registers.

The user must still set the values - they are used by the clock reference points

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.307 Parameter McuClkMonitorEn

Enables/Disables the clock monitor (CMU_FC_GCR[FCE]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.308 Parameter McuCmuName

This is the name of the CMU.

With name convention: CMU_FC_[Number Of CMU Unit]_[Name of Monitored clock].

Note: The CMU_FC_5 is monitoring HSE_CLK, it accessible only by HSE_B Cortex-M0+ core.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	CMU_FC_0_FXOSC_CLK
literals	['CMU_FC_0_FXOSC_CLK', 'CMU_FC_3_CORE_CLK', 'CMU_FC_4_↩ AIPS_PLAT_CLK', 'CMU_FC_5_HSE_CLK']

4.309 Parameter McuFHHAsyncEventEn

This field is used to enable/disable FHH asynchronous event at the module boundary. (CMU_FC_IER[FHHAEE]).

0 - Asynchronous FHH Event is Disabled

1 - Asynchronous FHH Event is Enabled

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.310 Parameter McuFLLAsyncEventEn

This field is used to enable/disable FLL asynchronous event at the module boundary. (CMU_FC_IER[FLLAEE]).

0 - Asynchronous FLL Event is Disabled

1 - Asynchronous FLL Event is Enabled

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.311 Parameter McuFHHInterruptEn

This field is used to enable/disable FHH interrupt at the module boundary. (CMU_FC_IER[FHHIE]).

0 - FHH Interrupt is Disabled

1 - FHH Interrupt is Enabled

This field is only supported for CMU_FC_0.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.312 Parameter McuFLLInterruptEn

This field is used to enable/disable FLL interrupt at the module boundary. (CMU_FC_IER[FLLIE]).

0 - FLL Interrupt is Disabled

1 - FLL Interrupt is Enabled

This field is only supported for CMU_FC_0.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.313 Container McuClockReferencePoint

This container defines a reference point in the Mcu Clock tree. It defines the frequency which then can be used by other modules as an input value. Lower multiplictiy is 1, as even in the simpliest case (only one frequency is used), there is one frequency to be defined.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.314 Parameter McuClockReferencePointFrequency

This is the frequency for the specific instance of the McuClockReferencePoint container.

It shall be given in Hz.

Calculated value.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true

Property	Value
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.8E7
max	5.0E9
min	0.0

4.315 Parameter McuClockFrequencySelect

Select clock source for the specific instance of the McuClockReferencePoint container.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	CORE_CLK
literals	['CUSTOM', 'CORE_CLK', 'AIPS_PLAT_CLK', 'AIPS_SLOW_CLK', 'HSE_CLK', 'DCM_CLK', 'STM0_CLK', 'FLEXCAN_PE_CLK0_2', 'CLKOUT_STANDBY', 'CLKOUT_RUN', 'TRACE_CLK', 'RTC_CLK', 'FIRC_CLK', 'SIRC_CLK', 'FXOSC_CLK', 'SCS_CLK']

4.316 Container McuDemEventParameterRefs

Container for the references to DemEventParameter elements which shall be invoked using the API Dem_SetEventStatus API in case the corresponding error occurs.

The EventId is taken from the referenced DemEventParameter's DemEventId value.

The standardized errors are provided in the container and can be extended by vendor specific error references.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.317 Reference MCU_E_TIMEOUT_FAILURE

Reference to configured DEM event to report Timeout failure.

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	true
destination	/AUTOSAR/EcucDefs/Dem/DemConfigSet/DemEventParameter

4.318 Reference MCU_E_INVALIDFXOSC_CONFIG

Reference to configured DEM event to report a FXOSC invalid configuration event.

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE

Property	Value
requiresSymbolicNameValue	true
destination	/AUTOSAR/EcucDefs/Dem/DemConfigSet/DemEventParameter

4.319 Reference MCU_E_CLOCKMUXSWITCH_FAILURE

Reference to configured DEM event to report a failed clock switch request.

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	true
destination	/AUTOSAR/EcucDefs/Dem/DemConfigSet/DemEventParameter

4.320 Reference MCU_E_CLOCK_FAILURE

Reference to configured DEM event to report Clock source failure.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	true
destination	/AUTOSAR/EcucDefs/Dem/DemConfigSet/DemEventParameter

4.321 Container McuModeSettingConf

This container contains the configuration for the Mode setting of the MCU.

Note: Implementation Specific Parameter.

Included subcontainers:

- McuPartitionConfiguration
- McuDcmGprConfiguration

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.322 Parameter McuMode

This parameter shall represent the ID of the MCU mode.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	true
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	0
max	255
min	0

4.323 Parameter McuPowerMode

This parameter selects the Power Mode to be used.

For valid Mode transitions refers to "MC_ME Mode Diagram" from Reference Manual.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	RUN
literals	['RUN', 'CORE_STANDBY', 'SOC_PREPARE_STANDBY', 'SOC_↩ STANDBY', 'STANDBY', 'SOC_PREPARE_FAST_STANDBY', 'SOC↩ _FAST_STANDBY', 'FAST_STANDBY', 'FUNC_RESET', 'DEST_RESET']

4.324 Parameter McuMainCoreSelect

This field is used to select which core will be designated as the Main Core.

The driver will configure the MC_ME_MAIN_COREID[PIDX] and MC_ME_MAIN_COREID[CIDX]

register fields based on this parameter.

This field is modifiable only when McuPowerMode = 'SOC_STANDBY'.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	CM7_0
literals	['CM7_0']

4.325 Parameter McuCoreLockStepEnable

Configures the MC_ME_PRTN0_CORE_LOCKSTEP[LS2] register field.

This bit provides the lockstep indication to Core 4 & Core 5 in partition 0.

0 - Lockstep Disabled.

1 - Lockstep Enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.326 Parameter McuEnableSleepOnExit

Indicates sleep-on-exit when returning from Handler mode to Thread mode:

0 - Do not sleep when returning to Thread mode.

1 - Enter sleep, or deep sleep, on return from an ISR.

Setting this bit to 1 enables an interrupt driven application to avoid returning to an empty main application.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.327 Container McuPartitionConfiguration

This section generates control signals based on the logic partitions

implemented inside it. The logic partition refers to SoC blocks controlled by single

partition of MC_ME. Each of the MC_ME partition implements or control a set of logic

functionality using the MC_ME partition processes hardware.

Note: Implementation Specific Parameter.

Included subcontainers:

- McuPartition0Config
- McuPartition1Config
- McuPartition2Config
- McuPartition3Config
- McuPeripheral

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.328 Container McuPartition0Config

This container contains the configuration for Partition 0.

Included subcontainers:

- McuCore0Configuration

- McuCore1Configuration
- McuCore4Configuration
- McuCore3Configuration

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.329 Parameter McuPartitionUnderMcuControl

Set this to TRUE if Partition 0 is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN0 or PRST0).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.330 Parameter McuPartitionPowerUnderMcuControl

Set this to TRUE if Partition 0 Power Management is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN0_PCONF).

This means that the setting configured by node "McuPartitionClockEnable" will be left untouched

(i.e. the configuration will not have any effect in hardware).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.331 Parameter McuPrtnCofb0UnderMcuControl

Set this to TRUE if PRTN0_COFB0 is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN0_COFB0).

This means that all the peripherals under the 'McuPeripheral' list prefixed by 'PRTN0_COFB0'

will be left untouched (i.e. the configuration will not have any effect in hardware).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.332 Parameter McuPrtnCofb1UnderMcuControl

Set this to TRUE if PRTN0_COFB1 is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN0_COFB1).

This means that all the peripherals under the 'McuPeripheral' list prefixed by 'PRTN0_COFB1'

will be left untouched (i.e. the configuration will not have any effect in hardware).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.333 Parameter McuPartitionClockEnable

Configures the MC_ME_PRTN0_PCONF[PCE] register field.

This bit controls whether the clock to IPs (other than core(s)) in the partition are enabled or disabled.

0b - Disable the clock to IPs.

1b - Enable the clock to IPs.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.334 Container McuCore0Configuration

This container contains the configuration for the CM7_0 core within Partition 0.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.335 Parameter McuCoreUnderMcuControl

Set this to TRUE if CM7_0 is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN0_CORE0 and PRST0_0).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.336 Parameter McuCoreClockEnable

Configures the MC_ME_PRTN0_CORE0_PCONF[CCE] register field.

This bit controls whether the clock to CM7_0 is enabled or disabled.

0 - CM7_0 Core Clock is Disabled.

1 - CM7_0 Core Clock is Enabled.

Warning: When disabling the core clock, the driver

will busy-wait until the corresponding MC_ME_PRTN0_CORE0_STAT[WFI]

flag is set. This means that the core to be shutdown must always

execute an WFI as its last instruction.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.337 Parameter McuCoreBootAddress

Configures the MC_ME_PRTN0_CORE0_ADDR[ADDR] register field.

This register controls the boot address of the CM7_0 core.

The value from this field will be masked with 0xFFFFFFFC (i.e.

the boot address must be aligned on a 4-byte boundary).

This field is modifiable only when the corresponding CM7_0 core

clock is enabled and McuCoreControlConfiguration/McuCoreBootAddressControl

is 'true'.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	4294967292
min	0

4.338 Parameter McuCoreBootAddressLinkerSym

Configures the MC_ME_PRTN0_CORE0_ADDR[ADDR] register field.

This register controls the boot address of the CM7_0 core.

The value from this field will be masked with 0xFFFFFFFC (i.e.

the boot address must be aligned on a 4-byte boundary).

This field is modifiable only when the corresponding CM7_0 core

clock is enabled and McuCoreControlConfiguration/McuCoreBootAddressControl

is 'true'.

Note: If this parameter is empty, then the boot address

(integer value) defined by 'McuCoreBootAddress' will be used.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-STRING-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	

4.339 Container McuCore1Configuration

This container contains the configuration for the CM7_1 core within Partition 0.

This node is not supported on S32K344/S32K314/S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.340 Parameter McuCoreUnderMcuControl

Set this to TRUE if CM7_1 is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN0_CORE1 and PRST0_0).

This node is not supported on S32K344/S32K314/S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.341 Parameter McuCoreClockEnable

Configures the MC_ME_PRTN0_CORE1_PCONF[CCE] register field.

This bit controls whether the clock to CM7_1 is enabled or disabled.

0 - CM7_1 Core Clock is Disabled.

1 - CM7_1 Core Clock is Enabled.

Warning: When disabling the core clock, the driver

will busy-wait until the corresponding MC_ME_PRTN0_CORE1_STAT[WFI]

flag is set. This means that the core to be shutdown must always

execute an WFI as its last instruction.

This node is not supported on S32K344/S32K314/S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.342 Parameter McuCoreBootAddress

Configures the MC_ME_PRTN0_CORE1_ADDR[ADDR] register field.

This register controls the boot address of the CM7_1 core.

The value from this field will be masked with 0xFFFFFFFC (i.e.

the boot address must be aligned on a 4-byte boundary).

This field is modifiable only when the corresponding CM7_1 core

clock is enabled and McuCoreControlConfiguration/McuCoreBootAddressControl

is 'true'.

This node is not supported on S32K344/S32K314/S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	4294967292
min	0

4.343 Parameter McuCoreBootAddressLinkerSym

Configures the MC_ME_PRTN0_CORE1_ADDR[ADDR] register field.

This register controls the boot address of the CM7_1 core.

The value from this field will be masked with 0xFFFFFFFC (i.e.

the boot address must be aligned on a 4-byte boundary).

This field is modifiable only when the corresponding CM7_1 core

clock is enabled and McuCoreControlConfiguration/McuCoreBootAddressControl

is 'true'.

Note: If this parameter is empty, then the boot address

(integer value) defined by 'McuCoreBootAddress' will be used.

This node is not supported on S32K344/S32K314/S32K312/S32K311/S32K310/S32M27x derivatives.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-STRING-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	

4.344 Container McuCore4Configuration

This container contains the configuration for the CM7_2 core within Partition 0.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.345 Parameter McuCoreUnderMcuControl

Set this to TRUE if CM7_2 is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN0_CORE4).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF

Property	Value
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.346 Parameter McuCoreClockEnable

Configures the MC_ME_PRTN0_CORE4_PCONF[CCE] register field.

This bit controls whether the clock to CM7_2 is enabled or disabled.

0 - CM7_2 Core Clock is Disabled.

1 - CM7_2 Core Clock is Enabled.

Warning: When disabling the core clock, the driver

will busy-wait until the corresponding MC_ME_PRTN0_CORE4_STAT[WFI]

flag is set. This means that the core to be shutdown must always

execute an WFI as its last instruction.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.347 Parameter McuCoreBootAddress

Configures the MC_ME_PRTN0_CORE4_ADDR[ADDR] register field.

This register controls the boot address of the CM7_2 core.

The value from this field will be masked with 0xFFFFFFFC (i.e.

the boot address must be aligned on a 4-byte boundary).

This field is modifiable only when the corresponding CM7_2 core

clock is enabled and McuCoreControlConfiguration/McuCoreBootAddressControl

is 'true'.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	4294967292
min	0

4.348 Parameter McuCoreBootAddressLinkerSym

Configures the MC_ME_PRTN0_CORE4_ADDR[ADDR] register field.

This register controls the boot address of the CM7_2 core.

The value from this field will be masked with 0xFFFFFFFC (i.e.

the boot address must be aligned on a 4-byte boundary).

This field is modifiable only when the corresponding CM7_2 core

clock is enabled and McuCoreControlConfiguration/McuCoreBootAddressControl

is 'true'.

Note: If this parameter is empty, then the boot address

(integer value) defined by 'McuCoreBootAddress' will be used.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-STRING-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	

4.349 Container McuCore3Configuration

This container contains the configuration for the CM7_3 core within Partition 0.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.350 Parameter McuCoreUnderMcuControl

Set this to TRUE if CM7_3 is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN0_CORE3).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.351 Parameter McuCoreClockEnable

Configures the MC_ME_PRTN0_CORE3_PCONF[CCE] register field.

This bit controls whether the clock to CM7_3 is enabled or disabled.

0 - CM7_3 Core Clock is Disabled.

1 - CM7_3 Core Clock is Enabled.

Warning: When disabling the core clock, the driver

will busy-wait until the corresponding MC_ME_PRTN0_CORE3_STAT[WFI]

flag is set. This means that the core to be shutdown must always

execute an WFI as its last instruction.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.352 Parameter McuCoreBootAddress

Configures the MC_ME_PRTN0_CORE3_ADDR[ADDR] register field.

This register controls the boot address of the CM7_3 core.

The value from this field will be masked with 0xFFFFFFFC (i.e.

the boot address must be aligned on a 4-byte boundary).

This field is modifiable only when the corresponding CM7_3 core

clock is enabled and McuCoreControlConfiguration/McuCoreBootAddressControl

is 'true'.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	4294967292
min	0

4.353 Parameter McuCoreBootAddressLinkerSym

Configures the MC_ME_PRTN0_CORE3_ADDR[ADDR] register field.

This register controls the boot address of the CM7_3 core.

The value from this field will be masked with 0xFFFFFFFC (i.e.

the boot address must be aligned on a 4-byte boundary).

This field is modifiable only when the corresponding CM7_3 core

clock is enabled and McuCoreControlConfiguration/McuCoreBootAddressControl

is 'true'.

Note: If this parameter is empty, then the boot address

(integer value) defined by 'McuCoreBootAddress' will be used.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-STRING-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	

4.354 Container McuPartition1Config

This container contains the configuration for Partition 1.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.355 Parameter McuPartitionUnderMcuControl

Set this to TRUE if Partition 1 is under MCU control.
If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN1 or PRST1).
Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.356 Parameter McuPartitionPowerUnderMcuControl

Set this to TRUE if Partition 1 Power Management is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN1_PCONF or PRST1_0[PERIPH_64_RST] or RDC1).

This means that the settings configured by nodes "McuPartitionClockEnable" and "McuPartitionResetEnable" will be left untouched (i.e. the configuration will not have any effect in hardware).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.357 Parameter McuPrtnCofb0UnderMcuControl

Set this to TRUE if PRTN1_COFB0 is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN1_COFB0).

This means that all the peripherals under the 'McuPeripheral' list prefixed by 'PRTN1_COFB0'

will be left untouched (i.e. the configuration will not have any effect in hardware).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.358 Parameter McuPrtnCofb1UnderMcuControl

Set this to TRUE if PRTN1_COFB1 is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN1_COFB1).

This means that all the peripherals under the 'McuPeripheral' list prefixed by 'PRTN1_COFB1'

will be left untouched (i.e. the configuration will not have any effect in hardware).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.359 Parameter McuPrtnCofb2UnderMcuControl

Set this to TRUE if PRTN1_COFB2 is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN1_COFB2).

This means that all the peripherals under the 'McuPeripheral' list prefixed by 'PRTN1_COFB2'

will be left untouched (i.e. the configuration will not have any effect in hardware).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.360 Parameter McuPrtnCofb3UnderMcuControl

Set this to TRUE if PRTN1_COFB3 is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN1_COFB3).

This means that all the peripherals under the 'McuPeripheral' list prefixed by 'PRTN1_COFB3'

will be left untouched (i.e. the configuration will not have any effect in hardware).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.361 Parameter McuPartitionClockEnable

Configures the MC_ME_PRTN1_PCONF[PCE] register field.

This bit controls whether the clock to IPs (other than core(s)) in the partition are enabled or disabled.

0b - Disable the clock to IPs.

1b - Enable the clock to IPs.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.362 Container McuPartition2Config

This container contains the configuration for Partition 2.

This node is not supported on S32K312 derivative.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.363 Parameter McuPartitionUnderMcuControl

Set this to TRUE if Partition 2 is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN2 or PRST2).

This node is not supported on S32K312 derivative.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.364 Parameter McuPartitionPowerUnderMcuControl

Set this to TRUE if Partition 2 Power Management is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN2_PCONF or

PRST2_0[PERIPH_128_RST] or RDC2).

This means that the settings configured by nodes "McuPartitionClockEnable" and "McuPartitionResetEnable"

will be left untouched (i.e. the configuration will not have any effect in hardware).

This node is not supported on S32K312 derivative.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.365 Parameter McuPrtnCofb0UnderMcuControl

Set this to TRUE if PRTN2_COFB0 is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN2_COFB0).

This means that all the peripherals under the 'McuPeripheral' list prefixed by 'PRTN2_COFB0'

will be left untouched (i.e. the configuration will not have any effect in hardware).

This node is not supported on S32K312 derivative.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.366 Parameter McuPrtnCofb1UnderMcuControl

Set this to TRUE if PRTN2_COFB1 is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN2_COFB1).

This means that all the peripherals under the 'McuPeripheral' list prefixed by 'PRTN2_COFB1'

will be left untouched (i.e. the configuration will not have any effect in hardware).

This node is not supported on S32K312 derivative.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.367 Parameter McuPrtnCofb2UnderMcuControl

Set this to TRUE if PRTN2_COFB1 is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN2_COFB2).

This means that all the peripherals under the 'McuPeripheral' list prefixed by 'PRTN2_COFB2'

will be left untouched (i.e. the configuration will not have any effect in hardware).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1

Property	Value
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.368 Parameter McuPartitionClockEnable

Configures the MC_ME_PRTN2_PCONF[PCE] register field.

This bit controls whether the clock to IPs (other than core(s)) in the partition are enabled or disabled.

0b - Disable the clock to IPs.

1b - Enable the clock to IPs.

This node is not supported on S32K312 derivative.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.369 Container McuPartition3Config

This container contains the configuration for Partition 3.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.370 Parameter McuPartitionUnderMcuControl

Set this to TRUE if Partition 3 is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN3).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.371 Parameter McuPartitionPowerUnderMcuControl

Set this to TRUE if Partition 3 Power Management is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN3_PCONF).

This means that the settings configured by nodes "McuPartitionClockEnable"

will be left untouched (i.e. the configuration will not have any effect in hardware).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF

Property	Value
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.372 Parameter McuPrtnCofb0UnderMcuControl

Set this to TRUE if PRTN3_COFB0 is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN3_COFB0).

This means that all the peripherals under the 'McuPeripheral' list prefixed by 'PRTN3_COFB0'

will be left untouched (i.e. the configuration will not have any effect in hardware).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.373 Parameter McuPrtnCofb1UnderMcuControl

Set this to TRUE if PRTN3_COFB1 is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN3_COFB1).

This means that all the peripherals under the 'McuPeripheral' list prefixed by 'PRTN3_COFB1' will be left untouched (i.e. the configuration will not have any effect in hardware).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.374 Parameter McuPrtnCofb2UnderMcuControl

Set this to TRUE if PRTN3_COFB1 is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (PRTN3_COFB2).

This means that all the peripherals under the 'McuPeripheral' list prefixed by 'PRTN3_COFB2' will be left untouched (i.e. the configuration will not have any effect in hardware).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.375 Parameter McuPartitionClockEnable

Configures the MC_ME_PRTN3_PCONF[PCE] register field.

This bit controls whether the clock to IPs (other than core(s)) in the partition are enabled or disabled.

0b - Disable the clock to IPs.

1b - Enable the clock to IPs.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.376 Container McuPeripheral

This contains the power state configuration for the current peripheral.

Note: Implementation Specific Container.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	56
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE

4.377 Parameter McuPeripheralName

This is the name of the peripheral.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	TRGMUX
literals	['TRGMUX', 'BCTU', 'EMIOS_0', 'EMIOS_1', 'LCU_0', 'LCU_1', 'ADC_↩ 0', 'ADC_1', 'PIT_0', 'PIT_1', 'EDMA', 'EDMA_TCD_0', 'EDMA_TCD↩ _1', 'EDMA_TCD_2', 'EDMA_TCD_3', 'EDMA_TCD_4', 'EDMA_TCD↩ _5', 'EDMA_TCD_6', 'EDMA_TCD_7', 'EDMA_TCD_8', 'EDMA_TCD↩ _9', 'EDMA_TCD_10', 'EDMA_TCD_11', 'SDA_AP', 'EIM_0', 'ERM_↩ 0', 'MSCM', 'SWT_0', 'STM_0', 'INTM', 'DMAMUX_0', 'DMAMUX_1', 'RTC', 'SIUL2_VIRTWRAPPER_PDAC3', 'WKPU', 'CMUs', 'TSPC', 'FX-OSC', 'PLL', 'FlexCAN_0', 'FlexCAN_1', 'FlexCAN_2', 'FlexIO', 'LPUART_↩ 0', 'LPUART_1', 'LPUART_2', 'LPUART_3', 'LPI2C_1', 'LPSPI_0', 'LPSPI↩ _1', 'LPSPI_2', 'LPSPI_3', 'LPCMP_0', 'TEMPSENSE', 'CRC', 'STCU_2']

4.378 Parameter McuModeEntrySlot

This is the MC_ME slot corresponding to the peripheral.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false

Property	Value
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	PRTN0_COFB1_REQ32
literals	['PRTN0_COFB1_REQ32', 'PRTN0_COFB1_REQ33', 'PRTN0_COFB1_REQ34', 'PRTN0_COFB1_REQ35', 'PRTN0_COFB1_REQ38', 'PRTN0_COFB1_REQ39', 'PRTN0_COFB1_REQ40', 'PRTN0_COFB1_REQ41', 'PRTN0_COFB1_REQ44', 'PRTN0_COFB1_REQ45', 'PRTN1_COFB0_REQ3', 'PRTN1_COFB0_REQ4', 'PRTN1_COFB0_REQ5', 'PRTN1_COFB0_REQ6', 'PRTN1_COFB0_REQ7', 'PRTN1_COFB0_REQ8', 'PRTN1_COFB0_REQ9', 'PRTN1_COFB0_REQ10', 'PRTN1_COFB0_REQ11', 'PRTN1_COFB0_REQ12', 'PRTN1_COFB0_REQ13', 'PRTN1_COFB0_REQ14', 'PRTN1_COFB0_REQ15', 'PRTN1_COFB0_REQ21', 'PRTN1_COFB0_REQ22', 'PRTN1_COFB0_REQ23', 'PRTN1_COFB0_REQ24', 'PRTN1_COFB0_REQ28', 'PRTN1_COFB0_REQ29', 'PRTN1_COFB0_REQ31', 'PRTN1_COFB1_REQ32', 'PRTN1_COFB1_REQ33', 'PRTN1_COFB1_REQ34', 'PRTN1_COFB1_REQ42', 'PRTN1_COFB1_REQ45', 'PRTN1_COFB1_REQ47', 'PRTN1_COFB1_REQ49', 'PRTN1_COFB1_REQ53', 'PRTN1_COFB1_REQ56', 'PRTN1_COFB2_REQ65', 'PRTN1_COFB2_REQ66', 'PRTN1_COFB2_REQ67', 'PRTN1_COFB2_REQ73', 'PRTN1_COFB2_REQ74', 'PRTN1_COFB2_REQ75', 'PRTN1_COFB2_REQ76', 'PRTN1_COFB2_REQ77', 'PRTN1_COFB2_REQ85', 'PRTN1_COFB2_REQ86', 'PRTN1_COFB2_REQ87', 'PRTN1_COFB2_REQ88', 'PRTN1_COFB2_REQ89', 'PRTN1_COFB2_REQ92', 'PRTN1_COFB2_REQ95', 'PRTN1_COFB3_REQ96', 'PRTN1_COFB3_REQ104', 'NONE']

4.379 Parameter McuPeripheralClockEnable

Configures the MC_ME_PRTNx_COFBx_CLKEN[REQx] register field.

0 - Peripheral Clock is Gated.

1 - Peripheral Clock is Running.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.380 Container McuDcmGprConfiguration

This DCM_GPR Control Register contains the various control settings of the DCM_GPR block.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.381 Parameter McuDcmGprUnderMcuControl

Set this to TRUE if DCM_GPR is under MCU control.

If it is FALSE, then the MCU driver will not write the corresponding registers (DCMRWF2).

(i.e. the configuration will not have any effect in hardware).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.382 Parameter McuBootBaseAddress

Cortex-M7_0 base address of vector table to be used after exiting Standby mode (only to be considered in Fast Standby mode.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	4294967295
min	0

4.383 Parameter McuSIRC_TRIM_BYP_STDBY_EXTControl

DCM_GPR_DCMRWF2[SIRC_TRIM_BYP_STDBY_EXT] field configuration.

Control to bypass the SIRC trimming on standby exit.

Unchecked - Not bypassed.

Checked - Bypassed.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.384 Parameter McuPMC_TRIM_RGM_DCF_BYP_STDBY_EXTControl

DCM_GPR_DCMRWF2[PMC_TRIM_RGM_DCF_BYP_STDBY_EXT] field configuration.

Control to bypass the PMC trimming and RGM DCF loading on standby exit.

Unchecked - Not bypassed.

Checked - Bypassed.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.385 Parameter McuFIRC_TRIM_BYP_STDBY_EXTControl

DCM_GPR_DCMRWF2[FIRC_TRIM_BYP_STDBY_EXT] field configuration.

Control to bypass the FIRC trimming on standby exit.

Unchecked - Not bypassed.

Checked - Bypassed.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A

Property	Value
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.386 Parameter McuDCM_SCAN_BYP_STDBY_EXTControl

DCM_GPR_DCMRWF2[DCM_SCAN_BYP_STDBY_EXT] field configuration.

Control to bypass the DCM scanning on standby exit.

Unchecked - Not bypassed.

Checked - Bypassed.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.387 Parameter McuGlobalPadkeepingEnable

If this parameter has been configured to 'TRUE', the effect is that the DCMRWF1[STANDBY_IO_CONFIG] will be written to 0 before entering Standby mode.

If this parameter has been configured to 'FALSE', the effect is that the DCMRWF1[STANDBY_IO_CONFIG] will be written to 1

before entering Standby mode, the isolation removal hardware removes pad keeping on Standby mode exit. This is useful in case of low-power debug.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.388 Container McuRamSectorSettingConf

This container contains the configuration for the RAM Sector setting.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.389 Parameter McuRamSectorId

This parameter shall represent the ID of the MCU RAM Sector configuration.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	true
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A

Property	Value
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	0
max	4294967295
min	0

4.390 Parameter McuRamDefaultValue

This parameter shall represent the Data pre-setting to be initialized.

Default value is 0.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	255
min	0

4.391 Parameter McuRamSectionBaseAddress

This parameter represents the RAM section base address.

The address must be aligned to 4 bytes.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	541065216
max	541097984
min	541065216

4.392 Parameter McuRamSectionSize

This parameter represents the RAM section size in bytes.

The size must be a multiple of 4.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	32768
max	32768
min	0

4.393 Parameter McuRamSectionWriteSize

This parameter shall define the size in bytes of data which can be written into RAM at once.

The ram write size is currently restricted to {1, 2, 4, 8} bytes.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC

Property	Value
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	8
max	4294967295
min	0

4.394 Parameter McuRamSectionBaseAddrLinkerSym

This parameter represents the RAM section base address.

The address must be aligned to 4 bytes.

If this parameter is empty, then the integer values from "McuRamSectionBaseAddress" will be used.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-STRING-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	

4.395 Parameter McuRamSectionSizeLinkerSym

This parameter represents the RAM section size in bytes.

The size must be multiple of 4.

If this parameter is empty, then the integer values from "McuRamSectionSize" will be used.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-STRING-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	

4.396 Container McuResetConfig

The reset generation module (MC_RGM) centralizes the different reset sources and manages the reset sequence of the device.

Note: Implementation Specific Parameter.

Included subcontainers:

- McuResetSourcesConfig
- McuResetGeneratorConfiguration

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.397 Parameter McuResetType

This parameter selects the type of the reset to be performed through the McuPerformReset API.

A 'destructive' reset source is associated with an event related to a critical - usually

hardware - error or dysfunction. When a 'destructive' reset event occurs, the full reset

sequence is applied to the chip. This resets the full chip ensuring a safe start-up state for

both digital and analog modules, and the memory content must be considered to be unknown.

A 'functional' reset source is associated with an event related to a less-critical - usually non-hardware - error or dysfunction. When a 'functional' reset event occurs, a partial reset sequence is applied to the chip. In this case, most digital modules are reset normally, while the state of analog modules or specific digital modules (e.g., debug modules, flash modules) as well as the system memory content is preserved.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FunctionalReset
literals	['FunctionalReset', 'DestructiveReset']

4.398 Parameter McuFuncResetEscThreshold

RGM_FRET[FRET] field configuration.

If the value of this field is 0, the functional reset escalation function is disabled.

Any other value is the number of 'functional' resets which will cause a 'destructive' reset.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A

Property	Value
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	15
max	15
min	0

4.399 Parameter McuDestResetEscThreshold

RGM_DRET[DRET] field configuration.

If the value of this field is 0, the destructive reset escalation function is disabled.

Any other value is the number of 'destructive' resets which will keep the chip in the reset state

until the next power-on reset.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	15
min	0

4.400 Container McuResetSourcesConfig

Configuration of reset sources.

Note: Implementation Specific Parameter.

Included subcontainers:

- McuFCCU_RST_ResetSource
- McuSWT0_RST_ResetSource
- McuSWT1_RST_ResetSource
- McuSWT2_RST_ResetSource
- McuSWT3_RST_ResetSource
- McuJTAG_RST_ResetSource
- McuDEBUG_FUNC_ResetSource

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.401 Container McuFCCU_RST_ResetSource

'FCCU Reset Reaction' reset source configuration.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.402 Parameter McuDisableReset

RGM_FERD[D_FCCU_RST] field configuration.

0 - Functional reset event FCCU_RST triggers a reset sequence.

1 - Functional reset event FCCU_RST generates an interrupt request.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.403 Container McuSWT0_RST_ResetSource

'SWT Timeout 0' reset source configuration.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.404 Parameter McuDisableReset

RGM_FERD[D_SWT0_RST] field configuration.

0 - Functional reset event SWT0_RST triggers a reset sequence.

1 - Functional reset event SWT0_RST generates an interrupt request.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.405 Container McuSWT1_RST_ResetSource

'SWT Timeout 1' reset source configuration.

This node is not supported on S32K344/S32K314/S32K312 derivatives.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.406 Parameter McuDisableReset

RGM_FERD[D_SWT1_RST] field configuration.

0 - Functional reset event SWT1_RST triggers a reset sequence.

1 - Functional reset event SWT1_RST generates an interrupt request.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.407 Container McuSWT2_RST_ResetSource

'SWT Timeout 2' reset source configuration.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.408 Parameter McuDisableReset

RGM_FERD[D_SWT2_RST] field configuration.

0 - Functional reset event SWT2_RST triggers a reset sequence.

1 - Functional reset event SWT2_RST generates an interrupt request.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.409 Container McuSWT3_RST_ResetSource

'SWT Timeout 3' reset source configuration.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.410 Parameter McuDisableReset

RGM_FERD[D_SWT3_RST] field configuration.

0 - Functional reset event SWT3_RST triggers a reset sequence.

1 - Functional reset event SWT3_RST generates an interrupt request.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.411 Container McuJTAG_RST_ResetSource

'JTAG Reset' reset source configuration.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.412 Parameter McuDisableReset

RGM_FERD[D_JTAG_RST] field configuration.

0 - Functional reset event JTAG_RST triggers a reset sequence.

1 - Functional reset event JTAG_RST generates an interrupt request.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.413 Container McuDEBUG_FUNC_ResetSource

'Debug Functional Reset' reset source configuration.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.414 Parameter McuDisableReset

RGM_FERD[D_DEBUG_FUNC] field configuration.

0 - Functional reset event 'Debug Functional Reset' triggers a reset sequence.

1 - Functional reset event 'Debug Functional Reset' generates an interrupt request.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.415 Container McuResetGeneratorConfiguration

Configures AEC_AE_RSTGEN_CFG[RSTGEN_CFG]- Configures reset generator.

Set this bit to activate IP (this deasserts hard reset input port of IP).

Individual control over:

0: regs_otp (resets OTP register interface, not mirror regs)

1: can_phy (do not activate CANPHY if VERID.VARIANT says CANPHY is disabled)

2: lin_phy HP (do not activate LINPHY if VERID.VARIANT says LINPHY is disabled)

3: lin_phy LP (do not activate LINPHY if VERID.VARIANT says LINPHY is disabled)

4: gdu

5: hvi

6: dpga

7: tempsensor

8: cxpi (do not activate CXPI if VERID.VARIANT says CXPI is disabled, or if using LINPHY in noncxpi mode)

Note: Implementation specific Container.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.416 Parameter McuRegsOtpReset

Regs_otp (resets OTP register interface, not mirror regs)

0 - Regs Otp Reset disabled.

1 - Regs Otp Reset enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.417 Parameter McuCanPhyReset

Can_phy (do not activate CANPHY if VERID.VARIANT says CANPHY is disabled)

0 - CANPHY Reset disabled.

1 - CANPHY Reset enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF

Property	Value
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.418 Parameter McuLinPhyHpReset

Lin_phy HP (do not activate LINPHY if VERID.VARIANT says LINPHY is disabled).

0 - LINPHY HP Reset disabled.

1 - LINPHY HP Reset enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.419 Parameter McuLinPhyLpReset

Lin_phy LP (do not activate LINPHY if VERID.VARIANT says LINPHY is disabled).

0 - LINPHY LP Reset disabled.

1 - LINPHY LP Reset enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.420 Parameter McuGduReset

GDU resset.

0 - GDU Reset disabled.

1 - GDU Reset enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.421 Parameter McuHviReset

HVI resset.

0 - HVI Reset disabled.

1 - HVI Reset enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.422 Parameter McuDpgaReset

DPGA resset.

0 - DPGA Reset disabled.

1 - DPGA Reset enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.423 Parameter McuTempsensorReset

Tempsensor resset.

0 - Tempsensor Reset disabled.

1 - Tempsensor Reset enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.424 Parameter McuCxpiReset

Cxpi (do not activate CXPI if VERID.VARIANT says CXPI is disabled, or if using LINPHY in noncxpi mode).

0 - Cxpi Reset disabled.

1 - Cxpi Reset enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.425 Container McuPowerControl

Note: Implementation Specific Parameter.

Included subcontainers:

- McuPMC_Config
- McuPMC_AE_Config

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.426 Container McuPMC_Config

This PMC Control Register contains the various control settings of the PMC block.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.427 Parameter McuLMAUTOENEnable

PMC_CONFIG[LMAUTOEN] field configuration.

Last Mile regulator auto turn over bit.

This bit enables to turn over automatically from Boot Regulator Mode to Last Mile regulator mode and vice versa depending on the V15 voltage status (LVD15S). As long as LMEN=0 software must make sure that the system clock is on FIRC clock or slower. To use higher clock speed software must set LMEN=1.

0 - Auto turnover disabled.

1 - Auto turnover enabled.

This node is not supported on S32K312 derivative.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.428 Parameter McuLVDIEEnable

PMC_CONFIG[LVDIE] field configuration.

This bit enables hardware interrupt requests if one of the following flags is set: LVD5AF, LVD15F.

LVD interrupt must be disabled before going in Low Power Mode (LPM).

0 - LVD hardware interrupt is disabled (use polling).

1 - Request a LVD hardware interrupt when LVDA5F=1 or LVD15F=1.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.429 Parameter McuHVDIEEnable

PMC_CONFIG[HVDIE] field configuration.

This bit enables hardware interrupt requests if one of the following flags is set: HVDAF, HVDBF, HVD25F, HVD11F.

0 - HVD hardware interrupt is disabled (use polling).

1 - Request a HVD hardware interrupt when HVDAF=1 or HVDBF=1 or HVD25F=1 or HVD11F=1.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.430 Parameter McuLMSMPSENEnable

PMC_CONFIG[LMSMPSEN] field configuration.

This bit V15 Switched-mode power supply enable bit.

0 - Switched-mode power supply (SMPS) for V15 disabled.

1 - Switched-mode power supply (SMPS) for V15 enabled.

Note: The configuration need to be corresponding to the jump setting (DC 1.5V supply configuration) on board:

- Switched-mode power supply (SMPS) for V15 disabled: 1.5V must be supplied from other source (external power screw connector, sourced from Motherboard (J56A), on board PMIC FS26).

- Switched-mode power supply (SMPS) for V15 enabled: 1.5V must be sourced from on chip SMPS.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.431 Parameter McuLVRBLPENEnable

PMC_CONFIG[LVRBLPEN] field configuration.

This bit controls whether the low voltage reset detection (LVRBLP) on VDD_HV_B power domain is

active or inactive in Low Power Mode (LPM).

0 - Low voltage reset detection (LVRBLP) is disabled in LPM.

1 - Low voltage reset detection (LVRBLP) is enabled in LPM.

This node is not supported on S32K312 derivative.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1

Property	Value
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.432 Parameter McuLPM25ENEnable

PMC_CONFIG[LPM25EN] field configuration.

This bit controls whether the V25 regulator and low voltage reset detection (LVR25LP) is active

or inactive in Low Power Mode (LPM).

0 - V25 regulator and LVR25LP is disabled in LPM.

1 - V25 regulator and LVR25LP is enabled in LPM.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.433 Parameter McuFASTRECEnable

PMC_CONFIG[FASTREC] field configuration.

Fast Recovery from LPM Enable Bit.

This bit controls the recovery time from Low Power Mode (LPM) to Full Performance Mode (FPM). At

recovery from Low Power Mode, all the tank capacitors from the secondary supplies have to re-recharged. This causes a high current demand, that might not be met by the supply driving the VDD_HV_A primary domain. When selecting the fast recovery time, the current for recharging is approximately 3 times higher than for FASTREC=0. Please refer to data sheet for normal and fast recovery time specification. Application must determine from the drive capability of the external VDD_HV_A regulator, the size of tank caps on the secondary supply pins and the selected recovery time if this is suffient to start up from Low Power Mode in time.

0 - Normal recovery time from LPM.

1 - Fast recovery time from LPM.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.434 Parameter McuLMBCTLENEnable

PMC_CONFIG[LMBCTLEN] field configuration.
Last Mile Regulator Base Control Enable Bit.
This bit must be set to 1 if external BJT between VDD_HV_A and V15 is used on the PCB.
The base of this BJT must be connected to the VRC_CTRL pin and is controlled by the PMC
to regulate a voltage of 1.5V on V15 pin.
0 - External BCTL regulator for V15 disabled.
1 - External BCTL regulator for V15 enabled.
This node is not supported on S32K312 derivative.
Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.435 Parameter McuLMENEnable

PMC_CONFIG[LMEN] field configuration.

Last Mile Regulator Enable Bit.

Enables the Last Mile regulator, which regulates an external 1.5V voltage on V15 down to the core and logic supply (V11 power domain), which is typically 1.1V. Setting LMEN=1 hands over the V11 voltage generation from the Boot regulator to the Last Mile regulator. The software must ensure that before enabling the Last Mile regulator, the voltage on V15 is sufficiently high as indicated by the LVD15S status field (LVD15S=0). To use external BJT between VDD_HV_A and V15, the LMBCTLEN field must be set before the LMEN field, and the software must wait until 1.5V is up (LVD15S=0). If LMAUTOEN=0 then to disable the Last Mile regulator, LMEN and LMBCTLEN must be cleared simultaneously (single register write). The software must disable (LMEN=0) the Last Mile regulator before going into LPM. After setting LMEN=1, software must wait a minimum time of 1.5us before changing clock rate.

0 - Last Mile Regulator disabled.

1 - Last Mile Regulator enabled.

This node is not supported on S32K312 derivative.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP

Property	Value
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.436 Parameter McuSMPSConfSelect

SMPSCONFIG[CFG] configuration select.

Select a configuration for period and duty cycle(ontime) of the Switched-mode power

supply driving via PMOS_CTRL the external power FET.

CFG[3:0] bits |PMOS_CTRL Frequency | PMOS_CTRL Duty cycle 3V |PMOS_CTRL Duty cycle 5V

0001 |470 kHz | 64.7% |52.9%

0010 |470 kHz | 52.9% |41.2%

0011 |533 kHz | 60% |46.7%

0100 |533 kHz | 66.7% |53.5%

0101 |533 kHz | 53.3% |40%

0110 |421 kHz | 57.9% |47.4%

0111 |421 kHz | 63.2% |52.6%

1000 |421 kHz | 52.6% |42.1%

1001 to 1110 |Reserved | Reserved |Reserved

1111 |8MHz/(PERIOD+1) | ONTIME3V/(PERIOD+1) |ONTIME5V/(PERIOD+1)

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	15
max	15
min	0

4.437 Parameter McuSMPSPeriod

SMPSCONFIG[PERIOD] field configuration.

Determine the period and frequency of the output signal at PMOS_CTRL pin.

It calculates as: Freq_pmos_ctrl = 8 MHz/(PERIOD + 1)

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	31
min	0

4.438 Parameter McuSMPSOnTime3V

SMPSCONFIG[ONTIME3V] Duty Cycle for 3V range.

Determine the duty cycle of the output signal at PMOS_CTRL pin.

This duty cycle is applied while VDD_DCDC is smaller than 4V.

It calculates as: Duty_pmos_ctrl = ONTIME3V/(PERIOD+1)

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	31
min	0

4.439 Parameter McuSMPSOnTime5V

SMPSCONFIG[ONTIME5V] Duty Cycle for 5V range.

Determine the duty cycle of the output signal at PMOS_CTRL pin.

This duty cycle is applied while VDD_DCDC is greater or equal than 4V.

It calculates as: Duty_pmos_ctrl = ONTIME5V/(PERIOD+1)

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	31
min	0

4.440 Parameter McuSMPSLPM15EN

SMPSCONFIG[LPM15EN] field configuration.

V15 domain enable bit during LPM.

Enables the V15 in LPM. This is useful for fast recovery to FPM when using the SMPS mode to generate the V15.

This feature is only active in LPM, in FPM it is turned off. For this purpose there is a small LDO inside the PMC

that regulates the V15 to target value during LPM.

Note: LPM15EN is only allowed to use, when using the SMPS to geenrate the V15. For other options to generate the V15

it is forbidden.

0 - V15 not kept on target in LPM.

1 - V15 kept on target in LPM.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.441 Parameter McuSMPSDitherEn

SMPSCONFIG[DITHEREN] field configuration.

IRC Dither Enable.

Enables dithering of the 8 MHz IRC.

0 - 8MHz IRC dithering disabled.

1 - 8MHz IRC dithering enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.442 Parameter McuSMPSDitherConf

SMPSCONFIG[DITHERCFG] field configuration.

Configuration to select IRC dithering amplitude and frequency.

This feature is to spread the spectrum of driving PMOS_CTRL pin to reduce radiated emission.

DITHERCFG[1:0] bits |Dither Amplitude | Dither Frequency

00 |+/-0.4 MHz | 400 kHz

01 |+/-0.6 MHz | 286 kHz

10 |+/-0.8 MHz | 222 kHz

11 |+/-1 MHz | 182 kHz

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	3
min	0

4.443 Container McuPMC_AE_Config

This PMC_AE Control Register contains the various control settings of the PMC_AE block.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.444 Parameter McuLowVoltageDetectInterruptsOnVLSEnable

PMC_MONITOR[LVDVLSIE] - LVD on VLS interrupt enable.

This is to enable interrupt for low voltage detect on VLS (GDU) power domain.

An interrupt will be requested in case of LVDVLSF=1.

0 - Low voltage detect interrupts on VLS disabled.

1 - Low voltage detect interrupts on VLS enabled.

Note:

- Implementation Specific Parameter.

- Only support for S32M2xx series of devices.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.445 Parameter McuLowVoltageDetectInterruptsOnVDDCEnable

PMC_MONITOR[LVDCIE] - LVD on VDDC interrupt enable.

This is to enable interrupt for low voltage detect on VDDC power domain.

An interrupt will be requested in case of LVDCF=1.

0 - Low voltage detect interrupts on VDDC disabled.

1 - Low voltage detect interrupts on VDDC enabled.

Note:

- Implementation Specific Parameter.

- Only support for S32M2xx series of devices.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.446 Parameter McuHighVoltageDetectInterruptOnVDDEnable

PMC_MONITOR[HVDVDDIE] - HVD on VDD interrupt enable.

This is to enable interrupt for high voltage detection on VDD supply.

An interrupt will be requested in case of HVDVDDF=1.

0 - High voltage detect interrupt disabled.

1 - High voltage detect interrupt enabled.

Note:

- Implementation Specific Parameter.

- Only support for S32M2xx series of devices.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.447 Parameter McuHighVoltageDetectInterruptOnVDDINTAndVDD15Enable

PMC_MONITOR[HVDINT15IE] - HVD on VDD interrupt enable.

This is to enable interrupt for high voltage detection on the A10 internal supplies VDDINT or VDD15.

An interrupt will be requested in case of HVD15F=1 or HVDINTF=1.

0 - High voltage detect interrupt disabled.

1 - High voltage detect interrupt enabled.

Note:

- Implementation Specific Parameter.

- Only support for S32M2xx series of devices.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.448 Parameter McuLinphySupplyEnable

PMC_CONFIG[LINSUPEN] - LINPHY supply enable bit.

Enables the LINPHY supply. If disabled LINPHY is turned high ohmic.

In case LINPHY is generally not used setting LINSUPEN=0 will save power.

0 - LINPHY supply is high ohmic (off).

1 - LINPHY supply is as selected by LINSUPSEL bit.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.449 Parameter McuVDDCEnable

PMC_CONFIG[VDDCEN] - VDDC enable bit.

This Bit enables the VDDC supply.

A PMC internal regulator generates the VDDC from VPRE supply.

VDDC is the 5V supply for the CAN physical interface IP.

0 - VDDC is disabled.

1 - VDDC is enabled and regulated to 5V.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP

Property	Value
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.450 Parameter McuLvdVlsSelect

PMC_CONFIG[LVDVLSSEL] - LVD VLS select.

The LVD (Low-Voltage-Dectection) threshold for generated VLS supply is selectable. There are two options.

5_5V - LVD threshold on VLS supply is 5.5V.

6_5V - LVD threshold on VLS supply is 6.5V.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	LVD_6_5V
literals	['LVD_5_5V', 'LVD_6_5V']

4.451 Parameter McuLinphySupplySelect

PMC_CONFIG[LINSUPSEL] - LINPHY supply select.

Selects LINPHY supply to be either connected to VSUP pin or HD pin (GDU).

In case using the boost option in GDU, the LINPHY supply must be on HD pin.

0b - LINPHY supply connects to VSUP pin.

1b - LINPHY supply connects to HD pin (of GDU).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	VSUP_PIN
literals	['VSUP_PIN', 'HD_PIN']

4.452 Parameter McuVddVoltageLevelSelect

PMC_CONFIG[VDDSEL5V] - VDD voltage level select.

Selects VDD (supply for MCU) to be either 3.3V or 5V.

3_3V - VDD is regulated to 3.3V.

5V - VDD is regulated to 5V.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	VDD_3_3V
literals	['VDD_3_3V', 'VDD_5V']

This chapter describes the Tresos configuration plug-in for the *driver* Driver. The most of the parameters are described below.

Chapter 5

Module Index

5.1 Software Specification

Here is a list of all modules:

Chapter 6

Module Documentation

6.1 Clock Ip Driver

6.1.1 Detailed Description

Data Structures

- struct Clock_Ip_ExtOSCType
- struct Clock_Ip_ClockMonitorType
- struct Clock_Ip_RegisterValueType

  Register value structure. Implements Clock_Ip_RegisterValueType_Class. More...
- struct Clock_Ip_RegisterIndexType

  Register index structure. Implements Clock_Ip_RegisterIndexType_Class. More...
- struct Clock_Ip_IrcoscConfigType

  Clock Source IRCOSC configuration structure. Implements Clock_Ip_IrcoscConfigType_Class. More...
- struct Clock_Ip_XoscConfigType

  CGM Clock Source XOSC configuration structure. Implements Clock_Ip_XoscConfigType_Class. More...
- struct Clock_Ip_PllConfigType

  CGM Clock Source PLLDIG configuration structure. Implements Clock_Ip_PllConfigType_Class. More...
- struct Clock_Ip_SelectorConfigType

  Clock selector configuration structure. Implements Clock_Ip_SelectorConfigType_Class. More...
- struct Clock_Ip_DividerConfigType

  Clock divider configuration structure. Implements Clock_Ip_DividerConfigType_Class. More...
- struct Clock_Ip_DividerTriggerConfigType

  Clock divider trigger configuration structure. Implements Clock_Ip_DividerTriggerConfigType_Class. More...
- struct Clock_Ip_FracDivConfigType

  Clock fractional divider configuration structure. Implements Clock_Ip_FracDivConfigType_Class. More...
- struct Clock_Ip_ExtClkConfigType

  Clock external clock configuration structure. Implements Clock_Ip_ExtClkConfigType_Class. More...
- struct Clock_Ip_PcfsConfigType

  Clock Source PCFS configuration structure. Implements Clock_Ip_PcfsConfigType_Class. More...
- struct Clock_Ip_GateConfigType

Clock gate clock configuration structure. Implements Clock_Ip_GateConfigType_Class. More...

- struct Clock_Ip_CmuConfigType

  Clock cmu configuration structure. Implements Clock_Ip_CmuConfigType_Class. More...

- struct Clock_Ip_ConfiguredFrequencyType

  Configured frequency structure. Implements Clock_Ip_ConfiguredFrequencyType_Class. More...

- struct Clock_Ip_ClockConfigType

  Clock configuration structure. Implements Clock_Ip_ClockConfigType_Class. More...

Types Reference

- typedef void(* Clock_Ip_NotificationsCallbackType) (Clock_Ip_NotificationType Error, Clock_Ip_NameType ClockName)

  Clock notifications callback type. Implements ClockNotificationsCallbackType_Class.

Enum Reference

- enum Clock_Ip_ClockNameSourceType

  Clock ip source type.

- enum Clock_Ip_PllStatusReturnType

  Clock pll status return codes.

- enum Clock_Ip_DfsStatusType

  Clock dfs status return codes.

- enum Clock_Ip_CommandType

  Clock ip specific commands.

- enum Clock_Ip_NameType

  Clock names.

- enum Clock_Ip_StatusType

  Clock ip status return codes.

- enum Clock_Ip_PllStatusType

  Clock ip pll status return codes.

- enum Clock_Ip_CmuStatusType

  Clock ip cmu status return codes.

- enum Clock_Ip_NotificationType

  Clock ip report error types.

- enum Clock_Ip_TriggerDividerType

  Clock ip trigger divider type.

Function Reference

- Clock_Ip_StatusType Clock_Ip_Init (Clock_Ip_ClockConfigType const *Config)
  Set clock configuration according to pre-defined structure.
- void Clock_Ip_InitClock (Clock_Ip_ClockConfigType const *Config)
  Set the PLL and other MCU specific clock options.
- Clock_Ip_PllStatusType Clock_Ip_GetPllStatus (void)
  Returns the lock status of the PLL.
- void Clock_Ip_DistributePll (void)
  Activates the PLL in MCU clock distribution.
- void Clock_Ip_InstallNotificationsCallback (Clock_Ip_NotificationsCallbackType Callback)
  Install a clock notifications callback.
- void Clock_Ip_DisableClockMonitor (Clock_Ip_NameType ClockName)
  Disables a clock monitor.
- void Clock_Ip_DisableModuleClock (Clock_Ip_NameType ClockName)
  Disables clock for a peripheral.
- void Clock_Ip_EnableModuleClock (Clock_Ip_NameType ClockName)
  Enables clock for a peripheral.
- uint32 Clock_Ip_GetClockFrequency (Clock_Ip_NameType ClockName)
  Gets the clock frequency for a specific clock name.
- void Clock_Ip_DisableFircInStandby (void)
  Disables FIRC in STANDBY mode.
- void Clock_Ip_EnableFircInStandby (void)
  Enables FIRC in STANDBY mode.
- void Clock_Ip_DisableSircInStandby (void)
  Disables SIRC in STANDBY mode.
- void Clock_Ip_EnableSircInStandby (void)
  Enables SIRC in STANDBY mode.
- void Clock_Ip_StartTimeout (uint32 *StartTimeOut, uint32 *ElapsedTimeOut, uint32 *TimeoutTicksOut, uint32 TimeoutUs)
  Initializes a starting reference point for timeout.
- boolean Clock_Ip_TimeoutExpired (uint32 *StartTimeInOut, uint32 *ElapsedTimeInOut, uint32 Timeout↩ Ticks)
  Checks for timeout condition.
- void Clock_Ip_SpecificSetUserAccessAllowed (void)
  Set user access allowed for some clock register.
- void Clock_Ip_SetRtcRtccClksel_TrustedCall (Clock_Ip_SelectorConfigType const *Config)
  Write Config RTCCLKSEL to register.
- uint32 Clock_Ip_Get_RTC_CLK_Frequency_TrustedCall (void)
  Return the frequency of RTC_CLK clock.
- void Clock_Ip_PowerClockIpModules (void)
  Set user access allowed for some clock register.
- void Clock_Ip_PRAMCSetRamIWS (void)
  Set Ram waitstate value base on core_clk.
- void Power_Ip_CM7_EnableSleepOnExit (void)
  The function enables SLEEPONEXIT bit.
- void Power_Ip_CM7_DisableSleepOnExit (void)

The function disables SLEEPONEXIT bit.

- void Power_Ip_CM7_DisableDeepSleep (void)

  The function disable SLEEPDEEP bit.
- void Power_Ip_CM7_EnableDeepSleep (void)

  The function enable SLEEPDEEP bit.
- void Power_Ip_MC_RGM_ResetInit (const Power_Ip_MC_RGM_ConfigType *ConfigPtr)

  This function initializes the Reset parameters.
- void Power_Ip_MC_RGM_PerformReset (const Power_Ip_MC_RGM_ConfigType *ConfigPtr)

  This function performs a microcontroller reset.
- Power_Ip_ResetType Power_Ip_MC_RGM_GetResetReason (void)

  This function returns the Reset reason.
- Power_Ip_RawResetType Power_Ip_MC_RGM_GetResetRawValue (void)

  This function returns the Raw Reset value.
- uint8 Power_Ip_MC_RGM_ResetDuringStandby (void)

  This function returns whether a reset occurred during standby.
- void Power_Ip_PMC_PrepareLowPowerEntry (void)

  This function prepares the PMC module for Standby/Low Power entry.

6.1.2 Data Structure Documentation

6.1.2.1 struct Clock_Ip_ExtOSCType

XOSC - Register Layout Typedef

Definition at line 415 of file Clock_Ip_Specific.h.

Data Fields

Type	Name	Description
uint32	CTRL	XOSC Control Register, offset: 0x0
const uint32	STAT	Oscillator Status Register, offset: 0x4

6.1.2.2 struct Clock_Ip_ClockMonitorType

CMU - Register Layout Typedef

Definition at line 421 of file Clock_Ip_Specific.h.

Data Fields

Type	Name	Description
uint32	GCR	Global Configuration Register, offset: 0x0
uint32	RCCR	Reference Count Configuration Register, offset: 0x4
uint32	HTCR	High Threshold Configuration Register, offset: 0x8

Data Fields

Type	Name	Description
uint32	LTCR	Low Threshold Configuration Register, offset: 0xC
volatile uint32	SR	Status Register, offset: 0x10
uint32	IER	Interrupt Enable Register, offset: 0x14

6.1.2.3 struct Clock_Ip_RegisterValueType

Register value structure. Implements Clock_Ip_RegisterValueType_Class.

Definition at line 2911 of file Clock_Ip_Types.h.

Data Fields

Type	Name	Description
uint32 *	RegisterAddr	Register address.
uint32	RegisterData	Register value.

6.1.2.4 struct Clock_Ip_RegisterIndexType

Register index structure. Implements Clock_Ip_RegisterIndexType_Class.

Definition at line 2922 of file Clock_Ip_Types.h.

Data Fields

Type	Name	Description
uint16	StartIndex	Start index in register array.
uint16	EndIndex	End index in register array.

6.1.2.5 struct Clock_Ip_IrcoscConfigType

Clock Source IRCOSC configuration structure. Implements Clock_Ip_IrcoscConfigType_Class.

Definition at line 2935 of file Clock_Ip_Types.h.

Data Fields

Type	Name	Description
Clock_Ip_NameType	Name	Clock name associated to ircosc
uint16	Enable	Enable ircosc.

Data Fields

Type	Name	Description
uint8	Regulator	Enable regulator.
uint8	Range	Ircosc range.
uint8	LowPowerModeEnable	Ircosc enable in VLP mode
uint8	StopModeEnable	Ircosc enable in STOP mode

6.1.2.6 struct Clock_Ip_XoscConfigType

CGM Clock Source XOSC configuration structure. Implements Clock_Ip_XoscConfigType_Class.

Definition at line 2951 of file Clock_Ip_Types.h.

Data Fields

Type	Name	Description
Clock_Ip_NameType	Name	Clock name associated to xosc
uint32	Freq	External oscillator frequency.
uint16	Enable	Enable xosc.
uint16	StartupDelay	Startup stabilization time.
uint8	BypassOption	XOSC bypass option
uint8	CompEn	Comparator enable
uint8	TransConductance	Crystal overdrive protection
uint8	Gain	Gain value
uint8	Monitor	Monitor type
uint8	AutoLevelController	Automatic level controller

6.1.2.7 struct Clock_Ip_PllConfigType

CGM Clock Source PLLDIG configuration structure. Implements Clock_Ip_PllConfigType_Class.

Definition at line 2974 of file Clock_Ip_Types.h.

Data Fields

Type	Name	Description
Clock_Ip_NameType	Name	Clock name associated to pll
uint16	Enable	Enable pll.
Clock_Ip_NameType	InputReference	Input reference.
uint8	Bypass	Bypass pll.
uint8	Predivider	Input clock predivider.
uint16	Multiplier	Clock multiplier.

Data Fields

Type	Name	Description
uint8	Postdivider	Clock postidivder.
uint16	NumeratorFracLoopDiv	Numerator of fractional loop division factor (MFN)
uint8	MulFactorDiv	Multiplication factor divider (MFD)
uint8	FrequencyModulationBypass	Enable/disable modulation
uint8	ModulationType	Modulation type
uint16	ModulationPeriod	Stepsize - modulation period
uint16	IncrementStep	Stepno - step no
uint8	SigmaDelta	Sigma Delta Modulation Enable
uint8	DitherControl	Dither control enable
uint8	DitherControlValue	Dither control value
uint8	Monitor	Monitor type
uint16	Dividers[3U]	Dividers values

6.1.2.8 struct Clock_Ip_SelectorConfigType

Clock selector configuration structure. Implements Clock_Ip_SelectorConfigType_Class.

Definition at line 3011 of file Clock_Ip_Types.h.

Data Fields

Type	Name	Description
Clock_Ip_NameType	Name	Clock name associated to selector
Clock_Ip_NameType	Value	Name of the selected input source

6.1.2.9 struct Clock_Ip_DividerConfigType

Clock divider configuration structure. Implements Clock_Ip_DividerConfigType_Class.

Definition at line 3022 of file Clock_Ip_Types.h.

Data Fields

Type	Name	Description
Clock_Ip_NameType	Name	Clock name associated to divider.
uint32	Value	Divider value - if value is zero then divider is disabled.
uint8	Options[1U]	Option divider value - this value depend hardware information.

6.1.2.10 struct Clock_Ip_DividerTriggerConfigType

Clock divider trigger configuration structure. Implements Clock_Ip_DividerTriggerConfigType_Class.

Definition at line 3033 of file Clock_Ip_Types.h.

Data Fields

Type	Name	Description
Clock_Ip_NameType	Name	Clock name associated to divider for which trigger is configured.
Clock_Ip_TriggerDividerType	TriggerType	Trigger value - if value is zero then divider is updated immediately, divider is not triggered.
Clock_Ip_NameType	Source	Clock name of the common input source of all dividers from the same group that support a common update

6.1.2.11 struct Clock_Ip_FracDivConfigType

Clock fractional divider configuration structure. Implements Clock_Ip_FracDivConfigType_Class.

Definition at line 3047 of file Clock_Ip_Types.h.

Data Fields

Type	Name	Description
Clock_Ip_NameType	Name	Clock name associated to fractional divider.
uint8	Enable	Enable control for port n
uint32	Value[2U]	Fractional dividers

6.1.2.12 struct Clock_Ip_ExtClkConfigType

Clock external clock configuration structure. Implements Clock_Ip_ExtClkConfigType_Class.

Definition at line 3059 of file Clock_Ip_Types.h.

Data Fields

Type	Name	Description
Clock_Ip_NameType	Name	Clock name of the external clock.
uint32	Value	Enable value - if value is zero then clock is gated, otherwise is enabled in different modes.

6.1.2.13 struct Clock_Ip_PcfsConfigType

Clock Source PCFS configuration structure. Implements Clock_Ip_PcfsConfigType_Class.

Definition at line 3070 of file Clock_Ip_Types.h.

Data Fields

Type	Name	Description
Clock_Ip_NameType	Name	Clock source from which ramp-down and to which ramp-up are processed.
uint32	MaxAllowableIDDchange	Maximum variation of current per time (mA/microsec) - max allowable IDD change is determined by the user's power supply design.
uint32	StepDuration	Step duration of each PCFS step
Clock_Ip_NameType	SelectorName	Name of the selector that supports PCFS and name is one the inputs that can be selected
uint32	ClockSourceFrequency	Frequency of the clock source from which ramp-down and to which ramp-up are processed.

6.1.2.14 struct Clock_Ip_GateConfigType

Clock gate clock configuration structure. Implements Clock_Ip_GateConfigType_Class.

Definition at line 3084 of file Clock_Ip_Types.h.

Data Fields

Type	Name	Description
Clock_Ip_NameType	Name	Clock name associated to clock gate.
uint16	Enable	Enable or disable clock

6.1.2.15 struct Clock_Ip_CmuConfigType

Clock cmu configuration structure. Implements Clock_Ip_CmuConfigType_Class.

Definition at line 3095 of file Clock_Ip_Types.h.

Data Fields

Type	Name	Description
Clock_Ip_NameType	Name	Clock name associated to clock monitor.
uint8	Enable	Enable/disable clock monitor
uint32	Interrupt	Enable/disable interrupt

Data Fields

Type	Name	Description
uint32	MonitoredClockFrequency	Frequency of the clock source from which ramp-down and to which ramp-up are processed.
Clock_Ip_RegisterIndexType	Indexes	Register index if register value optimization is enabled.

6.1.2.16 struct Clock_Ip_ConfiguredFrequencyType

Configured frequency structure. Implements Clock_Ip_ConfiguredFrequencyType_Class.

Definition at line 3108 of file Clock_Ip_Types.h.

Data Fields

Type	Name	Description
Clock_Ip_NameType	Name	Clock name of the configured frequency value
uint32	ConfiguredFrequencyValue	Configured frequency value

6.1.2.17 struct Clock_Ip_ClockConfigType

Clock configuration structure. Implements Clock_Ip_ClockConfigType_Class.

Definition at line 3118 of file Clock_Ip_Types.h.

Data Fields

Type	Name	Description
uint32	ClkConfigId	The ID for Clock configuration
const Clock_Ip_RegisterValueType(∗	RegValues)[]	Pointer to register values array
uint8	IrcoscsCount	IRCOSCs count
uint8	XoscsCount	XOSCs count
uint8	PllsCount	PLLs count
uint8	SelectorsCount	Selectors count
uint8	DividersCount	Dividers count
uint8	DividerTriggersCount	Divider triggers count
uint8	FracDivsCount	Fractional dividers count
uint8	ExtClksCount	External clocks count
uint8	GatesCount	Clock gates count
uint8	PcfsCount	Clock pcfs count
uint8	CmusCount	Clock cmus count
uint8	ConfigureFrequenciesCount	Configured frequencies count
const Clock_Ip_IrcoscConfigType(∗	Ircoscs)[]	IRCOSCs

Data Fields

Type	Name	Description
const Clock_Ip_XoscConfigType(*	Xoscs)[]	XOSCs
const Clock_Ip_PllConfigType(*	Plls)[]	PLLs
const Clock_Ip_SelectorConfigType(*	Selectors)[]	Selectors
const Clock_Ip_DividerConfigType(*	Dividers)[]	Dividers
const Clock_Ip_DividerTriggerConfigType(*	DividerTriggers)[]	Divider triggers
const Clock_Ip_FracDivConfigType(*	FracDivs)[]	Fractional dividers
const Clock_Ip_ExtClkConfigType(*	ExtClks)[]	External clocks
const Clock_Ip_GateConfigType(*	Gates)[]	Clock gates
const Clock_Ip_PcfsConfigType(*	Pcfs)[]	Progressive clock switching
const Clock_Ip_CmuConfigType(*	Cmus)[]	Clock cmus
const Clock_Ip_ConfiguredFrequencyType(*	ConfiguredFrequencies)[]	Configured frequency values

6.1.3 Types Reference

6.1.3.1 Clock_Ip_NotificationsCallbackType

```
typedef void(* Clock_Ip_NotificationsCallbackType) (Clock_Ip_NotificationType Error, Clock_Ip_NameType ClockName)
```

Clock notifications callback type. Implements ClockNotificationsCallbackType_Class.

Definition at line 2905 of file Clock_Ip_Types.h.

6.1.4 Enum Reference

6.1.4.1 Clock_Ip_ClockNameSourceType

```
enum Clock_Ip_ClockNameSourceType
```

Clock ip source type.

Enumerator

UKNOWN_TYPE	Clock path from source to this clock name has at least one selector.
IRCOSC_TYPE	Source is an internal oscillator.
XOSC_TYPE	Source is an external oscillator.
PLL_TYPE	Source is a pll.
EXT_CLK_TYPE	Source is an external clock.
SERDES_TYPE	Source is a SERDES.

Definition at line 256 of file Clock_Ip_Private.h.

6.1.4.2 Clock_Ip_PllStatusReturnType

enum Clock_Ip_PllStatusReturnType

Clock pll status return codes.

Enumerator

STATUS_PLL_NOT_ENABLED	Not enabled
STATUS_PLL_UNLOCKED	Unlocked
STATUS_PLL_LOCKED	Locked

Definition at line 270 of file Clock_Ip_Private.h.

6.1.4.3 Clock_Ip_DfsStatusType

enum Clock_Ip_DfsStatusType

Clock dfs status return codes.

Enumerator

STATUS_DFS_NOT_ENABLED	Not enabled
STATUS_DFS_UNLOCKED	Unlocked
STATUS_DFS_LOCKED	Locked

Definition at line 280 of file Clock_Ip_Private.h.

6.1.4.4 Clock_Ip_CommandType

enum Clock_Ip_CommandType

Clock ip specific commands.

Enumerator

CLOCK_IP_RESERVED_COMMAND	Reserved command

Enumerator

CLOCK_IP_INITIALIZE_PLATFORM_COMMAND	Specific platform objects
CLOCK_IP_INITIALIZE_CLOCK_OBJECTS_COMMAND	Initialize clock objects
CLOCK_IP_SET_USER_ACCESS_ALLOWED_COMMAND	User access allowed
CLOCK_IP_DISABLE_SAFE_CLOCK_COMMAND	Disable safe clock

Definition at line 290 of file Clock_Ip_Private.h.

6.1.4.5 Clock_Ip_NameType

enum Clock_Ip_NameType

Clock names.

Definition at line 125 of file Clock_Ip_Types.h.

6.1.4.6 Clock_Ip_StatusType

enum Clock_Ip_StatusType

Clock ip status return codes.

Enumerator

CLOCK_IP_SUCCESS	Clock tree was initialized successfully.
CLOCK_IP_ERROR	One of the elements timeout, clock tree couldn't be initialized.

Definition at line 2849 of file Clock_Ip_Types.h.

6.1.4.7 Clock_Ip_PllStatusType

enum Clock_Ip_PllStatusType

Clock ip pll status return codes.

Enumerator

CLOCK_IP_PLL_LOCKED	PLL is locked
CLOCK_IP_PLL_UNLOCKED	PLL is unlocked
CLOCK_IP_PLL_STATUS_UNDEFINED	PLL status is unknown

Definition at line 2857 of file Clock_Ip_Types.h.

6.1.4.8 Clock_Ip_CmuStatusType

enum Clock_Ip_CmuStatusType

Clock ip cmu status return codes.

Enumerator

CLOCK_IP_CMU_IN_RANGE	Frequency is in range
CLOCK_IP_CMU_HIGH_FREQ	Frequency is higher than high limit
CLOCK_IP_CMU_LOW_FREQ	Frequency is lower than low limit
CLOCK_IP_CMU_STATUS_UNDEFINED	CMU status is unknown

Definition at line 2866 of file Clock_Ip_Types.h.

6.1.4.9 Clock_Ip_NotificationType

enum Clock_Ip_NotificationType

Clock ip report error types.

Enumerator

CLOCK_IP_CMU_ERROR	Cmu Fccu notification.
CLOCK_IP_REPORT_TIMEOUT_ERROR	Report Timeout Error.
CLOCK_IP_REPORT_FXOSC_CONFIGURATION_ERROR	Report Fxosc Configuration Error.
CLOCK_IP_REPORT_CLOCK_MUX_SWITCH_ERROR	Report Clock Mux Switch Error.
CLOCK_IP_RAM_MEMORY_CONFIG_ENTRY	Ram config entry point.
CLOCK_IP_RAM_MEMORY_CONFIG_EXIT	Ram config exit point.
CLOCK_IP_FLASH_MEMORY_CONFIG_ENTRY	Flash config entry point.
CLOCK_IP_FLASH_MEMORY_CONFIG_EXIT	Flash config exit point.
CLOCK_IP_ACTIVE	Report Clock Active.
CLOCK_IP_INACTIVE	Report Clock Inactive.
CLOCK_IP_REPORT_WRITE_PROTECTION_ERROR	Report Write Protection Error.

Definition at line 2875 of file Clock_Ip_Types.h.

6.1.4.10 Clock_Ip_TriggerDividerType

```
enum Clock_Ip_TriggerDividerType
```

Clock ip trigger divider type.

Enumerator

IMMEDIATE_DIVIDER_UPDATE	Immediate divider update.
COMMON_TRIGGER_DIVIDER_UPDATE	Common trigger divider update.

Definition at line 2891 of file Clock_Ip_Types.h.

6.1.5 Function Reference

6.1.5.1 Clock_Ip_Init()

```
Clock_Ip_StatusType Clock_Ip_Init (
            Clock_Ip_ClockConfigType const * Config )
```

Set clock configuration according to pre-defined structure.

This function sets system to target clock configuration; It sets the clock modules registers for clock mode change.

Parameters

in	*Config*	Pointer to configuration structure.

Returns

void

Note

If external clock is used in the target mode, please make sure it is enabled, for example, if the external oscillator is used, please setup correctly.

6.1.5.2 Clock_Ip_InitClock()

```
void Clock_Ip_InitClock (
            Clock_Ip_ClockConfigType const * Config )
```

Set the PLL and other MCU specific clock options.

This function initializes the PLL and other MCU specific clock options. The clock configuration parameters are provided via the configuration structure.

This function shall start the PLL lock procedure (if PLL shall be initialized) and shall return without waiting until the PLL is locked.

Parameters

in	*Config*	Pointer to configuration structure.

Returns

void

6.1.5.3 Clock_Ip_GetPllStatus()

```
Clock_Ip_PllStatusType Clock_Ip_GetPllStatus (
            void  )
```

Returns the lock status of the PLL.

This function returns status of the PLL: undefined, unlocked or locked. This function returns undefined status if this function is called prior to calling of the function Clock_Ip_InitClock

Returns

Status. Pll lock status

6.1.5.4 Clock_Ip_DistributePll()

```
void Clock_Ip_DistributePll (
            void  )
```

Activates the PLL in MCU clock distribution.

This function activates the PLL clock to the MCU clock distribution.

This function removes the current clock source (for example internal oscillator clock) from MCU clock distribution.

Application layer calls this function after the status of the PLL has been detected as locked by the function Clock↩ _Ip_GetPllStatus.

The function Clock_Ip_DistributePll shall return without affecting the MCU hardware if the PLL clock has been automatically activated by the MCU hardware.

Returns

void

6.1.5.5 Clock_Ip_InstallNotificationsCallback()

```
void Clock_Ip_InstallNotificationsCallback (
            Clock_Ip_NotificationsCallbackType Callback )
```

Install a clock notifications callback.

This function installs a callback for reporting notifications from clock driver

Parameters

in	*Clock_Ip_NotificationsCallbackType*	notifications callback

Returns

void

6.1.5.6 Clock_Ip_DisableClockMonitor()

```
void Clock_Ip_DisableClockMonitor (
            Clock_Ip_NameType ClockName )
```

Disables a clock monitor.

This function disables a clock monitor.

Parameters

in	*ClockName*	Clock Name.

Returns

void

6.1.5.7 Clock_Ip_DisableModuleClock()

```
void Clock_Ip_DisableModuleClock (
            Clock_Ip_NameType ClockName )
```

Disables clock for a peripheral.

This function disables clock for a peripheral.

Parameters

in	*ClockName*	Clock Name.

Returns

void

6.1.5.8 Clock_Ip_EnableModuleClock()

```
void Clock_Ip_EnableModuleClock (
            Clock_Ip_NameType ClockName )
```

Enables clock for a peripheral.

This function enables clock for a peripheral.

Parameters

in	*ClockName*	Clock Name.

Returns

void

6.1.5.9 Clock_Ip_GetClockFrequency()

```
uint32 Clock_Ip_GetClockFrequency (
            Clock_Ip_NameType ClockName )
```

Gets the clock frequency for a specific clock name.

This function checks the current clock configurations and then calculates the clock frequency for a specific clock name defined in Clock_Ip_NameType. Clock modules must be properly configured before using this function. See features.h for supported clock names for different chip families. The returned value is in Hertz. If frequency is required for a peripheral and the module is not clocked, then 0 Hz frequency is returned.

Parameters

in	*ClockName*	Clock names defined in Clock_Ip_NameType

Returns

frequency Returned clock frequency value in Hertz

6.1.5.10 Clock_Ip_DisableFircInStandby()

```
void Clock_Ip_DisableFircInStandby (
            void  )
```

Disables FIRC in STANDBY mode.

Parameters

in	*void.*	

Returns

void

6.1.5.11 Clock_Ip_EnableFircInStandby()

```
void Clock_Ip_EnableFircInStandby (
            void  )
```

Enables FIRC in STANDBY mode.

Parameters

in	*void.*	

Returns

void

6.1.5.12 Clock_Ip_DisableSircInStandby()

```
void Clock_Ip_DisableSircInStandby (
            void  )
```

Disables SIRC in STANDBY mode.

Parameters

in	*void.*	

Returns

void

6.1.5.13 Clock_Ip_EnableSircInStandby()

```
void Clock_Ip_EnableSircInStandby (
            void  )
```

Enables SIRC in STANDBY mode.

Parameters

in	*void.*	

Returns

void

6.1.5.14 Clock_Ip_StartTimeout()

```
void Clock_Ip_StartTimeout (
            uint32 * StartTimeOut,
            uint32 * ElapsedTimeOut,
            uint32 * TimeoutTicksOut,
            uint32 TimeoutUs )
```

Initializes a starting reference point for timeout.

Parameters

out	*StartTimeOut*	The starting time from which elapsed time is measured
out	*ElapsedTimeOut*	The elapsed time to be passed to Clock_Ip_TimeoutExpired
out	*TimeoutTicksOut*	The timeout value (in ticks) to be passed to Clock_Ip_TimeoutExpired
in	*TimeoutUs*	The timeout value (in microseconds)

6.1.5.15 Clock_Ip_TimeoutExpired()

```
boolean Clock_Ip_TimeoutExpired (
            uint32 * StartTimeInOut,
            uint32 * ElapsedTimeInOut,
            uint32 TimeoutTicks )
```

Checks for timeout condition.

Parameters

in,out	*StartTimeInOut*	The starting time from which elapsed time is measured
in,out	*ElapsedTimeInOut*	The accumulated elapsed time from the starting time reference
in	*TimeoutTicks*	The timeout limit (in ticks)

6.1.5.16 Clock_Ip_SpecificSetUserAccessAllowed()

```
void Clock_Ip_SpecificSetUserAccessAllowed (
            void  )
```

Set user access allowed for some clock register.

Parameters

None	

Returns

None

6.1.5.17 Clock_Ip_SetRtcRtccClksel_TrustedCall()

```
void Clock_Ip_SetRtcRtccClksel_TrustedCall (
            Clock_Ip_SelectorConfigType const * Config )
```

Write Config RTCCLKSEL to register.

Parameters

Config	- pointer to configuration of RTC selector

Returns

None

6.1.5.18 Clock_Ip_Get_RTC_CLK_Frequency_TrustedCall()

```
uint32 Clock_Ip_Get_RTC_CLK_Frequency_TrustedCall (
            void  )
```

Return the frequency of RTC_CLK clock.

Parameters

None	

Returns

uint32

6.1.5.19 Clock_Ip_PowerClockIpModules()

```
void Clock_Ip_PowerClockIpModules (
            void  )
```

Set user access allowed for some clock register.

Parameters

None	

Returns

None

6.1.5.20 Clock_Ip_PRAMCSetRamIWS()

```
void Clock_Ip_PRAMCSetRamIWS (
             void  )
```

Set Ram waitstate value base on core_clk.

Parameters

None	

Returns

None

6.1.5.21 Power_Ip_CM7_EnableSleepOnExit()

```
void Power_Ip_CM7_EnableSleepOnExit (
             void  )
```

The function enables SLEEPONEXIT bit.

Parameters

None	

Returns

None

6.1.5.22 Power_Ip_CM7_DisableSleepOnExit()

```
void Power_Ip_CM7_DisableSleepOnExit (
             void  )
```

The function disables SLEEPONEXIT bit.

Parameters

None	

Returns

None

6.1.5.23 Power_Ip_CM7_DisableDeepSleep()

```
void Power_Ip_CM7_DisableDeepSleep (
            void  )
```

The function disable SLEEPDEEP bit.

Parameters

in	*none*	

Returns

void

6.1.5.24 Power_Ip_CM7_EnableDeepSleep()

```
void Power_Ip_CM7_EnableDeepSleep (
            void  )
```

The function enable SLEEPDEEP bit.

Parameters

in	*none*	

Returns

void

6.1.5.25 Power_Ip_MC_RGM_ResetInit()

```
void Power_Ip_MC_RGM_ResetInit (
            const Power_Ip_MC_RGM_ConfigType * ConfigPtr )
```

This function initializes the Reset parameters.

Parameters

ConfigPtr	Pointer to the MC_RGM configuration structure.

Returns

None

6.1.5.26 Power_Ip_MC_RGM_PerformReset()

```
void Power_Ip_MC_RGM_PerformReset (
            const Power_Ip_MC_RGM_ConfigType * ConfigPtr )
```

This function performs a microcontroller reset.

Parameters

ConfigPtr	Pointer to the MC_RGM configuration structure.

Returns

None

6.1.5.27 Power_Ip_MC_RGM_GetResetReason()

```
Power_Ip_ResetType Power_Ip_MC_RGM_GetResetReason (
            void  )
```

This function returns the Reset reason.

Parameters

None	

Returns

Reason of the Reset event.

6.1.5.28 Power_Ip_MC_RGM_GetResetRawValue()

```
Power_Ip_RawResetType Power_Ip_MC_RGM_GetResetRawValue (
            void  )
```

This function returns the Raw Reset value.

Parameters

None	

Returns

Implementation-specific value with the Reset status.

6.1.5.29 Power_Ip_MC_RGM_ResetDuringStandby()

```
uint8 Power_Ip_MC_RGM_ResetDuringStandby (
            void  )
```

This function returns whether a reset occurred during standby.

Parameters

None	

Returns

boolean

6.1.5.30 Power_Ip_PMC_PrepareLowPowerEntry()

```
void Power_Ip_PMC_PrepareLowPowerEntry (
            void  )
```

This function prepares the PMC module for Standby/Low Power entry.

Parameters

None	

Returns

None

6.2 Mcu Driver

6.2.1 Detailed Description

Data Structures

- struct Mcu_ConfigType

  Initialization data for the MCU driver. More...
- struct Mcu_MidrReturnType

  MIDR configuration. More...

Macros

- #define MCU_VENDOR_ID

  Import all data types from lower layers that should be exported. Mcu.h shall include Mcu_Cfg.h for the API pre-compiler switches.

Types Reference

- typedef Power_Ip_HwIPsConfigType Mcu_HwIPsConfigType

  Mcu driver configuration structure.
- typedef Clock_Ip_ClockConfigType Mcu_ClockConfigType

  Definition of a Clock configuration.
- typedef Ram_Ip_RamConfigType Mcu_RamConfigType

  Definition of a Clock configuration.
- typedef Power_Ip_ModeConfigType Mcu_ModeConfigType

  Definition of a Mode configuration.

Enum Reference

- enum Mcu_ClockNotificationType

  Mcu_ClockNotificationType.

Function Reference

- void Mcu_Init (const Mcu_ConfigType *ConfigPtr)

  MCU driver initialization function.
- Std_ReturnType Mcu_InitRamSection (Mcu_RamSectionType RamSection)

  MCU driver initialization of Ram sections.
- Std_ReturnType Mcu_InitClock (Mcu_ClockType ClockSetting)

  MCU driver clock initialization function.
- void Mcu_SetMode (Mcu_ModeType McuMode)

  This function sets the MCU power mode.
- Std_ReturnType Mcu_DistributePllClock (void)

  This function activates the PLL clock to the MCU clock distribution.
- Mcu_PllStatusType Mcu_GetPllStatus (void)

  This function returns the lock status of the PLL.
- Mcu_ResetType Mcu_GetResetReason (void)

  This function returns the Reset reason.
- Mcu_RawResetType Mcu_GetResetRawValue (void)

  This function returns the Raw Reset value.
- void Mcu_PerformReset (void)

  This function performs a microcontroller reset.
- void Mcu_GetVersionInfo (Std_VersionInfoType *versioninfo)

  This function returns the Version Information for the MCU module.
- Mcu_RamStateType Mcu_GetRamState (void)

  This function returns the actual state of the RAM.
- void Mcu_GetMidrStructure (Mcu_MidrReturnType MidrPtr[((uint8) 1U)])

  This function returns the value of the MIDR registers.
- void Mcu_EmiosConfigureGpren (uint8 Module, uint8 Value)

  eMios Global Prescaler Enable.
- void Mcu_DisableCmu (Clock_Ip_NameType ClockName)

  Disable clock monitoring unit.
- uint32 Mcu_GetClockFrequency (Clock_Ip_NameType ClockName)

  Return the frequency of a given clock.
- void Mcu_SleepOnExit (Mcu_SleepOnExitType SleepOnExit)

  This function disable/enable SleepOnExit.
- void Mcu_PmcAeConfig (void)

  This function configure the Power Management Controller AE. This function configure the Power Management Controller AE of the microcontroller.
- void Mcu_AecResetConfig (void)

  This function configure the Reset config AEC. This function configure the Reset config AEC of the microcontroller.
- void Mcu_ReportDemTimeoutError (void)

  Reports timeout error to DEM.
- void Mcu_ReportDemFxoscError (void)

  Reports incorrect FXOSC configuration to DEM.
- void Mcu_ReportDemClockMuxSwitchError (void)

  Reports failed clock multiplexer switch to DEM.

Variables

- const Mcu_ConfigType * Mcu_pConfigPtr
  Local copy of the pointer to the configuration data.

6.2.2 Data Structure Documentation

6.2.2.1 struct Mcu_ConfigType

Initialization data for the MCU driver.

A pointer to such a structure is provided to the MCU initialization routines for configuration.

Definition at line 169 of file Mcu.h.

Data Fields

- Mcu_ClockNotificationType ClkSrcFailureNotification
  Clock source failure notification enable configuration.
- const Mcu_DemConfigType * DemConfigPtr
  Total number of RAM sections.
- Mcu_RamSectionType NoRamConfigs
  Total number of MCU modes.
- Mcu_ModeType NoModeConfigs
  Total number of MCU clock configurations.
- const Mcu_RamConfigType(* RamConfigArrayPtr)[((uint32) 2U)]
  RAM data configuration.
- const Mcu_ModeConfigType(* ModeConfigArrayPtr)[((uint32) 3U)]
  Clock data configuration.
- const Mcu_HwIPsConfigType * HwIPsConfigPtr
  IPs data generic configuration.

6.2.2.1.1 Field Documentation

6.2.2.1.1.1 ClkSrcFailureNotification `Mcu_ClockNotificationType ClkSrcFailureNotification`

Clock source failure notification enable configuration.

<

DEM error reporting configuration.

Definition at line 173 of file Mcu.h.

6.2.2.1.1.2 DemConfigPtr `const Mcu_DemConfigType* DemConfigPtr`

Total number of RAM sections.

Definition at line 178 of file Mcu.h.

6.2.2.1.1.3 NoRamConfigs `Mcu_RamSectionType NoRamConfigs`

Total number of MCU modes.

Definition at line 181 of file Mcu.h.

6.2.2.1.1.4 NoModeConfigs `Mcu_ModeType NoModeConfigs`

Total number of MCU clock configurations.

Definition at line 184 of file Mcu.h.

6.2.2.1.1.5 RamConfigArrayPtr `const Mcu_RamConfigType(* RamConfigArrayPtr)[((uint32) 2U)]`

RAM data configuration.

<

Power Modes data configuration.

Definition at line 192 of file Mcu.h.

6.2.2.1.1.6 ModeConfigArrayPtr `const Mcu_ModeConfigType(* ModeConfigArrayPtr)[((uint32) 3U)]`

Clock data configuration.

Definition at line 196 of file Mcu.h.

6.2.2.1.1.7 HwIPsConfigPtr `const Mcu_HwIPsConfigType* HwIPsConfigPtr`

IPs data generic configuration.

<

Definition at line 203 of file Mcu.h.

6.2.2.2 struct Mcu_MidrReturnType

MIDR configuration.

Definition at line 236 of file Mcu_Ipw_Types.h.

Data Fields

Type	Name	Description
uint32	Midr1	SIUL2_MIDR1 Configuration register.
uint32	Midr2	SIUL2_MIDR2 Configuration register.
uint32	Midr3	SIUL2_MIDR3 Configuration register.
uint32	Midr4	SIUL2_MIDR4 Configuration register.

6.2.3 Macro Definition Documentation

6.2.3.1 MCU_VENDOR_ID

```
#define MCU_VENDOR_ID
```

Import all data types from lower layers that should be exported. Mcu.h shall include Mcu_Cfg.h for the API pre-compiler switches.

Definition at line 63 of file Mcu.h.

6.2.4 Types Reference

6.2.4.1 Mcu_HwIPsConfigType

```
typedef Power_Ip_HwIPsConfigType Mcu_HwIPsConfigType
```

Mcu driver configuration structure.

Configuration for SIU reset configuration module. Configuration for power management and SSCM. Configuration for FLASH controller. Used by "Mcu_ConfigType" structure.

Definition at line 193 of file Mcu_Ipw_Types.h.

6.2.4.2 Mcu_ClockConfigType

```
typedef Clock_Ip_ClockConfigType Mcu_ClockConfigType
```

Definition of a Clock configuration.

This configuration is transmitted as parameter to Mcu_Ipw_InitClock() API. Used by "Mcu_ConfigType" structure.

Note

The structure Mcu_ConfigType shall provide a configurable (enable/ disable) clock failure notification if the MCU provides an interrupt for such detection.

Definition at line 206 of file Mcu_Ipw_Types.h.

6.2.4.3 Mcu_RamConfigType

```
typedef Ram_Ip_RamConfigType Mcu_RamConfigType
```

Definition of a Clock configuration.

This configuration is transmitted as parameter to Mcu_Ipw_InitClock() API. Used by "Mcu_ConfigType" structure.

Note

The structure Mcu_ConfigType shall provide a configurable (enable/ disable) clock failure notification if the MCU provides an interrupt for such detection.

Definition at line 219 of file Mcu_Ipw_Types.h.

6.2.4.4 Mcu_ModeConfigType

```
typedef Power_Ip_ModeConfigType Mcu_ModeConfigType
```

Definition of a Mode configuration.

This configuration is transmitted as parameter to Mcu_Ipw_SetMode() API. Used by "Mcu_ConfigType" structure.

Definition at line 228 of file Mcu_Ipw_Types.h.

6.2.5 Enum Reference

6.2.5.1 Mcu_ClockNotificationType

```
enum Mcu_ClockNotificationType
```

Mcu_ClockNotificationType.

Clock failure notification. Enable/disable clock failure interrupt generated by the MCU.

Enumerator

MCU_CLK_NOTIF_DIS	Disable clock notification.
MCU_CLK_NOTIF_EN	Enable clock notification.

Definition at line 172 of file Mcu_Ipw_Types.h.

6.2.6 Function Reference

6.2.6.1 Mcu_Init()

```
void Mcu_Init (
            const Mcu_ConfigType * ConfigPtr )
```

MCU driver initialization function.

This routine initializes the MCU Driver. The intention of this function is to make the configuration setting for power down, clock and Ram sections visible within the MCU Driver.

Parameters

in	*ConfigPtr*	Pointer to configuration structure.

Returns

void

6.2.6.2 Mcu_InitRamSection()

```
Std_ReturnType Mcu_InitRamSection (
            Mcu_RamSectionType RamSection )
```

MCU driver initialization of Ram sections.

Function initializes the ram section selected by RamSection parameter. The section base address, size and value to be written are provided from the configuration structure. The function will write the value specified in the configuration structure indexed by RamSection. After the write it will read back the RAM to verify that the requested value was written.

Parameters

in	*RamSection*	Index of ram section from configuration structure to be initialized.

Returns

Command has or has not been accepted.

Return values

E_OK	Valid parameter, the driver state allowed execution and the RAM check was successful
E_NOT_OK	Invalid parameter, the driver state did not allowed execution or the RAM check was not successful

6.2.6.3 Mcu_InitClock()

```
Std_ReturnType Mcu_InitClock (
            Mcu_ClockType ClockSetting )
```

MCU driver clock initialization function.

This function intializes the PLL and MCU specific clock options. The clock setting is provided from the configuration structure.

Parameters

in	*ClockSetting*	Clock setting ID from config structure to be used.

Returns

Command has or has not been accepted.

Return values

E_OK	The driver state allowed the execution of the function and the provided parameter was in range
E_NOT_OK	The driver state did not allowed execution or the parameter was invalid

6.2.6.4 Mcu_SetMode()

```
void Mcu_SetMode (
            Mcu_ModeType McuMode )
```

This function sets the MCU power mode.

This function activates MCU power mode from config structure selected by McuMode parameter. If the driver state is invalid or McuMode is not in range the function will skip changing the mcu mode.

Parameters

in	*McuMode*	MCU mode setting ID from config structure to be set.

Returns

void

6.2.6.5 Mcu_DistributePllClock()

```
Std_ReturnType Mcu_DistributePllClock (
            void  )
```

This function activates the PLL clock to the MCU clock distribution.

Function completes the PLL configuration and then activates the PLL clock to MCU. If the MCU_NO_PLL is TRUE the Mcu_DistributePllClock has to be disabled. The function will not distribute the PLL clock if the driver state does not allow it, or the PLL is not stable.

Returns

Std_ReturnType

Return values

E_OK	Command has been accepted.
E_NOT_OK	Command has not been accepted.

6.2.6.6 Mcu_GetPllStatus()

```
Mcu_PllStatusType Mcu_GetPllStatus (
            void  )
```

This function returns the lock status of the PLL.

The user takes care that the PLL is locked by executing Mcu_GetPllStatus. If the MCU_NO_PLL is TRUE the MCU_GetPllStatus has to return MCU_PLL_STATUS_UNDEFINED. It will also return MCU_PLL_STATUS↩ _UNDEFINED if the driver state was invalid

Returns

Mcu_PllStatusType Provides the lock status of the PLL.

Return values

MCU_PLL_STATUS_UNDEFINED	PLL Status is unknown.
MCU_PLL_LOCKED	PLL is locked.
MCU_PLL_UNLOCKED	PLL is unlocked.

6.2.6.7 Mcu_GetResetReason()

```
Mcu_ResetType Mcu_GetResetReason (
            void  )
```

This function returns the Reset reason.

This routine returns the Reset reason that is read from the hardware.

Returns

Mcu_ResetType Reason of the Reset event.

6.2.6.8 Mcu_GetResetRawValue()

```
Mcu_RawResetType Mcu_GetResetRawValue (
            void  )
```

This function returns the Raw Reset value.

This routine returns the Raw Reset value that is read from the hardware.

Returns

Mcu_RawResetType Description of the returned value.

Return values

uint32	Code of the Raw reset value.

6.2.6.9 Mcu_PerformReset()

```
void Mcu_PerformReset (
            void  )
```

This function performs a microcontroller reset.

This function performs a microcontroller reset by using the hardware feature of the microcontroller. In case the function returns, the user must reset the platform using an alternate reset mechanism

Returns

void

6.2.6.10 Mcu_GetVersionInfo()

```
void Mcu_GetVersionInfo (
            Std_VersionInfoType * versioninfo )
```

This function returns the Version Information for the MCU module.

This function returns the vendor id, module id, major, minor and patch version.

Parameters

in,out	*versioninfo*	A pointer to a variable to store version info.

Returns

void

6.2.6.11 Mcu_GetRamState()

```
Mcu_RamStateType Mcu_GetRamState (
            void  )
```

This function returns the actual state of the RAM.

This function returns if the Ram Status is valid after a reset. The report is get from STCU as a result of MBIST (Memory Built-In Self Tests).

Returns

Mcu_RamStateType Status of the Ram Content.

Return values

MCU_RAMSTATE_INVALID	Ram state is not valid or unknown (default), or the driver state does not allow this call.
MCU_RAMSTATE_VALID	Ram state is valid.

6.2.6.12 Mcu_GetMidrStructure()

```
void Mcu_GetMidrStructure (
            Mcu_MidrReturnType MidrPtr[((uint8) 1U)] )
```

This function returns the value of the MIDR registers.

This function returns the platform dependent Mcu_MidrReturnType structure witch contains the MIDRn registers.

Parameters

in,out	*MidrPtr*	A pointer to a variable to store the Mcu_MidrReturnType structure.

Returns

void

6.2.6.13 Mcu_EmiosConfigureGpren()

```
void Mcu_EmiosConfigureGpren (
            uint8 Module,
            uint8 Value )
```

eMios Global Prescaler Enable.

This function enables or disables the GPREN bit of the EMIOSMCR register of an addressed eMIOS instance.

Precondition

Function requires an execution of Mcu_Init() before it can be used, otherwise it reports error to DET.

Parameters

in	*Module*	MCU_EMIOS_MODULE_0 --> Select eMios 0 MCU_EMIOS_MODULE_1 --> Select eMios 1 MCU_EMIOS_MODULE_2 --> Select eMios 2
in	*Value*	MCU_EMIOS_GPREN_BIT_ENABLE --> Global Prescaler Enabled MCU_EMIOS_GPREN_BIT_DISABLE --> Global Prescaler Disabled

Returns

void

6.2.6.14 Mcu_DisableCmu()

```
void Mcu_DisableCmu (
            Clock_Ip_NameType ClockName )
```

Disable clock monitoring unit.

This function disables the selected clock monitoring unit.

Precondition

Function requires an execution of Mcu_Init() before it can be used.

Parameters

in	*ClockName*	Name of the monitor clock for which CMU must be disabled.

Returns

void

6.2.6.15 Mcu_GetClockFrequency()

```
uint32 Mcu_GetClockFrequency (
            Clock_Ip_NameType ClockName )
```

Return the frequency of a given clock.

This function returns the frequency of a given clock which is request by user.

Precondition

Function requires an execution of Mcu_Init() before it can be used,

Parameters

in	*ClockName*	Name of the monitor clock for which CMU must be disabled.

Returns

uint32

6.2.6.16 Mcu_SleepOnExit()

```
void Mcu_SleepOnExit (
            Mcu_SleepOnExitType SleepOnExit )
```

This function disable/enable SleepOnExit.

Disable/enable Sleep on exit when returning from Handler mode to Thread mode.

Parameters

in	*Mcu_SleepOnExitType*	The value will be configured to SLEEPONEXIT bits. MCU_SLEEP_ON_EXIT_DISABLED - Disable SLEEPONEXIT bit. MCU_SLEEP_ON_EXIT_ENABLED - Enable SLEEPONEXIT bit.

Returns

void

6.2.6.17 Mcu_PmcAeConfig()

```
void Mcu_PmcAeConfig (
            void  )
```

This function configure the Power Management Controller AE. This function configure the Power Management Controller AE of the microcontroller.

Returns

void

6.2.6.18 Mcu_AecResetConfig()

```
void Mcu_AecResetConfig (
            void  )
```

This function configure the Reset config AEC. This function configure the Reset config AEC of the microcontroller.

Returns

void

6.2.6.19 Mcu_ReportDemTimeoutError()

```
void Mcu_ReportDemTimeoutError (
            void  )
```

Reports timeout error to DEM.

Checks if the timeout expired and reports the timeout error to DEM if that is the case.

Parameters

in	*u32Timeout*	Maximum timeout to be waited.

Returns

void.

6.2.6.20 Mcu_ReportDemFxoscError()

```
void Mcu_ReportDemFxoscError (
            void  )
```

Reports incorrect FXOSC configuration to DEM.

Directly reports the FXOSC configuration error to DEM.

Parameters

in	*void.*	

Returns

void.

6.2.6.21 Mcu_ReportDemClockMuxSwitchError()

```
void Mcu_ReportDemClockMuxSwitchError (
            void  )
```

Reports failed clock multiplexer switch to DEM.

Directly reports the clock multiplexer switch error to DEM.

Parameters

in	*void.*	

Returns

void.

6.2.7 Variable Documentation

6.2.7.1 Mcu_pConfigPtr

```
const Mcu_ConfigType* Mcu_pConfigPtr  [extern]
```

Local copy of the pointer to the configuration data.

6.3 Power Ip Driver

6.3.1 Detailed Description

Data Structures

- struct Power_Ip_AEC_ConfigType
  Configuration for AEC. More...
- struct Power_Ip_DCM_GPR_ConfigType
  Configuration for DCM_GPR. More...
- struct Power_Ip_MC_ME_CoreConfigType
  MC_ME Core Configuration. More...
- struct Power_Ip_MC_ME_CofbConfigType
  MC_ME COFB Configuration. More...
- struct Power_Ip_MC_ME_PartitionConfigType
  MC_ME Partition Configuration. More...
- struct Power_Ip_MC_ME_ModeConfigType
  MC_ME IP Configuration. More...
- struct Power_Ip_MC_RGM_ConfigType
  Configuration of MC_RGM hardware IP. More...
- struct Power_Ip_PMC_ConfigType
  Configuration for PMC. More...

Macros

- #define IP_CM_AIRCR
- #define CM_AIRCR_VECTKEY(x)
- #define POWER_IP_FIRST_RESET_REASON_POS
  This macro is used to define the position of the first reset reason.

Types Reference

- typedef MC_RGM_Type Power_Ip_MC_RGM_Type
- typedef uint32 Power_Ip_RawResetType

  The type Mcu_RawResetType specifies the reset reason in raw register format, read from a reset status register.
- typedef uint32 Power_Ip_ModeType

  The Mcu_ModeType specifies the identification (ID) for a MCU mode, configured via configuration structure.
- typedef void(* Power_Ip_ReportErrorsCallbackType) (Power_Ip_ReportErrorType Error, uint8 ErrorCode)

  Power report error callback structure. Implements PowerReportErrorCallbackType_Class.

Enum Reference

- enum Power_Ip_MC_RGM_ResetType

  Reset type to be performed through the Mcu_PerformReset() API.
- enum Power_MC_RGM_StatusType
- enum Power_Ip_PMC_StatusType
- enum Power_Ip_ResetType

  The type Power_Ip_ResetType, represents the different reset that a specified POWER_IP can have.
- enum Power_Ip_PowerModeType

  Power Modes encoding.
- enum Power_Ip_ReportErrorType

  Power ip report error types.

Function Reference

- void Power_Ip_Init (const Power_Ip_HwIPsConfigType *HwIPsConfigPtr)

  Power initialization.
- void Power_Ip_SetMode (const Power_Ip_ModeConfigType *ModeConfigPtr)

  Sets mode.
- Power_Ip_PowerModeType Power_Ip_GetPreviousMode (void)

  This function returns the previous mode.
- void Power_Ip_PerformReset (const Power_Ip_HwIPsConfigType *HwIPsConfigPtr)

  Performs reset.
- Power_Ip_ResetType Power_Ip_GetResetReason (void)

  Returns reset type.
- Power_Ip_RawResetType Power_Ip_GetResetRawValue (void)

  Returns raw reset type.
- void Power_Ip_InstallNotificationsCallback (Power_Ip_ReportErrorsCallbackType ReportErrorsCallback)

  Install report error callback.
- void Power_Ip_EnableSleepOnExit (void)

  The function enable SLEEPONEXIT bit.
- void Power_Ip_DisableSleepOnExit (void)

  The function disable SLEEPONEXIT bit.
- void Power_Ip_StartTimeout (uint32 *StartTimeOut, uint32 *ElapsedTimeOut, uint32 *TimeoutTicksOut, uint32 TimeoutUs)

  Initializes a starting reference point for timeout.
- boolean Power_Ip_TimeoutExpired (uint32 *StartTimeInOut, uint32 *ElapsedTimeInOut, uint32 Timeout↩ Ticks)

  Checks for timeout condition.

6.3.1.1 MISRA-C:2012 violations

6.3.2 Data Structure Documentation

6.3.2.1 struct Power_Ip_AEC_ConfigType

Configuration for AEC.

The Reset generator configuration (AEC_RSTGEN_CFG)

Definition at line 136 of file Power_Ip_AEC_Types.h.

Data Fields

Type	Name	Description
uint16	Rstgencfg	Reset generator configuration (AEC_RSTGEN_CFG).

6.3.2.2 struct Power_Ip_DCM_GPR_ConfigType

Configuration for DCM_GPR.

The power control unit (DCM_GPR) acts as a bridge for mapping the DCM_GPR peripheral to the DCM_GPR address space.

Definition at line 108 of file Power_Ip_DCM_GPR_Types.h.

Data Fields

Type	Name	Description
boolean	DcmGprUnderMcuControl	Specifies whether the the DCM_GPR registers is under MCU control.
uint8	BootMode	The boot mode after exiting Standby mode.
uint32	BootAddress	Cortex-M7_0 base address of vector table.
uint32	ConfigRegister	DCM_GPR configuration register (DCMRWF2)
boolean	GlobalPadkeeping	Global Padkeeping enablement (DCMRWF1[STANBY_IO_CONFIG])

6.3.2.3 struct Power_Ip_MC_ME_CoreConfigType

MC_ME Core Configuration.

This structure contains information for configuring the cores. The definitions for each Core setting within the structure Power_Ip_MC_ME_CoreConfigType shall contain:

- The index of the Core (within its partition).

- The boot address of the Core.
- Power management information (i.e. start or shutdown the Core).

Definition at line 162 of file Power_Ip_MC_ME_Types.h.

Data Fields

Type	Name	Description
boolean	CoreUnderMcuControl	Specifies whether the given core is under MCU control.
uint8	CoreIndex	The index of the core within the partition.
uint32 *	CoreBootAddress	The boot address of the core.
uint32	CorePconfRegValue	The process configuration register value of the core.

6.3.2.4 struct Power_Ip_MC_ME_CofbConfigType

MC_ME COFB Configuration.

This structure contains information for configuring the COFBs (Collection of Functional Blocks). The definitions for each COFB setting within the structure Power_Ip_MC_ME_CofbConfigType shall contain:

- The index of the COFB (within its partition).
- The list of peripherals enable/disable (i.e. the value of the PRTNx_COFBx_CLKEN register).

Definition at line 185 of file Power_Ip_MC_ME_Types.h.

Data Fields

Type	Name	Description
boolean	CofbUnderMcuControl	Specifies whether the given COFB set is under MCU control.
uint8	CofbIndex	The index of the COFB set within the partition.
uint32	CofbClkenRegValue	The clock enable register value of the COFB set.
uint32	CofbBlocksToUpdateMask	Mask containing the COFB blocks to be updated.

6.3.2.5 struct Power_Ip_MC_ME_PartitionConfigType

MC_ME Partition Configuration.

This structure contains information for configuring the Partitions. The definitions for each Partition setting within the structure Power_Ip_MC_ME_PartitionConfigType shall contain:

- The index of the Partition.

- The configuration settings for the COFBs contained within the Partition.
- The configuration settings for the Cores contained within the Partition.

Definition at line 209 of file Power_Ip_MC_ME_Types.h.

Data Fields

Type	Name	Description
boolean	PartitionUnderMcuControl	Specifies whether the given partition is under MCU control.
boolean	PartitionPowerUnderMcuControl	Specifies whether the given partition's power management is under MCU control.
uint8	PartitionIndex	The index of the partition.
uint32	PartitionPconfRegValue	The process configuration register value of the partition.
uint32	PartitionTriggerMask	Mask containing the Partition triggers (PCE/OSSE/etc) to be updated.
uint8	NumberOfCofbs	Number of COFBs within the partition.
const Power_Ip_MC_ME_CofbConfigType(∗	ArrayPartitionCofbConfigPtr)[]	The configuration of the COFBs.
uint8	NumberOfCores	Number of cores within the partition.
const Power_Ip_MC_ME_CoreConfigType(∗	ArrayPartitionCoreConfigPtr)[]	The configuration of the cores.

6.3.2.6 struct Power_Ip_MC_ME_ModeConfigType

MC_ME IP Configuration.

This structure contains information for configuring the entire MC_ME IP.

Definition at line 242 of file Power_Ip_MC_ME_Types.h.

Data Fields

Type	Name	Description
uint32	MainCoreIdRegValue	MC_ME Main Core ID register.
const Power_Ip_MC_ME_PartitionConfigType(∗	ArrayPartitionConfigPtr)[((uint8) [2U]]	MC_ME Mode Partition Settings.

6.3.2.7 struct Power_Ip_MC_RGM_ConfigType

Configuration of MC_RGM hardware IP.

This data configuration is set at module initialization phase.

Definition at line 131 of file Power_Ip_MC_RGM_Types.h.

Data Fields

Type	Name	Description
Power_Ip_MC_RGM_ResetType	ResetType	RESET type: Functional vs Destructive.
uint32	FuncResetOpt	Enable/Disable functional reset sources (RGM_FERD register).
uint32	FesThresholdReset	Functional Reset Escalation Threshold (RGM_FRET register).
uint32	DesThresholdReset	Destructive Reset Escalation Threshold (RGM_DRET register).

6.3.2.8 struct Power_Ip_PMC_ConfigType

Configuration for PMC.

The power control unit (PMC) acts as a bridge for mapping the PMC peripheral to the PMC address space.

Definition at line 222 of file Power_Ip_PMC_Types.h.

Data Fields

Type	Name	Description
uint32	ConfigRegister	PMC configuration register (CONFIG)
uint32	PmcAeConfig	Trimming Register (PMC_AE_CONFIG).
uint32	PmcAeMonitor	Trimming Register (PMC_AE_MONITOR).

6.3.3 Macro Definition Documentation

6.3.3.1 IP_CM_AIRCR

```
#define IP_CM_AIRCR
```

CM7 AIRCR base pointer

Definition at line 123 of file Power_Ip_CortexM7.h.

6.3.3.2 CM_AIRCR_VECTKEY

```
#define CM_AIRCR_VECTKEY(
             x )
```

Reg_eSys_CortexM_H_REF_1 A function should be used in preference to a function-like macro where they are interchangeable.

Definition at line 133 of file Power_Ip_CortexM7.h.

6.3.3.3 POWER_IP_FIRST_RESET_REASON_POS

```
#define POWER_IP_FIRST_RESET_REASON_POS
```

This macro is used to define the position of the first reset reason.

Definition at line 340 of file Power_Ip_Specific.h.

6.3.4 Types Reference

6.3.4.1 Power_Ip_MC_RGM_Type

```
typedef MC_RGM_Type Power_Ip_MC_RGM_Type
```

MC_RGM - Register Layout Typedef

Definition at line 433 of file Power_Ip_Specific.h.

6.3.4.2 Power_Ip_RawResetType

```
typedef uint32 Power_Ip_RawResetType
```

The type Mcu_RawResetType specifies the reset reason in raw register format, read from a reset status register.

The type shall be uint8, uint16 or uint32 based on best performance.

Destructive and Functional Reset Events Log.

Definition at line 251 of file Power_Ip_Types.h.

6.3.4.3 Power_Ip_ModeType

```
typedef uint32 Power_Ip_ModeType
```

The Mcu_ModeType specifies the identification (ID) for a MCU mode, configured via configuration structure.

The type shall be uint8, uint16 or uint32.

Definition at line 260 of file Power_Ip_Types.h.

6.3.4.4 Power_Ip_ReportErrorsCallbackType

```
typedef void(* Power_Ip_ReportErrorsCallbackType) (Power_Ip_ReportErrorType Error, uint8 ErrorCode)
```

Power report error callback structure. Implements PowerReportErrorCallbackType_Class.

Definition at line 309 of file Power_Ip_Types.h.

6.3.5 Enum Reference

6.3.5.1 Power_Ip_MC_RGM_ResetType

```
enum Power_Ip_MC_RGM_ResetType
```

Reset type to be performed through the Mcu_PerformReset() API.

Destructive Reset:

- Flash is always reset, so an updated value of the option bits is reloaded in volatile registers outside of the Flash array.
- Trimming is lost.
- STCU is reset and configured BISTs are executed Functional Reset:
- Starts the reset sequence from PHASE1 or from PHASE3.
- The volatile registers are not reset; in case of a reset event, the trimming is maintained.
- No BISTs shall be executed after functional resets.

Enumerator

MCU_FUNC_RESET	Functional Reset type.
MCU_DEST_RESET	Destructive Reset type.

Definition at line 114 of file Power_Ip_MC_RGM_Types.h.

6.3.5.2 Power_MC_RGM_StatusType

enum Power_MC_RGM_StatusType

Enumerator

POWER_MC_RGM_UNINIT	The MC_RGM driver is uninitialized.
POWER_MC_RGM_INIT	The MC_RGM driver is initialized.

Definition at line 162 of file Power_Ip_MC_RGM_Types.h.

6.3.5.3 Power_Ip_PMC_StatusType

enum Power_Ip_PMC_StatusType

Enumerator

PMC_UNINIT	The PMC driver is uninitialized.
PMC_INIT	The PMC driver is initialized.

Definition at line 238 of file Power_Ip_PMC_Types.h.

6.3.5.4 Power_Ip_ResetType

enum Power_Ip_ResetType

The type Power_Ip_ResetType, represents the different reset that a specified POWER_IP can have.

The POWER_IP shall provide at least the values MCU_POWER_ON_RESET and MCU_RESET_UNDEFINED for the enumeration Power_Ip_ResetType.

Enumerator

MCU_POWER_ON_RESET	Power on reset event. RGM_DES[F_DR0].
MCU_FCCU_FTR_RESET	Non-critical supply presence detector fail. RGM_DES[F_DR1].
MCU_STCU_URF_RESET	FCCU failure to react. RGM_DES[F_DR3].
MCU_MC_RGM_FRE_RESET	STCU unrecoverable fault. RGM_DES[F_DR4].

Enumerator

MCU_FXOSC_FAIL_RESET	Functional reset escalation. RGM_DES[F_DR6].
MCU_PLL_LOL_RESET	FXOSC failure. RGM_DES[F_DR8].
MCU_CORE_CLK_FAIL_RESET	CORE_PLL and related DFS loss of lock. RGM_DES[F_DR9].
MCU_AIPS_PLAT_CLK_FAIL_RESET	PERIPH_PLL and related DFS loss of lock. RGM_DES[F_DR10].
MCU_HSE_CLK_FAIL_RESET	DDR_PLL loss of lock. RGM_DES[F_DR11].
MCU_SYS_DIV_FAIL_RESET	ACCEL_PLL loss of lock. RGM_DES[F_DR12].
MCU_CM7_CORE_CLK_FAIL_RESET	CM7_CORE_CLK failure.
MCU_HSE_TMPR_RST_RESET	XBAR_DIV3_CLK failure. RGM_DES[F_DR13].
MCU_HSE_SNVS_RST_RESET	Life-cycle error. RGM_DES[F_DR16].
MCU_SW_DEST_RESET	HSE SNVS tamper detected. RGM_DES[F_DR17].
MCU_DEBUG_DEST_RESET	HSE SWT timeout. RGM_DES[F_DR18].
MCU_F_EXR_RESET	Software destructive reset. RGM_DES[F_DR30].
MCU_FCCU_RST_RESET	FCCU Reset Reaction. RGM_FES[F_FR3].
MCU_ST_DONE_RESET	Self-Test Done. RGM_FES[F_FR4].
MCU_SWT0_RST_RESET	SWT0 Timeout. RGM_FES[F_FR6].
MCU_SWT1_RST_RESET	SWT1 Timeout. RGM_FES[F_FR6].
MCU_SWT2_RST_RESET	SWT2 Timeout.
MCU_JTAG_RST_RESET	HSE Memory ECC Error. RGM_FES[F_FR18].
MCU_SWT3_RST_RESET	SWT3 Timeout.
MCU_PLL_AUX_RESET	PLL_AUX_CLK failure.
MCU_HSE_SWT_RST_RESET	HSE Boot Failure Error. RGM_FES[F_FR20].
MCU_HSE_BOOT_RST_RESET	HSE M7 Core Lock. RGM_FES[F_FR21].
MCU_SW_FUNC_RESET	Software functional reset. RGM_FES[F_FR30].
MCU_DEBUG_FUNC_RESET	Debug functional reset. RGM_FES[F_FR31].
MCU_WAKEUP_REASON	Wake-up event detected.
MCU_NO_RESET_REASON	No reset reason found.
MCU_MULTIPLE_RESET_REASON	More than one reset events are logged except "Power on event".
MCU_RESET_UNDEFINED	Undefined reset source.

Definition at line 353 of file Power_Ip_Specific.h.

6.3.5.5 Power_Ip_PowerModeType

```
enum Power_Ip_PowerModeType
```

Power Modes encoding.

Supported power modes for the MCU.

Enumerator

POWER_IP_DEST_RESET_MODE	Destructive Reset Mode.
POWER_IP_FUNC_RESET_MODE	Functional Reset Mode.
POWER_IP_RESET_MODE	Any reset mode. Used when the particular type of reset doesn't matter.
POWER_IP_CORE_WARM_RESET_MODE	Core Warm Reset Mode.
POWER_IP_CORE_STANDBY_MODE	Core Standby Mode.
POWER_IP_SOC_PREPARE_STANDBY_MODE	Prepare Standby Mode.
POWER_IP_SOC_STANDBY_MODE	StandBy Mode.
POWER_IP_STANDBY_MODE	Prepare Standby and StandBy Mode.
POWER_IP_SOC_PREPARE_FAST_↩ STANDBY_MODE	Prepare Fast Standby Mode.
POWER_IP_SOC_FAST_STANDBY_MODE	Fast StandBy Mode.
POWER_IP_FAST_STANDBY_MODE	Prepare Fast Standby and Fast StandBy Mode.
POWER_IP_RUN_MODE	Run Mode.

Definition at line 223 of file Power_Ip_Types.h.

6.3.5.6 Power_Ip_ReportErrorType

enum Power_Ip_ReportErrorType

Power ip report error types.

Enumerator

POWER_IP_REPORT_TIMEOUT_ERROR	Report Timeout Error.
POWER_IP_ISR_ERROR	Notification Error.
POWER_IP_PMC_ERROR	Notification PMC.
POWER_IP_ISR_VOLTAGE_HVD_VDDINT_↩ DETECT	Report the Voltage on VDDINT is high-voltage detected.
POWER_IP_ISR_VOLTAGE_HVD_15_DETECT	Report the Voltage on VDD15 is high-voltage detected.
POWER_IP_ISR_VOLTAGE_HVD_VDD_↩ DETECT	Report the Voltage on VDD is high-voltage detected.
POWER_IP_ISR_VOLTAGE_LVD_VDDC_↩ DETECT	Report the Voltage on VDDC is low-voltage detected.
POWER_IP_ISR_VOLTAGE_LVD_VLS_↩ DETECT	Report the Voltage on VLS is low-voltage detected.

Definition at line 293 of file Power_Ip_Types.h.

6.3.6 Function Reference

6.3.6.1 Power_Ip_Init()

```
void Power_Ip_Init (
            const Power_Ip_HwIPsConfigType * HwIPsConfigPtr )
```

Power initialization.

This function power initialization

Parameters

in	*HwIPsConfigPtr*	power initialization configuration.

Returns

void

6.3.6.2 Power_Ip_SetMode()

```
void Power_Ip_SetMode (
            const Power_Ip_ModeConfigType * ModeConfigPtr )
```

Sets mode.

This function sets mode.

Parameters

in	*ModeConfigPtr*	power set mote configuration.

Returns

void

6.3.6.3 Power_Ip_GetPreviousMode()

```
Power_Ip_PowerModeType Power_Ip_GetPreviousMode (
            void  )
```

This function returns the previous mode.

This function returns the previous mode.

Returns

Status of the previous mode.

6.3.6.4 Power_Ip_PerformReset()

```
void Power_Ip_PerformReset (
            const Power_Ip_HwIPsConfigType * HwIPsConfigPtr )
```

Performs reset.

This function performs reset.

Parameters

in	*HwIPsConfigPtr*	reset initialization configuration.

Returns

void

6.3.6.5 Power_Ip_GetResetReason()

```
Power_Ip_ResetType Power_Ip_GetResetReason (
            void  )
```

Returns reset type.

This function returns reset type.

Returns

Power_Ip_ResetType Reset type

6.3.6.6 Power_Ip_GetResetRawValue()

```
Power_Ip_RawResetType Power_Ip_GetResetRawValue (
            void  )
```

Returns raw reset type.

This function returns raw reset type.

Returns

Power_Ip_RawResetType Raw reset type

6.3.6.7 Power_Ip_InstallNotificationsCallback()

```
void Power_Ip_InstallNotificationsCallback (
            Power_Ip_ReportErrorsCallbackType ReportErrorsCallback )
```

Install report error callback.

This function installs a callback for reporting errors from power driver

Parameters

in	*ReportErrorsCallback*	Callback to be installed.

Returns

void

6.3.6.8 Power_Ip_EnableSleepOnExit()

```
void Power_Ip_EnableSleepOnExit (
            void  )
```

The function enable SLEEPONEXIT bit.

The function enable SLEEPONEXIT bit.

Returns

void

6.3.6.9 Power_Ip_DisableSleepOnExit()

```
void Power_Ip_DisableSleepOnExit (
            void  )
```

The function disable SLEEPONEXIT bit.

The function disable SLEEPONEXIT bit.

Returns

void

6.3.6.10 Power_Ip_StartTimeout()

```
void Power_Ip_StartTimeout (
             uint32 * StartTimeOut,
             uint32 * ElapsedTimeOut,
             uint32 * TimeoutTicksOut,
             uint32 TimeoutUs )
```

Initializes a starting reference point for timeout.

Parameters

out	*StartTimeOut*	The starting time from which elapsed time is measured
out	*ElapsedTimeOut*	The elapsed time to be passed to Power_Ip_TimeoutExpired
out	*TimeoutTicksOut*	The timeout value (in ticks) to be passed to Power_Ip_TimeoutExpired
in	*TimeoutUs*	The timeout value (in microseconds)

6.3.6.11 Power_Ip_TimeoutExpired()

```
boolean Power_Ip_TimeoutExpired (
            uint32 * StartTimeInOut,
            uint32 * ElapsedTimeInOut,
            uint32 TimeoutTicks )
```

Checks for timeout condition.

Parameters

in,out	*StartTimeInOut*	The starting time from which elapsed time is measured
in,out	*ElapsedTimeInOut*	The accumulated elapsed time from the starting time reference
in	*TimeoutTicks*	The timeout limit (in ticks)

6.4 Ram Ip Driver

6.4.1 Detailed Description

Data Structures

- struct Ram_Ip_RamConfigType

  Definition of a RAM section within the configuration structure. The definitions for each RAM section within the structure Ram_Ip_ConfigType shall contain: More...

Types Reference

- typedef void(* Ram_Ip_ReportErrorsCallbackType) (Ram_Ip_RamReportErrorType Error, uint8 Error↩ Code)

  Ram report error callback structure. Implements RamReportErrorCallbackType_Class.
- typedef uint32 Ram_Ip_RamSectionType

The Ram_Ip_RamSectionType specifies the identification (ID) for a RAM section, configured via the configuration structure. The type shall be uint8, uint16 or uint32, based on best performance.

- typedef uint32 Ram_Ip_RamIndexType

  The Ram_Ip_RamIndexType specifies the variable for indexing RAM sections. The type shall be uint8, uint16 or uint32, based on best performance.
- typedef Ram_Ip_uintPtrType Ram_Ip_RamSizeType

  The Ram_Ip_RamSizeType specifies the RAM section size. The type shall be uint8, uint16 or uint32, based on best performance.
- typedef uint32 Ram_Ip_RamWriteSizeType

  The Ram_Ip_RamWriteSizeType specifies the RAM section write size. The type shall be uint8, uint16 or uint32, based on best performance.

Enum Reference

- enum Ram_Ip_RamReportErrorType

  Ram ip report error types.
- enum Ram_Ip_RamStateType

  Ram State of the microcontroller.
- enum Ram_Ip_StatusType

  Ram ip status return codes.

Function Reference

- Ram_Ip_StatusType Ram_Ip_InitRamSection (const Ram_Ip_RamConfigType *RamConfigPtr)

  Initializes RAM section.
- Ram_Ip_RamStateType Ram_Ip_GetRamState (void)

  Returns RAM state.
- void Ram_Ip_InstallNotificationsCallback (Ram_Ip_ReportErrorsCallbackType ReportErrorsCallback)

  Install report error callback. This function installs a callback for reporting errors from Ram driver.

6.4.2 Data Structure Documentation

6.4.2.1 struct Ram_Ip_RamConfigType

Definition of a RAM section within the configuration structure. The definitions for each RAM section within the structure Ram_Ip_ConfigType shall contain:

- RAM section base address
- Section size
- Data pre-setting to be initialized
- RAM write size

Definition at line 199 of file Ram_Ip_Types.h.

Data Fields

Type	Name	Description
Ram_Ip_RamSectionType	RamSectorId	The ID for Ram Sector configuration.
uint8(*	RamBaseAddrPtr)[1U]	RAM section base address.
Ram_Ip_RamSizeType *	RamSize	RAM section size.
uint64	RamDefaultValue	RAM default value for initialization.
Ram_Ip_RamWriteSizeType	RamWriteSize	RAM section write size.

6.4.3 Types Reference

6.4.3.1 Ram_Ip_ReportErrorsCallbackType

```
typedef void(* Ram_Ip_ReportErrorsCallbackType) (Ram_Ip_RamReportErrorType Error, uint8 ErrorCode)
```

Ram report error callback structure. Implements RamReportErrorCallbackType_Class.

Definition at line 133 of file Ram_Ip_Types.h.

6.4.3.2 Ram_Ip_RamSectionType

```
typedef uint32 Ram_Ip_RamSectionType
```

The Ram_Ip_RamSectionType specifies the identification (ID) for a RAM section, configured via the configuration structure. The type shall be uint8, uint16 or uint32, based on best performance.

Definition at line 141 of file Ram_Ip_Types.h.

6.4.3.3 Ram_Ip_RamIndexType

```
typedef uint32 Ram_Ip_RamIndexType
```

The Ram_Ip_RamIndexType specifies the variable for indexing RAM sections. The type shall be uint8, uint16 or uint32, based on best performance.

Definition at line 148 of file Ram_Ip_Types.h.

6.4.3.4 Ram_Ip_RamSizeType

```
typedef Ram_Ip_uintPtrType Ram_Ip_RamSizeType
```

The Ram_Ip_RamSizeType specifies the RAM section size. The type shall be uint8, uint16 or uint32, based on best performance.

Definition at line 155 of file Ram_Ip_Types.h.

6.4.3.5 Ram_Ip_RamWriteSizeType

```
typedef uint32 Ram_Ip_RamWriteSizeType
```

The Ram_Ip_RamWriteSizeType specifies the RAM section write size. The type shall be uint8, uint16 or uint32, based on best performance.

Definition at line 162 of file Ram_Ip_Types.h.

6.4.4 Enum Reference

6.4.4.1 Ram_Ip_RamReportErrorType

```
enum Ram_Ip_RamReportErrorType
```

Ram ip report error types.

Enumerator

RAM_IP_REPORT_TIMEOUT_ERROR	Report Timeout Error.

Definition at line 109 of file Ram_Ip_Types.h.

6.4.4.2 Ram_Ip_RamStateType

```
enum Ram_Ip_RamStateType
```

Ram State of the microcontroller.

This is the Ram State data type returned by the function Mcu_GetRamState() of the Mcu module.

Enumerator

RAM_IP_RAMSTATE_INVALID	RAM content is not valid or unknown (default).
RAM_IP_RAMSTATE_VALID	RAM content is valid.

Definition at line 169 of file Ram_Ip_Types.h.

6.4.4.3 Ram_Ip_StatusType

```
enum Ram_Ip_StatusType
```

Ram ip status return codes.

This is the Ram State data type returned by the function Mcu_GetRamState() of the Mcu module.

Enumerator

RAM_IP_STATUS_OK	RAM_IP Ok status
RAM_IP_STATUS_NOT_OK	RAM_IP Not ok status
RAM_IP_STATUS_UNDEFINED	RAM_IP Status is unknown

Definition at line 182 of file Ram_Ip_Types.h.

6.4.5 Function Reference

6.4.5.1 Ram_Ip_InitRamSection()

```
Ram_Ip_StatusType Ram_Ip_InitRamSection (
            const Ram_Ip_RamConfigType * RamConfigPtr )
```

Initializes RAM section.

This function initializes RAM section.

Parameters

in	*RamConfigPtr*	Ram section configuration.

Returns

Ram_Ip_StatusType Ram status

6.4.5.2 Ram_Ip_GetRamState()

```
Ram_Ip_RamStateType Ram_Ip_GetRamState (
            void  )
```

Returns RAM state.

This function returns RAM section.

Returns

Ram_Ip_RamStateType Ram state

6.4.5.3 Ram_Ip_InstallNotificationsCallback()

```
void Ram_Ip_InstallNotificationsCallback (
            Ram_Ip_ReportErrorsCallbackType ReportErrorsCallback )
```

Install report error callback. This function installs a callback for reporting errors from Ram driver.

Parameters

in	*ReportErrorsCallback*	Callback to be installed.

Returns

void

How to Reach Us:

Home Page:
nxp.com

Web Support:
nxp.com/support