User Manual

for S32K1_S32M24x MCU Driver

Document Number: UM2MCUASRR21-11 Rev0000R2.0.0 Rev. 1.0

Chapter 1

Revision History

Revision	Date	Author	Description
1.0	04.08.2023	NXP RTD Team	S32K1_S32M24X Real-Time Drivers AUTOSAR 4.4 & R21-11 Version 2.0.0

Chapter 2

Introduction

This User Manual describes NXP Semiconductors' AUTOSAR Mcu Driver for S32K1_S32M24x.

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:

- s32k116_qfn32
- s32k116_lqfp48
- s32k118_lqfp48
- s32k118_lqfp64
- s32k142_lqfp48
- s32k142_lqfp64
- s32k142_lqfp100
- s32k142w_lqfp48

- s32k142w_lqfp64
- s32k144_lqfp48
- s32k144_lqfp64 / MWCT1014S_lqfp64
- s32k144_lqfp100 / MWCT1014S_lqfp100
- s32k144_mapbga100
- s32k144w_lqfp48
- s32k144w_lqfp64
- s32k146_lqfp64
- s32k146_lqfp100 / MWCT1015S_lqfp100
- s32k146_mapbga100 / MWCT1015S_mapbga100
- s32k146_lqfp144
- s32k148_lqfp100
- s32k148_mapbga100 / MWCT1016S_mapbga100
- s32k148_lqfp144
- s32k148_lqfp176
- s32m241_lqfp64
- s32m242_lqfp64
- s32m243_lqfp64
- s32m244_lqfp64

All of the above microcontroller devices are collectively named as S32K1_S32M24X. 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 Release R21-11
2	S32K1xx Reference Manual	S32K1xx Series Reference Manual, Rev. 14, 09/2021
3	S32K1xx Data Sheet	S32K1xx Data Sheet, Rev. 14, 08/2021
4	S32M24x Reference Manual	S32M24x Reference Manual, Rev. 2 Draft A, 05/2023
5	S32M24x Data Sheet	Supports S32M24x and S32M27x, Rev. 3 DraftA, 05/2023
6	S32M244 Errata Document	S32M244_P64A+P73G, Rev. 0
6	S32M242 Errata Document	S32M242_N33V+P73G, Rev. 0, 6/2023
7	S32K116 Errata Document	S32K116_0N96V Rev. 22/OCT/2021
8	S32K118 Errata Document	S32K118_0N97V Rev. 22/OCT/2021
9	S32K142 Errata Document	S32K142_0N33V Rev. 22/OCT/2021
10	S32K144 Errata Document	S32K144_0N57U Rev. 22/OCT/2021
11	S32K144W Errata Document	S32K144W_0P64A Rev. 22/OCT/2021
12	S32K146 Errata Document	S32K146_0N73V Rev. 22/OCT/2021
13	S32K148 Errata Document	S32K148_0N20V Rev. 22/OCT/2021

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 S32K1_S32M24x 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 (CMU,PCC,PRAMC,SIM,SCG)
2. Power IPs (PMC,RCM,SIM,SMC,AE_PMC,AEC)
3. Ram IPs (PRAMC)

3.4 Deviations from Requirements

The driver deviates from the AUTOSAR Mcu Driver Software Specification in some places.

The table Mcu Requirements Deviations identifies the AUTOSAR requirements that are not fully implemented, implemented differently, or out of scope for the Mcu Driver.

The table Status Column Description provides the "Status" column description.

3.4.1 Status Column Description

Table 3.1 Status Column Description Areas

Term	Definition
N/S	Not In Scope
N/I	Not Implemented
N/F	Not Fully Implemented

3.4.2 Mcu Requirements Deviations

Table 3.3 Mcu Requirements Deviations

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.

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. (BSW12336)	The function Mcu_Distribute↩PllClock will change the Mcu hardware. The clock switching to PLL is not completed by Mcu_InitClock.
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. (BSW12125, BSW12461)	There is a separate plug-in that will cover shared ip's.
SWS_Mcu_00257	N/S	Fail criteria for MCU_E_CL↩OCK_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_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.	For S32K3XX,S32K1, CLOCK IP can't be supported because it will break compatibility.

3.5 Driver Limitations

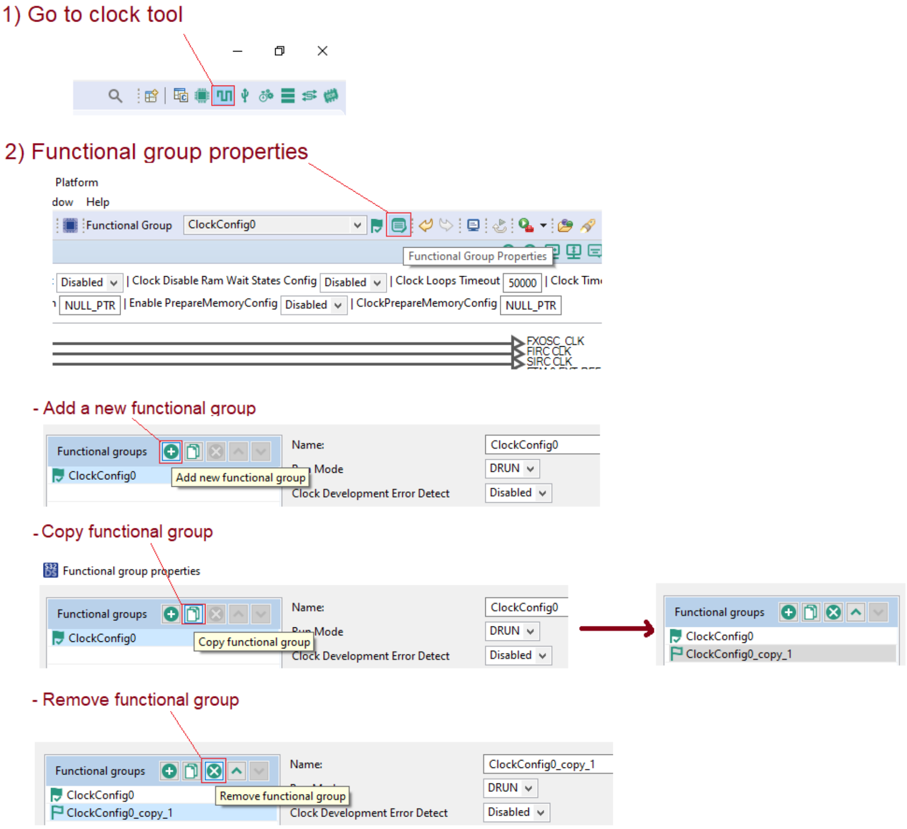
- Can't disable FIRC clock.
- The notifications did not support in IPL.
- In IPL, the interrupt handler function for PMC and CMU are implemented separately: an interrupt handler function for only PMC and a similar one for CMU. Interrupt hander function that controls both CMU and PMC isn't supported.
- Errata e011063 SMC: An asynchronous wakeup event during VLPS mode entry may result in possible system hang scenario (exist only on S32K14x).

- Errata e011114 and e050383(exist on S32K144w, S32M24x) SMC: invalid data might be fetched while accessing Flash in VLP modes.
- The power mode change notification function will not work without a previously initialized Clock driver. If done otherwise the system clock will not be updated correctly when power mode is changed.

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.



1) Go to clock tool

2) Functional group properties

- Add a new functional group

- Copy functional group

- Remove functional group

Functional Group: ClockConfig0

Disabled | Clock Disable Ram Wait States Config: Disabled | Clock Loops Timeout: 50000 | Clock Tim

NULL_PTR | Enable PrepareMemoryConfig: Disabled | ClockPrepareMemoryConfig: NULL_PTR

Name: ClockConfig0

Run Mode: DRUN

Clock Development Error Detect: Disabled

Name: ClockConfig0_copy_1

Figure 3.1 Clock tool snapshot for Functional Group Properties

- For reconfiguring the PLLs using Mcu_InitClock and Mcu_DistributePllClock the peripherals that are clocked using the PLL that needs to be reconfigured should be turned OFF using Mcu_SetMode to transition in a mode where that peripheral is OFF.

- 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.
- If you want to use peripheral module, you must enable clock gate corresponding to the module in Mcu↩ PeripheralClockConfig container.

1. Mcu_Init
2. Mcu_InitClock (This function is only called in Run mode.)
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

3.6.2 Specific clock use case:

- The protocol clock of FlexCan has two clock source SOSC and SYS_CLK, which control by CAN.CTRL1[C↩ LKSRC]. When SOSC is selected, SYS_CLK must be greater than 1.5x the protocol clock. When SYS_CLK is selected, the ratio of SYS_CLK and FLEXCAN_CLK can be 1:1.
- Maximum supported clock out frequency for this device is 20 MHz. There is no restrict corresponding to this range in EB Tresos due to frequency of CLKOUT is depended on working mode.

3.6.3 MCU Mode Management

- NA

3.6.4 MCU RAM Configuration

- NA

3.7 Runtime errors

The driver generates the following DET errors at runtime.

Table 3.4 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_RAMSECTI↩ ON	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_INITIALI↩ ZED	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 3.6 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 McuVeryLowPowerStopAbortNotification
    - Parameter McuPerformResetCallout
    - Parameter McuCmuNotification
    - Parameter McuCmuErrorIsrUsed
    - Parameter McuErrorIsrNotification
    - Parameter McuDisableRcmInit
    - Parameter McuDisablePmcInit
    - Parameter McuDisableSmcInit
    - Parameter McuEnableModeChangeNotification
    - Parameter McuTimeoutMethod
    - Parameter McuRegisterValuesOptimization
    - Reference McuEcucPartitionRef
  - Container McuDebugConfiguration
    - Parameter McuDisableDemReportErrorStatus
    - Parameter McuGetMidrStructureApi
    - Parameter McuDisableCmuApi
    - Parameter McuEnablePeripheralCMU
    - Parameter McuSRAMRetentionConfigApi

  - Parameter McuGetClockFrequencyApi
  - Parameter McuGetPowerModeStatetApi
  - Parameter McuPmcAeConfigApi
  - Parameter McuAecResetConfigApi
- 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 McuRTC_CLKINFrequencyHz
  - Parameter McuTCLK0_REF_CLKFrequencyHz
  - Parameter McuTCLK1_REF_CLKFrequencyHz
  - Parameter McuTCLK2_REF_CLKFrequencyHz
  - Parameter McuClockSrcFailureNotification
  - Container McuAllowedModes
    - Parameter McuAllowHighSpeedRunMode
    - Parameter McuAllowVeryLowPowerModes
  - Container McuClockSettingConfig
    - Parameter McuClockSettingId
    - Parameter McuSysClockUnderMcuControl
    - Parameter McuScgClkOutSelect
    - Container McuRunClockConfig
    - Parameter McuPreDivSystemClockFrequency
    - Parameter McuCoreClockFrequency
    - Parameter McuSystemClockFrequency
    - Parameter McuBusClockFrequency
    - Parameter McuFlashClockFrequency
    - Parameter McuSystemClockSwitch
    - Parameter McuCoreClockDivider
    - Parameter McuBusClockDivider
    - Parameter McuSlowClockDivider
    - Parameter McuScgClkOutFrequency
    - Container McuVlprClockConfig
    - Parameter McuPreDivSystemClockFrequency

- Parameter McuCoreClockFrequency
- Parameter McuSystemClockFrequency
- Parameter McuBusClockFrequency
- Parameter McuFlashClockFrequency
- Parameter McuSystemClockSwitch
- Parameter McuCoreClockDivider
- Parameter McuBusClockDivider
- Parameter McuSlowClockDivider
- Parameter McuScgClkOutFrequency
- Container McuHsrunClockConfig
- Parameter McuPreDivSystemClockFrequency
- Parameter McuCoreClockFrequency
- Parameter McuSystemClockFrequency
- Parameter McuBusClockFrequency
- Parameter McuFlashClockFrequency
- Parameter McuSystemClockSwitch
- Parameter McuCoreClockDivider
- Parameter McuBusClockDivider
- Parameter McuSlowClockDivider
- Parameter McuScgClkOutFrequency
- Container McuSystemOSCClockConfig
- Parameter McuSOSCUnderMcuControl
- Parameter McuSOSCFrequency
- Parameter McuSOSCDiv2Frequency
- Parameter McuSOSCDiv1Frequency
- Parameter McuSOSCEnable
- Parameter McuSOSCClockMonitorResetEnable
- Parameter McuSOSCClockMonitorEnable
- Parameter McuSOSCDiv2
- Parameter McuSOSCDiv1
- Parameter McuSOSCRangeSelect
- Parameter McuSOSCHighGainOscillatorSelect
- Parameter McuSOSCExternalReferenceSelect
- Container McuSIRCClockConfig
- Parameter McuSIRCUnderMcuControl
- Parameter McuSIRCFrequency
- Parameter McuSIRCDiv2Frequency
- Parameter McuSIRCDiv1Frequency
- Parameter McuSIRCEnable
- Parameter McuSIRCLowPowerEnable
- Parameter McuSIRCStopEnable
- Parameter McuSIRCDiv2
- Parameter McuSIRCDiv1
- Parameter McuSIRCRangeSelect
- Container McuFIRCClockConfig
- Parameter McuFIRCUnderMcuControl
- Parameter McuFIRCFrequency

- Parameter McuFIRCDiv2Frequency
- Parameter McuFIRCDiv1Frequency
- Parameter McuFIRCEnable
- Parameter McuFIRCRegulatorEnable
- Parameter McuFIRCDiv2
- Parameter McuFIRCDiv1
- Parameter McuFIRCRangeSelect
- Container McuSystemPll
- Parameter McuSystemPllUnderMcuControl
- Parameter McuSPLLFrequency
- Parameter McuSPLLDiv2Frequency
- Parameter McuSPLLDiv1Frequency
- Parameter McuSPLLEnable
- Parameter McuSPLLClockMonitorResetEnable
- Parameter McuSPLLClockMonitorEnable
- Parameter McuSPLLDiv2
- Parameter McuSPLLDiv1
- Parameter McuSPLLInputClkPreDivider
- Parameter McuSPLLReferenceFrequency
- Parameter McuSPLLInputFrequency
- Parameter McuSPLLMultiplier
- Parameter McuSPLLSelectSourceClock
- Container McuSIMClockConfig
- Parameter McuSIMUnderMcuControl
- Parameter McuEIMClockGatingEnable
- Parameter McuERMClockGatingEnable
- Parameter McuDMAClockGatingEnable
- Parameter McuMPUClockGatingEnable
- Parameter McuMSCMClockGatingEnable
- Parameter McuGPIOClockGatingEnable
- Container McuSimChipConfiguration
- Parameter McuDebugTraceDividerEnable
- Parameter McuTRACECLKDivider
- Parameter McuTRACECLKFraction
- Parameter McuTRACECLKSelect
- Parameter McuCLKOUTEnable
- Parameter McuCLKOUTDivider
- Parameter McuCLKOUTSelect
- Container McuSimLpoConfiguration
- Parameter McuRTCClkSelect
- Parameter McuLPOClkSelect
- Parameter McuLPO_32KClockEnable
- Parameter McuLPO_1KClockEnable
- Container McuSimFtmConfiguration
- Parameter McuFTM3ExternalClockPinSelect
- Parameter McuFTM2ExternalClockPinSelect
- Parameter McuFTM1ExternalClockPinSelect

  - Parameter McuFTM0ExternalClockPinSelect
  - Parameter McuFTM7ExternalClockPinSelect
  - Parameter McuFTM6ExternalClockPinSelect
  - Parameter McuFTM5ExternalClockPinSelect
  - Parameter McuFTM4ExternalClockPinSelect
  - Container McuClkMonitor
  - Container McuClkMonitor_0
  - Parameter McuClockMonitorUnderMcuControl
  - Parameter McuClkMonitorEn
  - Parameter McuCmuName
  - Parameter McuAsyncFHHInterruptEn
  - Parameter McuAsyncFLLInterruptEn
  - Container McuClkMonitor_1
  - Parameter McuClockMonitorUnderMcuControl
  - Parameter McuClkMonitorEn
  - Parameter McuCmuName
  - Parameter McuSyncFHHInterruptEn
  - Parameter McuSyncFLLInterruptEn
  - Container McuPeripheralClockConfig
  - Parameter McuPeripheralClockUnderMcuControl
  - Parameter McuPerName
  - Parameter McuPeripheralClockEnable
  - Parameter McuPeripheralClockSelect
  - Parameter McuPeripheralClockDivider
  - Parameter McuPeripheralFractionalDivider
  - Parameter McuPeripheralClockFrequency
  - Container McuClockReferencePoint
  - Parameter McuClockReferencePointFrequency
  - Parameter McuClockFrequencySelect
- Container McuDemEventParameterRefs
  - Reference MCU_E_TIMEOUT_FAILURE
  - Reference MCU_E_CLOCK_FAILURE
  - Reference MCU_E_SWITCHMODE_FAILURE
- Container McuModeSettingConf
  - Parameter McuMode
  - Parameter McuPowerMode
  - Parameter McuEnableSleepOnExit
- Container McuRamSectorSettingConf
  - Parameter McuRamSectorId
  - Parameter McuRamDefaultValue
  - Parameter McuRamSectionBaseAddress
  - Parameter McuRamSectionSize
  - Parameter McuRamSectionWriteSize
  - Parameter McuRamSectionBaseAddrLinkerSym
  - Parameter McuRamSectionSizeLinkerSym
- Container McuInterruptEvents
  - Parameter McuVoltageErrorEvent

    - Parameter McuAlternateResetEvent
  - Container McuResetConfig
    - Parameter McuResetPinFilterBusClockSelect
    - Parameter McuResetPinFilterInStopMode
    - Parameter McuResetPinFilterInRunAndWait
    - Container McuSystemInterruptEnable
    - Parameter McuResetDelayTime
    - Parameter McuStopAcknowledgeErrorInterrupt
    - Parameter McuMDMAPSystemResetInterrupt
    - Parameter McuSoftwareInterrupt
    - Parameter McuCoreLockupInterrupt
    - Parameter McuJTAGResetInterrupt
    - Parameter McuGlobalInterrupt
    - Parameter McuExternalResetPinInterrupt
    - Parameter McuWatchdogInterrupt
    - Parameter McuCMULossOfClockResetInterrupt
    - Parameter McuLossOfLockInterrupt
    - Parameter McuLossOfClockInterrupt
    - 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 McuLowVoltageDetectInterruptEnable
    - Parameter McuLowVoltageDetectResetEnable
    - Parameter McuLowVoltageWarningInterruptEnable
    - Parameter McuLPODisable
    - Parameter McuClockBiasDisable
    - Parameter McuLowPowerBiasEnable
    - 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
- 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:

- None

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 of 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	true

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.

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	0

4.10 Parameter McuEnableUserModeSupport

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

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

b) 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 McuVeryLowPowerStopAbortNotification

Function name of callout. This function will be called when entering VLPS is aborted.

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

4.14 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

4.15 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 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

4.17 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

4.18 Parameter McuDisableRcmInit

To allow integration with non-Autosar Application Initialization so that settings done by AppInit will not to be done again by MCU driver.

If this parameter is set to TRUE, the Reset Control Module (RMC) initialization has to be disabled in the MCU driver.

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

#define POWER_IP_DISABLE_RCM_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

4.19 Parameter McuDisablePmcInit

To allow integration with non-Autosar Application Initialization so that settings done by AppInit will not to be done again by MCU driver.

If this parameter is set to TRUE, the Power Management Controller (PMC) initialization has to 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

4.20 Parameter McuDisableSmcInit

To allow integration with non-Autosar Application Initialization so that settings done by AppInit will not to be done again by MCU driver.

If this parameter is set to TRUE, the System Mode Controller (SMC) initialization has to be disabled in the MCU driver.

If this parameter is set to FALSE, the MCU driver is responsible for the System Mode Controller (SMC) initialization.

#define POWER_IP_DISABLE_SMC_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

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

4.21 Parameter McuEnableModeChangeNotification

If this parameter is set to TRUE, the Power Driver will send notifications to Clock driver when power mode is changed.

If this parameter is set to FALSE, the Power Driver will not send notifications to Clock driver when power mode is changed.

#define POWER_MODE_CHANGE_NOTIFICATION (STD_ON)/(STD_OFF) will be generated in Power_Ip_Cfg_Defines. 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 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_COU↩ NTER_CUSTOM']

4.23 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
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	DISABLED
literals	['ENABLED', 'DISABLED']

4.24 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

4.25 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

4.26 Parameter McuDisableDemReportErrorStatus

Enable/Disable the API for reporting the Dem Error.

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	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.27 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

4.28 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

Property	Value
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

4.29 Parameter McuEnablePeripheralCMU

Enable/Disable Peripheral CMU for S32K11X

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

4.30 Parameter McuSRAMRetentionConfigApi

Enable/Disable the API for SRAM retention configuration.

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	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.31 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

4.32 Parameter McuGetPowerModeStatetApi

Enable/Disable the API for Get Power Mode state: Mcu_GetPowerMode_State().

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

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	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.33 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

4.34 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

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

4.35 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

4.36 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
lowerMultiplicity	1

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

4.37 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	5
max	255
min	0

4.38 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

4.39 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

4.40 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

4.41 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

4.42 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

4.43 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	2
max	2
min	2

4.44 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

4.45 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

4.46 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	

4.47 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

4.48 Container McuModuleConfiguration

This container contains the configuration for the MCU driver.

Included subcontainers:

- McuAllowedModes
- McuClockSettingConfig
- McuDemEventParameterRefs
- McuModeSettingConf
- McuRamSectorSettingConf
- McuInterruptEvents
- McuResetConfig
- McuPowerControl

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

4.49 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

4.50 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

4.51 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

4.52 Parameter McuRTC_CLKINFrequencyHz

RTC_CLKIN 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-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD

Property	Value
defaultValue	32768.0
max	1000000.0
min	0.0

4.53 Parameter McuTCLK0_REF_CLKFrequencyHz

TCLK0_REF_CLK 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-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	32000.0
max	2.0E7
min	0.0

4.54 Parameter McuTCLK1_REF_CLKFrequencyHz

TCLK1_REF_CLK 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

Property	Value
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	32000.0
max	2.0E7
min	0.0

4.55 Parameter McuTCLK2_REF_CLKFrequencyHz

TCLK2_REF_CLK 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-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	32000.0
max	2.0E7
min	0.0

4.56 Parameter McuClockSrcFailureNotification

Enables/Disables clock failure notification.

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

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
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	DISABLED
literals	['ENABLED', 'DISABLED']

4.57 Container McuAllowedModes

Configures SMC_PMPROT register. The PMPROT register can be written only once after any system reset.

If the MCU is configured for a disallowed or reserved power mode, the MCU remains in its current power mode.

For example, if the MCU is in normal RUN mode and AVLP is 0, an attempt to enter VLPR mode using PMCTRL[RUNM] is blocked and PMCTRL[RUNM] remains 00b, indicating the MCU is still in Normal Run 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

4.58 Parameter McuAllowHighSpeedRunMode

This is a write-once parameter

SMC_PMPROT[AHSRUN] - Allow High Speed Run mode

Provided the appropriate control bits are set up in PMCTRL, this write-once field allows the MCU to enter High Speed Run mode (HSRUN).

0 - HSRUN is not allowed

1 - HSRUN is allowed

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

4.59 Parameter McuAllowVeryLowPowerModes

This is a write-once parameter

Provided the appropriate control bits are set up in PMCTRL, this write-once field allows the MCU to enter any very-low-power mode (VLPR, and VLPS).

0 - VLPR and VLPS are not allowed.

1 - VLPR and VLPS are allowed.

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

4.60 Container McuClockSettingConfig

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

Included subcontainers:

- McuRunClockConfig
- McuVlprClockConfig
- McuHsrunClockConfig
- McuSystemOSCClockConfig
- McuSIRCClockConfig
- McuFIRCClockConfig
- McuSystemPll
- McuSIMClockConfig
- McuClkMonitor
- McuPeripheralClockConfig
- 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

4.61 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

4.62 Parameter McuSysClockUnderMcuControl

0 - System clock tree is NOT under mcu control.

1 - System clock is under mcu control.

If this is set to false, the MCU code will not configure the SCG_xCCR register when Mcu_InitClock is called

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

4.63 Parameter McuScgClkOutSelect

SCG_CLKOUTCNFG[CLKOUTSEL] - SCG Clkout Select.

This register controls which SCG clock source is selected to be ported out to the CLKOUT pin.

0 - SCG SLOW Clock (FLASH_CLK Clock).

1 - System OSC

2 - Slow IRC

3 - Fast IRC

6 - System PLL

The selected clock must be enabled.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-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	SLOW_CLK
literals	['SLOW_CLK', 'SOSC_CLK', 'SIRC_CLK', 'FIRC_CLK', 'SPLL_CLK']

4.64 Container McuRunClockConfig

This container configures the system clock source and the system clock dividers

for the core, platform, external and bus clock domains when in Run mode only.

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

4.65 Parameter McuPreDivSystemClockFrequency

Run Core clock - Pre Divide System Clock Frequency.

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

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

Note: PREDIV_SYS_CLK is only available in S32K148.

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	1.6E8
min	0.0

4.66 Parameter McuCoreClockFrequency

Run Core clock - Clocks the ARM core, divided by DIVCORE bits inside SCG

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	4.8E7
max	8.0E7
min	0.0

4.67 Parameter McuSystemClockFrequency

Run System clock - Clocks the Crossbar, NVIC, Flash controller, FTM and PDB,etc.

RUN_SYS_CLK can run up to CORE_CLK.

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	8.0E7
max	8.0E7
min	0.0

4.68 Parameter McuBusClockFrequency

Run Bus clock - BUS_CLK Clocks the Peripherals, divided by DIVBUS bits inside SCG

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

Property	Value
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.0E7
max	4.8E7
min	0.0

4.69 Parameter McuFlashClockFrequency

Run Flash clock - Clocks the flash module, divided by DIVSLOW bits inside SCG

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

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

Configuring RUN_FLASH_CLK to lower frequencies than 24MHz (S32K11X) and 20Mhz (S32K14xW and S32M242) and 26.67MHz (S32K14X and S32M241) adds wait states and no power saving. It is recommended to configure it as close to 24MHz or 20Mhz or 26.67MHz.

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	2.0E7
max	2.0E7
min	0.0

4.70 Parameter McuSystemClockSwitch

Run System Clock Select. Configure the SCG_RCCR[SCS] register field.

The system clock is either:

- System OSC (SCG_RCCR[SCS]=1)

- Slow IRC (SCG_RCCR[SCS]=2)

- Fast IRC (SCG_RCCR[SCS]=3)

- System PLL (SCG_RCCR[SCS]=6)

Value extracted from Resource: MCU.RunSystemClkSource.List

The selected clock must be enabled.

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	['SOSC_CLK', 'SIRC_CLK', 'FIRC_CLK', 'SPLL_CLK']

4.71 Parameter McuCoreClockDivider

Configures the SCG_RCCR[DIVCORE] bitfield

This parameter represents the core clock 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

Property	Value
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1
max	16
min	1

4.72 Parameter McuBusClockDivider

Configures the SCG_RCCR[DIVBUS] bitfield

This parameter represents the bus clock 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	16
min	1

4.73 Parameter McuSlowClockDivider

Configures the SCG_RCCR[DIVSLOW] bitfield.

This parameter represents the flash clock divider.

Note: Configuring DIVSLOW to lower frequencies than supported flash frequencies mentioned

in datasheet (fFLASH) adds wait states and no power saving. It is recommended to configure

DIVSLOW as close to fFLASH.

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	8
min	1

4.74 Parameter McuScgClkOutFrequency

This is frequency of SCG clockout. 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	1000000.0
max	1.6E8
min	0.0

4.75 Container McuVlprClockConfig

Selects the clock source generating the system clock in VLPR mode.

The clock dividers cannot be changed while in VLPR mode.

They must be programmed prior to entering VLPR mode to guarantee

- the core/system and bus clocks are less than or equal to 4 MHz

- the flash memory clock is less than or equal to 1 MHz.

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

4.76 Parameter McuPreDivSystemClockFrequency

VLPR Core clock - Pre Divide System Clock Frequency.

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

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

Note: PREDIV_SYS_CLK is only available in S32K148.

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	8000000.0
max	8000000.0
min	0.0

4.77 Parameter McuCoreClockFrequency

VLPR Core clock - Clocks the ARM core, divided by DIVCORE bits inside SCG

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	1000000.0
max	1000000.0
min	0.0

4.78 Parameter McuSystemClockFrequency

VLPR System clock - Clocks the Crossbar, NVIC, Flash controller, FTM and PDB,etc.

VLPR_SYS_CLK can run up to CORE_CLK.

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

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1000000.0
max	1000000.0
min	0.0

4.79 Parameter McuBusClockFrequency

VLPR Bus clock - BUS_CLK Clocks the Peripherals, divided by DIVBUS bits inside SCG

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	1000000.0
max	1000000.0
min	0.0

4.80 Parameter McuFlashClockFrequency

VLPR Flash clock - Clocks the flash module, divided by DIVSLOW bits inside SCG

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

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

Configuring VLPR_FLASH_CLK to lower frequencies than 1MHz (S32K14X and S32K11X and S32M241) and 0.25MHz (S32K14xW and S32M242) adds wait states and no power saving. It is recommended to configure it as close to 1MHz or 0.25MHz.

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	250000.0
max	250000.0
min	0.0

4.81 Parameter McuSystemClockSwitch

VLPR System Clock Select. Configure the SCG_VCCR[SCS] register field.

The system clock is either:

- Slow IRC (SCG_VCCR[SCS]=2)

Value extracted from Resource: MCU.VlprSystemClkSource.List

The selected clock must be enabled.

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	SIRC_CLK
literals	['SIRC_CLK']

4.82 Parameter McuCoreClockDivider

Configures the SCG_VCCR[DIVCORE] bitfield

This parameter represents the core clock 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	8
max	16
min	1

4.83 Parameter McuBusClockDivider

Configures the SCG_VCCR[DIVBUS] bitfield

This parameter represents the bus clock 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	16
min	1

4.84 Parameter McuSlowClockDivider

Configures the SCG_VCCR[DIVSLOW] bitfield.

This parameter represents the flash clock divider.

Note: Configuring DIVSLOW to lower frequencies than supported flash frequencies mentioned

in datasheet (fFLASH) adds wait states and no power saving. It is recommended to configure

DIVSLOW as close to fFLASH.

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	4
max	8
min	1

4.85 Parameter McuScgClkOutFrequency

This is frequency of SCG clockout. It is given in Hz.

In VLPR mode, the FIRC, SOSC, SPLL clocks are disabled.

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

Property	Value
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1000000.0
max	1.6E8
min	0.0

4.86 Container McuHsrunClockConfig

This container configures the system clock source and the system clock dividers

for the core, platform, external and bus clock domains when in HSRUN mode only.

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

4.87 Parameter McuPreDivSystemClockFrequency

HSRUN System clock - Pre Divide System Clock Frequency.

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

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

Note: PREDIV_SYS_CLK is only available in S32K148.

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	4.8E7
max	1.6E8
min	0.0

4.88 Parameter McuCoreClockFrequency

HSRUN Core clock - Clocks the ARM core, divided by DIVCORE bits inside SCG

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	4.8E7
max	1.12E8
min	0.0

4.89 Parameter McuSystemClockFrequency

HSRUN System clock - Clocks the Crossbar, NVIC, Flash controller, FTM and PDB,etc.

HSRUN_SYS_CLK can run up to CORE_CLK.

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	1.12E8
max	1.12E8
min	0.0

4.90 Parameter McuBusClockFrequency

HSRUN Bus clock - BUS_CLK Clocks the Peripherals, divided by DIVBUS bits inside SCG

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	5.6E7
max	5.6E7
min	0.0

4.91 Parameter McuFlashClockFrequency

HSRUN Flash clock - Clocks the flash module, divided by DIVSLOW bits inside SCG

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

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

Configuring HSRUN_FLASH_CLK to lower frequencies than 28MHz adds wait states and no power saving. It is recommended to configure it as close to 28MHz.

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	2.8E7
max	2.8E7
min	0.0

4.92 Parameter McuSystemClockSwitch

HSRUN System Clock Select. Configure the SCG_HCCR[SCS] register field.

The system clock is either:

- Fast IRC (SCG_HCCR[SCS]=3)

- System PLL (SCG_HCCR[SCS]=6)

Value extracted from Resource: MCU.HsrunSystemClkSource.List

The selected clock must be enabled.

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', 'SPLL_CLK']

4.93 Parameter McuCoreClockDivider

Configures the SCG_HCCR[DIVCORE] bitfield

This parameter represents the core clock 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	16
min	1

4.94 Parameter McuBusClockDivider

Configures the SCG_HCCR[DIVBUS] bitfield

This parameter represents the bus clock 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	16
min	1

4.95 Parameter McuSlowClockDivider

Configures the SCG_HCCR[DIVSLOW] bitfield.

This parameter represents the flash clock divider.

Note: Configuring DIVSLOW to lower frequencies than supported flash frequencies mentioned

in datasheet (fFLASH) adds wait states and no power saving. It is recommended to configure

DIVSLOW as close to fFLASH.

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	8
min	1

4.96 Parameter McuScgClkOutFrequency

This is frequency of SCG clockout. 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	1000000.0
max	1.6E8
min	0.0

4.97 Container McuSystemOSCClockConfig

Configures System OSC registers.

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

4.98 Parameter McuSOSCUnderMcuControl

0 - System OSC is NOT under mcu control.

1 - System OSC is under mcu control.

If this is set to false, the MCU code will not configure the SOSC registers when Mcu_InitClock is called

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

4.99 Parameter McuSOSCFrequency

This is the SOSC frequency for the specific instance of the

McuClockReferencePoint container. It is expressed in Hz.

If PLL is used, then oscillator needs to be in high range only, SCG_SOSCCFG[RANGE] on 11 as used in reference clock.

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	4.0E7
min	0.0

4.100 Parameter McuSOSCDiv2Frequency

This is the SOSC Divider 2 frequency for the specific instance of the

McuClockReferencePoint container. It is expressed in Hz.

This should be configured to 40 MHz or less in RUN/HSRUN mode.

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	8000000.0
max	4.0E7
min	0.0

4.101 Parameter McuSOSCDiv1Frequency

This is the SOSC Divider 1 frequency for the specific instance of the

McuClockReferencePoint container. It is expressed in Hz.

This should be configured to 40 MHz or less in RUN/HSRUN mode.

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

Property	Value
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	8000000.0
max	4.0E7
min	0.0

4.102 Parameter McuSOSCEnable

SCG_SOSCCSR[SOSCEN] - System OSC Enable

0 - System OSC is disabled.

1 - System OSC 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

4.103 Parameter McuSOSCClockMonitorResetEnable

SCG_SOSCCSR[SOSCCMRE] - System OSC Clock Monitor Reset Enable

0 - Clock Monitor generates interrupt when error detected.

1 - Clock Monitor generates reset when error detected.

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

4.104 Parameter McuSOSCClockMonitorEnable

SCG_SOSCCSR[SOSCCM] - System OSC Clock Monitor

Enables the clock monitor, if the clock source is disabled in a low power mode then the clock monitor is also disabled in the low power mode. When the clock monitor is disabled in a low power mode, it remains disabled until the clock valid flag is set following exit from the low power mode.

0 - System OSC Clock Monitor is disabled.

1 - System OSC Clock Monitor 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

4.105 Parameter McuSOSCDiv2

Configures SCG_SOSCDIV[SOSCDIV2].

System OSC Clock Divide 2.

Clock divider 2 for System OSC. Used by bus clock modules that need an asynchronous clock source.

0 - Output disabled.

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	64
min	0

4.106 Parameter McuSOSCDiv1

Configures SCG_SOSCDIV[SOSCDIV1]

System OSC Clock Divide 1.

Clock divider 1 for System OSC. Used to generate the system clock source and by platform clock modules that need an asynchronous clock source.

0 - Output disabled.

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	1
max	64
min	0

4.107 Parameter McuSOSCRangeSelect

SCG_SOSCCFG[RANGE] - System OSC Range Select

Selects the frequency range for the system crystal oscillator (OSC)

NOTE The following constraints are not checked by the xdm schema:

If PLL is used, then oscillator needs to be in high range only, SCG_SOSCCFG[RANGE] on 11 as used in reference clock.

- Medium frequency range selected for the crytstal oscillator of 4 MHz to 8 MHz.

- High frequency range selected for the crystal oscillator of 8 MHz to 40 MHz.

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	HIGH_FREQ_RANGE
literals	['MEDIUM_FREQ_RANGE', 'HIGH_FREQ_RANGE']

4.108 Parameter McuSOSCHighGainOscillatorSelect

SCG_SOSCCFG[HGO] - High Gain Oscillator Select

Controls the crystal oscillator power mode of operations.

unchecked - Configure crystal oscillator for low-power operation

checked - Configure crystal oscillator for high-gain operation

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

4.109 Parameter McuSOSCExternalReferenceSelect

SCG_SOSCCFG[EREFS] - External Reference Select

Selects the source for the external reference clock.

unchecked - Internal oscillator of OSC requested.

checked - External reference clock from PAD pin selected

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

4.110 Container McuSIRCClockConfig

Configures Slow IRC (SIRC) registers.

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

4.111 Parameter McuSIRCUnderMcuControl

0 - Slow IRC is NOT under mcu control.

1 - Slow IRC is under mcu control.

If this is set to false, the MCU code will not configure the SIRC registers when Mcu_InitClock is called

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	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.112 Parameter McuSIRCFrequency

This is the SIRC frequency for the specific instance of the

McuClockReferencePoint container. It is expressed 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	0.0
max	8000000.0
min	0.0

4.113 Parameter McuSIRCDiv2Frequency

This is the SIRC Divider 2 frequency for the specific instance of the

McuClockReferencePoint container. It is expressed in Hz.

This should be configured to 8 MHz or less.

Note: Implementation Specific Parameter.

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	0.0
max	8000000.0
min	0.0

4.114 Parameter McuSIRCDiv1Frequency

This is the SIRC Divider 1 frequency for the specific instance of the

McuClockReferencePoint container. It is expressed in Hz.

This should be configured to 8 MHz or less in RUN/HSRUN mode and to 4 MHz or less in VLPR mode.

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	8000000.0
min	0.0

4.115 Parameter McuSIRCEnable

SCG_SIRCCSR[SIRCEN] - Slow IRC Enable

0 - Slow IRC is disabled.

1 - Slow IRC 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

4.116 Parameter McuSIRCLowPowerEnable

SCG_SIRCCSR[SIRCLPEN] - Slow IRC Low Power Enable

0 - Slow IRC is disabled in VLP modes.

1 - Slow IRC is enabled in VLP modes.

SCG_SIRCCSR[SIRCLPEN] bit field is applicable for VLPS mode only.

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 McuSIRCStopEnable

SCG_SIRCCSR[SIRCSTEN] - Slow IRC Enable

0 - Slow IRC is disabled.

1 - Slow IRC is enabled.

SCG__SIRCCSR[SIRCSTEN] bit field is not applicable and should be ignored.

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

4.118 Parameter McuSIRCDiv2

Configures SCG__SIRCDIV[SIRCDIV2].

Slow IRC Clock Divider 2.

Clock divider 2 for Slow IRC. Used by bus clock modules that need an asynchronous clock source.

0 - Output disabled.

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	64
min	0

4.119 Parameter McuSIRCDiv1

Configures SCG_SIRCDIV[SIRCDIV1].

Slow IRC Clock Divider 1.

Clock divider 1 for Slow IRC. Used to generate the system clock source and by platform clock modules

that need an asynchronous clock source.

0 - Output disabled.

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	64
min	0

4.120 Parameter McuSIRCRangeSelect

SCG_SIRCCFG[RANGE] - Selects the Frequency Range

Slow IRC low range clock (2 MHz)

Slow IRC high range clock (8 MHz)

Note: The SIRC clock is chosen as source clock that must be sacrificed to be ON at all times.

Add addition, Software should not configure the SCG_SIRCCFG[RANGE] to any value other than HIGH_RANGE_CLOCK.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-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	HIGH_RANGE_CLOCK
literals	['HIGH_RANGE_CLOCK']

4.121 Container McuFIRCClockConfig

Configures Fast IRC (FIRC) registers.

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

4.122 Parameter McuFIRCUnderMcuControl

0 - Fast IRC is NOT under mcu control.

1 - Fast IRC is under mcu control.

If this is set to false, the MCU code will not configure the FIRC registers when Mcu_InitClock is called

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

4.123 Parameter McuFIRCFrequency

This is the FIRC frequency for the specific instance of the

McuClockReferencePoint container. It is expressed in Hz.

Note: Implementation specific Container.

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

4.124 Parameter McuFIRCDiv2Frequency

This is the FIRC Divider 2 frequency for the specific instance of the

McuClockReferencePoint container. It is expressed in Hz.

This should be configured to 48 MHz or less in RUN/HSRUN mode.

Note: Implementation specific Container.

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

4.125 Parameter McuFIRCDiv1Frequency

This is the FIRC Divider 1 frequency for the specific instance of the

McuClockReferencePoint container. It is expressed in Hz.

This should be configured to 48 MHz or less in RUN/HSRUN mode.

Note: Implementation specific Container.

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

4.126 Parameter McuFIRCEnable

SCG_FIRCCSR[FIRCEN] - Fast IRC Enable

0 - Fast IRC is disabled.

1 - Fast IRC is enabled.

Note: Implementation specific Container.

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

4.127 Parameter McuFIRCRegulatorEnable

Fast IRC Regulator Enable

0 - Fast IRC Regulator is disabled.(SCG_FIRCCSR[FIRCREGOFF] = 1)

1 - Fast IRC Regulator is enabled.(SCG_FIRCCSR[FIRCREGOFF] = 0)

When Fast IRC is used, FIRCREGOFF must be 0. Fast IRC cannot be operated with FIRCREGOFF=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	true

4.128 Parameter McuFIRCDiv2

Configures SCG_FIRCDIV[FIRCDIV2]

Fast IRC Clock Divider 2.

Clock divider 2 for the Fast IRC. Used by bus clock modules that need an asynchronous clock source.

0 - Output disabled.

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	64
min	0

4.129 Parameter McuFIRCDiv1

Configures SCG_FIRCDIV[FIRCDIV1]

Fast IRC Clock Divider 1.

Clock divider 1 for Fast IRC. Used to generate the system clock source and by platform clock modules

that need an asynchronous clock source.

0 - Output disabled.

Note: Implementation specific Parameter.

Property	Value
type	ECUC-INTEGER-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	1
max	64
min	0

4.130 Parameter McuFIRCRangeSelect

SCG_FIRCCFG[RANGE] - Selects the Frequency Range.

00b - Fast IRC is trimmed to 48 MHz.

01b - Reserved.

10b - Reserved.

11b - Reserved.

Note: Software should not configure the SCG_FIRCCFG[RANGE] to any value other than Fast IRC is trimmed to 48 MHz.

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	TRIMMED_TO_48MHZ
literals	['TRIMMED_TO_48MHZ']

4.131 Container McuSystemPll

This container provides the specific configuration for the System PLL.

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

4.132 Parameter McuSystemPllUnderMcuControl

Set this to TRUE if System 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 refference 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	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.133 Parameter McuSPLLFrequency

This is the System PLL frequency for the specific instance of the

McuClockReferencePoint container. It is expressed in Hz.

Note: Pll_freq = (McuSPLLReferenceFrequency * McuSPLLMultiplier)/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	0.0
max	1.6E8
min	0.0

4.134 Parameter McuSPLLDiv2Frequency

This is the System PLL Divider 2 frequency for the specific instance of the

McuClockReferencePoint container. It is expressed in Hz.

This should be configured to 40 MHz or less in RUN mode and 56 MHz or less in HSRUN mode.

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

Property	Value
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	2.4E7
max	4.0E7
min	0.0

4.135 Parameter McuSPLLDiv1Frequency

This is the System PLL Divider 1 frequency for the specific instance of the

McuClockReferencePoint container. It is expressed in Hz.

This should be configured to 80MHz or less in RUN mode and to 112 MHz or less in HSRUN mode.

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	8.0E7
min	0.0

4.136 Parameter McuSPLLEnable

SCG_SPLLCSR[SPLLEN] - System PLL Enable

0 - System PLL is disabled.

1 - System PLL 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

4.137 Parameter McuSPLLClockMonitorResetEnable

SCG_SPLLCSR[SPLLCMRE] - System PLL Clock Monitor Reset Enable

0 - Clock Monitor generates interrupt when error detected.

1 - Clock Monitor generates reset when error detected.

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

4.138 Parameter McuSPLLClockMonitorEnable

SCG_SPLLCSR[SPLLCM] - System PLL Clock Monitor

Enables the clock monitor, if the clock source is disabled in a low power mode then the clock monitor is also disabled in the low power mode. When the clock monitor is disabled in a low power mode, it remains disabled until the clock valid flag is set following exit from the low power mode.

0 - RTC OSC Clock Monitor is disabled.

1 - RTC OSC Clock Monitor 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

4.139 Parameter McuSPLLDiv2

Configures SCG_SPLLDIV[SPLLDIV2].

System PLL Clock Divider 2.

Clock divider 2 for System PLL. Used by bus clock modules that need an asynchronous clock source.

0 - Output disabled.

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	4
max	64
min	0

4.140 Parameter McuSPLLDiv1

Configures SCG_SPLLDIV[SPLLDIV1]

System PLL Clock Divider 1.

Clock divider 1 for System PLL. Used to generate the system clock source and by platform clock modules

that need an asynchronous clock source.

0 - Output disabled.

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	64
min	0

4.141 Parameter McuSPLLInputClkPreDivider

PLL Reference Clock Divider.

Set the SPLL: SCG_SPLLCFG[PREDIV] field register.

Selects the amount to divide down the reference clock for the System PLL. The resulting frequency must be in the range of 8 MHz to 50 MHz.

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	1
max	8
min	1

4.142 Parameter McuSPLLReferenceFrequency

FSPLL_REF is PLL reference frequency range after the PREDIV.

F = McuSPLLInputFrequency / McuSPLLInputClkPreDivider.

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	8000000.0
max	1.6E7
min	0.0

4.143 Parameter McuSPLLInputFrequency

FSPLL_Input is PLL input frequency range before the PREDIV.

For S32K1XX, the valid range is [0 ... 40] MHz.

For S32R14xW, the valid range is [0 ... 48] MHz.

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	8000000.0
max	4.8E7
min	0.0

4.144 Parameter McuSPLLMultiplier

System PLL Multiplier.

Set the SCG_SPLLCFG[MULT] field register.

Valid range is in [16..47].

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	20
max	47
min	16

4.145 Parameter McuSPLLSelectSourceClock

SPLLCFG[SOURCE]: System PLL Clock Source.

Configures the input clock source for the System PLL.

The selected clock must be enabled.

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	SOSC_CLK
literals	['SOSC_CLK', 'FIRC_CLK']

4.146 Container McuSIMClockConfig

Configures SIM_CHIPCTL[TRACECLK_SEL], SIM_CHIPCTL[CLKOUTSEL] bits and SIM_PLATGC and SIM_CLKDIV4 registers.

Note: Implementation specific Container.

Included subcontainers:

- McuSimChipConfiguration
- McuSimLpoConfiguration
- McuSimFtmConfiguration

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

4.147 Parameter McuSIMUnderMcuControl

0 - SIM is NOT under mcu control.

1 - SIM is under mcu control.

If this is set to false, the MCU code will not configure the SIM registers when Mcu_InitClock is called

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

4.148 Parameter McuEIMClockGatingEnable

SIM_PLATCGC[CGCEIM] - EIM Clock Gating Control

Controls the clock gating to the EIM.

0 - Clock disabled.

1 - Clock 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

4.149 Parameter McuERMClockGatingEnable

SIM_PLATCGC[CGCERM] - ERM Clock Gating Control

Controls the clock gating to the ERM.

0 - Clock disabled.

1 - Clock 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

4.150 Parameter McuDMAClockGatingEnable

SIM_PLATCGC[CGCDMA] - DMA Clock Gating Control

Controls the clock gating to the DMA module.

0 - Clock disabled.

1 - Clock 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	true

4.151 Parameter McuMPUClockGatingEnable

SIM_PLATCGC[CGCMPU] - MPU Clock Gating Control

Controls the clock gating to the MPU module.

0 - Clock disabled.

1 - Clock 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

4.152 Parameter McuMSCMClockGatingEnable

SIM_PLATCGC[CGCMSCM] - MSCM Clock Gating Control

Controls the clock gating to the MSCM module.

0 - Clock disabled.

1 - Clock 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

4.153 Parameter McuGPIOClockGatingEnable

SIM_PLATCGC[CGCGPIO] - GPIO Clock Gating Control

Controls the clock gating to the GPIO module.

0 - Clock disabled.

1 - Clock enabled.

Note: Implementation Specific Parameter.This bit is available in S32K11X variants only

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

4.154 Container McuSimChipConfiguration

This container contains the configuration for the SIM_CHIPCTL registers.

Included subcontainers:

- None

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

4.155 Parameter McuDebugTraceDividerEnable

SIM_CLKDIV4[TRACEDIVEN] - Debug Trace Divider Control

0 - Debug trace divider disabled.

1 - Debug trace divider 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

4.156 Parameter McuTRACECLKDivider

Configures the SIM_CLKDIV4[TRACEDIV] bitfield

Trace clock divider divisor - This field sets the divide value for the fractional clock divider used as a source for trace clock. The source clock for the trace clock is set by the SIM_CHIPCTRL[TRACECLK_SEL]. Divider output clock = Divider input clock * [(TRACEFRAC+1)/(TRACEDIV+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	1
max	8
min	1

4.157 Parameter McuTRACECLKFraction

Configures the SIM_CLKDIV4[TRACEFRAC] bitfield

This field value is TRACEFRAC+1.

Trace clock divider fraction - This field sets the divide value for the fractional clock divider used as a source for trace clock. The source clock for the trace clock is set by the SIM_CHIPCTRL[TRACECLK_SEL]. Divider output clock = Divider input clock * [(TRACEFRAC+1)/(TRACEDIV+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	1
max	2
min	1

4.158 Parameter McuTRACECLKSelect

SIM_CHIPCTL[TRACECLK_SEL] - Debug trace clock select

Selects core clock or platform clock as the trace clock source.

The selected clock must be enabled.

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	CORE_CLK
literals	['CORE_CLK']

4.159 Parameter McuCLKOUTEnable

SIM_CHIPCTL[CLKOUTEN] - CLKOUT enable

unchecked - Clockout disabled.

checked - Clockout enabled.

Maximum supported clock out frequency for this device is 20 MHz.

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 McuCLKOUTDivider

Configures the SIM_CHIPCTL[CLKOUTDIV] bitfield

CLKOUT Divide Ratio.

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	8
min	1

4.161 Parameter McuCLKOUTSelect

SIM_CHIPCTL[CLKOUTSEL] - CLKOUT select

Selects the clock to output on the CLKOUT pin.

0 SCG_CLKOUT

2 SOSC_DIV2

4 SIRC_DIV2

5 For S32K148: QSPI_SFIF_CLK_HYP_PREMUX_CLK: Divide by 2 clock (configured through SCLKCONFIG[5]) for HyperRAM going to sfif clock to QSPI; For others: Reserved

6 FIRC_DIV2

7 HCLK

8 For S32K14x: SPLL_DIV2_CLK For S32K11x: Reserved

9 BUS_CLK

A LPO_128K_CLK

B For S32K148: QSPI_CLK; For others: Reserved

C LPO_CLK as selected by SIM_LPOCLKS[LPOCLKSEL]

D For S32K148: QSPI_SFIF_CLK; For others: Reserved

E RTC_CLK as selected by SIM CLK 32 KHz Select

F For S32K148: QSPI_2xSFIF_CLK; For others: Reserved

The selected clock must be enabled.

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	SCG_CLKOUT_CLK
literals	['SCG_CLKOUT_CLK', 'SOSCDIV2_CLK', 'SIRCDIV2_CLK', 'FIRCDIV2↩_CLK', 'HCLK', 'SPLLDIV2_CLK', 'BUS_CLK', 'LPO_128K_CLK', 'LPO↩_CLK', 'RTC_CLK']

4.162 Container McuSimLpoConfiguration

This container contains the configuration for the SIM_LPOCLKS registers.

Included subcontainers:

- None

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

4.163 Parameter McuRTCClkSelect

This is a write-once parameter

SIM_LPOCLKS[RTCCLKSEL] - 32 kHz clock source select.

Selects 32 kHz clock source for peripherals.

0 - SOSCDIV1_CLK

1 - 32 kHz LPO clock

2 - 32 kHz RTC_CLKIN clock

3 - FIRCDIV1_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	SOSCDIV1_CLK
literals	['SOSCDIV1_CLK', 'LPO_32K_CLK', 'RTC_CLKIN', 'FIRCDIV1_CLK']

4.164 Parameter McuLPOClkSelect

This is a write-once parameter

SIM_LPOCLKS[LPOCLKSEL] - LPO clock source select

Selects LPO clock source for peripherals

0 - 128 kHz LPO clock

1 - No clock

2 - 32 kHz LPO clock which is divided by the 128 kHz LPO clock

3 - 1 kHz LPO clock which is divided by the 128 kHz LPO clock

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	LPO_128K_CLK
literals	['LPO_128K_CLK', 'NO_CLOCK', 'LPO_32K_CLK', 'LPO_1K_CLK']

4.165 Parameter McuLPO_32KClockEnable

This is a write-once parameter

SIM_LPOCLKS[LPO32KCLKEN] - 32 kHz LPO clock enable

0 - Disable 32 kHz LPO clock output

1 - Enable 32 kHz LPO clock output

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

4.166 Parameter McuLPO_1KClockEnable

This is a write-once parameter

SIM_LPOCLKS[LPO1KCLKEN] - 1 kHz LPO clock enable

0 - Disable 1 kHz LPO clock output

1 - Enable 1 kHz LPO clock output

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

4.167 Container McuSimFtmConfiguration

This container contains the configuration for the SIM_FTMOPT0 registers.

Included subcontainers:

- None

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

4.168 Parameter McuFTM3ExternalClockPinSelect

SIM_FTMOPT0[FTM3CLKSEL] - FTM3 External Clock Pin Select

Selects the external pin used to drive the clock to the FTM3 module.

The selected pin must also be configured for the FTM external clock function through the appropriate Pin Control Register in the Port Control module.

0 - FTM3 external clock driven by TCLK0 pin.

1 - FTM3 external clock driven by TCLK1 pin.

2 - FTM3 external clock driven by TCLK2 pin.

3 - No clock input.

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	TCLK0_PIN
literals	['TCLK0_PIN', 'TCLK1_PIN', 'TCLK2_PIN', 'NO_CLOCK']

4.169 Parameter McuFTM2ExternalClockPinSelect

SIM_FTMOPT0[FTM2CLKSEL] - FTM2 External Clock Pin Select

Selects the external pin used to drive the clock to the FTM2 module.

The selected pin must also be configured for the FTM external clock function through the appropriate Pin Control Register in the Port Control module.

0 - FTM2 external clock driven by TCLK0 pin.

1 - FTM2 external clock driven by TCLK1 pin.

2 - FTM2 external clock driven by TCLK2 pin.

3 - No clock input.

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	TCLK0_PIN
literals	['TCLK0_PIN', 'TCLK1_PIN', 'TCLK2_PIN', 'NO_CLOCK']

4.170 Parameter McuFTM1ExternalClockPinSelect

SIM_FTMOPT0[FTM1CLKSEL] - FTM1 External Clock Pin Select

Selects the external pin used to drive the clock to the FTM1 module.

The selected pin must also be configured for the FTM external clock function through the appropriate Pin Control Register in the Port Control module.

0 - FTM1 external clock driven by TCLK0 pin.

1 - FTM1 external clock driven by TCLK1 pin.

2 - FTM1 external clock driven by TCLK2 pin.

3 - No clock input.

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	TCLK0_PIN
literals	['TCLK0_PIN', 'TCLK1_PIN', 'TCLK2_PIN', 'NO_CLOCK']

4.171 Parameter McuFTM0ExternalClockPinSelect

SIM_FTMOPT0[FTM0CLKSEL] - FTM0 External Clock Pin Select

Selects the external pin used to drive the clock to the FTM0 module.

The selected pin must also be configured for the FTM external clock function through the appropriate Pin Control Register in the Port Control module.

0 - FTM0 external clock driven by TCLK0 pin.

1 - FTM0 external clock driven by TCLK1 pin.

2 - FTM0 external clock driven by TCLK2 pin.

3 - No clock input.

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	TCLK0_PIN
literals	['TCLK0_PIN', 'TCLK1_PIN', 'TCLK2_PIN', 'NO_CLOCK']

4.172 Parameter McuFTM7ExternalClockPinSelect

SIM_FTMOPT0[FTM7CLKSEL] - FTM7 External Clock Pin Select

Selects the external pin used to drive the clock to the FTM7 module.

The selected pin must also be configured for the FTM external clock function through the appropriate Pin Control Register in the Port Control module.

0 - FTM7 external clock driven by TCLK0 pin.

1 - FTM7 external clock driven by TCLK1 pin.

2 - FTM7 external clock driven by TCLK2 pin.

3 - No clock input.

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	TCLK0_PIN
literals	['TCLK0_PIN', 'TCLK1_PIN', 'TCLK2_PIN', 'NO_CLOCK']

4.173 Parameter McuFTM6ExternalClockPinSelect

SIM_FTMOPT0[FTM6CLKSEL] - FTM6 External Clock Pin Select

Selects the external pin used to drive the clock to the FTM6 module.

The selected pin must also be configured for the FTM external clock function through the appropriate Pin Control Register in the Port Control module.

0 - FTM6 external clock driven by TCLK0 pin.

1 - FTM6 external clock driven by TCLK1 pin.

2 - FTM6 external clock driven by TCLK2 pin.

3 - No clock input.

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	TCLK0_PIN
literals	['TCLK0_PIN', 'TCLK1_PIN', 'TCLK2_PIN', 'NO_CLOCK']

4.174 Parameter McuFTM5ExternalClockPinSelect

SIM_FTMOPT0[FTM5CLKSEL] - FTM5 External Clock Pin Select

Selects the external pin used to drive the clock to the FTM5 module.

The selected pin must also be configured for the FTM external clock function through the appropriate Pin Control Register in the Port Control module.

0 - FTM5 external clock driven by TCLK0 pin.

1 - FTM5 external clock driven by TCLK1 pin.

2 - FTM5 external clock driven by TCLK2 pin.

3 - No clock input.

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	TCLK0_PIN
literals	['TCLK0_PIN', 'TCLK1_PIN', 'TCLK2_PIN', 'NO_CLOCK']

4.175 Parameter McuFTM4ExternalClockPinSelect

SIM_FTMOPT0[FTM7CLKSEL] - FTM4 External Clock Pin Select

Selects the external pin used to drive the clock to the FTM4 module.

The selected pin must also be configured for the FTM external clock function through the appropriate Pin Control Register in the Port Control module.

0 - FTM4 external clock driven by TCLK0 pin.

1 - FTM4 external clock driven by TCLK1 pin.

2 - FTM4 external clock driven by TCLK2 pin.

3 - No clock input.

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	TCLK0_PIN
literals	['TCLK0_PIN', 'TCLK1_PIN', 'TCLK2_PIN', 'NO_CLOCK']

4.176 Container McuClkMonitor

This container contains the specific configuration (parameters) of the Clock Monitor Unit.

Each CMU is independently programmed. SIRC is 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.

This container only for S32K11x derivatives.

Note: Implementation Specific Parameter.

Included subcontainers:

- McuClkMonitor_0
- McuClkMonitor_1

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

4.177 Container McuClkMonitor_0

This container contains the specific configuration (parameters) of the Monitor_0.

Clock Monitor Unit for Motor Clock.

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	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.178 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

This container only for S32K11x 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

4.179 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

4.180 Parameter McuCmuName

This is the name of the CMU0.

With name convention: CMU_FC_[Number Of CMU Unit]_[Name of Monitored clock].

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	CMU_FC_0_FIRC_MON1_CLK
literals	['CMU_FC_0_FIRC_MON1_CLK']

4.181 Parameter McuAsyncFHHInterruptEn

This field is used to enable/disable FHH asynchronous interrupt at the module boundary. (CMU_FC_IER[FHHAIE]).

0 - Asynchronous FHH Interrupt is Disabled

1 - Asynchronous FHH Interrupt 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

4.182 Parameter McuAsyncFLLInterruptEn

This field is used to enable/disable FLL asynchronous interrupt at the module boundary. (CMU_FC_IER[FLLAIE]).

0 - Asynchronous FLL Interrupt is Disabled

1 - Asynchronous FLL Interrupt 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

4.183 Container McuClkMonitor_1

This container contains the specific configuration (parameters) of the Monitor_1.

Clock Monitor Unit for Motor Clock.

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	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.184 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

This container only for S32K11x 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

4.185 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

4.186 Parameter McuCmuName

This is the name of the CMU1.

With name convention: CMU_FC_[Number Of CMU Unit]_[Name of Monitored clock].

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	CMU_FC_1_FIRC_MON2_CLK
literals	['CMU_FC_1_FIRC_MON2_CLK']

4.187 Parameter McuSyncFHHInterruptEn

This field is used to enable/disable FHH synchronous interrupt at the module boundary. (CMU_FC_IER[FHHIE]).

0 - Synchronous FHH Interrupt is Disabled

1 - Synchronous FHH Interrupt 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

4.188 Parameter McuSyncFLLInterruptEn

This field is used to enable/disable FLL synchronous interrupt at the module boundary. (CMU_FC_IER[FLLIE]).

0 - Synchronous FLL Interrupt is Disabled

1 - Synchronous FLL Interrupt 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

4.189 Container McuPeripheralClockConfig

This contains the combination for current peripheral in Run and LowPower Mode.

Note: Implementation Specific Container.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	30
upperMultiplicity	30
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.190 Parameter McuPeripheralClockUnderMcuControl

0 - Peripheral Clock is NOT under mcu control.

1 - Peripheral Clock is under mcu control.

If this is set to false, the MCU code will not configure the PCC registers when Mcu_InitClock is called

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

4.191 Parameter McuPerName

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	FTFM
literals	['FTFM', 'DMAMUX0', 'FLEXCAN0', 'FLEXCAN1', 'FTM3', 'ADC1', 'LPSPI0', 'LPSPI1', 'LPSPI2', 'PDB1', 'CRC0', 'PDB0', 'LPIT0', 'FTM0', 'FTM1', 'FTM2', 'ADC0', 'RTC0', 'LPTMR0', 'PORTA', 'PORTB', 'PORTC', 'PORTD', 'PORTE', 'FlexIO', 'EWM0', 'LPI2C0', 'LPUART0', 'LPUART1', 'CMP0']

4.192 Parameter McuPeripheralClockEnable

Sets PCC_[peripheral][CGC] bit.

This read/write bit enables the clock for the peripheral.

0 - Clock disabled.

1 - Clock 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

4.193 Parameter McuPeripheralClockSelect

Configures PCC_[peripheral][PCS].

This is used for peripherals that support various clock selections.

If the peripheral does not support various clock selections the field won't be editable.

This field can only be written when the CGC bit is 0 (clock disabled). Likewise, if the INUSE flag is set, this field is locked.

0 - Clock is off (or external clock as selected by FTMnCLKSEL for FTM modules).

1 - SOSCDIV2_CLK (SOCDIV1_CLK for FTM modules)

2 - SIRCDIV2_CLK (SIRCDIV1_CLK for FTM modules)

3 - FIRCDIV2_CLK (FIRCDIV1_CLK for FTM modules)

6 - SPLLDIV2_CLK (SPLLDIV1_CLK for FTM modules)

The selected clock must be enabled.

Note: Please make sure that the divider of clock source is not 0 .

If the field is not editable, it means the bit field is read only and the value will not be used.

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	CLOCK_IS_OFF
literals	['CLOCK_IS_OFF', 'SOSC', 'SIRC', 'FIRC', 'SPLL', 'LPO_128K_CLK']

4.194 Parameter McuPeripheralClockDivider

Configures PCC_[peripheral][PCD].

This is used for peripherals that require a clock divider. At SOC integration, each peripheral is assigned either a divider or not.

This field can only be written when the CGC bit is 0 (clock disabled). Likewise, if the INUSE flag is set, this field is locked.

Allowed values are from 1 to 8.

Note: Implementation Specific Parameter.

If the field is not editable, it means the bit field is read only and the value will not be used.

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	8
min	1

4.195 Parameter McuPeripheralFractionalDivider

Configures PCC_[peripheral][FRAC].

This field value is FRAC+1.

This sets the fraction multiply value for the fractional clock divider used as a clock source. Divider output clock = Divider input clock x [(FRAC+1)/(DIV+1)].

This field can only be written when the CGC bit is 0 (clock disabled). Likewise, if the INUSE flag is set, this field is locked.

1 - Fractional Multiply value is 1.

2 - Fractional Multiply value is 2.

Note: Implementation Specific Parameter.

If the field is not editable, it means the bit field is read only and the value will not be used.

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	2
min	1

4.196 Parameter McuPeripheralClockFrequency

Divider output clock = Divider input clock x [(FRAC+1)/(DIV+1)].

This is only calculated if the clock source is selectable and if the peripheral is enabled.

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

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

Note: The maximum frequency of LPUARTx, LPSPIx, LPI2Cx, FlexIO, LPTMR0 and LPIT0 are governed by BUS_CLK,

FTMx are governed by SYS_CLK, ADCx are 50MHz but always less than BUS_CLK.

So please check configuration value is fit for SYS_CLK and BUS_CLK values correspond to MCU mode.

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	8.0E7
min	0.0

4.197 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

4.198 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
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4.8E7
max	3.2E8
min	0.0

4.199 Parameter McuClockFrequencySelect

Select clock source for the specific instance of the McuClockReferencePoint container.

Note: The clock frequency configured in McuPeripheralClockConfig should be used to export the clock frequency through McuClockReferencePoint.

This reference point should be used in the configuration of the module that uses it (SPI, I2C, GPT, etc.).

If the configured module has also an internal clock selection (like FlexTimer for example),

the clock reference point should be configured taking the internal clock selection into account and the reference used should reflect the clock that finally enters the used peripheral.

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_SYS_CLK
literals	['CUSTOM', 'FIRC_CLK', 'FIRCDIV2_CLK', 'FIRCDIV1_CLK', 'SIRC_C↩ LK', 'SIRCDIV2_CLK', 'SIRCDIV1_CLK', 'SOSC_CLK', 'SOSCDIV2_CLK', 'SOSCDIV1_CLK', 'LPO_128K_CLK', 'LPO_32K_CLK', 'LPO_1K_CLK', 'SPLL_CLK', 'SPLLDIV2_CLK', 'SPLLDIV1_CLK', 'RUN_PRI_DIV_SY↩ S_CLK', 'RUN_CORE_CLK', 'RUN_SYS_CLK', 'RUN_FLASH_CLK', 'R↩ UN_BUS_CLK', 'VLPR_PRI_DIV_SYS_CLK', 'VLPR_CORE_CLK', 'V↩ LPR_SYS_CLK', 'VLPR_FLASH_CLK', 'VLPR_BUS_CLK', 'FTM3_CLK', 'ADC1_CLK', 'LPSPI0_CLK', 'LPSPI1_CLK', 'LPSPI2_CLK', 'LPIT0_CLK', 'FTM0_CLK', 'FTM1_CLK', 'FTM2_CLK', 'ADC0_CLK', 'LPTMR0_CLK', 'FlexIO_CLK', 'LPI2C0_CLK', 'LPUART0_CLK', 'LPUART1_CLK']

4.200 Container McuDemEventParameterRefs

Container for the references to DemEventParameter elements which shall be invoked using the API Dem_ReportErrorStatus 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

4.201 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

4.202 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

Property	Value
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Dem/DemConfigSet/DemEventParameter

4.203 Reference MCU_E_SWITCHMODE_FAILURE

Reference to configured DEM event to report Switch Mode 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

4.204 Container McuModeSettingConf

This container contains the configuration for the Mode setting of the MCU.

Note: Implementation Specific Parameter.

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

4.205 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

4.206 Parameter McuPowerMode

This parameter selects the Power Mode to be used.

For valid Mode transitions refers to Power mode state 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', 'HSRUN', 'VLPR', 'VLPS', 'STOP1', 'STOP2']

4.207 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
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.208 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

4.209 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
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	0
max	4294967295
min	0

4.210 Parameter McuRamDefaultValue

This parameter shall represent the Data pre-setting to be initialized.

Default value is 0xbababababa.

Note: Implementation Specific Parameter.

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

4.211 Parameter McuRamSectionBaseAddress

This parameter represents the RAM section base address.

The address must be aligned to 4 bytes.

Note: Implementation Specific Parameter.

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	536838144
max	536899583
min	536838144

4.212 Parameter McuRamSectionSize

This parameter represents the RAM section size in bytes.

The size must be multiple of 4.

Note: Implementation Specific Parameter.

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	1024
max	61440
min	0

4.213 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
symbolicNameValue	false
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

4.214 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	

4.215 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	

4.216 Container McuInterruptEvents

Configuration for different interrupts handled by MCU.

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

4.217 Parameter McuVoltageErrorEvent

Power Management Unit Fault Monitoring Interrupts.

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

4.218 Parameter McuAlternateResetEvent

Some events can generate an interrupt from RCM.

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

4.219 Container McuResetConfig

The Reset Control Module (MC_RCM) centralizes the different reset sources and manages the reset sequence of the device.

Note: Implementation Specific Parameter.

Included subcontainers:

- McuSystemInterruptEnable
- McuResetGeneratorConfiguration

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

4.220 Parameter McuResetPinFilterBusClockSelect

RCM_RPC[RSTFLTSEL] - Reset Pin Filter Bus Clock Select.

Selects the reset pin bus clock filter width. Transitions for less than (RSTFLTSEL+1) bus clock cycles are always filtered, transitions equal to (RSTFLTSEL+1) bus clock cycles may be filtered.

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

4.221 Parameter McuResetPinFilterInStopMode

RCM_RPC[RSTFLTSS] - Reset Pin Filter Select in Stop Mode.

Selects how the reset pin filter is enabled in any stop mode.

0 - All filtering disabled.

1 - LPO clock filter enabled.

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	ALL_FILTERING_DISABLE
literals	['ALL_FILTERING_DISABLE', 'LPO_CLOCK_FILTER_ENABLE']

4.222 Parameter McuResetPinFilterInRunAndWait

RCM_RPC[RSTFLTSRW] - Reset Pin Filter Select in Run and Wait Modes.

Selects how the reset pin filter is enabled in run and wait modes.

0 - All filtering disabled.

1 - Bus clock filter enabled for normal operation.

2 - LPO clock filter enabled for normal operation.

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

Property	Value
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	ALL_FILTERING_DISABLE
literals	['ALL_FILTERING_DISABLE', 'BUS_CLOCK_FILTER_ENABLE', 'LPO↩ _CLOCK_FILTER_ENABLE']

4.223 Container McuSystemInterruptEnable

Configures RCM_SRIE

This registers delays the assertion of a system reset for a period of time (DELAY field) while an interrupt is generated.

This allows software to perform a graceful shutdown.

A Chip POR source cannot be delayed by this feature, and entering Stop mode will terminate the delay.

The SRS will only update after the system reset occurs.

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

4.224 Parameter McuResetDelayTime

RCM_SRIE[DELAY] - Reset Delay Time.

Configures the maximum reset delay time from when the interrupt is asserted and the system reset occurs.

0 - 10 LPO cycles.

1 - 34 LPO cycles.

2 - 130 LPO cycles.

2 - 514 LPO cycles.

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	DELAY_10_LPO_CYCLES
literals	['DELAY_10_LPO_CYCLES', 'DELAY_34_LPO_CYCLES', 'DELAY_130↩ _LPO_CYCLES', 'DELAY_514_LPO_CYCLES']

4.225 Parameter McuStopAcknowledgeErrorInterrupt

RCM_SRIE[SACKERR] - Stop Acknowledge Error Interrupt.

0 - Interrupt disabled.

1 - Interrupt 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

4.226 Parameter McuMDMAPSystemResetInterrupt

RCM_SRIE[MDM_AP] - MDM-AP System Reset Request.

0 - Interrupt disabled.

1 - Interrupt 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

4.227 Parameter McuSoftwareInterrupt

RCM_SRIE[SW] - Software Interrupt.

0 - Interrupt disabled.

1 - Interrupt 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

4.228 Parameter McuCoreLockupInterrupt

RCM_SRIE[LOCKUP] - Core Lockup Interrupt.

0 - Interrupt disabled.

1 - Interrupt 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

4.229 Parameter McuJTAGResetInterrupt

RCM_SRIE[JTAG] - JTAG generated reset.

0 - Interrupt disabled.

1 - Interrupt 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

4.230 Parameter McuGlobalInterrupt

RCM_SRIE[GIE] - Global Interrupt Enable.

0 - All interrupt sources disabled.

1 - All interrupt sources 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

4.231 Parameter McuExternalResetPinInterrupt

RCM_SRIE[PIN] - External Reset Pin Interrupt.

0 - Interrupt disabled.

1 - Interrupt 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

4.232 Parameter McuWatchdogInterrupt

RCM_SRIE[WDOG] - Watchdog Interrupt.

0 - Interrupt disabled.

1 - Interrupt 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

4.233 Parameter McuCMULossOfClockResetInterrupt

RCM_SRIE[CMU_LOC] - CMU Loss-of-Clock Reset Interrupt.

0 - Interrupt disabled.

1 - Interrupt 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

4.234 Parameter McuLossOfLockInterrupt

RCM_SRIE[LOL] - Loss of Lock Interrupt.

0 - Interrupt disabled.

1 - Interrupt 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

4.235 Parameter McuLossOfClockInterrupt

RCM_SRIE[LOC] - Loss of Clock Interrupt.

0 - Interrupt disabled.

1 - Interrupt 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

4.236 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

4.237 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

4.238 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
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

4.239 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

4.240 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

4.241 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

4.242 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

4.243 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

4.244 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

4.245 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

4.246 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

4.247 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

4.248 Parameter McuLowVoltageDetectInterruptEnable

PMC_LVDSC1[LVDIE] - Low Voltage Detect Interrupt Enable.

This bit enables hardware interrupt requests for LVDF.

0 - Hardware interrupt disabled (use polling).

1 - Request a hardware interrupt when LVDF = 1.

Note:

- Implementation Specific Parameter.

- Only support for 1xx 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

4.249 Parameter McuLowVoltageDetectResetEnable

PMC_LVDSC1[LVDRE] - Low Voltage Detect Reset Enable.

This bit enables the low voltage detect events to generate a system reset.

0 - No system resets on low voltage detect events.

1 - If the supply voltage falls below VLVD, a system reset will be generated.

Note:

- Implementation Specific Parameter.

- Only support for 1xx 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

4.250 Parameter McuLowVoltageWarningInterruptEnable

PMC_LVDSC2[LVWIE] - Low-Voltage Warning Interrupt Enable.

This bit enables hardware interrupt requests for LVWF.

0 - Hardware interrupt disabled (use polling).

1 - Request a hardware interrupt when LVWF = 1.

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	false

4.251 Parameter McuLPODisable

PMC_REGSC[LPODIS] - This bit enables or disable the low power oscillator.

After disabling the LPO a time of 2 LPO clock cycles is required before it is allowed to enable it

again. Violating this waiting time of 2 cycles can result in malfunction of the LPO.

unchecked - Low power oscillator enabled.

checked - Low power oscillator disabled.

Note: The reset delay feature requires the LPO clock to remain active.

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

4.252 Parameter McuClockBiasDisable

PMC_REGSC[CLKBIASDIS] - Clock Bias Disable Bit.

This bit disables the bias currents and reference voltages for some clock modules in order to further

reduce power consumption in VLPS mode.

Note: While using this bit, it must be ensured that respective clock modules are disabled in VLPS mode.

Else, severe malfunction of clock modules will happen.

unchecked - No effect.

checked - In VLPS mode, the bias currents and reference voltages for the following clock modules are disabled: SIRC, FIRC, PLL.

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

4.253 Parameter McuLowPowerBiasEnable

PMC_REGSC[BIASEN] - Bias Enable Bit.

This bit enables source and well biasing for the core logic in low power mode. In full performance mode

this bit has no effect. This is useful to further reduce MCU power consumption in low power mode.

unchecked - Biasing disabled, core logic can run in full performance.

checked - Biasing enabled, core logic is slower and there are restrictions in allowed system clock speed.

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	false

4.254 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

4.255 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

4.256 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

4.257 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

4.258 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

4.259 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

4.260 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
symbolicNameValue	false
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

4.261 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

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	LVD_6_5V
literals	['LVD_5_5V', 'LVD_6_5V']

4.262 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']

4.263 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_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_SpecificPerpihParamType

  Clock Specific peripheral configure. Implements Clock_Ip_SpecificPerpihParamType_Class. More...
- struct Clock_IP_SpecificPeriphConfigType

  Clock Specific peripheral structure. Implements Clock_IP_SpecificPeriphConfigType_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_PowerModesType

  Power modes.
- enum Clock_Ip_PowerNotificationType

  Power mode notification.
- 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.
- enum Clock_Ip_SpecificPeriphParamType

  specific peripheral.

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.
- void Clock_Ip_PowerModeChangeNotification (Clock_Ip_PowerModesType PowerMode, Clock_Ip_PowerNotificationTy Notification)

  Sends notifications regarding power mode transition.
- uint32 Clock_Ip_GetClockFrequency (Clock_Ip_NameType ClockName)

  Gets the clock frequency for a specific clock name.
- 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 TimeoutTicks)

  Checks for timeout condition.

6.1.2 Data Structure Documentation

6.1.2.1 struct Clock_Ip_RegisterValueType

Register value structure. Implements Clock_Ip_RegisterValueType_Class.

Definition at line 3013 of file Clock_Ip_Types.h.

Data Fields

Type	Name	Description
uint32 *	RegisterAddr	Register address.
uint32	RegisterData	Register value.

6.1.2.2 struct Clock_Ip_RegisterIndexType

Register index structure. Implements Clock_Ip_RegisterIndexType_Class.

Definition at line 3024 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.

6.1.2.3 struct Clock_Ip_IrcoscConfigType

Clock Source IRCOSC configuration structure. Implements Clock_Ip_IrcoscConfigType_Class.

Definition at line 3037 of file Clock_Ip_Types.h.

Data Fields

Type	Name	Description
Clock_Ip_NameType	Name	Clock name associated to `ircosc`
uint16	Enable	Enable `ircosc`.
uint8	Regulator	Enable regulator.
uint8	Range	Ircosc range.
uint8	LowPowerModeEnable	Ircosc enable in VLP mode
uint8	StopModeEnable	Ircosc enable in STOP mode

6.1.2.4 struct Clock_Ip_XoscConfigType

CGM Clock Source XOSC configuration structure. Implements Clock_Ip_XoscConfigType_Class.

Definition at line 3053 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

Data Fields

Type	Name	Description
uint8	Gain	Gain value
uint8	Monitor	Monitor type
uint8	AutoLevelController	Automatic level controller

6.1.2.5 struct Clock_Ip_PllConfigType

CGM Clock Source PLLDIG configuration structure. Implements Clock_Ip_PllConfigType_Class.

Definition at line 3076 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.
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

6.1.2.6 struct Clock_Ip_SelectorConfigType

Clock selector configuration structure. Implements Clock_Ip_SelectorConfigType_Class.

Definition at line 3113 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

6.1.2.7 struct Clock_Ip_DividerConfigType

Clock divider configuration structure. Implements Clock_Ip_DividerConfigType_Class.

Definition at line 3124 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.

6.1.2.8 struct Clock_Ip_DividerTriggerConfigType

Clock divider trigger configuration structure. Implements Clock_Ip_DividerTriggerConfigType_Class.

Definition at line 3135 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

6.1.2.9 struct Clock_Ip_FracDivConfigType

Clock fractional divider configuration structure. Implements Clock_Ip_FracDivConfigType_Class.

Definition at line 3149 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

6.1.2.10 struct Clock_Ip_ExtClkConfigType

Clock external clock configuration structure. Implements Clock_Ip_ExtClkConfigType_Class.

Definition at line 3161 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.

6.1.2.11 struct Clock_Ip_PcfsConfigType

Clock Source PCFS configuration structure. Implements Clock_Ip_PcfsConfigType_Class.

Definition at line 3172 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.

6.1.2.12 struct Clock_Ip_GateConfigType

Clock gate clock configuration structure. Implements Clock_Ip_GateConfigType_Class.

Definition at line 3186 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

6.1.2.13 struct Clock_Ip_CmuConfigType

Clock cmu configuration structure. Implements Clock_Ip_CmuConfigType_Class.

Definition at line 3197 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
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.

6.1.2.14 struct Clock_Ip_ConfiguredFrequencyType

Configured frequency structure. Implements Clock_Ip_ConfiguredFrequencyType_Class.

Definition at line 3210 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

6.1.2.15 struct Clock_Ip_SpecificPerpihParamType

Clock Specific peripheral configure. Implements Clock_Ip_SpecificPerpihParamType_Class.

Definition at line 3220 of file Clock_Ip_Types.h.

6.1.2.16 struct Clock_IP_SpecificPeriphConfigType

Clock Specific peripheral structure. Implements Clock_IP_SpecificPeriphConfigType_Class.

Definition at line 3230 of file Clock_Ip_Types.h.

6.1.2.17 struct Clock_Ip_ClockConfigType

Clock configuration structure. Implements Clock_Ip_ClockConfigType_Class.

Definition at line 3240 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
Clock_Ip_IrcoscConfigType	Ircoscs[(2U)]	IRCOSCs
Clock_Ip_XoscConfigType	Xoscs[(1U)]	XOSCs
Clock_Ip_PllConfigType	Plls[(1U)]	PLLs
Clock_Ip_SelectorConfigType	Selectors[(26U)]	Selectors
Clock_Ip_DividerConfigType	Dividers[(17U)]	Dividers
Clock_Ip_DividerTriggerConfigType	DividerTriggers[1U]	Divider triggers
Clock_Ip_FracDivConfigType	FracDivs[1U]	Fractional dividers
Clock_Ip_ExtClkConfigType	ExtClks[(4U)]	External clocks
Clock_Ip_GateConfigType	Gates[(39U)]	Clock gates
Clock_Ip_PcfsConfigType	Pcfs[1U]	Progressive clock switching
Clock_Ip_CmuConfigType	Cmus[1U]	Clock cmus
Clock_IP_SpecificPeriphConfigType	SpecificPeriphalConfiguration	Clock specific peripheral configuration
Clock_Ip_ConfiguredFrequencyType	ConfiguredFrequencies[(1U)]	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 3007 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
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_PowerModesType

```
enum Clock_Ip_PowerModesType
```

Power modes.

Definition at line 194 of file Clock_Ip_Types.h.

6.1.4.6 Clock_Ip_PowerNotificationType

```
enum Clock_Ip_PowerNotificationType
```

Power mode notification.

Definition at line 205 of file Clock_Ip_Types.h.

6.1.4.7 Clock_Ip_NameType

```
enum Clock_Ip_NameType
```

Clock names.

Definition at line 215 of file Clock_Ip_Types.h.

6.1.4.8 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 2945 of file Clock_Ip_Types.h.

6.1.4.9 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 2953 of file Clock_Ip_Types.h.

6.1.4.10 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 2962 of file Clock_Ip_Types.h.

6.1.4.11 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 2971 of file Clock_Ip_Types.h.

6.1.4.12 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 2987 of file Clock_Ip_Types.h.

6.1.4.13 Clock_Ip_SpecificPeriphParamType

```
enum Clock_Ip_SpecificPeriphParamType
```

specific peripheral.

Definition at line 2995 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_PowerModeChangeNotification()

```
void Clock_Ip_PowerModeChangeNotification (
            Clock_Ip_PowerModesType PowerMode,
            Clock_Ip_PowerNotificationType Notification )
```

Sends notifications regarding power mode transition.

This function sends notifications regarding power mode transition. It is called by power driver each time power mode is changed.

Parameters

in	*PowerMode*	Power mode.
in	*PowerMode*	Power mode notification.

Returns

void

6.1.5.10 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

6.1.5.11 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)

6.1.5.12 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)

6.2 Mcu Driver

6.2.1 Detailed Description

Data Structures

- struct Mcu_ConfigType

  Initialization data for the MCU driver. 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_PowerModeStateType Mcu_GetPowerModeState (void)

  This function returns the System Status (power mode, clock settings) from MC_ME hw IP. The return value is the content register read from hardware (ME_GS). The return value shall evidentiate the current running mode, clock settings and activation, etc.
- 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_ReportDemSwitchModeError (void)

  Reports failed switch mode to DEM.
- void Mcu_Ipw_SRAMRetentionConfig (Mcu_SRAMRetenConfigType SRAMRetenConfig)

  Configuration for SRAM retention.

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

  DEM error reporting configuration.
- Mcu_RamSectionType NoRamConfigs

  Total number of RAM sections.
- Mcu_ModeType NoModeConfigs

  Total number of MCU modes.
- Mcu_ClockType NoClkConfigs

  Total number of MCU clock configurations.
- const Mcu_RamConfigType(∗ RamConfigArrayPtr)[((uint32) 1U)]

  RAM data configuration.
- const Mcu_ModeConfigType(∗ ModeConfigArrayPtr)[((uint32) 6U)]

  Power Modes 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.

Definition at line 173 of file Mcu.h.

6.2.2.1.1.2 DemConfigPtr `const Mcu_DemConfigType* DemConfigPtr`

DEM error reporting configuration.

Definition at line 178 of file Mcu.h.

6.2.2.1.1.3 NoRamConfigs `Mcu_RamSectionType NoRamConfigs`

Total number of RAM sections.

Definition at line 181 of file Mcu.h.

6.2.2.1.1.4 NoModeConfigs `Mcu_ModeType NoModeConfigs`

Total number of MCU modes.

Definition at line 184 of file Mcu.h.

6.2.2.1.1.5 NoClkConfigs `Mcu_ClockType NoClkConfigs`

Total number of MCU clock configurations.

Definition at line 188 of file Mcu.h.

6.2.2.1.1.6 RamConfigArrayPtr `const Mcu_RamConfigType(* RamConfigArrayPtr)[((uint32) 1U)]`

RAM data configuration.

Definition at line 192 of file Mcu.h.

6.2.2.1.1.7 ModeConfigArrayPtr `const Mcu_ModeConfigType(* ModeConfigArrayPtr)[((uint32) 6U)]`

Power Modes data configuration.

Clock data configuration.

Definition at line 196 of file Mcu.h.

6.2.2.1.1.8 HwIPsConfigPtr `const Mcu_HwIPsConfigType* HwIPsConfigPtr`

IPs data generic configuration.

Definition at line 203 of file Mcu.h.

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 220 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 230 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_STATU↩ S_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_GetPowerModeState()

```
Mcu_PowerModeStateType Mcu_GetPowerModeState (
            void  )
```

This function returns the System Status (power mode, clock settings) from MC_ME hw IP. The return value is the content register read from hardware (ME_GS). The return value shall evidentiate the current running mode, clock settings and activation, etc.

Returns

Mcu_PowerModeStateType Get the state of the power mode.

Return values

uint32	Content of ME_GS register.

6.2.6.12 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

6.2.6.13 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

6.2.6.14 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

6.2.6.15 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

6.2.6.16 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

6.2.6.17 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.

6.2.6.18 Mcu_ReportDemSwitchModeError()

```
void Mcu_ReportDemSwitchModeError (
             void  )
```

Reports failed switch mode to DEM.

Directly reports the clock multiplexer switch error to DEM.

Parameters

in	*void.*	

Returns

void.

6.2.6.19 Mcu_Ipw_SRAMRetentionConfig()

```
void Mcu_Ipw_SRAMRetentionConfig (
             Mcu_SRAMRetenConfigType SRAMRetenConfig )
```

Configuration for SRAM retention.

This function configure for both SRAML_RETEN and SRAMU_RETEN bits.

Returns

void

6.3 Power Ip Driver

6.3.1 Detailed Description

Data Structures

- struct Power_Ip_AEC_ConfigType

  Configuration for AEC. More...
- struct Power_Ip_PMC_ConfigType

  Configuration for PMC. More...
- struct Power_Ip_RCM_ConfigType

  RCM IP configuration. More...
- struct Power_Ip_SMC_ConfigType

  SMC IP configuration. More...
- struct Power_Ip_HwIPsConfigType

  More...
- struct Power_Ip_ModeConfigType

  Definition of a MCU mode section in the configuration structure. More...

Types Reference

- typedef uint32 Power_Ip_ModeType

  The Power_Ip_ModeType specifies the identification (ID) for a MCU mode, configured via configuration structure.
- typedef uint32 Power_Ip_RawResetType

  The type Mcu_RawResetType specifies the reset reason in raw register format, read from a reset status register.
- typedef void(∗ Power_Ip_ReportErrorsCallbackType) (Power_Ip_ReportErrorType Error, uint8 ErrorCode)

  Power report error callback structure. Implements PowerReportErrorCallbackType_Class.

Enum Reference

- enum Power_Ip_PMC_StatusType
- 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.
- 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_CM4_SystemReset (void)

  The function initiates a system reset request to reset the SoC.
- 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.

6.3.2 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_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 232 of file Power_Ip_PMC_Types.h.

Data Fields

Type	Name	Description
uint8	Lvdsc2	Trimming Register (PMC_LVDSC2).
uint8	Regsc	Trimming Register (PMC_REGSC).
uint32	PmcAeConfig	Trimming Register (PMC_AE_CONFIG).
uint32	PmcAeMonitor	Trimming Register (PMC_AE_MONITOR).

6.3.2.3 struct Power_Ip_RCM_ConfigType

RCM IP configuration.

This structure contains information for peripheral configuration

Definition at line 211 of file Power_Ip_RCM_Types.h.

6.3.2.4 struct Power_Ip_SMC_ConfigType

SMC IP configuration.

This structure contains information for allowed modes

Definition at line 176 of file Power_Ip_SMC_Types.h.

6.3.2.5 struct Power_Ip_HwIPsConfigType

Definition at line 217 of file Power_Ip_Types.h.

Data Fields

Type	Name	Description
const Power_Ip_RCM_ConfigType *	RCMConfigPtr	Configuration for RCM (Reset Generation Module) hardware IP.
const Power_Ip_PMC_ConfigType *	PMCConfigPtr	Configuration for PMC (Power Management Unit) hardware IP, part of PMC.
const Power_Ip_SMC_ConfigType *	SMCConfigPtr	Configuration for SMC hardware IP.
const Power_Ip_AEC_ConfigType *	AECConfigPtr	Configuration for AEC hardware IP.

6.3.2.6 struct Power_Ip_ModeConfigType

Definition of a MCU mode section in the configuration structure.

Specifies the system behaviour during the selected target mode. Data set and configured by Mcu_SetMode call.

Definition at line 251 of file Power_Ip_Types.h.

Data Fields

Type	Name	Description
Power_Ip_ModeType	ModeConfigId	The ID for Power Mode configuration.
Power_Ip_PowerModeType	PowerMode	Power modes control configuration.
uint8	SleepOnExit	Indicates sleep-on-exit configuration.

6.3.3 Types Reference

6.3.3.1 Power_Ip_ModeType

```
typedef uint32 Power_Ip_ModeType
```

The Power_Ip_ModeType specifies the identification (ID) for a MCU mode, configured via configuration structure.

The type shall be uint8, uint16 or uint32.

Definition at line 211 of file Power_Ip_Types.h.

6.3.3.2 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 268 of file Power_Ip_Types.h.

6.3.3.3 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 289 of file Power_Ip_Types.h.

6.3.4 Enum Reference

6.3.4.1 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 247 of file Power_Ip_PMC_Types.h.

6.3.4.2 Power_Ip_PowerModeType

```
enum Power_Ip_PowerModeType
```

Power Modes encoding.

Supported power modes for SMC hw IP.

Enumerator

POWER_IP_RUN_MODE	Run Mode.
POWER_IP_HSRUN_MODE	High Speed Mode.
POWER_IP_VLPR_MODE	Very Low Power Run Mode.
POWER_IP_VLPS_MODE	Very Low Power Stop Mode.
POWER_IP_STOP1_MODE	Stop 1 Mode.
POWER_IP_STOP2_MODE	Stop 2 Mode.

Definition at line 236 of file Power_Ip_Types.h.

6.3.4.3 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_REPORT_SWITCH_MODE_ERROR	Report switch mode Error.
POWER_IP_REPORT_VLPSA_NOTIFICATION	Report the VLPS transition was aborted.
POWER_IP_ISR_VOLTAGE_HVD_VDDINT_↩ DETECT	Report the Voltage on VDDINT is high-voltage detected.

Enumerator

POWER_IP_ISR_VOLTAGE_HVD_15_DETECT	Report the Voltage on VDD15 is high-voltage detected.
POWER_IP_ISR_VOLTAGE_HVD_VDD_DE↵ TECT	Report the Voltage on VDD is high-voltage detected.
POWER_IP_ISR_VOLTAGE_LVD_VDDC_DE↵ TECT	Report the Voltage on VDDC is low-voltage detected.
POWER_IP_ISR_VOLTAGE_LVD_VLS_DET↵ ECT	Report the Voltage on VLS is low-voltage detected.

Definition at line 271 of file Power_Ip_Types.h.

6.3.5 Function Reference

6.3.5.1 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

6.3.5.2 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

6.3.5.3 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

6.3.5.4 Power_Ip_GetResetReason()

```
Power_Ip_ResetType Power_Ip_GetResetReason (
            void )
```

Returns reset type.

This function returns reset type.

Returns

Power_Ip_ResetType Reset type

6.3.5.5 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

6.3.5.6 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

6.3.5.7 Power_Ip_CM4_SystemReset()

```
void Power_Ip_CM4_SystemReset (
            void  )
```

The function initiates a system reset request to reset the SoC.

The function initiates a system reset request to reset the SoC

Parameters

in	*none*	

Returns

void

6.3.5.8 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)

6.3.5.9 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 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_StatusType

  Ram ip status return codes.

Function Reference

- Ram_Ip_StatusType Ram_Ip_InitRamSection (const Ram_Ip_RamConfigType ∗RamConfigPtr)

  Initializes RAM section.

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 171 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_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 127 of file Ram_Ip_Types.h.

6.4.3.2 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 134 of file Ram_Ip_Types.h.

6.4.3.3 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 141 of file Ram_Ip_Types.h.

6.4.3.4 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 148 of file Ram_Ip_Types.h.

6.4.4 Enum Reference

6.4.4.1 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 154 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

How to Reach Us:

Home Page:
nxp.com

Web Support:
nxp.com/support