User Manual

for S32 PMIC Driver

Document Number: UM11PMICASR4.4 Rev0000R4.0.0 Rev. 1.0

Chapter 1

Revision History

Revision	Date	Author	Description
1.0	31.10.2022	NXP AASW Team	Prepared for release S32 RTD AUTOSAR 4.4 Version 4.0.0 Release

Chapter 2

Introduction

This User Manual describes NXP Semiconductor Non-AUTOSAR Pmic for S32CC platform. Non-AUTOSAR Pmic driver configuration parameters and deviations from the specification are described in Driver chapter of this document. Non-AUTOSAR Pmic driver requirements and APIs are described in the Non-AUTOSAR Pmic driver software specification document.

2.1 Supported Derivatives

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

- s32g274a_bga525
- s32g254a_bga525
- s32g233a_bga525
- s32g234m_bga525
- s32g378a_bga525
- s32g379a_bga525
- s32g398a_bga525
- s32g399a_bga525
- s32g338m_bga525
- s32g339m_bga525
- s32g358a_bga525
- s32g359a_bga525

All of the above microcontroller devices are collectively named as S32.

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	S32G2 Reference Manual	Rev. 5, May 2022
2	S32G3 Reference Manual	Rev. 2 Draft C, June 2022
3	S32G2 Mask Set Errata for Mask 0P77B	Rev. 2.4
4	S32G3 Mask Set Errata for Mask 0P72B	Rev. 1.1
5	S32G2 Data Sheet	Rev. 5, May 2022
6	S32G3 Data Sheet	Rev. 2 Draft B, June 2022
7	VR5510 Data Sheet	Rev. 5, April 2022

Chapter 3

Driver

3.1 Requirements

For CDD: PMIC is a Complex Device Driver (CDD), so there are no AUTOSAR requirements regarding this module.

It has vendor-specific requirements and implementation.

3.2 Driver Design Summary

- The PMIC driver is implemented as an Non-AUTOSAR Complex Device Driver. It provides support for implementing power management control.
- It uses the underlying hardware communication peripherals (e.g. I2C) to access the external registers on PMIC devices.
- The driver offers a hardware independent API to the upper layer that can be used to configure PMIC devices.
- Hardware and software settings can be configured using an AUTOSAR standard configuration tool.
- The driver reports errors to the Diagnostic Event Manager (DEM) and Default Error Tracer (DET) as defined in AUTOSAR.

3.3 Hardware Resources

The hardware configured by the PMIC driver is VR5510 revision B1 which is a NXP device power control management.

3.4 Deviations from Requirements

The driver deviates from the Non-AUTOSAR Pmic Driver software specification in some places. The table identifies the AUTOSAR requirements that are not fully implemented, implemented differently, not available, not testable or out of scope for the Pmic Driver.

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

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

Requirement	Status	Description	Notes
Req_Id	N/S	These requirements are not applicable to this specification.	Not a requirement.

3.5 Driver Limitations

1. Pmic driver does not have all MISRA violations resolved.

2. Pmic_InitDevice function might not work with the Watchdog Window 1ms, 2ms... because the I2c command takes time and then Watchdog will not feed in the open duration. To avoid this problem, user should set Watchdog Window bigger and Watchdog Closed Window Duty Cycle minimum value (31,25%) or disable the Watchdog Window.

3. S32G2-RDB2 board does not support the emulation OTP feature.

4. Pmic driver does not support I2c asynchronous type because of the competition between I2c interrupt and timer(Gpt) interrupt. Then the watchdog will not trigger in the open state. To ensure safety, the driver just supports I2c synchronous transfer.

5. The watchdog is triggered via I2c command, it depends on the I2c speed. When I2c Speed has low speed, the watchdog might not trigger successfully in the watchdog's window open time. To avoid this problem, we recommend that the user should configure I2c with high speed, using a big window period when it is enabled.

6. Pmic driver does not support Ip configuration for Non-AUTOSAR applications. However, the user also uses the HLD configuration for Non-AUTOSAR applications by selecting the Non-AUTOSAR mode in the S32DS configuration.

7. Pmic driver may not initialize the device successfully in case the fault error counter limit is 2. The user avoids configuring the PmicFaultErrorCounterLimit as MAX_2 value in the configuration tool.

8. When a fault is triggered by the MCU via its FCCU pins, the device asserts the FS0B pin if the reaction is configured. Pmic_ReleaseFs0b() is called, if the FCCU error is always reported in the "Error phase", Pmic_ReleaseFs0b() will report E_NOT_OK without the error signal FS0B. The error signal FS0B might not report because the FS0B_SNS bit in the FS_SAFE_IOS register did not report the state immediately. The process takes more time to update state.

3.6 Driver usage and configuration tips

Driver usage and configuration tips

This chapter is showcasing how certain features need to be configured/enabled.

1. Enable Watchdog by using Pmic to configure OTP fail-safe with device burn OTP without a watchdog.

2. If the device enables Watchdog, the user must enable watchdog monitoring in Pmic Configuration as the node PmicOtpConfiguration/PmicOtpConfiguration_0/PmicOtpFailSafeUnitConfiguration/PmicOtpSafety↩ IOConfiguration/PmicWatchdogEnable and select the watchdog type according to watchdog OTP.

3. Pmic_Init() function must be called before another function of the Pmic driver.

4. User can emulate OTP Pmic device by Pmic_EmulateDeviceOTP() API. However, users must ensure the device is in OTP mode. Refer to Pmic VR5510 datasheet to understand this feature.

5. User can initialize device by Pmic_InitDevice() function. This will configure user settings and switch the device to a Normal FS state. Pmic_InitDevice() function shall not configure any Failsafe register if the device already was Initialized and moved to Normal_FS state. Eg: If the device was initialized and switched to Normal_FS in a boot phase, the Pmic_InitDevice() function will return successfully without any failsafe configuration in the application phase. And if users want to re-configure the device in this case, users should use the Pmic_GotoInitFS() function to switch the device back to InitFS mode. When Pmic_GotoInitFS() is called, the watchdog service will be disabled until you call Pmic_InitDevice(). This way makes sure that the driver can ignore trigger activity while the user is changing the user_configuration (failsafe register value). To change the configuration, Pmic_GotoInitFS() must be called and followed by Pmic_InitDevice().
In case the DIE_PROCESS feature is enabled, this function will configure SVS if the DIE_PROCESS OTP of MCU is enabled. It will not configure SVS if the DIE_PROCESS OTP of MCU is disabled. Refer to MCU S32G reference manual to check DIE_PROCESS OTP bit.

6. Pmic_ReadRegister(), Pmic_WriteRegister() are used to reading, writing Pmic's register value.

7. Pmic_InitClock will configure the clock of the Pmic device according to the configuration index. The Analog/↩ F-out pin will show the frequency of the clock source. But users need to check the AMUX_FOUT_OTP bit before the calls function. AMUX_FOUT_OTP must be FOUT_MODE.

8. Pmic_SetMode() will switch the device to the mode according to a configuration index.

9. Pmic_SetAnalogMux() will set analog mux assignment according to a channel configuration index. The Analog/F-out pin will show a voltage of channel mux. But users need to check the AMUX_FOUT_OTP bit before the calls function. AMUX_FOUT_OTP must be AMUX_MODE.

10. Pmic_SwitchSVS() will re-configure the static voltage scale (SVS) according to the SVS configuration index. Firstly, this function will switch the device back to InitFS, configure SVS and switch the device to Normal_FS. In case the DIE_PROCESS feature is enabled, this function will configure SVS if the DIE_PROCESS OTP of MCU enables it. It will not configure SVS if the DIE_PROCESS OTP of MCU is disabled. Refer to Mcu S32G reference manual to check DIE_PROCESS bit.

11. The DIE_PROCESS feature only supports on S32G2xx devices. S32G3xx will not be available.

12. Pmic_ConfigureWatchdog() will configure the Watchdog Window according to the configuration index. Users must use watchdog_notification_task to trigger watchdog when notification is called. The Pmic_TriggerWatchdog() needs to integrate into the notification task and switch to the new Watchdog Window.

13. Pmic_TriggerWatchdog() should trigger the watchdog of the device. 2 options for watchdog type: Challenge or Simple, refer to Pmic datasheet. In case an external watchdog is used, trigger activity will occur if Pmic_InitDevice() is already called before. If the Watchdog Window is enabled, this function should integrate into the watchdog notification task service. This way will make sure the synchronous when the driver changes the Watchdog Window setting. If the Watchdog Window is disabled, this trigger activity is controller by the driver.

14. Pmic_DisableWatchdog() will disable watchdog in run time. This function also switches the device back to InitFS, configure the Watchdog Window disabling, and switch the device to Normal-FS. User should ignore the trigger watchdog by watchdog_notification_task of Pmic driver in case watchdog window already enabled before you call the Pmic_DisableWatchdog().

15. Pmic_ReleaseFs0b() is used to release the safety FS0B pin. The API only execute command if the PMIC device is not in the ASSERT_FS0B state. This function has powerful in case PMIC reacted to assert the FS0B pin and is stuck in the ASSERT_FS0B state. If users want PMIC to release FS0B and switch to the NORMAL_FS state.

16. Pmic_GetRawFaultEvents() will get raw fault events from main-unit and fail-safe-unit flag registers.

17. Pmic_SetReactions() will set reactions when the system events occur.

18. Pmic_GetDeviceInfo() will get device information.

19. Pmic driver supported both AUTOSAR and Non-AUTOSAR. This module has only a 'Pmic' component into S32DS IDE for configuration AUTOSAR and Non-AUTOSAR. Change 'AUTOSAR Mode' or 'Non-AUT↩ OSAR' on Pmic component according to the user's application type. When the user configures 'AUTOSAR Mode', Pmic component requires to import CDD_I2c, CDD_Ocotp, Port, Dio, Det, Dem component. When the user configures 'Non-AUTOSAR Mode', Pmic component requires to import ip component I2c, Ocotp_Ip, Siul2_Port, Siul2_Dio, Det, Dem component. With Non-AUTOSAR case, the driver will call the I2c, Port, Dio and Ocotp Ip API. Users also need to fill data to some fields 'I2C IP Channel Reference, Siul2 instance, I2C SCL Pin reference, Siul2 I2c SCL Pin Mux for using. The project must add the macro definition for using Pmic Non-AUTOSAR.

    - Add the macro 'USE_IPV_IIC' to use I2c Ip in the example.
    - Add the macro 'USE_IPV_SIUL2' to use Port and Dio Ip in the example.

- Add the macro 'USE_IPV_OCOTP' to use Ocotp Ip in the example. Refer to the Pmic_Configuration_Non_AUTOSAR_Example_S32G274A_M7, Pmic_Configuration_Non_AUTOSAR_Example_S32G399A_M7 example to understand how to use Pmic driver Non-AUTOSAR.

20. When User_Mode is enabled in Non-AUTOSAR application, users should set the priority of any to interrupt more than 1. This way will avoid the hard fault handler by any *module_EXCLUSIVE_AREA*.
21. Recommendations on watchdog triggering and error checking:

- Given that transfers are slow even at 1Mbit/s (>40us for a write, > 48us for a read), no API should be called from an interrupt context as it will make the system unresponsive for a good chunk of time.
- Instead, the application software shall log the need for a watchdog trigger or the occurrence of an INTB event in their respective interrupt handlers and defer the "heavy" work to a non-interrupt context (i.e. task-context).
- The priorities of the tasks shall be set in such that the watchdog task could preempt every other task in the system.
- The exclusive areas within Pmic_ReadRegister and Pmic_WriteRegister shall be implemented such that the tasks calling any PMIC API will never preempt each other in the middle of a transfer (also applicable for the watchdog/INTB tasks). Interrupts do not necessarily have to be disabled if these are short enough to not interfere with the underlying synchronous communication activity.
- Given the extremely hardware-specific nature of PMICs and to keep the latencies of the communication channels at a minimum, there is no unified error handling interface available in the API. It is the responsibility of the application software to use the Pmic_GetRawFaultEvents function judiciously.

3.7 Runtime errors

The driver generates the following DEM errors at runtime.

Function	Error Code	Condition triggering the error
Pmic_InitDevice()	PMIC_E_ACESS_FAILURE	Issued when the PMIC device is not accessible.
Pmic_InitDevice()	PMIC_E_TIMEOUT_FAILURE	Issued when the uC/SoC/SiP times-outs while waiting for an event to occur in the PMIC device.
Pmic_InitDevice()	PMIC_E_INTEGRITY_CORRUPTED	Issued when the PMIC device reports BIST (LBIST/ABIST) or OTP failures.
Pmic_InitDevice()	PMIC_E_SIGNAL_SHORTED	Issued when the PMIC device does not successfully pass the safety signal/path checks.
Pmic_InitClock()	PMIC_E_CLOCK_FAILURE	Issued when the PMIC device reports clock failures (e.g. oscillator drifts, unlocked PLL, etc).
Pmic_SetMode()	PMIC_E_TIMEOUT_FAILURE	Issued when the uC/SoC/SiP times-outs while waiting for an event to occur in the PMIC device.

Function	Error Code	Condition triggering the error
Pmic_SwitchSVS()	PMIC_E_TIMEOUT_FAILURE	Issued when the uC/SoC/SiP timeouts while waiting for an event to occur in the PMIC device.
Pmic_SwitchSVS()	PMIC_E_INTEGRITY_CORR↩ UPTED	Issued when the PMIC device reports BIST (LBIST/ABIST) or O↩ TP failures.
Pmic_SwitchSVS()	PMIC_E_SIGNAL_SHORTED	Issued when the PMIC device does not successfully pass the safety signal/path checks.
Pmic_DisableWatchdog()	PMIC_E_TIMEOUT_FAILURE	Issued when the uC/SoC/SiP timeouts while waiting for an event to occur in the PMIC device.
Pmic_DisableWatchdog()	PMIC_E_INTEGRITY_CORR↩ UPTED	Issued when the PMIC device reports BIST (LBIST/ABIST) or O↩ TP failures.
Pmic_DisableWatchdog()	PMIC_E_SIGNAL_SHORTED	Issued when the PMIC device does not successfully pass the safety signal/path checks.
All Functions (Except Pmic_Init and Pmic_GetVersionInfo)	PMIC_E_COMM_FAILURE	Issued when the underlying communication fails, e.g. bus blocked or PMIC device not responding.
All Functions (Except Pmic_Init and Pmic_GetVersionInfo)	PMIC_E_INVALID_DATA	When the CRC verification fails on the master-side.

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 Pmic
  - Container PmicGeneralConfiguration
    - Parameter PmicDevErrorDetect
    - Parameter PmicDisableDemReportErrorStatus
    - Parameter PmicTimeoutMechanism
    - Parameter PmicTimeoutMethod
    - Parameter PmicTimeoutDuration
    - Parameter PmicVersionInfoApi
    - Parameter PmicDeviceInfoApi
    - Parameter PmicSetAnalogMuxApi
    - Parameter PmicWatchdogApi
    - Parameter PmicOtpEmulationModeApi
    - Parameter PmicSwitchSVSApi
    - Parameter PmicExternalWatchdog
    - Parameter PmicWatchdogTaskNotification
    - Parameter PmicDieProcessEnable
    - Parameter PmicEnableUserModeSupport
    - Reference PmicOcotplink
  - Container PmicGlobalConfig
    - Container PmicDevice
      - Parameter PmicDeviceId
      - Parameter PmicDeviceMainI2cAddress
      - Parameter PmicDeviceSafetyI2cAddress
      - Container PmicOtpConfiguration
      - Container PmicOtpMainUnitConfiguration
      - Container PmicOtpDeviceConfiguration
      - Parameter PmicMainI2cAddress
      - Parameter PmicVsupLockoutThreshold
      - Parameter PmicAutoRetryEnable
      - Parameter PmicAutoRetryTimeout

- Parameter PmicAutoRetryLimit
- Parameter PmicPllEnable
- Parameter PmicClk1Divider
- Parameter PmicClk2Divider
- Parameter PmicCenterDieTempThreshold
- Parameter PmicDeepSleepEnable
- Container PmicOtpMainIOConfiguration
- Parameter PmicPwron2GateEnable
- Parameter PmicAmuxFoutPinMode
- Parameter PmicPsyncEnable
- Parameter PmicPsyncMode
- Parameter PmicPsyncPowerDownControlEnable
- Parameter PmicStandbyTimerEnable
- Parameter PmicStandbyDischargeThreshold
- Parameter PmicStandbyPolarity
- Parameter PmicStandbyPGoodEnable
- Parameter PmicVddioSupplySelect
- Container PmicOtpPowerSequenceConfiguration
- Parameter PmicSlotWidth
- Container PmicVpreRegulator
- Parameter PmicVpreEnableRegulator
- Parameter PmicVprePowerupMode
- Parameter PmicVprePhaseDelay
- Parameter PmicVpreClockSelect
- Container PmicBoostRegulator
- Parameter PmicBoostEnableRegulator
- Parameter PmicBoostSlotSelect
- Parameter PmicBoostPhaseDelay
- Parameter PmicBoostClockSelect
- Parameter PmicBoostTsdBehavior
- Container PmicHVLdoRegulator
- Parameter PmicHVLdoEnableRegulator
- Parameter PmicHVLdoSequenceControl
- Parameter PmicHVLdoSlotSelect
- Parameter PmicHVLdoTsdBehavior
- Container PmicBuck1Regulator
- Parameter PmicBuck1EnableRegulator
- Parameter PmicBuck1SlotSelect
- Parameter PmicBuck1PhaseDelay
- Parameter PmicBuck1ClockSelect
- Parameter PmicBuck1TsdBehavior
- Container PmicBuck2Regulator
- Parameter PmicBuck2EnableRegulator
- Parameter PmicBuck2SlotSelect
- Parameter PmicBuck2PhaseDelay
- Parameter PmicBuck2ClockSelect
- Parameter PmicBuck2TsdBehavior

- Container PmicBuck3Regulator
- Parameter PmicBuck3EnableRegulator
- Parameter PmicBuck3SlotSelect
- Parameter PmicBuck3PhaseDelay
- Parameter PmicBuck3ClockSelect
- Parameter PmicBuck3TsdBehavior
- Container PmicLdo1Regulator
- Parameter PmicLdo1EnableRegulator
- Parameter PmicLdo1SlotSelect
- Parameter PmicLdo1TsdBehavior
- Container PmicLdo2Regulator
- Parameter PmicLdo2EnableRegulator
- Parameter PmicLdo2SlotSelect
- Parameter PmicLdo2TsdBehavior
- Container PmicLdo3Regulator
- Parameter PmicLdo3EnableRegulator
- Parameter PmicLdo3SlotSelect
- Parameter PmicLdo3TsdBehavior
- Container PmicOtpRegulatorsConfiguration
- Container PmicVpreRegulator
- Parameter PmicVpreOutputVoltage
- Parameter PmicVpreSlopeCompensation
- Parameter PmicVpreStdbyVoltageControlEnable
- Parameter PmicVpreCsaThreshold
- Parameter PmicVpreHsCurrentCapability
- Parameter PmicVpreLsCurrentCapability
- Parameter PmicVpreSoftStartRamp
- Parameter PmicVpreHsOffTime
- Parameter PmicVpreHsOnTime
- Parameter PmicVpreShutdownDelay
- Container PmicVBoostRegulator
- Parameter PmicVBoostOutputVoltage
- Parameter PmicVBoostSlopeCompensation
- Parameter PmicVBoostLsOnTime
- Parameter PmicVBoostCurrentLimit
- Parameter PmicVBoostSlewRate
- Parameter PmicVBoostCompensationCapacitor
- Parameter PmicVBoostCompensationResistor
- Parameter PmicVBoostToVBos
- Container PmicBuck1Regulator
- Parameter PmicBuck1OutputVoltage
- Parameter PmicBuck1CurrentLimit
- Parameter PmicBuck1InductorSelect
- Parameter PmicBuck1Transconductance
- Parameter PmicBuck1DVSRamp
- Parameter PmicBuck1PhaseMode
- Container PmicBuck2Regulator

- Parameter PmicBuck2OutputVoltage
- Parameter PmicBuck2CurrentLimit
- Parameter PmicBuck2InductorSelect
- Parameter PmicBuck2Transconductance
- Container PmicBuck3Regulator
- Parameter PmicBuck3OutputVoltage
- Parameter PmicBuck3CurrentLimit
- Parameter PmicBuck3InductorSelect
- Parameter PmicBuck3Transconductance
- Parameter PmicBuck3CompensationResistor
- Container PmicLdo1Regulator
- Parameter PmicLdo1OutputVoltage
- Parameter PmicLdo1CurrentLimit
- Container PmicLdo2Regulator
- Parameter PmicLdo2OutputVoltage
- Parameter PmicLdo2OperatingMode
- Container PmicLdo3Regulator
- Parameter PmicLdo3OutputVoltage
- Parameter PmicLdo3OperatingMode
- Container PmicHVLdoRegulator
- Parameter PmicHVLdoOutputVoltage
- Parameter PmicHVLdoTransitionMode
- Container PmicOtpFailSafeUnitConfiguration
- Container PmicOtpSafetyConfiguration
- Parameter PmicSafetyI2cAddress
- Parameter PmicRSTBTimerEnable
- Parameter PmicFaultRecoveryEnable
- Parameter PmicRstbDelay
- Parameter PmicSVSLimit
- Parameter PmicSVSOffsetType
- Parameter PmicRSTBAssertPGOOD
- Parameter PmicHVLDOMONMode
- Parameter PmicLBISTEnable
- Container PmicOtpSafetyIOConfiguration
- Parameter PmicWatchdogEnable
- Parameter PmicWatchdogType
- Parameter PmicWdiPolarity
- Parameter PmicFccu1OperatingMode
- Parameter PmicFccuEnable
- Parameter PmicStandbyModeEnable
- Parameter PmicStandbyPolarity
- Parameter PmicStandbyEntryControl
- Parameter PmicSafetyStandbyWindowEnable
- Parameter PmicSafetyWindowInitTimeout
- Container PmicOtpVoltageMonitorConfiguration
- Container PmicVcoreMonitor
- Parameter PmicVcoreEnableMonitor

- Parameter PmicVcoreMonitorVoltage
- Parameter PmicVcoreMonitorUVThreshold
- Parameter PmicVcoreMonitorOVThreshold
- Parameter PmicVcoreMonitorUVDebounce
- Parameter PmicVcoreMonitorOVDebounce
- Parameter PmicVcoreMonitorPGOODAssert
- Parameter PmicVcoreMonitorABIST1Enable
- Container PmicVddioMonitor
- Parameter PmicVddioEnableMonitor
- Parameter PmicVddioMonitorVoltage
- Parameter PmicVddioMonitorUVThreshold
- Parameter PmicVddioMonitorOVThreshold
- Parameter PmicVddioMonitorUVDebounce
- Parameter PmicVddioMonitorOVDebounce
- Parameter PmicVddioMonitorPGOODAssert
- Parameter PmicVddioMonitorABIST1Enable
- Container PmicHVLdoMonitor
- Parameter PmicHVLdoEnableMonitor
- Parameter PmicHVLdoMonitorVoltage
- Parameter PmicHVLdoMonitorUVThreshold
- Parameter PmicHVLdoMonitorOVThreshold
- Parameter PmicHVLdoMonitorUVDebounce
- Parameter PmicHVLdoMonitorOVDebounce
- Parameter PmicHVLdoMonitorPGOODAssert
- Parameter PmicHVLdoMonitorABIST1Enable
- Container PmicVmon1Monitor
- Parameter PmicVmon1EnableMonitor
- Parameter PmicVmon1MonitorVoltage
- Parameter PmicVmon1MonitorUVThreshold
- Parameter PmicVmon1MonitorOVThreshold
- Parameter PmicVmon1MonitorUVDebounce
- Parameter PmicVmon1MonitorOVDebounce
- Parameter PmicVmon1MonitorPGOODAssert
- Parameter PmicVmon1MonitorABIST1Enable
- Container PmicVmon2Monitor
- Parameter PmicVmon2EnableMonitor
- Parameter PmicVmon2MonitorVoltage
- Parameter PmicVmon2MonitorUVThreshold
- Parameter PmicVmon2MonitorOVThreshold
- Parameter PmicVmon2MonitorUVDebounce
- Parameter PmicVmon2MonitorOVDebounce
- Parameter PmicVmon2MonitorPGOODAssert
- Parameter PmicVmon2MonitorABIST1Enable
- Container PmicVmon3Monitor
- Parameter PmicVmon3EnableMonitor
- Parameter PmicVmon3MonitorVoltage
- Parameter PmicVmon3MonitorUVThreshold

- Parameter PmicVmon3MonitorOVThreshold
- Parameter PmicVmon3MonitorUVDebounce
- Parameter PmicVmon3MonitorOVDebounce
- Parameter PmicVmon3MonitorPGOODAssert
- Parameter PmicVmon3MonitorABIST1Enable
- Container PmicVmon4Monitor
- Parameter PmicVmon4EnableMonitor
- Parameter PmicVmon4MonitorVoltage
- Parameter PmicVmon4MonitorUVThreshold
- Parameter PmicVmon4MonitorOVThreshold
- Parameter PmicVmon4MonitorUVDebounce
- Parameter PmicVmon4MonitorOVDebounce
- Parameter PmicVmon4MonitorPGOODAssert
- Parameter PmicVmon4MonitorABIST1Enable
- Container PmicCommunicationConfiguration
- Parameter PmicI2cCommunicationMethod
- Reference PmicI2cChannelRef
- Reference PmicI2CSCLPinRef
- Reference PmicI2CSCLDioRef
- Container PmicClockSettingConfig
- Parameter PmicClockSettingId
- Parameter PmicIrcoscFrequencyHz
- Parameter PmicLowPowerOscFrequencyHz
- Parameter PmicFinEnable
- Parameter PmicExternalPIN_Fin_FrequencyHz
- Parameter PmicPllClockSelection
- Parameter PmicSpectrumModulationEnable
- Parameter PmicTriangularCarrierFrequencyHz
- Parameter PmicFoutMuxSelection
- Parameter PmicFoutClockSelection
- Parameter PmicFoutPhaseDelay
- Parameter PmicClk1Frequency
- Parameter PmicClk2Frequency
- Container PmicClockReferencePoint
- Parameter PmicClockFrequencySelect
- Parameter PmicClockReferencePointFrequency
- Container PmicModeSettingConf
- Parameter PmicModeID
- Parameter PmicModeSelection
- Parameter PmicPWRON2DeepSleepModeEnable
- Parameter PmicPWRON1WakeUpEnable
- Parameter PmicPWRON2WakeUpEnable
- Parameter PmicStandbyTimerEnable
- Parameter PmicStandbyTimerWindowDuration
- Container PmicRegulatorsConfiguration
- Container PmicVpreRegulator
- Parameter PmicVpreEnableRegulator

- Parameter PmicVpreStandbyOutputVoltage
- Parameter PmicVpreHsCurrentCapability
- Parameter PmicVpreLsCurrentCapability
- Container PmicVBoostRegulator
- Parameter PmicBoostEnableRegulator
- Parameter PmicBoostSlewRate
- Container PmicHVLdoRegulator
- Parameter PmicHVLdoEnableRegulator
- Parameter PmicHVLdoStandbyEnableRegulator
- Container PmicBuck1Regulator
- Parameter PmicBuck1EnableRegulator
- Parameter PmicBuck1StandbyEnableRegulator
- Parameter PmicBuck1StandbyOutputVoltage
- Container PmicBuck2Regulator
- Parameter PmicBuck2EnableRegulator
- Parameter PmicBuck2StandbyEnableRegulator
- Container PmicBuck3Regulator
- Parameter PmicBuck3EnableRegulator
- Parameter PmicBuck3StandbyEnableRegulator
- Container PmicLdo1Regulator
- Parameter PmicLdo1EnableRegulator
- Parameter PmicLdo1StandbyEnableRegulator
- Container PmicLdo2Regulator
- Parameter PmicLdo2EnableRegulator
- Parameter PmicLdo2StandbyEnableRegulator
- Container PmicLdo3Regulator
- Parameter PmicLdo3EnableRegulator
- Parameter PmicLdo3StandbyEnableRegulator
- Container PmicVMONConfiguration
- Parameter PmicVMON4RegulatorAssignment
- Parameter PmicVMON3RegulatorAssignment
- Parameter PmicVMON2RegulatorAssignment
- Parameter PmicVMON1RegulatorAssignment
- Container PmicReactionsSettingConf
- Parameter PmicReactionsSettingId
- Container PmicMainUnitReactionsConf
- Container PmicMainInterruptMasks
- Parameter PmicHvldoOverCurrentIntDisable
- Parameter PmicBuck1OverCurrentIntDisable
- Parameter PmicBuck2OverCurrentIntDisable
- Parameter PmicBuck3OverCurrentIntDisable
- Parameter PmicLdo1OverCurrentIntDisable
- Parameter PmicLdo2OverCurrentIntDisable
- Parameter PmicLdo3OverCurrentIntDisable
- Parameter PmicHvldoTempShutdownIntDisable
- Parameter PmicBoostTempShutdownIntDisable
- Parameter PmicBuck1TempShutdownIntDisable

- Parameter PmicBuck2TempShutdownIntDisable
- Parameter PmicBuck3TempShutdownIntDisable
- Parameter PmicLdo1TempShutdownIntDisable
- Parameter PmicLdo2TempShutdownIntDisable
- Parameter PmicLdo3TempShutdownIntDisable
- Parameter PmicCenterDieTempIntDisable
- Parameter PmicCommunicationErrorIntDisable
- Parameter PmicBosUndervoltageHighIntDisable
- Parameter PmicBoostUndervoltageHighIntDisable
- Parameter PmicBoostOvervoltageHighIntDisable
- Parameter PmicBistTempShutdownIntDisable
- Parameter PmicHvldoInputUndervoltageLowIntDisable
- Parameter PmicVpreOvervoltageIntDisable
- Parameter PmicVpreOvercurrentIntDisable
- Parameter PmicVpreUndervoltageLowIntDisable
- Parameter PmicVpreUndervoltageHighIntDisable
- Parameter PmicVsupUndervoltage7VIntDisable
- Parameter PmicVsupUndervoltageLowIntDisable
- Parameter PmicVsupUndervoltageHighIntDisable
- Parameter PmicPwron2TransitionIntDisable
- Parameter PmicPwron1TransitionIntDisable
- Container PmicMainThermalShutdownBehaviors
- Parameter PmicCenterDieTempThreshold
- Parameter PmicBoostTsdBehavior
- Parameter PmicBuck1TsdBehavior
- Parameter PmicBuck2TsdBehavior
- Parameter PmicBuck3TsdBehavior
- Parameter PmicLdo1TsdBehavior
- Parameter PmicLdo2TsdBehavior
- Parameter PmicLdo3TsdBehavior
- Parameter PmicLHVLdoTsdBehavior
- Container PmicFailSafeUnitReactionsConf
- Container PmicFailSafeInterruptMasks
- Parameter PmicVmon4OverUnderVoltageIntDisable
- Parameter PmicVmon3OverUnderVoltageIntDisable
- Parameter PmicVmon2OverUnderVoltageIntDisable
- Parameter PmicVmon1OverUnderVoltageIntDisable
- Parameter PmicVddioOverUnderVoltageIntDisable
- Parameter PmicVcoremonOverUnderVoltageIntDisable
- Parameter PmicWatchdogBadRefreshIntDisable
- Parameter PmicHvldoOverUnderVoltageIntDisable
- Parameter PmicFccu2EventIntDisable
- Parameter PmicFccu1EventIntDisable
- Container PmicAMUXConfiguration
- Parameter PmicSetAnalogMuxApi
- Container PmicAmuxChannel
- Parameter PmicAmuxChannelId

- Parameter PmicAmuxChannelSelect
- Parameter PmicAmuxRatio
- Container PmicFailSafeConfiguration
- Container PmicSVSConfiguration
- Parameter PmicSwitchSVSApi
- Parameter PmicSVSOffset
- Parameter PmicSVSOffsetSign
- Parameter PmicBuck1SVSOutputVoltage
- Container PmicSVSSettingConf
- Parameter PmicSVSSettingId
- Parameter PmicSVSOffsetSetting
- Parameter PmicSVSOffsetSignSetting
- Parameter PmicBuck1SVSOutputVoltageSetting
- Container PmicFailSafePinReactions
- Parameter PmicVcoremonOvervoltageImpact
- Parameter PmicVcoremonUndervoltageImpact
- Parameter PmicHvldoOvervoltageImpact
- Parameter PmicHvldoUndervoltageImpact
- Parameter PmicVddioOvervoltageImpact
- Parameter PmicVddioUndervoltageImpact
- Parameter PmicVmon4OvervoltageImpact
- Parameter PmicVmon4UndervoltageImpact
- Parameter PmicVmon3OvervoltageImpact
- Parameter PmicVmon3UndervoltageImpact
- Parameter PmicVmon2OvervoltageImpact
- Parameter PmicVmon2UndervoltageImpact
- Parameter PmicVmon1OvervoltageImpact
- Parameter PmicVmon1UndervoltageImpact
- Container PmicWatchdogConfiguration
- Parameter PmicWatchdogApi
- Parameter PmicWatchdogErrorCounterLimit
- Parameter PmicWatchdogRefreshCounterLimit
- Parameter PmicWatchdogErrorImpact
- Parameter PmicWatchdogWindowPeriodEnable
- Parameter PmicWatchdogWindowPeriod_Setting
- Parameter PmicWatchdogClosedWindowDutyCycle_Setting
- Parameter PmicWatchdogRecoveryWindowPeriod
- Parameter PmicWatchdogSeed
- Container PmicWatchdogSettingConf
- Parameter PmicWatchdogSettingId
- Parameter PmicWatchdogWindowPeriod
- Parameter PmicWatchdogClosedWindowDutyCycle
- Container PmicABIST2Configuration
- Parameter PmicHVLdoMonitorABIST2Enable
- Parameter PmicVcoreMonitorABIST2Enable
- Parameter PmicVddioMonitorABIST2Enable
- Parameter PmicVmon1MonitorABIST2Enable

        - Parameter PmicVmon2MonitorABIST2Enable
        - Parameter PmicVmon3MonitorABIST2Enable
        - Parameter PmicVmon4MonitorABIST2Enable
        - Container PmicSafeInputsConfiguration
        - Parameter PmicSafetyStandbyWindowDuration
        - Parameter PmicFccuMonitoringConfiguration
        - Parameter PmicFccu12FaultPolarity
        - Parameter PmicFccu12FaultImpact
        - Parameter PmicFccu1FaultPolarity
        - Parameter PmicFccu1FaultImpact
        - Parameter PmicFccu2FaultPolarity
        - Parameter PmicFccu2FaultImpact
        - Container PmicStateMachineConfiguration
        - Parameter PmicFaultErrorCounterLimit
        - Parameter PmicFaultErrorCounterImpact
        - Parameter PmicResetDuration
        - Parameter PmicBackupSafetyPath
        - Parameter PmicClockMonitoringEnable
        - Parameter PmicRSTBTimerEnable
        - Parameter PmicLowPowerClockMonitoringEnable
      * Container PmicDemEventParameterRefs
        - Reference PMIC_E_ACESS_FAILURE
        - Reference PMIC_E_INTEGRITY_CORRUPTED
        - Reference PMIC_E_SIGNAL_SHORTED
        - Reference PMIC_E_CLOCK_FAILURE
        - Reference PMIC_E_TIMEOUT_FAILURE
  – Container CommonPublishedInformation
      * Parameter ArReleaseMajorVersion
      * Parameter ArReleaseMinorVersion
      * Parameter ArReleaseRevisionVersion
      * Parameter ModuleId
      * Parameter SwMajorVersion
      * Parameter SwMinorVersion
      * Parameter SwPatchVersion
      * Parameter VendorApiInfix
      * Parameter VendorId

4.1 Module Pmic

Configuration of the Power Management Integrated Circuit (PMIC) module.

Included containers:

- PmicGeneralConfiguration
- PmicGlobalConfig
- CommonPublishedInformation

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

4.2 Container PmicGeneralConfiguration

This container contains the global configuration parameters of the Non-Autosar PMIC driver.

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

4.3 Parameter PmicDevErrorDetect

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

This field 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 PMIC_DEV_ERROR_DETECT (STD_ON)/(STD_OFF) will be generated in Pmic_Cfg.h file.

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	true

4.4 Parameter PmicDisableDemReportErrorStatus

Enable/Disable the API for reporting the Dem Error.

#define PMIC_DISABLE_DEM_REPORT_ERROR_STATUS (STD_OFF)/(STD_ON) will be generated in Pmic_Cfg.h file.

STD_ON will be applied if PmicDisableDemReportErrorStatus is true or PmicDemEventParameterRefs is not 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.5 Parameter PmicTimeoutMechanism

This parameter is used to select between two different timeout mechanisms.

When PmicTimeoutMechanism = LOOPS, the driver will use loop counting for timeouts

and the maximum number of iterations will be given by the PmicTimeoutDuration parameter.

#define PMIC_TIMEOUT_MECHANISM_LOOPS (STD_OFF)/(STD_ON) will be generated in Pmic_Cfg.h file.

When PmicTimeoutMechanism = TICKS, the driver will use the Time Service module for deterministic

timeouts and the maximum number of microseconds will be given by the PmicTimeoutDuration parameter.

Internally, it is using the 1us32bit set of API services.

#define PMIC_TIMEOUT_MECHANISM_TICKS (STD_OFF)/(STD_ON) will be generated in Pmic_Cfg.h file.

Note: The latter choice generates an additional dependency.

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	LOOPS
literals	['LOOPS', 'TICKS']

4.6 Parameter PmicTimeoutMethod

Vendor specific: Counter type used in timeout .

Based on selected counter type the timeout value will be interpreted as follows:

OSIF_COUNTER_SYSTEM - Use system counter.

OSIF_COUNTER_CUSTOM - Use custom counter.

OSIF_COUNTER_DUMMY - Use dummy counter.

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

Property	Value
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_SYSTEM
literals	['OSIF_COUNTER_SYSTEM', 'OSIF_COUNTER_CUSTOM', 'OSIF_CO↩ UNTER_DUMMY']

4.7 Parameter PmicTimeoutDuration

When PmicTimeoutMechanism = LOOPS, the unit of measurement is given in number of iterations.

When PmicTimeoutMechanism = TICKS, the unit of measurement is given in number of microseconds.

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

Note: Implementation Specific Parameter.

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

4.8 Parameter PmicVersionInfoApi

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

#define PMIC_VERSION_INFO_API (STD_ON)/(STD_OFF) will be generated in Pmic_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

4.9 Parameter PmicDeviceInfoApi

Pre-processor switch to enable/disable the API to read out the device information (Pmic_GetDeviceInfo).

#define PMIC_DEVICE_INFO_API (STD_ON)/(STD_OFF) will be generated in Pmic_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

4.10 Parameter PmicSetAnalogMuxApi

Pre-processor switch to enable/disable the API to set the analog multiplexer (AMUX) pin.

#define PMIC_SET_ANALOG_MUX_API (STD_ON)/(STD_OFF) will be generated in Pmic_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

4.11 Parameter PmicWatchdogApi

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

#define PMIC_WATCHDOG_API (STD_ON)/(STD_OFF) will be generated in CDD_Pmic_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

4.12 Parameter PmicOtpEmulationModeApi

Pre-processor switch to enable/disable the OTP Emulation Mode API.

#define PMIC_OTP_EMULATION_MODE_API (STD_ON)/(STD_OFF) will be generated in CDD_Pmic_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 PmicSwitchSVSApi

API shall re-configure static voltage scaler (SVS) after device was initialized.

In case DIE_PROCESS feature is enable, this function will configure SVS if the DIE_PROCESS otp of mcu enables.

It will not configure SVS if the DIE_PROCESS otp of mcu disable. Refer Mcu reference manual to check DIE_PROCESS bit.

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

4.14 Parameter PmicExternalWatchdog

External watchdog - true if external watchdog is used (Wdg_43_VR5510), all other watchdog settings in this driver will be ignored.

The driver will assume that the application is using an external watchdog driver instead of directly calling the 'Pmic_TriggerWatchdog' API.

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

4.15 Parameter PmicWatchdogTaskNotification

Function name of callback. This function will be called when watchdog task is required and watchdog window is not DISABLE option.

If this notification is enabled, the driver will assume that the application will service the external watchdog

by directly calling the 'Pmic_TriggerWatchdog' API.

Note: The notification can only be enabled if the Watchdog API is provided. (i.e. PmicGeneralConfiguration/PmicWatchdogAp = 'true').

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 PmicDieProcessEnable

This node allow to enable/disable the checking DIE_PROCESS of MCU before setting static voltage scaler (SVS).

#define PMIC_DIE_PROCESS_ENABLE (STD_ON)/(STD_OFF) will be generated in Pmic_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-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.17 Parameter PmicEnableUserModeSupport

Enable/Disable user mode support.

True: Enabled

False: Disabled

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

4.18 Reference PmicOcotplink

Reference to the Ocotp channel configuration used to check the die_process of MCU.

Ocotp Channel must be configured the eFuse word as OCOTP_BANK0_WORD6

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	true
destination	/AUTOSAR/EcucDefs/Ocotp/OcotpChannelList/OcotpChannel

4.19 Container PmicGlobalConfig

This container contains the global configuration parameter of the PMIC driver. This container is a MultipleConfigurationContai
i.e. this container and its sub-containers exist once per configuration set.

Included subcontainers:

- PmicDevice
- PmicDemEventParameterRefs

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

4.20 Container PmicDevice

This container contains the configuration parameters of a particular PMIC device.

Note: "Users shall use unique names for naming the PMIC devices."

Included subcontainers:

- PmicOtpConfiguration
- PmicCommunicationConfiguration
- PmicClockSettingConfig
- PmicModeSettingConf
- PmicReactionsSettingConf
- PmicAMUXConfiguration
- PmicFailSafeConfiguration

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

4.21 Parameter PmicDeviceId

Identifies the PMIC device.

Note: Implementation Specific Parameter.

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	1
max	255
min	0

4.22 Parameter PmicDeviceMainI2cAddress

The Main I2C address of the device.

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	A_0x20
literals	['A_0x20', 'A_0x22', 'A_0x24', 'A_0x26', 'A_0x28', 'A_0x2A', 'A_0x2C', 'A↩ _0x2E', 'A_0x30', 'A_0x32', 'A_0x34', 'A_0x36', 'A_0x38', 'A_0x3A', 'A_↩ 0x3C', 'A_0x3E']

4.23 Parameter PmicDeviceSafetyI2cAddress

The Fail-Safe Unit I2C address of the device.

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	A_0x21
literals	['A_0x21', 'A_0x23', 'A_0x25', 'A_0x27', 'A_0x29', 'A_0x2B', 'A_0x2D', 'A↩ _0x2F', 'A_0x31', 'A_0x33', 'A_0x35', 'A_0x37', 'A_0x39', 'A_0x3B', 'A_↩ 0x3D', 'A_0x3F']

4.24 Container PmicOtpConfiguration

This container contains the configuration parameters for the OTP registers.

Note: Implementation Specific Parameter.

Included subcontainers:

- PmicOtpMainUnitConfiguration
- PmicOtpFailSafeUnitConfiguration

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

4.25 Container PmicOtpMainUnitConfiguration

This container contains the configuration parameters for the OTP registers on the Main Unit.

Note: Implementation Specific Parameter.

Included subcontainers:

- PmicOtpDeviceConfiguration
- PmicOtpMainIOConfiguration
- PmicOtpPowerSequenceConfiguration
- PmicOtpRegulatorsConfiguration

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

4.26 Container PmicOtpDeviceConfiguration

This container contains the configuration parameters for the general device configuration of the Main Unit.

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

4.27 Parameter PmicMainI2cAddress

Configures the I2C address of the Main Unit (I2CDEVADDR_OTP[3:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	A_0x20
literals	['A_0x20', 'A_0x22', 'A_0x24', 'A_0x26', 'A_0x28', 'A_0x2A', 'A_0x2C', 'A↩ _0x2E', 'A_0x30', 'A_0x32', 'A_0x34', 'A_0x36', 'A_0x38', 'A_0x3A', 'A_↩ 0x3C', 'A_0x3E']

4.28 Parameter PmicVsupLockoutThreshold

Configures the under-voltage low and high thresholds of the VSUP (VSUPCFG_OTP).

It is imperative that VSUP > VSUP_UVH for the device to start.

When VSUP < VSUP_UVL, the device stops operation, disables HVLDO, and goes

directly to OFF mode without the proper power down sequence.

LOW_VOLTAGE_THRESHOLD - VSUP_UVL = 4.2V...4.4V and VSUP_UVH = 4.7V...5.1V

HIGH_VOLTAGE_THRESHOLD - VSUP_UVL = 5.3V...5.7V and VSUP_UVH = 6.0V...6.4V.

Legend:

VSUP_UVL - VSUP under-voltage threshold low (during power-up and VSUP falling)

VSUP_UVH - VSUP under-voltage threshold high (during power-up and VSUP rising)

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	LOW_VOLTAGE_THRESHOLD
literals	['LOW_VOLTAGE_THRESHOLD', 'HIGH_VOLTAGE_THRESHOLD']

4.29 Parameter PmicAutoRetryEnable

This field is used to enable/disable the autoretry feature mechanism used to exit from the DEEP-FS (Deep Fail-Safe) mode (AUTORETRY_EN_OTP).

In case of VPRE_FB_OV detection, or TSD detection on a regulator (if configured), or DFS request from the FS (Fail-Safe) State Machine,

the device will stop and go directly to DEEP-FS mode without the proper power down sequence.

0 - AUTORETRY is Disabled

1 - AUTORETRY 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.30 Parameter PmicAutoRetryTimeout

Configures the autoretry timeout (AUTORETRY_TIMEOUT_OTP).

Exit of DEEP-FS (Deep Fail-Safe) mode is only possible by PWRON1 = 0 or after either

4s or 100ms (the choice depending on the configuration of this field) if the autoretry

feature is activated.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_4S
literals	['TIME_4S', 'TIME_100MS']

4.31 Parameter PmicAutoRetryLimit

Configures the autoretry limit (AUTORETRY_INFINITE_OTP).

The number of autoretries can be limited to 15 or infinite (the choice depending on the

configuration of this field).

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	NO_LIMIT
literals	['TIMES_15_LIMIT', 'NO_LIMIT']

4.32 Parameter PmicPllEnable

This field is used to enable/disable the PLL of the Main Unit (PLL_SEL_OTP).

0 - PLL is Disabled

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

4.33 Parameter PmicClk1Divider

Configures the CLK1 divider of the Main Unit (CLK_DIV1_OTP[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	DIV_9
literals	['DIV_9', 'DIV_8']

4.34 Parameter PmicClk2Divider

Configures the CLK2 divider of the Main Unit (CLK_DIV2_OTP[2:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	DIV_44
literals	['DIV_8', 'DIV_10', 'DIV_44']

4.35 Parameter PmicCenterDieTempThreshold

Configures the center die temperature threshold (DIE_CENTER_TEMP_OTP[2:0]).

A thermal sensor is placed on the center of the die (device) which is used to

generate interrupts for the MCU whenever the temperature exceeds the configured

threshold.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TEMP_105_oC
literals	['TEMP_75_oC', 'TEMP_90_oC', 'TEMP_105_oC', 'TEMP_120_oC', 'TEMP_135_oC', 'TEMP_150_oC']

4.36 Parameter PmicDeepSleepEnable

This field is used to enable/disable the Deep Sleep Mode of the Main Unit (DSM_EN_OTP).

0 - DSM (Deep Sleep Mode) is Disabled

1 - DSM (Deep Sleep Mode) is Enabled

When transitioning to DSM, the device goes through the proper power down sequence set by OTP to reach the deep sleep state where only the HVLDO is kept alive.

The DSM shuts down most of the device, except allows the HVLDO to be in the LDO mode if it is programmed to operate in Deep Sleep. The PWRON2 input detector is active in Deep Sleep and can trigger a turn-on event.

When DSM is enabled, the PWRON2 pin can be programmed to be used to transition to DSM from NORMAL_M.

Warning: Only the PWRON2 input detector is alive during DSM, so only this pin can be used to exit from DSM.

Note: Implementation Specific Parameter.

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

4.37 Container PmicOtpMainIOConfiguration

This container contains the configuration parameters for the I/O capabilities of the Main Unit.

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

4.38 Parameter PmicPwron2GateEnable

Configures the impact of the PWRON2 on the logic that controls turning on

and turning off the regulators (PWRON2_GATE_EN_OTP).

0 - PWRON2 Gate is Disabled (PWRON2 is NOT required for PWRUP)

1 - PWRON2 Gate is Enabled (PWRON2 is required for PWRUP)

Note: Implementation Specific Parameter.

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

4.39 Parameter PmicAmuxFoutPinMode

Configures the internal function of the FOUT/AMUX pin (AMUX_FOUT_OTP).

The AMUX pin delivers 32 analog voltage channels outputs to the MCU ADC input.

The AMUX output is buffered out through the AMUX/FOUT pin. The voltage channels

delivered to AMUX pin can be selected by I2C.

Maximum AMUX output vultage range is VDDIO.

The FOUT pin can be used to send different clocks to synchronize an external IC

or for diagnostic.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	AMUX_MODE
literals	['AMUX_MODE', 'FOUT_MODE']

4.40 Parameter PmicPsyncEnable

This field is used to enable/disable the PSYNC mechanism used to synchronize multiple power management ICs (PSYNC_EN_OTP).

When disabled, the PSYNC pin shall be kept either open or short to GND.

0 - PSYNC Mechanism is Disabled

1 - PSYNC Mechanism 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.41 Parameter PmicPsyncMode

Configures the mode of the PSYNC mechanism (PSYNC_CFG_OTP).

The PSYNC mechanism allows to manage complex start-up sequence with multiple

power management ICs like 2x VR5510 or 1x VR5510 + 1x PF82 (the choice

depending on the configuration of this field).

When PSYNC is used to synchronize two VR5510, the PSYNC pin of each device

must be connected and pulled up to VBOS pin of the VR5510 master device.

When PSYNC is used to synchronize one VR5510 and one PF82, the PSYNC pin of

VR5510 must be connected to the PGOOD pin of PF82 and can be pulled up to

VBOS or VSNS pin.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SYNC_2X_VR5510
literals	['SYNC_2X_VR5510', 'SYNC_VR5510_AND_PF82']

4.42 Parameter PmicPsyncPowerDownControlEnable

This field is used to enable/disable the synchronization of power downs of multiple power management ICs (PSYNC_PWRDW

When enabled, the VR5510 device will power down when PSYNC is low level.

0 - PSYNC PowerDown Control is Disabled

1 - PSYNC PowerDown Control 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	false

4.43 Parameter PmicStandbyTimerEnable

This field is used to enable/disable the standby transition timer (STBY_TIMER_EN_OTP).

0 - Standby Transition Timer is Disabled

1 - Standby Transition Timer is Enabled

The Standby Timer is implemented in the Main-Unit logic to automatically transition

the device to Deep Fail-Safe (DFS) mode in case of timeout during Standby mode.

Note: Implementation Specific Parameter.

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

4.44 Parameter PmicStandbyDischargeThreshold

Configures the standby discharge threshold (STBY_DISCH_OTP).

The device goes to standby mode after verifying that all the disabled

regulators have been discharged to less than 75mV or 150mV (the choice

depending on the configuration of this field).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-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	MAX_75_MV
literals	['MAX_75_MV', 'MAX_150_MV']

4.45 Parameter PmicStandbyPolarity

Configures the polarity of the standby signal (STBY_POLARITY_OTP).

STBY is an input signal that can be connected in the application to the MCU.

The standby mode is entered by toggling the STBY pin to either low or high

(the choice depending on the configuration of this field) or by a software

command from the MCU when conditions are programmed correctly with the

STBY_EN_OTP and STBY_WINDOW_EN_OTP bits.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	HIGH_IN_NORMAL_LOW_IN_STANDBY
literals	['HIGH_IN_NORMAL_LOW_IN_STANDBY', 'LOW_IN_NORMAL_HIG↩ H_IN_STANDBY']

4.46 Parameter PmicStandbyPGoodEnable

This field is used to enable/disable the standby PGOOD output signal (STBY_PGOOD_EN_OTP).

0 - STBY_PGOOD signal is Disabled

1 - STBY_PGOOD signal is Enabled

STBY_PGOOD is an output signal that can be connected in the application to the MCU.

The signal is high in normal mode and it will be asserted low in Standby mode to indicate

a safe transition to the Standby mode when the regulators are discharged below the

OTP configured threshold.

Note: Implementation Specific Parameter.

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

4.47 Parameter PmicVddioSupplySelect

Configures the VDDIO supply selection (VDDIO_REG_ASSIGN_OTP[2:0]).

The VDDIO input pin can be connected to VPRE, LDO1-3, BUCK2-3, or an

external regulator (the choice depending on the configuration of this field).

The regulator connected to VDDIO MUST be at 1.8V or 3.3V

to be compatible with over-voltage and under-voltage monitoring thresholds.

Note: Implementation Specific Parameter.

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

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	LDO3
literals	['EXTERNAL_REGULATOR', 'VPRE', 'LDO1', 'LDO2', 'BUCK2', 'BUCK3', 'LDO3']

4.48 Container PmicOtpPowerSequenceConfiguration

This container contains the configuration parameters for the power sequencing of the Main Unit.

Excepting VPRE, all the other regulator start according to the OTP power sequencing configuration.

Eight slots (SLOT_0 to SLOT_7) are available to program the start-up sequence of the regulators.

Additionally, HVLDO can be programmed to start up in a slot or not.

The power-up sequence starts at SLOT_0 and ends at SLOT_7, while the power-down sequence is executed in reverse order. If not all the available 7 slots are used, the state machine skips past the unused slots.

The regulators assigned to SLOT_7 are not started during the power-up sequence. Instead, they can be started later in NORMAL mode through an I2C command.

Warning: For applications that require the HVLDO to track BUCK1, BUCK1 and HVLDO must be separated by one slot, where HVLDO will start first followed by BUCK1.

Note: Implementation Specific Parameter.

Included subcontainers:

- PmicVpreRegulator
- PmicBoostRegulator
- PmicHVLdoRegulator
- PmicBuck1Regulator
- PmicBuck2Regulator
- PmicBuck3Regulator
- PmicLdo1Regulator
- PmicLdo2Regulator
- PmicLdo3Regulator

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

4.49 Parameter PmicSlotWidth

Controls the width of each slot (SLOT_WIDTH_OTP[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_250US
literals	['TIME_250US', 'TIME_500US', 'TIME_1000US', 'TIME_2000US']

4.50 Container PmicVpreRegulator

This container contains the configuration parameters for the power sequencing of the VPRE (HVBUCK) regulator.

VPRE is the first regulator to start automatically, if enabled, before SLOT_0.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

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

4.51 Parameter PmicVpreEnableRegulator

This field is used to enable/disable the VPRE (HVBUCK) regulator (VPREDIS_OTP).

0 - VPRE is Disabled

1 - VPRE 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.52 Parameter PmicVprePowerupMode

Controls the power-up behavior of the VPRE (HVBUCK) regulator (VPRE_AUTO_ON_OTP).

Note: Implementation Specific Parameter.

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

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	AUTO_ENABLED
literals	['AUTO_ENABLED', 'GATE_ENABLED']

4.53 Parameter PmicVprePhaseDelay

Controls the phase delay of the VPRE (HVBUCK) regulator (VPRE_PH_OTP[2:0]).

The clocks of all the switching regulators (VPRE, BOOST, and BUCK1-3) can be delayed

to avoid all the regulators to turn ON at the same time to reduce peak current and

improve EMC performance.

VPRE has a peak current detection architecture. The PWM synchronizes the turn ON

of the High-Side switch.

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: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	7
min	0

4.54 Parameter PmicVpreClockSelect

Controls the clock selection of the VPRE (HVBUCK) regulator (VPRE_CLK_SEL_OTP).

Warning: The VPRE operation at 2.2MHz is known to be unstable. Output ripple and PWM jitter may occur.

Consider using CLK2 for stability purposes.

There is a known bug when using the CLK1 for VPRE Clock, which results in buggy regulation

for high input voltages (VSUP over 30V).

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	CLK2
literals	['CLK2', 'CLK1']

4.55 Container PmicBoostRegulator

This container contains the configuration parameters for the power sequencing of the BOOST regulator.

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

4.56 Parameter PmicBoostEnableRegulator

This field is used to enable/disable the BOOST regulator (BOOSTEN_OTP).

0 - BOOST is Disabled

1 - BOOST is Enabled

Warning: When disabled, the VBOOST pin must be pulled up

to VPRE and the BOOST_LS pin must be left open.

Note: Implementation Specific Parameter.

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

4.57 Parameter PmicBoostSlotSelect

Controls the slot assignment of the BOOST regulator (BOOSTS_OTP[2:0]).

The regulators assigned to SLOT_7 are not started during the power-up sequence. Instead,

they can be started later in NORMAL mode through an I2C command.

Note: Implementation Specific Parameter.

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

Property	Value
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SLOT_0
literals	['SLOT_0', 'SLOT_1', 'SLOT_2', 'SLOT_3', 'SLOT_4', 'SLOT_5', 'SLOT_6', 'SLOT_7']

4.58 Parameter PmicBoostPhaseDelay

Controls the phase delay of the BOOST regulator (VBST_PH_OTP[2:0]).

The clocks of all the switching regulators (VPRE, BOOST, and BUCK1-3) can be delayed

to avoid all the regulators to turn ON at the same time to reduce peak current and

improve EMC performance.

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: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	7
min	0

4.59 Parameter PmicBoostClockSelect

Controls the clock selection of the BOOST regulator (VBST_CLK_SEL_OTP).

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	CLK1
literals	['CLK2', 'CLK1']

4.60 Parameter PmicBoostTsdBehavior

Controls the reaction/behavior of the BOOST regulator in case of a thermal shutdown event (BOOST_TSDCFG_OTP).

Can be configured to either shutdown only the regulator OR shutdown the regulator and transition the

device into Deep Fail-State (DFS) mode when the thermal shutdown threshold is exceeded.

In case SHUTDOWN_ONLY is selected, the regulator will startup automatically when the temperature goes down.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SHUTDOWN_AND_DFS
literals	['SHUTDOWN_ONLY', 'SHUTDOWN_AND_DFS']

4.61 Container PmicHVLdoRegulator

This container contains the configuration parameters for the power sequencing of the HVLDO regulator.

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

4.62 Parameter PmicHVLdoEnableRegulator

This field is used to enable/disable the HVLDO regulator (HVLDOEN_OTP).

0 - HVLDO is Disabled

1 - HVLDO 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.63 Parameter PmicHVLdoSequenceControl

Controls the power-up behavior of the HVLDO regulator (HVLDO_SLOT_EN_OTP).

HVLDO can be configured to be the first (before VPRE) regulator to start automatically

before SLOT_0 (HV_HVLDO_IN must be supplied by VSUP), or to start according to the assigned slot.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	START_IN_SLOT
literals	['START_FIRST', 'START_IN_SLOT']

4.64 Parameter PmicHVLdoSlotSelect

Controls the slot assignment of the HVLDO regulator (HVLDOS_OTP[2:0]).

The regulators assigned to SLOT_7 are not started during the power-up sequence. Instead,

they can be started later in NORMAL mode through an I2C command.

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-POST-BUILD: PRE-COMPILE

Property	Value
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SLOT_0
literals	['SLOT_0', 'SLOT_1', 'SLOT_2', 'SLOT_3', 'SLOT_4', 'SLOT_5', 'SLOT_6', 'SLOT_7']

4.65 Parameter PmicHVLdoTsdBehavior

Controls the reaction/behavior of the HVLDO regulator in case of a thermal shutdown event (HVLDO_TSDCFG_OTP).

Can be configured to either shutdown only the regulator OR shutdown the regulator and transition the

device into Deep Fail-State (DFS) mode when the thermal shutdown threshold is exceeded.

In case SHUTDOWN_ONLY is selected, the regulator will startup automatically when the temperature goes down.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SHUTDOWN_AND_DFS
literals	['SHUTDOWN_ONLY', 'SHUTDOWN_AND_DFS']

4.66 Container PmicBuck1Regulator

This container contains the configuration parameters for the power sequencing of the BUCK1 regulator.

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

4.67 Parameter PmicBuck1EnableRegulator

This field is used to enable/disable the BUCK1 regulator.

0 - BUCK1 is Disabled

1 - BUCK1 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.68 Parameter PmicBuck1SlotSelect

Controls the slot assignment of the BUCK1 regulator (BUCK1S_OTP[2:0]).

The regulators assigned to SLOT_7 are not started during the power-up sequence. Instead,

they can be started later in NORMAL mode through an I2C command.

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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SLOT_0
literals	['SLOT_0', 'SLOT_1', 'SLOT_2', 'SLOT_3', 'SLOT_4', 'SLOT_5', 'SLOT_6', 'SLOT_7']

4.69 Parameter PmicBuck1PhaseDelay

Controls the phase delay of the BUCK1 regulator (BUCK1_PH_OTP[2:0]).

The clocks of all the switching regulators (VPRE, BOOST, and BUCK1-3) can be delayed

to avoid all the regulators to turn ON at the same time to reduce peak current and

improve EMC performance.

BUCK1 has a valley current detection architecture. The PWM synchronizes the turn

ON of the Low-Side switch.

In dual-phase mode operation, the phase delay of the BUCK2 regulator is automatically

shifted an additional 180 degrees (i.e. BUCK2 Phase Delay (cycles) = PmicBuck1PhaseDelay + 1/2)

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: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	7
min	0

4.70 Parameter PmicBuck1ClockSelect

Controls the clock selection of the BUCK1 regulator (BUCK1_CLK_SEL_OTP).

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	CLK1
literals	['CLK2', 'CLK1']

4.71 Parameter PmicBuck1TsdBehavior

Controls the reaction/behavior of the BUCK1 regulator in case of a thermal shutdown event (BUCK1_TSDCFG_OTP).

Can be configured to either shutdown only the regulator OR shutdown the regulator and transition the

device into Deep Fail-State (DFS) mode when the thermal shutdown threshold is exceeded.

In case SHUTDOWN_ONLY is selected, the regulator will startup automatically when the temperature goes down.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SHUTDOWN_AND_DFS
literals	['SHUTDOWN_ONLY', 'SHUTDOWN_AND_DFS']

4.72 Container PmicBuck2Regulator

This container contains the configuration parameters for the power sequencing of the BUCK2 regulator.

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

4.73 Parameter PmicBuck2EnableRegulator

This field is used to enable/disable the BUCK2 regulator (BUCK2EN_OTP).

0 - BUCK2 is Disabled

1 - BUCK2 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.74 Parameter PmicBuck2SlotSelect

Controls the slot assignment of the BUCK2 regulator (BUCK2S_OTP[2:0]).

The regulators assigned to SLOT_7 are not started during the power-up sequence. Instead,

they can be started later in NORMAL mode through an I2C command.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SLOT_0
literals	['SLOT_0', 'SLOT_1', 'SLOT_2', 'SLOT_3', 'SLOT_4', 'SLOT_5', 'SLOT_6', 'SLOT_7']

4.75 Parameter PmicBuck2PhaseDelay

Controls the phase delay of the BUCK2 regulator (BUCK2_PH_OTP[2:0]).

The clocks of all the switching regulators (VPRE, BOOST, and BUCK1-3) can be delayed

to avoid all the regulators to turn ON at the same time to reduce peak current and

improve EMC performance.

BUCK2 has a valley current detection architecture. The PWM synchronizes the turn

ON of the Low-Side switch.

In dual-phase mode operation, the phase delay of the BUCK2 regulator is automatically

shifted an additional 180 degrees (i.e. BUCK2 Phase Delay (cycles) = PmicBuck1PhaseDelay + 1/2)

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: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	7
min	0

4.76 Parameter PmicBuck2ClockSelect

Controls the clock selection of the BUCK2 regulator (BUCK2_CLK_SEL_OTP).

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	CLK1
literals	['CLK2', 'CLK1']

4.77 Parameter PmicBuck2TsdBehavior

Controls the reaction/behavior of the BUCK2 regulator in case of a thermal shutdown event (BUCK2_TSDCFG_OTP).

Can be configured to either shutdown only the regulator OR shutdown the regulator and transition the

device into Deep Fail-State (DFS) mode when the thermal shutdown threshold is exceeded.

In case SHUTDOWN_ONLY is selected, the regulator will startup automatically when the temperature goes down.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SHUTDOWN_AND_DFS
literals	['SHUTDOWN_ONLY', 'SHUTDOWN_AND_DFS']

4.78 Container PmicBuck3Regulator

This container contains the configuration parameters for the power sequencing of the BUCK3 regulator.

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

4.79 Parameter PmicBuck3EnableRegulator

This field is used to enable/disable the BUCK3 regulator (BUCK3EN_OTP).

0 - BUCK3 is Disabled

1 - BUCK3 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.80 Parameter PmicBuck3SlotSelect

Controls the slot assignment of the BUCK3 regulator (BUCK3S_OTP[2:0]).

The regulators assigned to SLOT_7 are not started during the power-up sequence. Instead,

they can be started later in NORMAL mode through an I2C command.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SLOT_0
literals	['SLOT_0', 'SLOT_1', 'SLOT_2', 'SLOT_3', 'SLOT_4', 'SLOT_5', 'SLOT_6', 'SLOT_7']

4.81 Parameter PmicBuck3PhaseDelay

Controls the phase delay of the BUCK3 regulator (BUCK3_PH_OTP[2:0]).

The clocks of all the switching regulators (VPRE, BOOST, and BUCK1-3) can be delayed

to avoid all the regulators to turn ON at the same time to reduce peak current and

improve EMC performance.

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: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	7
min	0

4.82 Parameter PmicBuck3ClockSelect

Controls the clock selection of the BUCK3 regulator (BUCK3_CLK_SEL_OTP).

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	CLK1
literals	['CLK2', 'CLK1']

4.83 Parameter PmicBuck3TsdBehavior

Controls the reaction/behavior of the BUCK3 regulator in case of a thermal shutdown event (BUCK3_TSDCFG_OTP).

Can be configured to either shutdown only the regulator OR shutdown the regulator and transition the

device into Deep Fail-State (DFS) mode when the thermal shutdown threshold is exceeded.

In case SHUTDOWN_ONLY is selected, the regulator will startup automatically when the temperature goes down.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SHUTDOWN_AND_DFS
literals	['SHUTDOWN_ONLY', 'SHUTDOWN_AND_DFS']

4.84 Container PmicLdo1Regulator

This container contains the configuration parameters for the power sequencing of the LDO1 regulator.

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

4.85 Parameter PmicLdo1EnableRegulator

This field is used to enable/disable the LDO1 regulator (LDO1EN_OTP).

0 - LDO1 is Disabled

1 - LDO1 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.86 Parameter PmicLdo1SlotSelect

Controls the slot assignment of the LDO1 regulator (LDO1S_OTP[2:0]).

The regulators assigned to SLOT_7 are not started during the power-up sequence. Instead,

they can be started later in NORMAL mode through an I2C command.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SLOT_0
literals	['SLOT_0', 'SLOT_1', 'SLOT_2', 'SLOT_3', 'SLOT_4', 'SLOT_5', 'SLOT_6', 'SLOT_7']

4.87 Parameter PmicLdo1TsdBehavior

Controls the reaction/behavior of the LDO1 regulator in case of a thermal shutdown event (LDO1_TSDCFG_OTP).

Can be configured to either shutdown only the regulator OR shutdown the regulator and transition the

device into Deep Fail-State (DFS) mode when the thermal shutdown threshold is exceeded.

In case SHUTDOWN_ONLY is selected, the regulator will startup automatically when the temperature goes down.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SHUTDOWN_AND_DFS
literals	['SHUTDOWN_ONLY', 'SHUTDOWN_AND_DFS']

4.88 Container PmicLdo2Regulator

This container contains the configuration parameters for the power sequencing of the LDO2 regulator.

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

4.89 Parameter PmicLdo2EnableRegulator

This field is used to enable/disable the LDO2 regulator (LDO2EN_OTP).

0 - LDO2 is Disabled

1 - LDO2 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.90 Parameter PmicLdo2SlotSelect

Controls the slot assignment of the LDO2 regulator (LDO2S_OTP[2:0]).

The regulators assigned to SLOT_7 are not started during the power-up sequence. Instead,

they can be started later in NORMAL mode through an I2C command.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SLOT_0
literals	['SLOT_0', 'SLOT_1', 'SLOT_2', 'SLOT_3', 'SLOT_4', 'SLOT_5', 'SLOT_6', 'SLOT_7']

4.91 Parameter PmicLdo2TsdBehavior

Controls the reaction/behavior of the LDO2 regulator in case of a thermal shutdown event (LDO2_TSDCFG_OTP).

Can be configured to either shutdown only the regulator OR shutdown the regulator and transition the

device into Deep Fail-State (DFS) mode when the thermal shutdown threshold is exceeded.

In case SHUTDOWN_ONLY is selected, the regulator will startup automatically when the temperature goes down.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SHUTDOWN_AND_DFS
literals	['SHUTDOWN_ONLY', 'SHUTDOWN_AND_DFS']

4.92 Container PmicLdo3Regulator

This container contains the configuration parameters for the power sequencing of the LDO3 regulator.

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

4.93 Parameter PmicLdo3EnableRegulator

This field is used to enable/disable the LDO3 regulator (LDO3EN_OTP).

0 - LDO3 is Disabled

1 - LDO3 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.94 Parameter PmicLdo3SlotSelect

Controls the slot assignment of the LDO3 regulator (LDO3S_OTP[2:0]).

The regulators assigned to SLOT_7 are not started during the power-up sequence. Instead,

they can be started later in NORMAL mode through an I2C command.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SLOT_0
literals	['SLOT_0', 'SLOT_1', 'SLOT_2', 'SLOT_3', 'SLOT_4', 'SLOT_5', 'SLOT_6', 'SLOT_7']

4.95 Parameter PmicLdo3TsdBehavior

Controls the reaction/behavior of the LDO3 regulator in case of a thermal shutdown event (LDO3_TSDCFG_OTP).

Can be configured to either shutdown only the regulator OR shutdown the regulator and transition the

device into Deep Fail-State (DFS) mode when the thermal shutdown threshold is exceeded.

In case SHUTDOWN_ONLY is selected, the regulator will startup automatically when the temperature goes down.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SHUTDOWN_AND_DFS
literals	['SHUTDOWN_ONLY', 'SHUTDOWN_AND_DFS']

4.96 Container PmicOtpRegulatorsConfiguration

This container contains the configuration parameters for the OTP registers controlling the functionality of regulators.

Note: Implementation Specific Parameter.

Included subcontainers:

- PmicVpreRegulator
- PmicVBoostRegulator
- PmicBuck1Regulator
- PmicBuck2Regulator
- PmicBuck3Regulator

- PmicLdo1Regulator
- PmicLdo2Regulator
- PmicLdo3Regulator
- PmicHVLdoRegulator

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

4.97 Container PmicVpreRegulator

This container contains the configuration parameters of the VPRE (HVBUCK) regulator.

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

4.98 Parameter PmicVpreOutputVoltage

Controls the output voltage of the VPRE (HVBUCK) regulator (VPREV_OTP[5:0]).

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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	OUT_3V3
literals	['OUT_3V3', 'OUT_3V4', 'OUT_3V5', 'OUT_3V7', 'OUT_4V0', 'OUT_4V5', 'OUT_5V0', 'OUT_5V1', 'OUT_5V2']

4.99 Parameter PmicVpreSlopeCompensation

Controls the slope compensation of the VPRE (HVBUCK) regulator (VPRESC_OTP[5:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SLOPE_60
literals	['SLOPE_60', 'SLOPE_70', 'SLOPE_140', 'SLOPE_200']

4.100 Parameter PmicVpreStdbyVoltageControlEnable

This field is used to enable/disable the 3V output voltage selection for VPRE in Standby Mode (VPREV_STDY_EN_OTP).

0 - The output voltage of VPRE in Standby Mode is controlled by VPREV_OTP[5:0]

(i.e. 'PmicVpreOutputVoltage').

1 - The output voltage of VPRE in Standby Mode is controlled according to the M_REG_CTRL3[VPREV_STBY] selection

(i.e. 'PmicModeSettingConf/PmicRegulatorsConfiguration/PmicVpreRegulator/PmicVpreStandbyOutputVoltage').

Note: Implementation Specific Parameter.

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

4.101 Parameter PmicVpreCsaThreshold

Controls the current sense amplifier peak detection threshold of the VPRE (HVBUCK) regulator (VPREILIM_OTP[1:0]).

Warning: The 150mV setting is not available for 2.22MHz. (VPRE Clock Select must be set to CLK2).

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	MAX_120_MV
literals	['MAX_50_MV', 'MAX_80_MV', 'MAX_120_MV', 'MAX_150_MV']

4.102 Parameter PmicVpreHsCurrentCapability

Controls the pull-up current capability of the HS (High-Side) gate driver of the VPRE (HVBUCK) regulator (VPRESRHS_OTP[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	I_130_MA
literals	['I_130_MA', 'I_260_MA', 'I_520_MA', 'I_900_MA']

4.103 Parameter PmicVpreLsCurrentCapability

Controls the pull-down current capability of the LS (Low-Side) gate driver of the VPRE (HVBUCK) regulator (VPRESRLS_OTP[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	I_130_MA
literals	['I_130_MA', 'I_260_MA', 'I_520_MA', 'I_900_MA']

4.104 Parameter PmicVpreSoftStartRamp

Controls the soft start ramp (from 0V to 1V) of the VPRE (HVBUCK) regulator (VPRE_SSRAMP_OTP).

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: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	2
max	2
min	1

4.105 Parameter PmicVpreHsOffTime

Controls the minimum OFF time of the HS (High-Side) gate driver of the VPRE (HVBUCK) regulator (VPRETOFF_OTP[1:0

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_80NS
literals	['TIME_80NS', 'TIME_60NS', 'TIME_40NS', 'TIME_20NS']

4.106 Parameter PmicVpreHsOnTime

Controls the minimum ON time of the HS (High-Side) gate driver of the VPRE (HVBUCK) regulator (VPRE_PFM_TON_O

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_120NS
literals	['TIME_120NS', 'TIME_210NS', 'TIME_300NS', 'TIME_560NS']

4.107 Parameter PmicVpreShutdownDelay

Controls the shutdown delay between VBOOST and VPRE (VPRE_OFF_DLY_OTP).

When shutting down, the device follows the power down sequence to stop all the regulators in the reverse order of the power up sequence. The value of this field is an additional delay added to the Tslot time (PmicSlotWidth) between "SLOT_0" and "VPRE_OFF".

To be used in case VPRE is supplying an external PMIC to wait its power down sequence completion.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_250US
literals	['TIME_250US', 'TIME_32MS']

4.108 Container PmicVBoostRegulator

This container contains the configuration parameters of the VBOOST regulator.

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

4.109 Parameter PmicVBoostOutputVoltage

Controls the output voltage of the VBOOST regulator (VBSTV_OTP[3:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	OUT_5V00
literals	['OUT_4V50', 'OUT_5V00', 'OUT_5V09', 'OUT_5V19', 'OUT_5V40', 'OU↩ T_5V74', 'OUT_6V00']

4.110 Parameter PmicVBoostSlopeCompensation

Controls the slope compensation of the VBOOST regulator (VBSTSC_OTP[4:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SLOPE_67
literals	['SLOPE_160', 'SLOPE_125', 'SLOPE_79', 'SLOPE_67']

4.111 Parameter PmicVBoostLsOnTime

Controls the minimum ON time of the LS (Low-Side) gate driver of the VBOOST regulator (VBSTTONTIME_OTP[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_60NS
literals	['TIME_60NS', 'TIME_50NS', 'TIME_70NS', 'TIME_80NS']

4.112 Parameter PmicVBoostCurrentLimit

Controls (limits) the output current of the VBOOST regulator (VBSTILIM_OTP[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	I_3_A
literals	['I_2_A', 'I_3_A']

4.113 Parameter PmicVBoostSlewRate

Controls the slew rate of the VBOOST regulator (VBSTSR_OTP[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	RATE_500
literals	['RATE_100', 'RATE_300', 'RATE_500']

4.114 Parameter PmicVBoostCompensationCapacitor

Controls the compensation capacitor of the VBOOST regulator (VBSTCCOMP_OTP[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	C_125_PF
literals	['C_125_PF', 'C_75_PF', 'C_175_PF', 'C_225_PF']

4.115 Parameter PmicVBoostCompensationResistor

Controls the compensation resistor of the VBOOST regulator (VBSTRCOMP_OTP[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	R_500_KOHM
literals	['R_750_KOHM', 'R_500_KOHM', 'R_1000_KOHM', 'R_250_KOHM']

4.116 Parameter PmicVBoostToVBos

This field is used to enable/disable the input path to VBOS of the VBOOST regulator (VBOS_VBOOST_OTP).

0 - VBOOST Input Path to BOS is Disabled (BOS will regulate only from VSUP or VPRE)

1 - VBOOST Input Path to BOS is Enabled (BOS will regulate from VBOOST or VSUP or VPRE)

Note: Implementation Specific Parameter.

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

4.117 Container PmicBuck1Regulator

This container contains the configuration parameters of the BUCK1 regulator.

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

4.118 Parameter PmicBuck1OutputVoltage

Controls the output voltage of the BUCK1 regulator (BUCK1V_OTP[7:0]).

BUCK1 (BUCK1/2 in dual-phase mode) is intended to supply the MCU core voltage.

The output voltage must have a resolution of 6.25 mV.

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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0.8
max	1.8
min	0.4

4.119 Parameter PmicBuck1CurrentLimit

Controls (limits) the output current of the BUCK1 regulator (BUCK1_ILIM_OTP[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	I_3_6_A
literals	['I_2_4_A', 'I_3_6_A']

4.120 Parameter PmicBuck1InductorSelect

Controls the inductance (LBuck1) for VBuck1_SW of the BUCK1 regulator (BUCK1_LSELECT_OTP[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	L_1_0_UH
literals	['L_1_0_UH', 'L_0_47_UH', 'L_1_5_UH']

4.121 Parameter PmicBuck1Transconductance

Controls the transconductance of the BUCK1 regulator (BUCK1_COMP_OTP[2:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TRANS_65_uMho
literals	['TRANS_16_25_uMho', 'TRANS_32_5_uMho', 'TRANS_48_75_uMho', 'TRANS_65_uMho', 'TRANS_81_25_uMho', 'TRANS_97_5_uMho', 'TRANS_113_75_uMho']

4.122 Parameter PmicBuck1DVSRamp

Controls the DVS (Dynamic Voltage Scaling) ramp of the BUCK1 regulator (BUCK1DVS_RAMP_OTP).

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	DVS_7_8
literals	['DVS_7_8', 'DVS_15_6']

4.123 Parameter PmicBuck1PhaseMode

Controls the phase mode of the BUCK1 and BUCK2 regulators (VB12MULTIPH_OTP).

BUCK1 and BUCK2 can work independently or in dual phase mode to double the output current capability.

When BUCK1 and BUCK2 are used in single phase, they have independent outputs, working independently.

When BUCK1 and BUCK2 are used in dual phase, they must have the same output voltage

configuration. Any action like TSD, OV, disable by I2C, on BUCK1 will affect BUCK2 and vice versa.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	DUAL_PHASE_MODE
literals	['SINGLE_PHASE_MODE', 'DUAL_PHASE_MODE']

4.124 Container PmicBuck2Regulator

This container contains the configuration parameters of the BUCK2 regulator.

This container is configurable only when the BUCK1 and BUCK2 regulators are

not in double phase mode (i.e. PmicBuck1Regulator/PmicBuck1PhaseMode = 'SINGLE_PHASE_MODE').

Otherwise, BUCK2 will implicitly use the same configuration as BUCK1.

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

4.125 Parameter PmicBuck2OutputVoltage

Controls the output voltage of the BUCK2 regulator (BUCK2V_OTP[7:0]).

The output voltage must have a resolution of 6.25 mV.

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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0.8
max	1.8
min	0.4

4.126 Parameter PmicBuck2CurrentLimit

Controls (limits) the output current of the BUCK2 regulator (BUCK2_ILIM_OTP[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	I_3_6_A
literals	['I_2_4_A', 'I_3_6_A']

4.127 Parameter PmicBuck2InductorSelect

Controls the inductance (LBuck2) for VBuck2_SW of the BUCK2 regulator (BUCK2_LSELECT_OTP[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	L_1_0_UH
literals	['L_1_0_UH', 'L_0_47_UH', 'L_1_5_UH']

4.128 Parameter PmicBuck2Transconductance

Controls the transconductance of the BUCK2 regulator (BUCK2_COMP_OTP[2:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TRANS_65_uMho
literals	['TRANS_16_25_uMho', 'TRANS_32_5_uMho', 'TRANS_48_75_uMho', 'TRANS_65_uMho', 'TRANS_81_25_uMho', 'TRANS_97_5_uMho', 'TRA↩ NS_113_75_uMho']

4.129 Container PmicBuck3Regulator

This container contains the configuration parameters of the BUCK3 regulator.

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

4.130 Parameter PmicBuck3OutputVoltage

Controls the output voltage of the BUCK3 regulator (BUCK3V_OTP[4:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	OUT_1V1
literals	['OUT_1V0', 'OUT_1V1', 'OUT_1V2', 'OUT_1V25', 'OUT_1V3', 'OUT_1↩V35', 'OUT_1V5', 'OUT_1V6', 'OUT_1V8', 'OUT_1V85', 'OUT_2V0', 'O↩UT_2V1', 'OUT_2V15', 'OUT_2V25', 'OUT_2V3', 'OUT_2V4', 'OUT_2V5', 'OUT_2V8', 'OUT_3V15', 'OUT_3V2', 'OUT_3V25', 'OUT_3V3', 'OUT_3↩V35', 'OUT_3V4', 'OUT_3V5', 'OUT_3V8', 'OUT_4V0', 'OUT_4V1']

4.131 Parameter PmicBuck3CurrentLimit

Controls (limits) the output current of the BUCK3 regulator (BUCK3_ILIM_OTP[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	I_3_6_A
literals	['I_2_4_A', 'I_3_6_A']

4.132 Parameter PmicBuck3InductorSelect

Controls the inductance (LBuck3) for VBuck3_SW of the BUCK3 regulator (BUCK3_LSELECT_OTP[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	L_1_0_UH
literals	['L_1_0_UH', 'L_0_47_UH', 'L_1_5_UH']

4.133 Parameter PmicBuck3Transconductance

Controls the transconductance of the BUCK3 regulator (BUCK3_GM_OTP).

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TRANS_65_uMho
literals	['TRANS_65_uMho', 'TRANS_32_5_uMho']

4.134 Parameter PmicBuck3CompensationResistor

Controls the compensation resistor of the BUCK3 regulator (BUCK3_RC_OTP).

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	R_56_KOHM
literals	['R_56_KOHM', 'R_106_KOHM']

4.135 Container PmicLdo1Regulator

This container contains the configuration parameters of the LDO1 regulator.

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

4.136 Parameter PmicLdo1OutputVoltage

Controls the output voltage of the LDO1 regulator (LDO1V_OTP[2:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	OUT_1V8
literals	['OUT_1V1', 'OUT_1V2', 'OUT_1V6', 'OUT_1V8', 'OUT_2V5', 'OUT_2V8', 'OUT_3V3', 'OUT_5V0']

4.137 Parameter PmicLdo1CurrentLimit

Controls (limits) the output current of the LDO1 regulator (LDO1ILIM_OTP).

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	I_400_MA
literals	['I_400_MA', 'I_150_MA']

4.138 Container PmicLdo2Regulator

This container contains the configuration parameters of the LDO2 regulator.

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

4.139 Parameter PmicLdo2OutputVoltage

Controls the output voltage of the LDO2 regulator (LDO2V_OTP[3:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	OUT_1V8
literals	['OUT_1V5', 'OUT_1V6', 'OUT_1V8', 'OUT_1V85', 'OUT_2V15', 'OUT_↩ 2V5', 'OUT_2V8', 'OUT_3V0', 'OUT_3V1', 'OUT_3V15', 'OUT_3V2', 'OU↩ T_3V3', 'OUT_3V35', 'OUT_4V0', 'OUT_4V9', 'OUT_5V0']

4.140 Parameter PmicLdo2OperatingMode

Controls the operating mode (Regulator / Load Switch) of the LDO2 regulator (LDO2_LS_OTP).

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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	REGULATOR_MODE
literals	['REGULATOR_MODE', 'LOAD_SWITCH_MODE']

4.141 Container PmicLdo3Regulator

This container contains the configuration parameters of the LDO3 regulator.

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

4.142 Parameter PmicLdo3OutputVoltage

Controls the output voltage of the LDO3 regulator (LDO3V_OTP[3:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-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	OUT_1V8
literals	['OUT_1V5', 'OUT_1V6', 'OUT_1V8', 'OUT_1V85', 'OUT_2V15', 'OUT_↩ 2V5', 'OUT_2V8', 'OUT_3V0', 'OUT_3V1', 'OUT_3V15', 'OUT_3V2', 'OU↩ T_3V3', 'OUT_3V35', 'OUT_4V0', 'OUT_4V9', 'OUT_5V0']

4.143 Parameter PmicLdo3OperatingMode

Controls the operating mode (Regulator / Load Switch) of the LDO3 regulator (LDO3_LS_OTP).

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	LOAD_SWITCH_MODE
literals	['REGULATOR_MODE', 'LOAD_SWITCH_MODE']

4.144 Container PmicHVLdoRegulator

This container contains the configuration parameters of the HVLDO regulator.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

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

4.145 Parameter PmicHVLdoOutputVoltage

Controls the output voltage of the HVLDO regulator (HVLDOV_OTP[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	OUT_0V8
literals	['OUT_0V8', 'OUT_3V3']

4.146 Parameter PmicHVLdoTransitionMode

Controls the transition mode (LDO in both NORMAL_M and STBY_M / LOAD_SWITCH in NORMAL_M and LDO in STBY_M) of the HVLDO regulator (HVLDO_TRANS_MODE_OTP).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-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	SWITCH_NORMAL_LDO_STDBY
literals	['LDO_MODE_ONLY', 'SWITCH_NORMAL_LDO_STDBY']

4.147 Container PmicOtpFailSafeUnitConfiguration

This container contains the configuration parameters for the OTP registers on the Fail-Safe Unit.

Note: Implementation Specific Parameter.

Included subcontainers:

- PmicOtpSafetyConfiguration
- PmicOtpSafetyIOConfiguration
- PmicOtpVoltageMonitorConfiguration

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

4.148 Container PmicOtpSafetyConfiguration

This container contains the configuration parameters for the general safety configuration of the Fail-Safe Unit.

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

4.149 Parameter PmicSafetyI2cAddress

Configures the I2C address of the Fail-Safe Unit (I2CDEVID_OTP[3:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	A_0x21
literals	['A_0x21', 'A_0x23', 'A_0x25', 'A_0x27', 'A_0x29', 'A_0x2B', 'A_0x2D', 'A↩ _0x2F', 'A_0x31', 'A_0x33', 'A_0x35', 'A_0x37', 'A_0x39', 'A_0x3B', 'A_↩ 0x3D', 'A_0x3F']

4.150 Parameter PmicRSTBTimerEnable

This field is used to enable/disable the 8s RSTB timer (DIS8S_OTP).

If enabled, the 8s RSTB timer starts each time the RSTB pin is asserted low.

When the LBIST is done, the 8s timer monitoring the RSTB pin starts and

the ABIST1 is automatically executed when all the regulators assigned to

ABIST1 have passed their UV and OV checks. When the ABIST1 is done,

RSTB and PGOOD pins are released, and the initialization of the device

is opened for 256ms. ABIST1 fail does not prevent RSTB and PGOOD release.

If RSTB is not released before the RSTB timer expires (i.e. within

8 seconds), the device will go into Deep Fail-Safe.

0 - RSTB Timer is Disabled

1 - RSTB Timer 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.151 Parameter PmicFaultRecoveryEnable

This field is used to enable/disable the fault recovery strategy (FLT_RECOVERY_EN_OTP).
0 - Fault Recovery Strategy is Disabled
1 - Fault Recovery Strategy is Enabled
This function extends the watchdog window to allow the MCU to perform a fault recovery strategy.
The goal is to not reset the MCU while it is trying to recover the application after a failure event.
When a fault is triggered by the MCU via its FCCU pins, the FS0B is asserted by the device and the watchdog window duration becomes automatically an open window (no more duty cycle), whose duration is given by the WDW_RECOVERY[3:0] bits.
The transition from the normal watchdog window duration to this new open window duration happens when the FCCU pin indicates an error and FS0B is asserted. If the MCU sends a good watchdog refresh before the end of the open window duration, the device switches bask to the normal watchdog window duration and associated duty cycle if the FCCU pins do not indicate an error anymore. Otherwise, a new open window period is started. If the MCU does not send a good watchdog refresh before the end of this second open window duration, then a reset pulse is generated and the Fail-Safe State Machine moves back to INIT_FS.
Note: Implementation Specific Parameter.

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

4.152 Parameter PmicRstbDelay

Configures the delay between RSTB assertion and RSTB release (RSTB_DELAY_OTP).

This means that even if the MCU has released its RSTB signal, the VR5510 device

will still wait for this configured delay to expire before releasing

the RSTB signal.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	NO_DELAY
literals	['NO_DELAY', 'DELAY_5MS']

4.153 Parameter PmicSVSLimit

Configures the maximum number of SVS steps allowed (VCORE_SVS_CLAMP_OTP[5:0]).

A SVS (Static Voltage Scaling) mechanism is implemented to allow the MCU to reduce the output voltage initially configured at start-up of BUCK1 (and BUCK2 if used in multiphase) by OTP, to optimize MCU voltage working points, and compensate part to part process variation.

The offset value is configurable with the SVS_OFFSET[5:0] bits. The number of allowed SVS steps configured in this field effectively limits the applicable offset amount.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	MAX_STEP_16
literals	['MAX_STEP_0', 'MAX_STEP_2', 'MAX_STEP_4', 'MAX_STEP_8', 'MAX_STEP_16', 'MAX_STEP_32', 'MAX_STEP_64']

4.154 Parameter PmicSVSOffsetType

Configures the offset type applicable by the SVS mechanism (VCORE_SVS_FULL_OFFSET_OTP).

A SVS (Static Voltage Scaling) mechanism is implemented to allow the MCU to reduce the output voltage initially configured at start-up of BUCK1 (and BUCK2 if used in multiphase) by OTP, to optimize MCU voltage working points, and compensate part to part process variation.

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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	NEGATIVE_OFFSET
literals	['NEGATIVE_OFFSET', 'NEGATIVE_AND_POSITIVE_OFFSET']

4.155 Parameter PmicRSTBAssertPGOOD

Configures whether the PGOOD output signal is also asserted low or not when the RSTB output signal is asserted (RSTB2PGOOD_OTP).

RSTB is an open-drain output that can be connected in the application to the RESET of the MCU.

RSTB requires an external pull-up resistor to VDDIO and a filtering capacitor to GND for immunity.

When RSTB is asserted low, FS0B is also asserted low.

If the RSTB timer is enabled, when RSTB is stuck low for more than 8s, the device transition to DEEP-FS mode.

0 - PGOOD assertion is Disabled

1 - PGOOD assertion 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.156 Parameter PmicHVLDOMONMode

Used to let the Fail-Safe Unit know whether the HVLDO regulator is

operating in LOAD_SWITCH mode or in LDO mode (HVLDO_MODE_OTP).

In LOAD_SWITCH mode, the reference will track internally the BUCK1

DVS DAC.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	LOAD_SWITCH_MODE
literals	['LOAD_SWITCH_MODE', 'LDO_MODE']

4.157 Parameter PmicLBISTEnable

Configures whether LBIST is performed (LBIST_DIS_OTP[7:0]).

The Fail-Safe State Machine includes a LBIST to verify the correct functionality

of the safety logic monitoring. The LBIST is performed after each POR, or after

each wake-up from Standby.

In case of LBIST fail, RSTB and PGOOD are released, but FS0B remains stuck low

and cannot be released.

The typical LBIST duration is 3ms and the maximum LBIST duration is 5ms.

0 - LBIST is Disabled

1 - LBIST 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.158 Container PmicOtpSafetyIOConfiguration

This container contains the configuration parameters for the I/O capabilities of the Fail-Safe Unit.

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

4.159 Parameter PmicWatchdogEnable

This field is used to enable/disable the watchdog monitoring (WD_DIS_OTP).

0 - Watchdog Monitoring is Disabled

1 - Watchdog Monitoring is Enabled

The watchdog is a windowed watchdog. The first half of the windows is said to

be CLOSED, and the second half is said to be OPEN.

A good watchdog refresh is a good watchdog answer during the OPEN window.

A bad watchdog refresh is a bad watchdog answer during the OPEN window,

no watchdog refresh during the OPEN window, or a good watchdog answer

during the CLOSED window.

After a good or a bad watchdog refresh, a new window period starts

immediately for the MCU to keep the synchronization with the windowed

watchdog.

If enabled, the MCU must refresh the watchdog in the OPEN window of the

watchdog window period.

The duration of the watchdog window is configurable from 1ms to 1024ms.

The duty cycle of the watchdog window is configurable from 31.25% to

68.75%.

Note: Implementation Specific Parameter.

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

4.160 Parameter PmicWatchdogType

Configures the type of watchdog (WD_SELECTION_OTP).

The watchdog can be either a simple watchdog or a challenger watchdog.

The simple watchdog monitoring is based on a single configurable watchdog key.

The challenger watchdog monitoring is based on a 16-bits pseudo-random word

generated by a LFSR, whose seed is configurable.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	CHALLENGER_WATCHDOG
literals	['SIMPLE_WATCHDOG', 'CHALLENGER_WATCHDOG']

4.161 Parameter PmicWdiPolarity

Configures the polarity of the WDI signal (WDI_POL_OTP).

This field is modifiable only when PmicFccu1OperatingMode = 'FCCU_MODE'.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	FALLING_EDGE
literals	['FALLING_EDGE', 'RISING_EDGE']

4.162 Parameter PmicFccu1OperatingMode

Configures the operating mode of the FCCU1 pin (FCCU_OR_WDI_OTP).

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	FCCU_MODE
literals	['FCCU_MODE', 'WDI_MODE']

4.163 Parameter PmicFccuEnable

This field is used to enable/disable the FCCU monitoring (FCCU_EN_OTP).

0 - FCCU Monitoring is Disabled

1 - FCCU Monitoring is Enabled

The FCCU input pins are in charge of monitoring HW failure from the MCU.

The FCCU input pins can be configured by pair, or single independent inputs.

The FCCU monitoring is active as soon as the INIT_FS state is closed by the

first good watchdog refresh.

The polarity of the FCCU fault signals as well as the fail-safe reaction on

RSTB and/or FS0B is configurable.

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

4.164 Parameter PmicStandbyModeEnable

This field is used to enable/disable the Low-Power Standby Mode (STBY_EN_OTP).

0 - Standby Mode is Disabled

1 - Standby Mode is Enabled

The low current standby mode is used when the device is required to go into a

minimal supply current mode while maintaining minimal preset output voltages.

The main regulators which are on during LP Standby mode are VPRE and the HVLDO.

The VPRE will be forced to operate in PFM mode, while the HVLDO will operate in

LDO mode. There is an option to operate other regulators (except BOOST) as well,

but the switching regulators will be forced to operate only in PFM.

Note: Implementation Specific Parameter.

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

4.165 Parameter PmicStandbyPolarity

Configures the polarity of the standby signal (STBY_POLARITY_OTP).

STBY is an input signal that can be connected in the application to the MCU.

The standby mode is entered by toggling the STBY pin to either low or high

(the choice depending on the configuration of this field) or by a software

command from the MCU when conditions are programmed correctly with the

STBY_EN_OTP and STBY_WINDOW_EN_OTP bits.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	HIGH_IN_NORMAL_LOW_IN_STANDBY
literals	['HIGH_IN_NORMAL_LOW_IN_STANDBY', 'LOW_IN_NORMAL_HIG↩ H_IN_STANDBY']

4.166 Parameter PmicStandbyEntryControl

Configures the method of entering the LP Standby Mode (STBY_SAFE_DIS_OTP).

The standby mode is entered by either toggling the STBY pin or by a software

command from the MCU (the choice depending on the configuration of this field)

when conditions are programmed correctly with the STBY_EN_OTP and

STBY_WINDOW_EN_OTP bits.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	SW_AND_PIN_CONTROL
literals	['SW_AND_PIN_CONTROL', 'PIN_CONTROL_ONLY']

4.167 Parameter PmicSafetyStandbyWindowEnable

This field is used to enable/disable the Safety Standby Window (STBY_WINDOW_EN_OTP).

0 - Safety Standby Window is Disabled

1 - Safety Standby Window is Enabled

When enabled, a maximum timeout can be configured between the I2C request and

the STBY pin transition.

Note: Implementation Specific Parameter.

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

4.168 Parameter PmicSafetyWindowInitTimeout

When the ABIST1 completes, the RSTB and PGOOD pins are released and the initialization of the device is opened via a programmable window based on the WD_INIT_TIMEOUT_OTP[1:0] bit field (CFG_ 2_OTP register)

if WD_INIT_TIMEOUT_OTP is expire timeout, PMIC device will reset.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_1024MS
literals	['TIME_256MS', 'TIME_1024MS', 'TIME_32_5S', 'TIME_67S']

4.169 Container PmicOtpVoltageMonitorConfiguration

This container contains the configuration parameters for the OTP registers controlling the voltage monitoring aspect of the Fail-Safe Unit.

The voltage supervisor is in charge of over-voltage and under-voltage monitoring of VCOREMON, VDDIO, and VMONx input pins.

When an over-voltage occurs on a regulator monitored by one of these pins, the associated regulator is switched off until the fault is removed.

Note: Implementation Specific Parameter.

Included subcontainers:

- PmicVcoreMonitor
- PmicVddioMonitor
- PmicHVLdoMonitor
- PmicVmon1Monitor
- PmicVmon2Monitor
- PmicVmon3Monitor
- PmicVmon4Monitor

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

4.170 Container PmicVcoreMonitor

This container contains the configuration parameters for the voltage monitoring of the VCOREMON input pin.

VCOREMON input pin is dedicated to BUCK1 or BUCK1/2 in case of multiphase operation.

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

4.171 Parameter PmicVcoreEnableMonitor

This field is used to enable/disable the voltage monitoring of the VCOREMON input pin.

0 - VCOREMON is Disabled

1 - VCOREMON is Enabled

Note: Implementation Specific Parameter.

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

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

4.172 Parameter PmicVcoreMonitorVoltage

Controls the nominal voltage of the VCOREMON input pin (VCORE_V_OTP[7:0]).

VCOREMON must be connected to the BUCK1 (BUCK1/2 in dual-phase mode) regulator output.

The nominal voltage must have a resolution of 6.25 mV.

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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0.8
max	1.8
min	0.4

4.173 Parameter PmicVcoreMonitorUVThreshold

Controls the under-voltage threshold of the VCOREMON input pin (VCOREUVTH_OTP[3:0]).
The threshold is relative to the configured nominal voltage. For example, if this field
is configured as 93.5% and VCOREMON Voltage = 0.8V, then the nominal under-voltage
threshold will be equal to 0.8 * (93.5 / 100) = 0.748 V.
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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	MIN_95_5
literals	['MIN_95_5', 'MIN_95', 'MIN_94_5', 'MIN_94', 'MIN_93_5', 'MIN_93', 'M↩ IN_92_5', 'MIN_92', 'MIN_91_5', 'MIN_91', 'MIN_90_5', 'MIN_90', 'MIN↩ _97_5', 'MIN_97', 'MIN_96_5', 'MIN_96']

4.174 Parameter PmicVcoreMonitorOVThreshold

Controls the over-voltage threshold of the VCOREMON input pin (VCOREOVTH_OTP[3:0]).

The threshold is relative to the configured nominal voltage. For example, if this field

is configured as 106.5% and VCOREMON Voltage = 0.8V, then the nominal over-voltage

threshold will be equal to 0.8 * (106.5 / 100) = 0.852 V.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	MAX_104_5
literals	['MAX_104_5', 'MAX_105', 'MAX_105_5', 'MAX_106', 'MAX_106_5', 'M↩ AX_107', 'MAX_107_5', 'MAX_108', 'MAX_108_5', 'MAX_109', 'MAX_↩ 109_5', 'MAX_110', 'MAX_102_5', 'MAX_103', 'MAX_103_5', 'MAX_104']

4.175 Parameter PmicVcoreMonitorUVDebounce

Configures the under-voltage filtering time (debounce) of the VCOREMON input pin (UV_MCU_OTP[1:0]).

When the monitored voltage is below the nominal under-voltage threshold, this field represents the

amount of time to wait until the under-voltage condition is asserted.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_25US
literals	['TIME_5US', 'TIME_15US', 'TIME_25US', 'TIME_40US']

4.176 Parameter PmicVcoreMonitorOVDebounce

Configures the over-voltage filtering time (debounce) of the VCOREMON input pin (OV_MCU_OTP).

When the monitored voltage is above the nominal over-voltage threshold, this field represents the

amount of time to wait until the over-voltage condition is asserted.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_25US
literals	['TIME_25US', 'TIME_45US']

4.177 Parameter PmicVcoreMonitorPGOODAssert

Configures whether the PGOOD output signal is asserted low or not by the FS_LOGIC when the assigned regulator monitored by VCOREMON (i.e. BUCK1 or BUCK1/2) is in under-voltage or over-voltage (PGOOD_VCORE_OTP)

PGOOD is an open-drain output that can be connected in the application to the PORB of the MCU.

PGOOD requires an external pull-up resistor to VDDIO and a filtering capacitor to GND for immunity.

When PGOOD is asserted low, RSTB and FS0B are also asserted low.

0 - PGOOD assertion is Disabled

1 - PGOOD assertion 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.178 Parameter PmicVcoreMonitorABIST1Enable

Configures whether VCOREMON is checked by ABIST1 or not (ABIST1_VCORE_OTP).

The Fail-Safe State Machine includes two ABISTs to verify the correct functionality of the safety analog monitoring. ABIST1 is executed automatically after each POR or after each wake-up from Standby. In case of ABIST1 fail, RSTB and PGOOD are released, but FS0B remains stuck low and cannot be released.

It is recommended to verify latent faults on VCOREMON, HVLDOMON, and VDDIOMON during ABIST1, while the latent faults on VMONx can be verified either during ABIST1 or

ABIST2.

0 - VCOREMON check by ABIST1 is Disabled

1 - VCOREMON check by ABIST1 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.179 Container PmicVddioMonitor

This container contains the configuration parameters for the voltage monitoring of the VDDIO input pin.

VDDIO input pin can be connected to VPRE, LDO1-3, BUCK2-3, or an external regulator. The regulator connected to VDDIO must be at 1.8V or 3.3V to be compatible with over-voltage and undere-voltage monitoring thresholds.

If an external regulator is connected to VDDIO, then only a flag is reported to the MCU which then must decide the appropriate action to perform.

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

4.180 Parameter PmicVddioEnableMonitor

This field is used to enable/disable the voltage monitoring of the VDDIOMON input pin (VDDIO_VMON_EN_OTP).

0 - VDDIOMON is Disabled

1 - VDDIOMON 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.181 Parameter PmicVddioMonitorVoltage

Controls the nominal voltage of the VDDIO input pin (VDDIO_V_OTP).

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	OUT_3V3
literals	['OUT_1V8', 'OUT_3V3']

4.182 Parameter PmicVddioMonitorUVThreshold

Controls the under-voltage threshold of the VDDIO input pin (VDDIOUVTH_OTP[3:0]).

The threshold is relative to the configured nominal voltage. For example, if this field

is configured as 93.5% and VDDIOMON Voltage = 1.8V, then the nominal under-voltage

threshold will be equal to 1.8 * (93.5 / 100) = 1.683 V.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	MIN_95_5
literals	['MIN_95_5', 'MIN_95', 'MIN_94_5', 'MIN_94', 'MIN_93_5', 'MIN_93', 'M↩ IN_92_5', 'MIN_92', 'MIN_91_5', 'MIN_91', 'MIN_90_5', 'MIN_90', 'MIN↩ _97_5', 'MIN_97', 'MIN_96_5', 'MIN_96']

4.183 Parameter PmicVddioMonitorOVThreshold

Controls the over-voltage threshold of the VDDIO input pin (VDDIOOVTH_OTP[3:0]).

The threshold is relative to the configured nominal voltage. For example, if this field

is configured as 106.5% and VDDIOMON Voltage = 1.8V, then the nominal over-voltage

threshold will be equal to 1.8 * (106.5 / 100) = 1.917 V.

Note: Implementation Specific Parameter.

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

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	MAX_104_5
literals	['MAX_104_5', 'MAX_105', 'MAX_105_5', 'MAX_106', 'MAX_106_5', 'M↩ AX_107', 'MAX_107_5', 'MAX_108', 'MAX_108_5', 'MAX_109', 'MAX_↩ 109_5', 'MAX_110', 'MAX_102_5', 'MAX_103', 'MAX_103_5', 'MAX_104']

4.184 Parameter PmicVddioMonitorUVDebounce

Configures the under-voltage filtering time (debounce) of the VDDIO input pin (UV_VDDIO_OTP[1:0]).

When the monitored voltage is below the nominal under-voltage threshold, this field represents the

amount of time to wait until the under-voltage condition is asserted.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_25US
literals	['TIME_5US', 'TIME_15US', 'TIME_25US', 'TIME_40US']

4.185 Parameter PmicVddioMonitorOVDebounce

Configures the over-voltage filtering time (debounce) of the VDDIO input pin (OV_VDDIO_OTP).

When the monitored voltage is above the nominal over-voltage threshold, this field represents the

amount of time to wait until the over-voltage condition is asserted.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_25US
literals	['TIME_25US', 'TIME_45US']

4.186 Parameter PmicVddioMonitorPGOODAssert

Configures whether the PGOOD output signal is asserted low or not by the FS_LOGIC when the assigned regulator monitored by VDDIOMON is in under-voltage or over-voltage (PGOOD_VDDIO_OTP).

PGOOD is an open-drain output that can be connected in the application to the PORB of the MCU.

PGOOD requires an external pull-up resistor to VDDIO and a filtering capacitor to GND for immunity.

When PGOOD is asserted low, RSTB and FS0B are also asserted low.

0 - PGOOD assertion is Disabled

1 - PGOOD assertion 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.187 Parameter PmicVddioMonitorABIST1Enable

Configures whether VDDIOMON is checked by ABIST1 or not (ABIST1_VDDIO_OTP).

The Fail-Safe State Machine includes two ABISTs to verify the correct functionality

of the safety analog monitoring. ABIST1 is executed automatically after each POR

or after each wake-up from Standby. In case of ABIST1 fail, RSTB and PGOOD are

released, but FS0B remains stuck low and cannot be released.

It is recommended to verify latent faults on VCOREMON, HVLDOMON, and VDDIOMON during

ABIST1, while the latent faults on VMONx can be verified either during ABIST1 or

ABIST2.

0 - VDDIOMON check by ABIST1 is Disabled

1 - VDDIOMON check by ABIST1 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.188 Container PmicHVLdoMonitor

This container contains the configuration parameters for the voltage monitoring of the HVLDO regulator.

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

4.189 Parameter PmicHVLdoEnableMonitor

This field is used to enable/disable the voltage monitoring of the HVLDO regulator (HVLDO_VMON_EN_OTP).

0 - HVLDOMON is Disabled

1 - HVLDOMON 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.190 Parameter PmicHVLdoMonitorVoltage

Controls the nominal voltage of the HVLDO regulator (HVLDO_V_OTP[1:0]).

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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	OUT_0V8
literals	['OUT_0V8', 'OUT_3V3']

4.191 Parameter PmicHVLdoMonitorUVThreshold

Controls the under-voltage threshold of the HVLDO regulator (HVLDO_VMON_UVTH_OTP[3:0]).

The threshold is relative to the configured nominal voltage. For example, if this field

is configured as 93.5% and HVLDOMON Voltage = 0.8V, then the nominal under-voltage

threshold will be equal to 0.8 * (93.5 / 100) = 0.748 V.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	MIN_95_5
literals	['MIN_95_5', 'MIN_95', 'MIN_94_5', 'MIN_94', 'MIN_93_5', 'MIN_93', 'M↩ IN_92_5', 'MIN_92', 'MIN_91_5', 'MIN_91', 'MIN_90_5', 'MIN_90', 'MIN↩ _97_5', 'MIN_97', 'MIN_96_5', 'MIN_96']

4.192 Parameter PmicHVLdoMonitorOVThreshold

Controls the over-voltage threshold of the HVLDO regulator (HVLD_VMON_OVTH_OTP[3:0]).

The threshold is relative to the configured nominal voltage. For example, if this field

is configured as 106.5% and HVLDOMON Voltage = 0.8V, then the nominal over-voltage

threshold will be equal to 0.8 * (106.5 / 100) = 0.852 V.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	MAX_104_5
literals	['MAX_104_5', 'MAX_105', 'MAX_105_5', 'MAX_106', 'MAX_106_5', 'M↩ AX_107', 'MAX_107_5', 'MAX_108', 'MAX_108_5', 'MAX_109', 'MAX_↩ 109_5', 'MAX_110', 'MAX_102_5', 'MAX_103', 'MAX_103_5', 'MAX_104']

4.193 Parameter PmicHVLdoMonitorUVDebounce

Configures the under-voltage filtering time (debounce) of the HVLDO regulator (UV_HVLDO_OTP[1:0]).

When the monitored voltage is below the nominal under-voltage threshold, this field represents the

amount of time to wait until the under-voltage condition is asserted.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_25US
literals	['TIME_5US', 'TIME_15US', 'TIME_25US', 'TIME_40US']

4.194 Parameter PmicHVLdoMonitorOVDebounce

Configures the over-voltage filtering time (debounce) of the HVLDO regulator (OV_HVLDO_OTP).

When the monitored voltage is above the nominal over-voltage threshold, this field represents the amount of time to wait until the over-voltage condition is asserted.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_25US
literals	['TIME_25US', 'TIME_45US']

4.195 Parameter PmicHVLdoMonitorPGOODAssert

Configures whether the PGOOD output signal is asserted low or not by the FS_LOGIC when the assigned regulator monitored by HVLDOMON is in under-voltage or over-voltage (PGOOD_HVLDO_VMON_OTP).

PGOOD is an open-drain output that can be connected in the application to the PORB of the MCU.

PGOOD requires an external pull-up resistor to VDDIO and a filtering capacitor to GND for immunity.

When PGOOD is asserted low, RSTB and FS0B are also asserted low.

0 - PGOOD assertion is Disabled

1 - PGOOD assertion is Enabled

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

4.196 Parameter PmicHVLdoMonitorABIST1Enable

Configures whether HVLDOMON is checked by ABIST1 or not (ABIST1_HVLDO_VMON_OTP).

The Fail-Safe State Machine includes two ABISTs to verify the correct functionality

of the safety analog monitoring. ABIST1 is executed automatically after each POR

or after each wake-up from Standby. In case of ABIST1 fail, RSTB and PGOOD are

released, but FS0B remains stuck low and cannot be released.

It is recommended to verify latent faults on VCOREMON, HVLDOMON, and VDDIOMON during

ABIST1, while the latent faults on VMONx can be verified either during ABIST1 or

ABIST2.

0 - HVLDOMON check by ABIST1 is Disabled

1 - HVLDOMON check by ABIST1 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.197 Container PmicVmon1Monitor

This container contains the configuration parameters for the voltage monitoring of the VMON1 input pin.

VMON1 input pin can be be connected to VPRE, LDO1-3, BUCK2 (in case BUCK2 is not used in multiphase), BUCK3,

or an external regulator. The configuration of which regulator is connected to VMON1 is given in the

M_VMON_REG register.

If an external regulator is connected to VMON1, then only a flag is reported to the MCU which then

must decide the appropriate action to perform.

Warning: An external resistor bridge must be connected to VMON1 and it shall be calculated

to deliver a middle point of 0.8V. It is recommended to use +/- 1% or less resistor accuracy.

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

4.198 Parameter PmicVmon1EnableMonitor

This field is used to enable/disable the voltage monitoring of the VMON1 input pin (VMON1_EN_OTP).

0 - VMON1 is Disabled

1 - VMON1 is Enabled

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.199 Parameter PmicVmon1MonitorVoltage

Controls the nominal voltage of the VMON1 input pin.

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: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0.8

4.200 Parameter PmicVmon1MonitorUVThreshold

Controls the under-voltage threshold of the VMON1 input pin (VMON1UVTH_OTP[3:0]).

The threshold is relative to the configured nominal voltage. For example, if this field

is configured as 93.5% and VMON1 Voltage = 0.8V, then the nominal under-voltage

threshold will be equal to 0.8 * (93.5 / 100) = 0.748 V.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	MIN_95_5
literals	['MIN_95_5', 'MIN_95', 'MIN_94_5', 'MIN_94', 'MIN_93_5', 'MIN_93', 'M↩ IN_92_5', 'MIN_92', 'MIN_91_5', 'MIN_91', 'MIN_90_5', 'MIN_90', 'MIN↩ _97_5', 'MIN_97', 'MIN_96_5', 'MIN_96']

4.201 Parameter PmicVmon1MonitorOVThreshold

Controls the over-voltage threshold of the VMON1 input pin (VMON1OVTH_OTP[3:0]).

The threshold is relative to the configured nominal voltage. For example, if this field

is configured as 106.5% and VMON1 Voltage = 0.8V, then the nominal over-voltage

threshold will be equal to 0.8 * (106.5 / 100) = 0.852 V.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	MAX_104_5
literals	['MAX_104_5', 'MAX_105', 'MAX_105_5', 'MAX_106', 'MAX_106_5', 'M↩ AX_107', 'MAX_107_5', 'MAX_108', 'MAX_108_5', 'MAX_109', 'MAX_↩ 109_5', 'MAX_110', 'MAX_102_5', 'MAX_103', 'MAX_103_5', 'MAX_104']

4.202 Parameter PmicVmon1MonitorUVDebounce

Configures the under-voltage filtering time (debounce) of the VMON1 input pin (UV_VMON1_OTP[1:0]).

When the monitored voltage is below the nominal under-voltage threshold, this field represents the amount of time to wait until the under-voltage condition is asserted.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_25US
literals	['TIME_5US', 'TIME_15US', 'TIME_25US', 'TIME_40US']

4.203 Parameter PmicVmon1MonitorOVDebounce

Configures the over-voltage filtering time (debounce) of the VMON1 input pin (OV_VMON1_OTP).

When the monitored voltage is above the nominal over-voltage threshold, this field represents the amount of time to wait until the over-voltage condition is asserted.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_25US
literals	['TIME_25US', 'TIME_45US']

4.204 Parameter PmicVmon1MonitorPGOODAssert

Configures whether the PGOOD output signal is asserted low or not by the FS_LOGIC when the assigned regulator monitored by VMON1 is in under-voltage or over-voltage (PGOOD_VMON1_OTP).

PGOOD is an open-drain output that can be connected in the application to the PORB of the MCU.

PGOOD requires an external pull-up resistor to VDDIO and a filtering capacitor to GND for immunity.

When PGOOD is asserted low, RSTB and FS0B are also asserted low.

0 - PGOOD assertion is Disabled

1 - PGOOD assertion 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.205 Parameter PmicVmon1MonitorABIST1Enable

Configures whether VMON1 is checked by ABIST1 or not (ABIST1_VMON1_OTP).

The Fail-Safe State Machine includes two ABISTs to verify the correct functionality of the safety analog monitoring. ABIST1 is executed automatically after each POR or after each wake-up from Standby. In case of ABIST1 fail, RSTB and PGOOD are released, but FS0B remains stuck low and cannot be released.

It is recommended to verify latent faults on VCOREMON, HVLDOMON, and VDDIOMON during ABIST1, while the latent faults on VMONx can be verified either during ABIST1 or

ABIST2.

0 - VMON1 check by ABIST1 is Disabled

1 - VMON1 check by ABIST1 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.206 Container PmicVmon2Monitor

This container contains the configuration parameters for the voltage monitoring of the VMON2 input pin.

VMON2 input pin can be be connected to VPRE, LDO1-3, BUCK2 (in case BUCK2 is not used in multiphase), BUCK3,

or an external regulator. The configuration of which regulator is connected to VMON2 is given in the

M_VMON_REG register.

If an external regulator is connected to VMON2, then only a flag is reported to the MCU which then

must decide the appropriate action to perform.

Warning: An external resistor bridge must be connected to VMON2 and it shall be calculated

to deliver a middle point of 0.8V. It is recommended to use +/- 1% or less resistor accuracy.

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

4.207 Parameter PmicVmon2EnableMonitor

This field is used to enable/disable the voltage monitoring of the VMON2 input pin (VMON2_EN_OTP).

0 - VMON2 is Disabled

1 - VMON2 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.208 Parameter PmicVmon2MonitorVoltage

Controls the nominal voltage of the VMON2 input pin.

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: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0.8

4.209 Parameter PmicVmon2MonitorUVThreshold

Controls the under-voltage threshold of the VMON2 input pin (VMON2UVTH_OTP[3:0]).

The threshold is relative to the configured nominal voltage. For example, if this field

is configured as 93.5% and VMON2 Voltage = 0.8V, then the nominal under-voltage

threshold will be equal to 0.8 * (93.5 / 100) = 0.748 V.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	MIN_95_5
literals	['MIN_95_5', 'MIN_95', 'MIN_94_5', 'MIN_94', 'MIN_93_5', 'MIN_93', 'M↩ IN_92_5', 'MIN_92', 'MIN_91_5', 'MIN_91', 'MIN_90_5', 'MIN_90', 'MIN↩ _97_5', 'MIN_97', 'MIN_96_5', 'MIN_96']

4.210 Parameter PmicVmon2MonitorOVThreshold

Controls the over-voltage threshold of the VMON2 input pin (VMON2OVTH_OTP[3:0]).

The threshold is relative to the configured nominal voltage. For example, if this field

is configured as 106.5% and VMON2 Voltage = 0.8V, then the nominal over-voltage

threshold will be equal to 0.8 * (106.5 / 100) = 0.852 V.

Note: Implementation Specific Parameter.

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

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	MAX_104_5
literals	['MAX_104_5', 'MAX_105', 'MAX_105_5', 'MAX_106', 'MAX_106_5', 'M↩ AX_107', 'MAX_107_5', 'MAX_108', 'MAX_108_5', 'MAX_109', 'MAX_↩ 109_5', 'MAX_110', 'MAX_102_5', 'MAX_103', 'MAX_103_5', 'MAX_104']

4.211 Parameter PmicVmon2MonitorUVDebounce

Configures the under-voltage filtering time (debounce) of the VMON2 input pin (UV_VMON2_OTP[1:0]).

When the monitored voltage is below the nominal under-voltage threshold, this field represents the

amount of time to wait until the under-voltage condition is asserted.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_25US
literals	['TIME_5US', 'TIME_15US', 'TIME_25US', 'TIME_40US']

4.212 Parameter PmicVmon2MonitorOVDebounce

Configures the over-voltage filtering time (debounce) of the VMON2 input pin (OV_VMON2_OTP).

When the monitored voltage is above the nominal over-voltage threshold, this field represents the

amount of time to wait until the over-voltage condition is asserted.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_25US
literals	['TIME_25US', 'TIME_45US']

4.213 Parameter PmicVmon2MonitorPGOODAssert

Configures whether the PGOOD output signal is asserted low or not by the FS_LOGIC when the assigned regulator monitored by VMON2 is in under-voltage or over-voltage (PGOOD_VMON2_OTP).

PGOOD is an open-drain output that can be connected in the application to the PORB of the MCU.

PGOOD requires an external pull-up resistor to VDDIO and a filtering capacitor to GND for immunity.

When PGOOD is asserted low, RSTB and FS0B are also asserted low.

0 - PGOOD assertion is Disabled

1 - PGOOD assertion 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.214 Parameter PmicVmon2MonitorABIST1Enable

Configures whether VMON2 is checked by ABIST1 or not (ABIST1_VMON2_OTP).

The Fail-Safe State Machine includes two ABISTs to verify the correct functionality

of the safety analog monitoring. ABIST1 is executed automatically after each POR

or after each wake-up from Standby. In case of ABIST1 fail, RSTB and PGOOD are

released, but FS0B remains stuck low and cannot be released.

It is recommended to verify latent faults on VCOREMON, HVLDOMON, and VDDIOMON during

ABIST1, while the latent faults on VMONx can be verified either during ABIST1 or

ABIST2.

0 - VMON2 check by ABIST1 is Disabled

1 - VMON2 check by ABIST1 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.215 Container PmicVmon3Monitor

This container contains the configuration parameters for the voltage monitoring of the VMON3 input pin.

VMON3 input pin can be be connected to VPRE, LDO1-3, BUCK2 (in case BUCK2 is not used in multiphase), BUCK3,

or an external regulator. The configuration of which regulator is connected to VMON3 is given in the

M_VMON_REG register.

If an external regulator is connected to VMON3, then only a flag is reported to the MCU which then must decide the appropriate action to perform.

Warning: An external resistor bridge must be connected to VMON3 and it shall be calculated to deliver a middle point of 0.8V. It is recommended to use +/- 1% or less resistor accuracy.

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

4.216 Parameter PmicVmon3EnableMonitor

This field is used to enable/disable the voltage monitoring of the VMON3 input pin (VMON3_EN_OTP).

0 - VMON3 is Disabled

1 - VMON3 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.217 Parameter PmicVmon3MonitorVoltage

Controls the nominal voltage of the VMON3 input pin.

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: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0.8

4.218 Parameter PmicVmon3MonitorUVThreshold

Controls the under-voltage threshold of the VMON3 input pin (VMON3UVTH_OTP[3:0]).

The threshold is relative to the configured nominal voltage. For example, if this field

is configured as 93.5% and VMON3 Voltage = 0.8V, then the nominal under-voltage

threshold will be equal to 0.8 * (93.5 / 100) = 0.748 V.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	MIN_95_5
literals	['MIN_95_5', 'MIN_95', 'MIN_94_5', 'MIN_94', 'MIN_93_5', 'MIN_93', 'M↩IN_92_5', 'MIN_92', 'MIN_91_5', 'MIN_91', 'MIN_90_5', 'MIN_90', 'MIN↩_97_5', 'MIN_97', 'MIN_96_5', 'MIN_96']

4.219 Parameter PmicVmon3MonitorOVThreshold

Controls the over-voltage threshold of the VMON3 input pin (VMON3OVTH_OTP[3:0]).

The threshold is relative to the configured nominal voltage. For example, if this field

is configured as 106.5% and VMON3 Voltage = 0.8V, then the nominal over-voltage

threshold will be equal to 0.8 * (106.5 / 100) = 0.852 V.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	MAX_104_5
literals	['MAX_104_5', 'MAX_105', 'MAX_105_5', 'MAX_106', 'MAX_106_5', 'M↩ AX_107', 'MAX_107_5', 'MAX_108', 'MAX_108_5', 'MAX_109', 'MAX_↩ 109_5', 'MAX_110', 'MAX_102_5', 'MAX_103', 'MAX_103_5', 'MAX_104']

4.220 Parameter PmicVmon3MonitorUVDebounce

Configures the under-voltage filtering time (debounce) of the VMON3 input pin (UV_VMON3_OTP[1:0]).

When the monitored voltage is below the nominal under-voltage threshold, this field represents the

amount of time to wait until the under-voltage condition is asserted.

Note: Implementation Specific Parameter.

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

Property	Value
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_25US
literals	['TIME_5US', 'TIME_15US', 'TIME_25US', 'TIME_40US']

4.221 Parameter PmicVmon3MonitorOVDebounce

Configures the over-voltage filtering time (debounce) of the VMON3 input pin (OV_VMON3_OTP).

When the monitored voltage is above the nominal over-voltage threshold, this field represents the amount of time to wait until the over-voltage condition is asserted.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_25US
literals	['TIME_25US', 'TIME_45US']

4.222 Parameter PmicVmon3MonitorPGOODAssert

Configures whether the PGOOD output signal is asserted low or not by the FS_LOGIC when the assigned regulator monitored by VMON3 is in under-voltage or over-voltage (PGOOD_VMON3_OTP).

PGOOD is an open-drain output that can be connected in the application to the PORB of the MCU.

PGOOD requires an external pull-up resistor to VDDIO and a filtering capacitor to GND for immunity.

When PGOOD is asserted low, RSTB and FS0B are also asserted low.

0 - PGOOD assertion is Disabled

1 - PGOOD assertion 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.223 Parameter PmicVmon3MonitorABIST1Enable

Configures whether VMON3 is checked by ABIST1 or not (ABIST1_VMON3_OTP).

The Fail-Safe State Machine includes two ABISTs to verify the correct functionality

of the safety analog monitoring. ABIST1 is executed automatically after each POR

or after each wake-up from Standby. In case of ABIST1 fail, RSTB and PGOOD are

released, but FS0B remains stuck low and cannot be released.

It is recommended to verify latent faults on VCOREMON, HVLDOMON, and VDDIOMON during

ABIST1, while the latent faults on VMONx can be verified either during ABIST1 or

ABIST2.

0 - VMON3 check by ABIST1 is Disabled

1 - VMON3 check by ABIST1 is Enabled

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF

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

4.224 Container PmicVmon4Monitor

This container contains the configuration parameters for the voltage monitoring of the VMON4 input pin.

VMON4 input pin can be be connected to VPRE, LDO1-3, BUCK2 (in case BUCK2 is not used in multiphase), BUCK3,

or an external regulator. The configuration of which regulator is connected to VMON4 is given in the

M_VMON_REG register.

If an external regulator is connected to VMON4, then only a flag is reported to the MCU which then

must decide the appropriate action to perform.

Warning: An external resistor bridge must be connected to VMON4 and it shall be calculated

to deliver a middle point of 0.8V. It is recommended to use +/- 1% or less resistor accuracy.

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

4.225 Parameter PmicVmon4EnableMonitor

This field is used to enable/disable the voltage monitoring of the VMON4 input pin (VMON4_EN_OTP).

0 - VMON4 is Disabled

1 - VMON4 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.226 Parameter PmicVmon4MonitorVoltage

Controls the nominal voltage of the VMON4 input pin.

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: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0.8

4.227 Parameter PmicVmon4MonitorUVThreshold

Controls the under-voltage threshold of the VMON4 input pin (VMON4UVTH_OTP[3:0]).

The threshold is relative to the configured nominal voltage. For example, if this field

is configured as 93.5% and VMON4 Voltage = 0.8V, then the nominal under-voltage

threshold will be equal to 0.8 * (93.5 / 100) = 0.748 V.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	MIN_95_5
literals	['MIN_95_5', 'MIN_95', 'MIN_94_5', 'MIN_94', 'MIN_93_5', 'MIN_93', 'M↩ IN_92_5', 'MIN_92', 'MIN_91_5', 'MIN_91', 'MIN_90_5', 'MIN_90', 'MIN↩ _97_5', 'MIN_97', 'MIN_96_5', 'MIN_96']

4.228 Parameter PmicVmon4MonitorOVThreshold

Controls the over-voltage threshold of the VMON4 input pin (VMON4OVTH_OTP[3:0]).

The threshold is relative to the configured nominal voltage. For example, if this field

is configured as 106.5% and VMON4 Voltage = 0.8V, then the nominal over-voltage

threshold will be equal to 0.8 * (106.5 / 100) = 0.852 V.

Note: Implementation Specific Parameter.

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

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	MAX_104_5
literals	['MAX_104_5', 'MAX_105', 'MAX_105_5', 'MAX_106', 'MAX_106_5', 'M↩ AX_107', 'MAX_107_5', 'MAX_108', 'MAX_108_5', 'MAX_109', 'MAX_↩ 109_5', 'MAX_110', 'MAX_102_5', 'MAX_103', 'MAX_103_5', 'MAX_104']

4.229 Parameter PmicVmon4MonitorUVDebounce

Configures the under-voltage filtering time (debounce) of the VMON4 input pin (UV_VMON4_OTP[1:0]).

When the monitored voltage is below the nominal under-voltage threshold, this field represents the

amount of time to wait until the under-voltage condition is asserted.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_25US
literals	['TIME_5US', 'TIME_15US', 'TIME_25US', 'TIME_40US']

4.230 Parameter PmicVmon4MonitorOVDebounce

Configures the over-voltage filtering time (debounce) of the VMON4 input pin (OV_VMON4_OTP).

When the monitored voltage is above the nominal over-voltage threshold, this field represents the

amount of time to wait until the over-voltage condition is asserted.

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-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	TIME_25US
literals	['TIME_25US', 'TIME_45US']

4.231 Parameter PmicVmon4MonitorPGOODAssert

Configures whether the PGOOD output signal is asserted low or not by the FS_LOGIC when the assigned regulator monitored by VMON4 is in under-voltage or over-voltage (PGOOD_VMON4_OTP).

PGOOD is an open-drain output that can be connected in the application to the PORB of the MCU.

PGOOD requires an external pull-up resistor to VDDIO and a filtering capacitor to GND for immunity.

When PGOOD is asserted low, RSTB and FS0B are also asserted low.

0 - PGOOD assertion is Disabled

1 - PGOOD assertion 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.232 Parameter PmicVmon4MonitorABIST1Enable

Configures whether VMON4 is checked by ABIST1 or not (ABIST1_VMON4_OTP).

The Fail-Safe State Machine includes two ABISTs to verify the correct functionality

of the safety analog monitoring. ABIST1 is executed automatically after each POR

or after each wake-up from Standby. In case of ABIST1 fail, RSTB and PGOOD are

released, but FS0B remains stuck low and cannot be released.

It is recommended to verify latent faults on VCOREMON, HVLDOMON, and VDDIOMON during

ABIST1, while the latent faults on VMONx can be verified either during ABIST1 or

ABIST2.

0 - VMON4 check by ABIST1 is Disabled

1 - VMON4 check by ABIST1 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	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	true

4.233 Container PmicCommunicationConfiguration

This container contains the configuration for the communication.

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

4.234 Parameter PmicI2cCommunicationMethod

Configures the mechanism used to communicate with the PMIC device (synchronous or asynchronous).

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	SYNCHRONOUS
literals	['SYNCHRONOUS']

4.235 Reference PmicI2cChannelRef

Reference to the I2C channel configuration used to communicate with the PMIC, which is

set in the I2C driver configuration.

Warning: The driver will use the I2C addresses configured by OTP

(i.e. PmicOtpMainUnitConfiguration/PmicOtpDeviceConfiguration/PmicMainI2cAddress

for the Main Unit and PmicOtpFailSafeUnitConfiguration/PmicOtpSafetyConfiguration/

PmicSafetyI2cAddress for the Fail-Safe Unit).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/I2c/I2cGlobalConfig/I2cChannel

4.236 Reference PmicI2CSCLPinRef

Reference to the SCL pin of the I2C channel used to communicate with the PMIC. This is pin will

be manually toggled in order to unlock the I2C bus should it get erroneously stuck.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Port/PortConfigSet/PortContainer/PortPin

4.237 Reference PmicI2CSCLDioRef

Reference to the Dio channel of the SCL pin of the I2C channel used to communicate with the PMIC. This Dio channel

will be used to toggle the pin repeatedly in order to unlock the I2C bus should it get erroneously stuck.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Dio/DioConfig/DioPort/DioChannel

4.238 Container PmicClockSettingConfig

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

Note: Implementation Specific Parameter.

Included subcontainers:

- PmicClockReferencePoint

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

4.239 Parameter PmicClockSettingId

The ID of this PmicClockSettingConfig to be used as argument for the API call Pmic_InitClock.

Note: Implementation Specific Parameter.

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

Property	Value
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

4.240 Parameter PmicIrcoscFrequencyHz

Internal Oscillator Frequency [Hz].

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-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	F_20000000
literals	['F_16000000', 'F_17000000', 'F_18000000', 'F_19000000', 'F_20000000', 'F↩ _21000000', 'F_22000000', 'F_23000000', 'F_24000000']

4.241 Parameter PmicLowPowerOscFrequencyHz

Low Power Oscillator Frequency [Hz] (LOW_POWER_CLOCK[1:0]).

This oscillator is used only in the low power standby mode

and has an accuracy of +/-10%.

The main purpose of this oscillator is to reduce the current

consumption of the device during the standby mode.

It is recommended to use the 600KHz setting for DDR self-refresh.

Warning: A frequency change of LPOSC needs

to be programmed at least 40us prior to transitioning the

VR5510 device into Standby Mode.

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-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	F_100000
literals	['F_100000', 'F_300000', 'F_600000']

4.242 Parameter PmicFinEnable

This field is used to enable/disable the FIN pin (EXT_FIN_DIS) .

0 - FIN Pin is Disabled

1 - FIN Pin is Enabled

The PLL shall also be enabled by OTP in order to synchronize the switching regulators.

Note: Implementation Specific Parameter.

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

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

4.243 Parameter PmicExternalPIN_Fin_FrequencyHz

FIN Reference Frequency [Hz] (CLK_FIN_DIV).

The input frequency range of FIN must be between 333KHz-500kHz or 2MHz-3MHz.

If the external frequency coming from the FIN pin at runtime is out of range,

CLK moves back to the internal oscillator and an error is reported through

the CLK_FIN_DIV_OK bit.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	333000.0
max	3000000.0
min	333000.0

4.244 Parameter PmicPllClockSelection

Configures the PLL Source Selection (EXT_FIN_SEL).

Warning: The PLL is enabled/disabled via the OTP settings.

This field is not editable unless the PLL is enabled

(PmicOtpMainUnitConfiguration/PmicOtpDeviceConfiguration/PmicPllEnable = 'true').

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	IRCOSC_CLK
literals	['IRCOSC_CLK', 'FIN_CLK']

4.245 Parameter PmicSpectrumModulationEnable

This field is used to enable/disable the spread spectrum modulation feature (MOD_EN) .

0 - Spread Spectrum Modulation is Disabled

1 - Spread Spectrum Modulation is Enabled

The main purpose of the spread spectrum modulation is to improve the EMC performance

by spreading the energy of the internal oscillator and VPRE frequency on VBAT

frequency spectrum.

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 Parameter PmicTriangularCarrierFrequencyHz

Triangular Carrier Frequency [Hz] (MOD_CONF).

The spread spectrum modulation has a +/-5% deviation range around

the center of the oscillator frequency.

It is recommended to select 23.15KHz carrier frequency when VPRE is

configured at 455KHz and 92.6KHz when VPRE is configured at 2.22MHz

for best performance.

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	F_23150
literals	['F_23150', 'F_92600']

4.247 Parameter PmicFoutMuxSelection

Configures which clock to be sent to the FOUT pin (FOUT_MUX_SEL[3:0]).

The FOUT pin can be used to synchronize an external device with the VR5510.

Warning: The function of the AMUX/FOUT pin is selected via the OTP settings.

This field is not editable unless the AMUX/FOUT pin is configured as FOUT

(PmicOtpMainUnitConfiguration/PmicOtpMainIOConfiguration/PmicAmuxFoutPinMode = 'FOUT_MODE').

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	DISABLED
literals	['DISABLED', 'VPRE_CLK', 'BOOST_CLK', 'BUCK1_CLK', 'BUCK2_CLK', 'BUCK3_CLK', 'FOUT_CLK', 'CLK20M_MAIN_DIV48', 'CLK20M_FS_D↩IV48', 'CLK_FIN_DIV']

4.248 Parameter PmicFoutClockSelection

Configures the FOUT Source Selection (FOUT_CLK_SEL).

This field is editable only when PmicFoutMuxSelection = FOUT_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	CLK2
literals	['CLK1', 'CLK2']

4.249 Parameter PmicFoutPhaseDelay

Controls the phase delay of the FOUT_CLK (FOUT_PHASE[2:0]).

This field is editable only when PmicFoutMuxSelection = FOUT_CLK

Note: Implementation Specific Parameter.

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

4.250 Parameter PmicClk1Frequency

CLK1 Reference Frequency [Hz].

Note: This field must not be manually modified.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	2222000.0
max	2666666.0
min	1777777.0

4.251 Parameter PmicClk2Frequency

CLK2 Reference Frequency [Hz].

Note: This field must not be manually modified.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	455000.0
max	545454.0
min	363636.0

4.252 Container PmicClockReferencePoint

This container defines a reference point for every possible frequency of the PMIC Clock Tree.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	6
upperMultiplicity	6
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.253 Parameter PmicClockFrequencySelect

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

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	VPRE_CLK
literals	['VPRE_CLK', 'BOOST_CLK', 'BUCK1_CLK', 'BUCK2_CLK', 'BUCK3_↩ CLK', 'FOUT_CLK']

4.254 Parameter PmicClockReferencePointFrequency

This is the frequency for the specific instance of the PmicClockReferencePoint container.

Calculated value.

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	455000.0
max	3500000.0
min	0.0

4.255 Container PmicModeSettingConf

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

Note: Implementation Specific Parameter.

Included subcontainers:

- PmicRegulatorsConfiguration
- PmicVMONConfiguration

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

4.256 Parameter PmicModeID

This parameter shall represent the ID of the PMIC mode.

Note: Implementation Specific Parameter.

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	255
min	0

4.257 Parameter PmicModeSelection

This parameter selects the Mode to be used.

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	NORMAL
literals	['NORMAL', 'RESET', 'SHUTDOWN', 'STANDBY', 'DEEPSLEEP']

4.258 Parameter PmicPWRON2DeepSleepModeEnable

This field is used to enable/disable the entry to DeepSleep Mode by toggling the PWRON2 pin (M_MODE[PWRON2_DSM_I

0 - PWRON2 (high-to-low) transition cannot trigger DSM

1 - PWRON2 (high-to-low) transition can trigger DSM

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

4.259 Parameter PmicPWRON1WakeUpEnable

This field is used to enable/disable the wake-up functionality of the PWRON1 pin (M_MODE[PWRON1DIS]).

0 - PWRON1 Wakeup Disable

1 - PWRON1 Wakeup Enable

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 PmicPWRON2WakeUpEnable

This field is used to enable/disable the wake-up functionality of the PWRON2 pin (M_MODE[PWRON2DIS]).

0 - PWRON2 Wakeup Disable

1 - PWRON2 Wakeup Enable

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

4.261 Parameter PmicStandbyTimerEnable

This field is used to enable/disable the standby transition timer (M_SM_CTRL1[STBY_TIMER_EN]).

0 - Standby Transition Timer is Disabled

1 - Standby Transition Timer is Enabled

The Standby Timer is implemented in the Main-Unit logic to automatically transition

the device to Deep Fail-Safe (DFS) mode in case of timeout during Standby mode.

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

4.262 Parameter PmicStandbyTimerWindowDuration

Configures the standby timer window duration (M_SM_CTRL1[TIMER_STBY_WINDOW]).

Represents the amount of time to wait in Standby mode until a timeout is triggered,

after which the device will automatically transition to Deep Fail-Safe (DFS) mode.

This field is editable only when the standby timer is enabled

(i.e. PmicStandbyTimerEnable = 'true').

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-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	TIME_1024MS
literals	['TIME_16MS', 'TIME_32MS', 'TIME_128MS', 'TIME_512MS', 'TIME_↩ 1024MS', 'TIME_4096MS', 'TIME_8192MS', 'TIME_16384MS', 'TIME_↩ 65536MS', 'TIME_131072MS', 'TIME_262144MS', 'TIME_524288MS', 'TIM↩ E_1048576MS', 'TIME_2097152MS', 'TIME_4194304MS', 'TIME_8388608MS']

4.263 Container PmicRegulatorsConfiguration

This container contains the configuration parameters for the registers controlling the functionality of regulators.

Note: Implementation Specific Parameter.

Included subcontainers:

- PmicVpreRegulator
- PmicVBoostRegulator
- PmicHVLdoRegulator
- PmicBuck1Regulator
- PmicBuck2Regulator
- PmicBuck3Regulator
- PmicLdo1Regulator
- PmicLdo2Regulator
- PmicLdo3Regulator

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

4.264 Container PmicVpreRegulator

This container contains the configuration parameters of the VPRE (HVBUCK) regulator.

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

4.265 Parameter PmicVpreEnableRegulator

This field is used to control the power state the VPRE (HVBUCK) regulator (M_REG_CTRL1[VPREDIS/VPREEN]).

NO_EFFECT - Keep previous power state of VPRE

DISABLE - VPRE is Disabled

ENABLE - VPRE is Enabled

This field is modifiable only when PmicModeSelection = 'NORMAL'.

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	NO_EFFECT
literals	['NO_EFFECT', 'DISABLE', 'ENABLE']

4.266 Parameter PmicVpreStandbyOutputVoltage

Controls the output voltage of the VPRE (HVBUCK) regulator in STANDBY mode (M_REG_CTRL3[VPREV_STBY]).

This field is modifiable only when PmicModeSelection = 'STANDBY' and VPRE Standby Voltage is enabled by OTP (i.e. PmicOtpMainUnitConfiguration/PmicOtpRegulatorsConfiguration/PmicVpreRegulator/PmicVpreStdbyVoltageCont = 'true').

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	OUT_3V0
literals	['OUT_3V3', 'OUT_3V0']

4.267 Parameter PmicVpreHsCurrentCapability

Controls the pull-up current capability of the HS (High-Side) gate driver of the VPRE (HVBUCK) regulator (M_REG_CTRL2[VPRESRHS]).

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	I_130_MA
literals	['I_130_MA', 'I_260_MA', 'I_520_MA', 'I_900_MA']

4.268 Parameter PmicVpreLsCurrentCapability

Controls the pull-down current capability of the LS (Low-Side) gate driver of the VPRE (HVBUCK) regulator (M_REG_CTRL2[VPRESRLS]).

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	I_130_MA
literals	['I_130_MA', 'I_260_MA', 'I_520_MA', 'I_900_MA']

4.269 Container PmicVBoostRegulator

This container contains the configuration parameters of the BOOST regulator.

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

4.270 Parameter PmicBoostEnableRegulator

This field is used to control the power state the BOOST regulator (M_REG_CTRL1[BOOSTDIS/BOOSTEN]).

NO_EFFECT - Keep previous power state of BOOST

DISABLE - BOOST is Disabled

ENABLE - BOOST is Enabled

This field is modifiable only when PmicModeSelection = 'NORMAL'.

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	NO_EFFECT
literals	['NO_EFFECT', 'DISABLE', 'ENABLE']

4.271 Parameter PmicBoostSlewRate

Controls the slew rate of the BOOST regulator (M_REG_CTRL2[VBSTSR]).

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	RATE_500
literals	['RATE_100', 'RATE_300', 'RATE_500']

4.272 Container PmicHVLdoRegulator

This container contains the configuration parameters of the HVLDO regulator.

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

4.273 Parameter PmicHVLdoEnableRegulator

This field is used to control the power state the HVLDO regulator (M_REG_CTRL2[HVLDOEN/HVLDODIS]).

NO_EFFECT - Keep previous power state of HVLDO

DISABLE - HVLDO is Disabled

ENABLE - HVLDO is Enabled

This field is modifiable only when PmicModeSelection = 'NORMAL'.

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	NO_EFFECT
literals	['NO_EFFECT', 'DISABLE', 'ENABLE']

4.274 Parameter PmicHVLdoStandbyEnableRegulator

This field is used to enable/disable the HVLDO regulator in STANDBY mode (M_REG_CTRL3[HVLDO_STBY]).

0 - HVLDO in Standby Mode is Disabled

1 - HVLDO in Standby Mode is Enabled

This field is modifiable only when PmicModeSelection = 'STANDBY'.

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

4.275 Container PmicBuck1Regulator

This container contains the configuration parameters of the BUCK1 regulator.

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

4.276 Parameter PmicBuck1EnableRegulator

This field is used to control the power state the BUCK1 regulator (M_REG_CTRL1[BUCK1DIS/BUCK1EN]).

NO_EFFECT - Keep previous power state of BUCK1

DISABLE - BUCK1 is Disabled

ENABLE - BUCK1 is Enabled

This field is modifiable only when PmicModeSelection = 'NORMAL'.

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	NO_EFFECT
literals	['NO_EFFECT', 'DISABLE', 'ENABLE']

4.277 Parameter PmicBuck1StandbyEnableRegulator

This field is used to enable/disable the BUCK1 regulator in STANDBY mode (M_REG_CTRL3[BUCK1_STBY]).

0 - BUCK1 in Standby Mode is Disabled

1 - BUCK1 in Standby Mode is Enabled

This field is modifiable only when PmicModeSelection = 'STANDBY'.

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

4.278 Parameter PmicBuck1StandbyOutputVoltage

Controls the output voltage of the BUCK1 (and BUCK1/2 in dual-phase mode) regulator(s)

in STANDBY Mode (M_LVB1_STDBY_DVS[BUCK1_STBY]).

BUCK1 (BUCK1/2 in dual-phase mode) is intended to supply the MCU core voltage.

The output voltage must have a resolution of 6.25 mV.

This field is modifiable only when PmicModeSelection = 'STANDBY'

and PmicBuck1StandbyEnableRegulator = 'true'.

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	0.4
max	1.8
min	0.4

4.279 Container PmicBuck2Regulator

This container contains the configuration parameters of the BUCK2 regulator.

The container is modifiable only when the BUCK2 regulator is not configured

to operate in dual-phase mode.

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

4.280 Parameter PmicBuck2EnableRegulator

This field is used to control the power state the BUCK2 regulator (M_REG_CTRL1[BUCK2DIS/BUCK2EN]).

NO_EFFECT - Keep previous power state of BUCK2

DISABLE - BUCK2 is Disabled

ENABLE - BUCK2 is Enabled

This field is modifiable only when PmicModeSelection = 'NORMAL'.

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	NO_EFFECT
literals	['NO_EFFECT', 'DISABLE', 'ENABLE']

4.281 Parameter PmicBuck2StandbyEnableRegulator

This field is used to enable/disable the BUCK2 regulator in STANDBY mode (M_REG_CTRL3[BUCK2_STBY]).

0 - BUCK2 in Standby Mode is Disabled

1 - BUCK2 in Standby Mode is Enabled

This field is modifiable only when PmicModeSelection = 'STANDBY'.

Note: Implementation Specific Parameter.

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

4.282 Container PmicBuck3Regulator

This container contains the configuration parameters of the BUCK3 regulator.

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

4.283 Parameter PmicBuck3EnableRegulator

This field is used to control the power state the BUCK3 regulator (M_REG_CTRL1[BUCK3DIS/BUCK3EN]).

NO_EFFECT - Keep previous power state of BUCK3

DISABLE - BUCK3 is Disabled

ENABLE - BUCK3 is Enabled

This field is modifiable only when PmicModeSelection = 'NORMAL'.

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	NO_EFFECT
literals	['NO_EFFECT', 'DISABLE', 'ENABLE']

4.284 Parameter PmicBuck3StandbyEnableRegulator

This field is used to enable/disable the BUCK3 regulator in STANDBY mode (M_REG_CTRL3[BUCK3_STBY]).

0 - BUCK3 in Standby Mode is Disabled

1 - BUCK3 in Standby Mode is Enabled

This field is modifiable only when PmicModeSelection = 'STANDBY'.

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

4.285 Container PmicLdo1Regulator

This container contains the configuration parameters of the LDO1 regulator.

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

4.286 Parameter PmicLdo1EnableRegulator

This field is used to control the power state the LDO1 regulator (M_REG_CTRL1[LDO1DIS/LDO1EN]).

NO_EFFECT - Keep previous power state of LDO1

DISABLE - LDO1 is Disabled

ENABLE - LDO1 is Enabled

This field is modifiable only when PmicModeSelection = 'NORMAL'.

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	NO_EFFECT
literals	['NO_EFFECT', 'DISABLE', 'ENABLE']

4.287 Parameter PmicLdo1StandbyEnableRegulator

This field is used to enable/disable the LDO1 regulator in STANDBY mode (M_REG_CTRL3[LDO1_STBY]).

0 - LDO1 in Standby Mode is Disabled

1 - LDO1 in Standby Mode is Enabled

This field is modifiable only when PmicModeSelection = 'STANDBY'.

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

4.288 Container PmicLdo2Regulator

This container contains the configuration parameters of the LDO2 regulator.

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

4.289 Parameter PmicLdo2EnableRegulator

This field is used to control the power state the LDO2 regulator (M_REG_CTRL1[LDO2DIS/LDO2EN]).

NO_EFFECT - Keep previous power state of LDO2

DISABLE - LDO2 is Disabled

ENABLE - LDO2 is Enabled

This field is modifiable only when PmicModeSelection = 'NORMAL'.

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	NO_EFFECT
literals	['NO_EFFECT', 'DISABLE', 'ENABLE']

4.290 Parameter PmicLdo2StandbyEnableRegulator

This field is used to enable/disable the LDO2 regulator in STANDBY mode (M_REG_CTRL3[LDO2_STBY]).

0 - LDO2 in Standby Mode is Disabled

1 - LDO2 in Standby Mode is Enabled

This field is modifiable only when PmicModeSelection = 'STANDBY'.

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

4.291 Container PmicLdo3Regulator

This container contains the configuration parameters of the LDO3 regulator.

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

4.292 Parameter PmicLdo3EnableRegulator

This field is used to control the power state the LDO3 regulator (M_REG_CTRL1[LDO3DIS/LDO3EN]).

NO_EFFECT - Keep previous power state of LDO3

DISABLE - LDO3 is Disabled

ENABLE - LDO3 is Enabled

This field is modifiable only when PmicModeSelection = 'NORMAL'.

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	NO_EFFECT
literals	['NO_EFFECT', 'DISABLE', 'ENABLE']

4.293 Parameter PmicLdo3StandbyEnableRegulator

This field is used to enable/disable the LDO3 regulator in STANDBY mode (M_REG_CTRL3[LDO3_STBY]).

0 - LDO3 in Standby Mode is Disabled

1 - LDO3 in Standby Mode is Enabled

This field is modifiable only when PmicModeSelection = 'STANDBY'.

Note: Implementation Specific Parameter.

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

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

4.294 Container PmicVMONConfiguration

This container contains the configuration parameters for the registers controlling the regulator assignments of the VMONx input pins.

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

4.295 Parameter PmicVMON4RegulatorAssignment

Configures which regulator will be monitored by VMON4 (M_VMON_REGx[VMON4_REG_ASSIGN]).

The VMON4 input pin can be connected to VPRE, BOOST, LDO1-3, BUCK3, or an

external regulator (the choice depending on the configuration of this field).

The regulator connected to VMON4 MUST use an external resistor bridge

delivering a middle point of 0.8V to be compatible with over-voltage and under-voltage

monitoring thresholds.

Note: This field is modifiable only if VMON4 is enabled by OTP (i.e. PmicOtpFailSafeUnitConfiguration/PmicOtpVoltageMoni = 'true')

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	EXTERNAL_REGULATOR
literals	['EXTERNAL_REGULATOR', 'VPRE', 'LDO1', 'LDO2', 'BUCK3', 'BOOST', 'LDO3']

4.296 Parameter PmicVMON3RegulatorAssignment

Configures which regulator will be monitored by VMON3 (M_VMON_REGx[VMON3_REG_ASSIGN]).

The VMON3 input pin can be connected to VPRE, BOOST, LDO1-3, BUCK3, or an

external regulator (the choice depending on the configuration of this field).

The regulator connected to VMON3 MUST use an external resistor bridge

delivering a middle point of 0.8V to be compatible with over-voltage and under-voltage

monitoring thresholds.

Note: This field is modifiable only if VMON3 is enabled by OTP (i.e. PmicOtpFailSafeUnitConfiguration/PmicOtpVoltageMoni = 'true')

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	EXTERNAL_REGULATOR
literals	['EXTERNAL_REGULATOR', 'VPRE', 'LDO1', 'LDO2', 'BUCK3', 'BOOST', 'LDO3']

4.297 Parameter PmicVMON2RegulatorAssignment

Configures which regulator will be monitored by VMON2 (M_VMON_REGx[VMON2_REG_ASSIGN]).

The VMON2 input pin can be connected to VPRE, BOOST, LDO1-3, BUCK3, or an

external regulator (the choice depending on the configuration of this field).

The regulator connected to VMON2 MUST use an external resistor bridge

delivering a middle point of 0.8V to be compatible with over-voltage and under-voltage

monitoring thresholds.

Note: This field is modifiable only if VMON2 is enabled by OTP (i.e. PmicOtpFailSafeUnitConfiguration/PmicOtpVoltageMoni
= 'true')

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	EXTERNAL_REGULATOR
literals	['EXTERNAL_REGULATOR', 'VPRE', 'LDO1', 'LDO2', 'BUCK3', 'BOOST', 'LDO3']

4.298 Parameter PmicVMON1RegulatorAssignment

Configures which regulator will be monitored by VMON1 (M_VMON_REGx[VMON1_REG_ASSIGN]).

The VMON1 input pin can be connected to VPRE, BOOST, LDO1-3, BUCK3, or an

external regulator (the choice depending on the configuration of this field).

The regulator connected to VMON1 MUST use an external resistor bridge

delivering a middle point of 0.8V to be compatible with over-voltage and under-voltage

monitoring thresholds.

Note: This field is modifiable only if VMON1 is enabled by OTP (i.e. PmicOtpFailSafeUnitConfiguration/PmicOtpVoltageMoni
= 'true')

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	EXTERNAL_REGULATOR
literals	['EXTERNAL_REGULATOR', 'VPRE', 'LDO1', 'LDO2', 'BUCK3', 'BOOST', 'LDO3']

4.299 Container PmicReactionsSettingConf

This container contains the configuration for the reactions settings of the PMIC.

Note: Implementation Specific Parameter.

Included subcontainers:

- PmicMainUnitReactionsConf
- PmicFailSafeUnitReactionsConf

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

4.300 Parameter PmicReactionsSettingId

The ID of this PmicReactionsSettingConf to be used as argument for the API calls Pmic_SetReactions.

Note: Implementation Specific Parameter.

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	255
min	0

4.301 Container PmicMainUnitReactionsConf

This container contains the configuration parameters for the reactions settings of the Main Unit.

Note: Implementation Specific Parameter.

Included subcontainers:

- PmicMainInterruptMasks
- PmicMainThermalShutdownBehaviors

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

4.302 Container PmicMainInterruptMasks

This container contains the configuration parameters for the interrupt masks settings of the Main Unit.

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

4.303 Parameter PmicHvldoOverCurrentIntDisable

This field is used to unmask/mask the interrupt for HVLDO Overcurrent event (M_INT_MASK1[HVLDOOC_M]).

0 - HVLDO Overcurrent Interrupt is Unmasked

1 - HVLDO Overcurrent Interrupt is Masked

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

4.304 Parameter PmicBuck1OverCurrentIntDisable

This field is used to unmask/mask the interrupt for BUCK1 Overcurrent event (M_INT_MASK1[BUCK1OC_M]).

0 - BUCK1 Overcurrent Interrupt is Unmasked

1 - BUCK1 Overcurrent Interrupt is Masked

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

4.305 Parameter PmicBuck2OverCurrentIntDisable

This field is used to unmask/mask the interrupt for BUCK2 Overcurrent event (M_INT_MASK1[BUCK2OC_M]).

0 - BUCK2 Overcurrent Interrupt is Unmasked

1 - BUCK2 Overcurrent Interrupt is Masked

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

4.306 Parameter PmicBuck3OverCurrentIntDisable

This field is used to unmask/mask the interrupt for BUCK3 Overcurrent event (M_INT_MASK1[BUCK3OC_M]).

0 - BUCK3 Overcurrent Interrupt is Unmasked

1 - BUCK3 Overcurrent Interrupt is Masked

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

4.307 Parameter PmicLdo1OverCurrentIntDisable

This field is used to unmask/mask the interrupt for LDO1 Overcurrent event (M_INT_MASK1[LDO1OC_M]).

0 - LDO1 Overcurrent Interrupt is Unmasked

1 - LDO1 Overcurrent Interrupt is Masked

Note: Implementation Specific Parameter.

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

4.308 Parameter PmicLdo2OverCurrentIntDisable

This field is used to unmask/mask the interrupt for LDO2 Overcurrent event (M_INT_MASK1[LDO2OC_M]).

0 - LDO2 Overcurrent Interrupt is Unmasked

1 - LDO2 Overcurrent Interrupt is Masked

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

4.309 Parameter PmicLdo3OverCurrentIntDisable

This field is used to unmask/mask the interrupt for LDO3 Overcurrent event (M_INT_MASK1[LDO3OC_M]).

0 - LDO3 Overcurrent Interrupt is Unmasked

1 - LDO3 Overcurrent Interrupt is Masked

Note: Implementation Specific Parameter.

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

4.310 Parameter PmicHvldoTempShutdownIntDisable

This field is used to unmask/mask the interrupt for HVLDO Temperature Shutdown event (M_INT_MASK1[HVLDO_TSDFI

0 - HVLDO Temperature Shutdown Interrupt is Unmasked

1 - HVLDO Temperature Shutdown Interrupt is Masked

Note: Implementation Specific Parameter.

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

4.311 Parameter PmicBoostTempShutdownIntDisable

This field is used to unmask/mask the interrupt for BOOST Temperature Shutdown event (M_INT_MASK1[BOOST_TSDFL

0 - BOOST Temperature Shutdown Interrupt is Unmasked

1 - BOOST Temperature Shutdown Interrupt is Masked

Note: Implementation Specific Parameter.

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

4.312 Parameter PmicBuck1TempShutdownIntDisable

This field is used to unmask/mask the interrupt for BUCK1 Temperature Shutdown event (M_INT_MASK1[BUCK1_TSDFL

0 - BUCK1 Temperature Shutdown Interrupt is Unmasked

1 - BUCK1 Temperature Shutdown Interrupt is Masked

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

4.313 Parameter PmicBuck2TempShutdownIntDisable

This field is used to unmask/mask the interrupt for BUCK2 Temperature Shutdown event (M_INT_MASK1[BUCK2_TSDFL

0 - BUCK2 Temperature Shutdown Interrupt is Unmasked

1 - BUCK2 Temperature Shutdown Interrupt is Masked

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

4.314 Parameter PmicBuck3TempShutdownIntDisable

This field is used to unmask/mask the interrupt for BUCK3 Temperature Shutdown event (M_INT_MASK1[BUCK3_TSDFL

0 - BUCK3 Temperature Shutdown Interrupt is Unmasked

1 - BUCK3 Temperature Shutdown Interrupt is Masked

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

4.315 Parameter PmicLdo1TempShutdownIntDisable

This field is used to unmask/mask the interrupt for LDO1 Temperature Shutdown event (M_INT_MASK1[LDO1_TSDFLG_

0 - LDO1 Temperature Shutdown Interrupt is Unmasked

1 - LDO1 Temperature Shutdown Interrupt is Masked

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

4.316 Parameter PmicLdo2TempShutdownIntDisable

This field is used to unmask/mask the interrupt for LDO2 Temperature Shutdown event (M_INT_MASK1[LDO2_TSDFLG_

0 - LDO2 Temperature Shutdown Interrupt is Unmasked

1 - LDO2 Temperature Shutdown Interrupt is Masked

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

4.317 Parameter PmicLdo3TempShutdownIntDisable

This field is used to unmask/mask the interrupt for LDO3 Temperature Shutdown event (M_INT_MASK1[LDO3_TSDFLG_

0 - LDO3 Temperature Shutdown Interrupt is Unmasked

1 - LDO3 Temperature Shutdown Interrupt is Masked

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

4.318 Parameter PmicCenterDieTempIntDisable

This field is used to unmask/mask the interrupt for exceeding the Center Die Temperature Threshold (M_INT_MASK2[DIE_

0 - Center Die Overheat Interrupt is Unmasked

1 - Center Die Overheat Interrupt is Masked

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

4.319 Parameter PmicCommunicationErrorIntDisable

This field is used to unmask/mask the interrupt for any I2C communication error event (M_INT_MASK2[COM_ERR_M]).

0 - Communication Error Interrupt is Unmasked

1 - Communication Error Interrupt is Masked

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

4.320 Parameter PmicBosUndervoltageHighIntDisable

This field is used to unmask/mask the interrupt for VBOS Undervoltage High event (M_INT_MASK2[VBOSUVH_M]).

0 - VBOS Undervoltage High Interrupt is Unmasked

1 - VBOS Undervoltage High Interrupt is Masked

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

4.321 Parameter PmicBoostUndervoltageHighIntDisable

This field is used to unmask/mask the interrupt for VBOOST Undervoltage High event (M_INT_MASK2[VBOOSTUVH_M])

0 - VBOOST Undervoltage High Interrupt is Unmasked

1 - VBOOST Undervoltage High Interrupt is Masked

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

4.322 Parameter PmicBoostOvervoltageHighIntDisable

This field is used to unmask/mask the interrupt for VBOOST Overvoltage High event (M_INT_MASK2[VBOOSTOVH_M]).

0 - VBOOST Overvoltage High Interrupt is Unmasked

1 - VBOOST Overvoltage High Interrupt is Masked

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

4.323 Parameter PmicBistTempShutdownIntDisable

This field is used to unmask/mask the interrupt for Thermal Shutdown during BIST event (M_INT_MASK2[TSD_BIST_ERI

0 - Thermal Shutdown during BIST Interrupt is Unmasked

1 - Thermal Shutdown during BIST Interrupt is Masked

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

4.324 Parameter PmicHvldoInputUndervoltageLowIntDisable

This field is used to unmask/mask the interrupt for both LV_HVLDO_IN and HV_HVLDO_IN Undervoltage Low event (M_INT_MASK2[HVLDO_INPUT_UVL_M]).

0 - LV/HV_HVLDO_IN Undervoltage Low Interrupt is Unmasked

1 - LV/HV_HVLDO_IN Undervoltage Low Interrupt is Masked

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

4.325 Parameter PmicVpreOvervoltageIntDisable

This field is used to unmask/mask the interrupt for VPRE Overvoltage event (M_INT_MASK2[VPREOV2_M]).

0 - VPRE Overvoltage Interrupt is Unmasked

1 - VPRE Overvoltage Interrupt is Masked

Note: Implementation Specific Parameter.

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

4.326 Parameter PmicVpreOvercurrentIntDisable

This field is used to unmask/mask the interrupt for VPRE Overcurrent event (M_INT_MASK2[VPREOC_M]).

0 - VPRE Overcurrent Interrupt is Unmasked

1 - VPRE Overcurrent Interrupt is Masked

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

4.327 Parameter PmicVpreUndervoltageLowIntDisable

This field is used to unmask/mask the interrupt for VPRE Undervoltage Low event (M_INT_MASK2[VPREUVL_M]).

0 - VPRE Undervoltage Low Interrupt is Unmasked

1 - VPRE Undervoltage Low Interrupt is Masked

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

4.328 Parameter PmicVpreUndervoltageHighIntDisable

This field is used to unmask/mask the interrupt for VPRE Undervoltage High event (M_INT_MASK2[VPREUVH_M]).

0 - VPRE Undervoltage High Interrupt is Unmasked

1 - VPRE Undervoltage High Interrupt is Masked

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

4.329 Parameter PmicVsupUndervoltage7VIntDisable

This field is used to unmask/mask the interrupt for VSUP Undervoltage 7V event (M_INT_MASK2[VSUPUV7_M]).

0 - VSUP Undervoltage 7V Interrupt is Unmasked

1 - VSUP Undervoltage 7V Interrupt is Masked

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

4.330 Parameter PmicVsupUndervoltageLowIntDisable

This field is used to unmask/mask the interrupt for VSUP Undervoltage Low event (M_INT_MASK2[VSUPUVL_M]).

0 - VSUP Undervoltage Low Interrupt is Unmasked

1 - VSUP Undervoltage Low Interrupt is Masked

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

4.331 Parameter PmicVsupUndervoltageHighIntDisable

This field is used to unmask/mask the interrupt for VSUP Undervoltage High event (M_INT_MASK2[VSUPUVH_M]).

0 - VSUP Undervoltage High Interrupt is Unmasked

1 - VSUP Undervoltage High Interrupt is Masked

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

4.332 Parameter PmicPwron2TransitionIntDisable

This field is used to unmask/mask the interrupt for PWRON2 Transition (Toggling) event (M_INT_MASK2[PWRON2FLG_M

0 - PWRON2 Transition Interrupt is Unmasked

1 - PWRON2 Transition Interrupt is Masked

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

4.333 Parameter PmicPwron1TransitionIntDisable

This field is used to unmask/mask the interrupt for PWRON1 Transition (Toggling) event (M_INT_MASK2[PWRON1FLG_M

0 - PWRON1 Transition Interrupt is Unmasked

1 - PWRON1 Transition Interrupt is Masked

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

4.334 Container PmicMainThermalShutdownBehaviors

This container contains the configuration parameters for the thermal shutdown behaviors of the Main Unit.

Note: Implementation Specific Parameter.

Included subcontainers:

- None

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

4.335 Parameter PmicCenterDieTempThreshold

Configures the center die temperature threshold (M_TSD_CFG[DIE_CENTER_TEMP]).

A thermal sensor is placed on the center of the die (device) which is used to

generate interrupts for the MCU whenever the temperature exceeds the configured

threshold.

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	TEMP_105_oC
literals	['TEMP_75_oC', 'TEMP_90_oC', 'TEMP_105_oC', 'TEMP_120_oC', 'TE↩ MP_135_oC', 'TEMP_150_oC']

4.336 Parameter PmicBoostTsdBehavior

Controls the reaction/behavior of the BOOST regulator in case of a thermal shutdown event (M_TSD_CFG[BOOST_TSDCF

Can be configured to either shutdown only the regulator OR shutdown the regulator and transition the

device into Deep Fail-State (DFS) mode when the thermal shutdown threshold is exceeded.

In case SHUTDOWN_ONLY is selected, the regulator will startup automatically when the temperature goes down.

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	SHUTDOWN_AND_DFS
literals	['SHUTDOWN_ONLY', 'SHUTDOWN_AND_DFS']

4.337 Parameter PmicBuck1TsdBehavior

Controls the reaction/behavior of the BUCK1 regulator in case of a thermal shutdown event (M_TSD_CFG[BUCK1_TSDCF

Can be configured to either shutdown only the regulator OR shutdown the regulator and transition the

device into Deep Fail-State (DFS) mode when the thermal shutdown threshold is exceeded.

In case SHUTDOWN_ONLY is selected, the regulator will startup automatically when the temperature goes down.

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	SHUTDOWN_AND_DFS
literals	['SHUTDOWN_ONLY', 'SHUTDOWN_AND_DFS']

4.338 Parameter PmicBuck2TsdBehavior

Controls the reaction/behavior of the BUCK2 regulator in case of a thermal shutdown event (M_TSD_CFG[BUCK2_TSDCFG

Can be configured to either shutdown only the regulator OR shutdown the regulator and transition the

device into Deep Fail-State (DFS) mode when the thermal shutdown threshold is exceeded.

In case SHUTDOWN_ONLY is selected, the regulator will startup automatically when the temperature goes down.

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	SHUTDOWN_AND_DFS
literals	['SHUTDOWN_ONLY', 'SHUTDOWN_AND_DFS']

4.339 Parameter PmicBuck3TsdBehavior

Controls the reaction/behavior of the BUCK3 regulator in case of a thermal shutdown event (M_TSD_CFG[BUCK3_TSDCFG

Can be configured to either shutdown only the regulator OR shutdown the regulator and transition the

device into Deep Fail-State (DFS) mode when the thermal shutdown threshold is exceeded.

In case SHUTDOWN_ONLY is selected, the regulator will startup automatically when the temperature goes down.

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	SHUTDOWN_AND_DFS
literals	['SHUTDOWN_ONLY', 'SHUTDOWN_AND_DFS']

4.340 Parameter PmicLdo1TsdBehavior

Controls the reaction/behavior of the LDO1 regulator in case of a thermal shutdown event (M_TSD_CFG[LDO1_TSDCFG]).

Can be configured to either shutdown only the regulator OR shutdown the regulator and transition the

device into Deep Fail-State (DFS) mode when the thermal shutdown threshold is exceeded.

In case SHUTDOWN_ONLY is selected, the regulator will startup automatically when the temperature goes down.

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	SHUTDOWN_AND_DFS
literals	['SHUTDOWN_ONLY', 'SHUTDOWN_AND_DFS']

4.341 Parameter PmicLdo2TsdBehavior

Controls the reaction/behavior of the LDO2 regulator in case of a thermal shutdown event (M_TSD_CFG[LDO2_TSDCFG]).

Can be configured to either shutdown only the regulator OR shutdown the regulator and transition the

device into Deep Fail-State (DFS) mode when the thermal shutdown threshold is exceeded.

In case SHUTDOWN_ONLY is selected, the regulator will startup automatically when the temperature goes down.

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	SHUTDOWN_AND_DFS
literals	['SHUTDOWN_ONLY', 'SHUTDOWN_AND_DFS']

4.342 Parameter PmicLdo3TsdBehavior

Controls the reaction/behavior of the LDO3 regulator in case of a thermal shutdown event (M_TSD_CFG[LDO3_TSDCFG]).

Can be configured to either shutdown only the regulator OR shutdown the regulator and transition the

device into Deep Fail-State (DFS) mode when the thermal shutdown threshold is exceeded.

In case SHUTDOWN_ONLY is selected, the regulator will startup automatically when the temperature goes down.

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	SHUTDOWN_AND_DFS
literals	['SHUTDOWN_ONLY', 'SHUTDOWN_AND_DFS']

4.343 Parameter PmicLHVLdoTsdBehavior

Controls the reaction/behavior of the HVLDO regulator in case of a thermal shutdown event (M_TSD_CFG[HVLDO_TSDCF

Can be configured to either shutdown only the regulator OR shutdown the regulator and transition the

device into Deep Fail-State (DFS) mode when the thermal shutdown threshold is exceeded.

In case SHUTDOWN_ONLY is selected, the regulator will startup automatically when the temperature goes down.

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	SHUTDOWN_AND_DFS
literals	['SHUTDOWN_ONLY', 'SHUTDOWN_AND_DFS']

4.344 Container PmicFailSafeUnitReactionsConf

This container contains the configuration parameters for the reactions settings of the Fail-Safe Unit.

Note: Implementation Specific Parameter.

Included subcontainers:

- PmicFailSafeInterruptMasks

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

4.345 Container PmicFailSafeInterruptMasks

This container contains the configuration parameters for the interrupt masks settings of the Fail-Safe Unit.

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

4.346 Parameter PmicVmon4OverUnderVoltageIntDisable

This field is used to unmask/mask the interrupt for VMON4 Overvoltage and Undervoltage events (FS_INTB_MASK[INT_IN

0 - VMON4 Overvoltage and Undervoltage Interrupt is Unmasked

1 - VMON4 Overvoltage and Undervoltage Interrupt is Masked

Note: Implementation Specific Parameter.

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

4.347 Parameter PmicVmon3OverUnderVoltageIntDisable

This field is used to unmask/mask the interrupt for VMON3 Overvoltage and Undervoltage events (FS_INTB_MASK[INT_IN

0 - VMON3 Overvoltage and Undervoltage Interrupt is Unmasked

1 - VMON3 Overvoltage and Undervoltage Interrupt is Masked

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

4.348 Parameter PmicVmon2OverUnderVoltageIntDisable

This field is used to unmask/mask the interrupt for VMON2 Overvoltage and Undervoltage events (FS_INTB_MASK[INT_IN

0 - VMON2 Overvoltage and Undervoltage Interrupt is Unmasked

1 - VMON2 Overvoltage and Undervoltage Interrupt is Masked

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

4.349 Parameter PmicVmon1OverUnderVoltageIntDisable

This field is used to unmask/mask the interrupt for VMON1 Overvoltage and Undervoltage events (FS_INTB_MASK[INT_IN

0 - VMON1 Overvoltage and Undervoltage Interrupt is Unmasked

1 - VMON1 Overvoltage and Undervoltage Interrupt is Masked

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

4.350 Parameter PmicVddioOverUnderVoltageIntDisable

This field is used to unmask/mask the interrupt for VDDIO Overvoltage and Undervoltage events (FS_INTB_MASK[INT_IN

0 - VDDIO Overvoltage and Undervoltage Interrupt is Unmasked

1 - VDDIO Overvoltage and Undervoltage Interrupt is Masked

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

4.351 Parameter PmicVcoremonOverUnderVoltageIntDisable

This field is used to unmask/mask the interrupt for VCOREMON Overvoltage and Undervoltage events (FS_INTB_MASK[IN

0 - VCOREMON Overvoltage and Undervoltage Interrupt is Unmasked

1 - VCOREMON Overvoltage and Undervoltage Interrupt is Masked

Note: Implementation Specific Parameter.

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

4.352 Parameter PmicWatchdogBadRefreshIntDisable

This field is used to unmask/mask the interrupt for Bad Watchdog Refresh event (FS_INTB_MASK[INT_INH_BAD_WD_F

0 - Bad Watchdog Refresh Interrupt is Unmasked

1 - Bad Watchdog Refresh Interrupt is Masked

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

4.353 Parameter PmicHvldoOverUnderVoltageIntDisable

This field is used to unmask/mask the interrupt for HVLDO Overvoltage and Undervoltage events (FS_INTB_MASK[INT_IN

0 - HVLDO Overvoltage and Undervoltage Interrupt is Unmasked

1 - HVLDO Overvoltage and Undervoltage Interrupt is Masked

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

4.354 Parameter PmicFccu2EventIntDisable

This field is used to unmask/mask the interrupt for FCCU2 event (FS_INTB_MASK[INT_INH_FCCU2]).

0 - FCCU2 Event Interrupt is Unmasked

1 - FCCU2 Event Interrupt is Masked

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

4.355 Parameter PmicFccu1EventIntDisable

This field is used to unmask/mask the interrupt for FCCU1 event (FS_INTB_MASK[INT_INH_FCCU1]).

0 - FCCU1 Event Interrupt is Unmasked

1 - FCCU1 Event Interrupt is Masked

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

4.356 Container PmicAMUXConfiguration

This container contains the configuration for the AMUX pin.

Note: Implementation Specific Parameter.

Included subcontainers:

- PmicAmuxChannel

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

4.357 Parameter PmicSetAnalogMuxApi

Pre-processor switch to enable/disable the API to set the analog multiplexer (AMUX) pin.

#define PMIC_SET_ANALOG_MUX_API (STD_ON)/(STD_OFF) will be generated in Pmic_Cfg.h file.

This API cannot be enabled unless the AMUX/FOUT pin is configured as AMUX by OTP

(PmicOtpMainUnitConfiguration/PmicOtpMainIOConfiguration/PmicAmuxFoutPinMode = 'AMUX_MODE').

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

4.358 Container PmicAmuxChannel

This container defines a symbolic name for every possible AMUX channel.

Note: Implementation Specific Parameter.

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

4.359 Parameter PmicAmuxChannelId

This parameter shall represent the ID of the AMUX channel.

Note: Implementation Specific Parameter.

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	255
min	0

4.360 Parameter PmicAmuxChannelSelect

Select the AMUX channel for the specific instance of the PmicAmuxChannel container (M_AMUX[AMUX]).

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	GND
literals	['GND', 'VDDIO', 'TEMPERATURE', 'BANDGAP_MAIN', 'BANDGAP_F↩ AIL_SAFE', 'BUCK1', 'BUCK2', 'BUCK3', 'VPRE', 'VBOOST', 'LDO1', 'L↩ DO2', 'VBOS', 'VSUP1', 'PWRON1', 'PWRON2', 'HVLDO', 'LDO3']

4.361 Parameter PmicAmuxRatio

Controls the divider ratio for VSUP, PWRON1, and PWRON2 (M_AMUX[RATIO]).

This field is editable only when PmicAmuxChannelSelect is either VSUP, PWRON1 or PWRON2.

Warning: The input voltage range for VSUP, PWRON1, and PWRON2 must be

between 2.25V and 22.5V when RATIO = 20, and between 4.2V and 42V when RATIO = 34.

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	RATIO_20
literals	['RATIO_20', 'RATIO_34']

4.362 Container PmicFailSafeConfiguration

This container contains the initial configuration parameters (set in the INIT_FS state) for the overvoltage and undervoltage reactions.

Note: Implementation Specific Parameter.

Included subcontainers:

- PmicSVSConfiguration
- PmicFailSafePinReactions
- PmicWatchdogConfiguration
- PmicABIST2Configuration
- PmicSafeInputsConfiguration
- PmicStateMachineConfiguration

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

4.363 Container PmicSVSConfiguration

This container contains the initial configuration parameters (set in the INIT_FS state) for the SVS mechanism.

Note: Implementation Specific Parameter.

Included subcontainers:

- PmicSVSSettingConf

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

4.364 Parameter PmicSwitchSVSApi

Pre-processor allows to switch static voltage scaler value.

User need to configure SVS setting in tab PmicSVSSettingConf.

In case DIE_PROCESS feature is enable, this function will configure SVS if the DIE_PROCESS otp of mcu enables.

It will not configure SVS if the DIE_PROCESS otp of mcu disable. Refer Mcu reference manual to check DIE_PROCESS bit.

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

4.365 Parameter PmicSVSOffset

Statically scales the output voltage of BUCK1 (and BUCK2 if used in dual-phase mode)

by the configured amount in the direction specified by SVS_OFFSET_SIGN

in the INIT_FS state (FS_I_SVS[SVS_OFFSET]).

A SVS (Static Voltage Scaling) mechanism is implemented to allow the MCU to reduce

the output voltage initially configured at start-up of BUCK1 (and BUCK2 if

used in multiphase) by OTP, to optimize MCU voltage working points, and

compensate part to part process variation.

The offset amount is effectively limited by the allowed SVS steps configured

in VCORE_SVS_CLAMP_OTP[5:0] by OTP (i.e. PmicFailSafeConfiguration/PmicOtpSafetyConfiguration/

PmicSVSLimit)

Depending on the offset, it may be needed to change the voltage in multiple steps

to not trigger an OV/UV event.

Note: Implementation Specific Parameter.

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

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	OFFSET_0_0
literals	['OFFSET_0_0', 'OFFSET_6_25', 'OFFSET_12_5', 'OFFSET_18_75', 'O↩ FFSET_25_0', 'OFFSET_31_25', 'OFFSET_37_5', 'OFFSET_43_75', 'OF↩ FSET_50_0', 'OFFSET_56_25', 'OFFSET_62_5', 'OFFSET_68_75', 'OFF↩ SET_75_0', 'OFFSET_81_25', 'OFFSET_87_5', 'OFFSET_93_75', 'OFFS↩ ET_100_0', 'OFFSET_106_25', 'OFFSET_112_5', 'OFFSET_118_75', 'O↩ FFSET_125_0', 'OFFSET_131_25', 'OFFSET_137_5', 'OFFSET_143_75', 'OFFSET_150_0', 'OFFSET_156_25', 'OFFSET_162_5', 'OFFSET_168_75', 'OFFSET_175_0', 'OFFSET_181_25', 'OFFSET_187_5', 'OFFSET_193_75', 'OFFSET_200_0', 'OFFSET_206_25', 'OFFSET_212_5', 'OFFSET_218_75', 'OFFSET_225_0', 'OFFSET_231_25', 'OFFSET_237_5', 'OFFSET_243_75', 'OFFSET_250_0', 'OFFSET_256_25', 'OFFSET_262_5', 'OFFSET_268_75', 'OFFSET_275_0', 'OFFSET_281_25', 'OFFSET_287_5', 'OFFSET_293_75', 'OFFSET_300_0', 'OFFSET_306_25', 'OFFSET_312_5', 'OFFSET_318_75', 'OFFSET_325_0', 'OFFSET_331_25', 'OFFSET_337_5', 'OFFSET_343_75', 'OFFSET_350_0', 'OFFSET_356_25', 'OFFSET_362_5', 'OFFSET_368_75', 'OFFSET_375_0', 'OFFSET_381_25', 'OFFSET_387_5', 'OFFSET_393_75']

4.366 Parameter PmicSVSOffsetSign

Configures the offset type applicable by the SVS mechanism (FS_I_SVS[SVS_OFFSET_SIGN]).

The offset type is limited by the SVS offset type configured by OTP in VCORE_SVS_FULL_OFFSET_OTP

(i.e. PmicFailSafeConfiguration/PmicOtpSafetyConfiguration/PmicSVSOffsetType)

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	NEGATIVE_OFFSET
literals	['NEGATIVE_OFFSET', 'POSITIVE_OFFSET']

4.367 Parameter PmicBuck1SVSOutputVoltage

Represents the output voltage of the BUCK1 (BUCK1/2 in dual-phase mode) regulator

after the SVS offset is applied. This output voltage will take effect at the

end of the INIT_FS state.

The formula for calculating this field is:

1. If PmicSVSOffsetSign = NEGATIVE_OFFSET:

PmicBuck1SVSOutputVoltage = PmicOtpMainUnitConfiguration/PmicOtpRegulatorsConfiguration/PmicBuck1Regulator/Pmi
- PmicSVSOffset

2. If PmicSVSOffsetSign = POSITIVE_OFFSET:

PmicBuck1SVSOutputVoltage = PmicOtpMainUnitConfiguration/PmicOtpRegulatorsConfiguration/PmicBuck1Regulator/Pmi
+ PmicSVSOffset

Note: This field must not be manually modified.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0.4
max	1.8
min	0.4

4.368 Container PmicSVSSettingConf

This container contains the configuration for the static voltage scaler settings of the PMIC.

If the PmicSwitchSVSApi is enabled, the list must have at least one element.

Otherwise, the list must be empty.

Note: Implementation Specific Parameter.

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

4.369 Parameter PmicSVSSettingId

This parameter shall represent the ID of the SVS configuration.

Note: Implementation Specific Parameter.

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	255
min	0

4.370 Parameter PmicSVSOffsetSetting

Statically scales the output voltage of BUCK1 (and BUCK2 if used in dual-phase mode)

by the configured amount in the direction specified by SVS_OFFSET_SIGN

in the INIT_FS state (FS_I_SVS[SVS_OFFSET]).

A SVS (Static Voltage Scaling) mechanism is implemented to allow the MCU to reduce

the output voltage initially configured at start-up of BUCK1 (and BUCK2 if

used in multiphase) by OTP, to optimize MCU voltage working points, and

compensate part to part process variation.

The offset amount is effectively limited by the allowed SVS steps configured

in VCORE_SVS_CLAMP_OTP[5:0] by OTP (i.e. PmicFailSafeConfiguration/PmicOtpSafetyConfiguration/

PmicSVSLimit)

Depending on the offset, it may be needed to change the voltage in multiple steps

to not trigger an OV/UV event.

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	OFFSET_0_0
literals	['OFFSET_0_0', 'OFFSET_6_25', 'OFFSET_12_5', 'OFFSET_18_75', 'O↩ FFSET_25_0', 'OFFSET_31_25', 'OFFSET_37_5', 'OFFSET_43_75', 'OF↩ FSET_50_0', 'OFFSET_56_25', 'OFFSET_62_5', 'OFFSET_68_75', 'OFF↩ SET_75_0', 'OFFSET_81_25', 'OFFSET_87_5', 'OFFSET_93_75', 'OFFS↩ ET_100_0', 'OFFSET_106_25', 'OFFSET_112_5', 'OFFSET_118_75', 'O↩ FFSET_125_0', 'OFFSET_131_25', 'OFFSET_137_5', 'OFFSET_143_75', 'OFFSET_150_0', 'OFFSET_156_25', 'OFFSET_162_5', 'OFFSET_168_75', 'OFFSET_175_0', 'OFFSET_181_25', 'OFFSET_187_5', 'OFFSET_193_75', 'OFFSET_200_0', 'OFFSET_206_25', 'OFFSET_212_5', 'OFFSET_218_75', 'OFFSET_225_0', 'OFFSET_231_25', 'OFFSET_237_5', 'OFFSET_243_75', 'OFFSET_250_0', 'OFFSET_256_25', 'OFFSET_262_5', 'OFFSET_268_75', 'OFFSET_275_0', 'OFFSET_281_25', 'OFFSET_287_5', 'OFFSET_293_75', 'OFFSET_300_0', 'OFFSET_306_25', 'OFFSET_312_5', 'OFFSET_318_75', 'OFFSET_325_0', 'OFFSET_331_25', 'OFFSET_337_5', 'OFFSET_343_75', 'OFFSET_350_0', 'OFFSET_356_25', 'OFFSET_362_5', 'OFFSET_368_75', 'OFFSET_375_0', 'OFFSET_381_25', 'OFFSET_387_5', 'OFFSET_393_75']

4.371 Parameter PmicSVSOffsetSignSetting

Configures the offset type applicable by the SVS mechanism (FS_I_SVS[SVS_OFFSET_SIGN]).

The offset type is limited by the SVS offset type configured by OTP in VCORE_SVS_FULL_OFFSET_OTP

(i.e. PmicFailSafeConfiguration/PmicOtpSafetyConfiguration/PmicSVSOffsetType)

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	NEGATIVE_OFFSET
literals	['NEGATIVE_OFFSET', 'POSITIVE_OFFSET']

4.372 Parameter PmicBuck1SVSOutputVoltageSetting

Represents the output voltage of the BUCK1 (BUCK1/2 in dual-phase mode) regulator

after the SVS offset is applied. This output voltage will take effect at the

end of the INIT_FS state.

The formula for calculating this field is:

1. If PmicSVSOffsetSign = NEGATIVE_OFFSET:

PmicBuck1SVSOutputVoltage = PmicOtpMainUnitConfiguration/PmicOtpRegulatorsConfiguration/PmicBuck1Regulator/Pmi
- PmicSVSOffset

2. If PmicSVSOffsetSign = POSITIVE_OFFSET:

PmicBuck1SVSOutputVoltage = PmicOtpMainUnitConfiguration/PmicOtpRegulatorsConfiguration/PmicBuck1Regulator/Pmi
+ PmicSVSOffset

Note: This field must not be manually modified.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0.4
max	1.8
min	0.4

4.373 Container PmicFailSafePinReactions

This container contains the configuration parameters for the pin reactions settings of the Fail-Safe Unit.

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

4.374 Parameter PmicVcoremonOvervoltageImpact

Configures the Fail-Safe reaction on RSTB and/or FS0B when an over-voltage fault is detected on VCOREMON (VCOREMON_OV_FS_IMPACT[1:0]).

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	FS0B_AND_RSTB_ASSERT
literals	['DISABLED', 'FS0B_ASSERT', 'FS0B_AND_RSTB_ASSERT']

4.375 Parameter PmicVcoremonUndervoltageImpact

Configures the Fail-Safe reaction on RSTB and/or FS0B when an under-voltage fault is detected on VCOREMON (VCOREMON_UV_FS_IMPACT[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FS0B_ASSERT
literals	['DISABLED', 'FS0B_ASSERT', 'FS0B_AND_RSTB_ASSERT']

4.376 Parameter PmicHvldoOvervoltageImpact

Configures the Fail-Safe reaction on RSTB and/or FS0B when an over-voltage fault is detected on HVLDO (HVLDO_OV_FS_

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	FS0B_AND_RSTB_ASSERT
literals	['DISABLED', 'FS0B_ASSERT', 'FS0B_AND_RSTB_ASSERT']

4.377 Parameter PmicHvldoUndervoltageImpact

Configures the Fail-Safe reaction on RSTB and/or FS0B when an under-voltage fault is detected on HVLDO (HVLDO_UV_FS_IMPACT[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FS0B_ASSERT
literals	['DISABLED', 'FS0B_ASSERT', 'FS0B_AND_RSTB_ASSERT']

4.378 Parameter PmicVddioOvervoltageImpact

Configures the Fail-Safe reaction on RSTB and/or FS0B when an over-voltage fault is detected on VDDIO (VDDIO_OV_FS_

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	FS0B_AND_RSTB_ASSERT
literals	['DISABLED', 'FS0B_ASSERT', 'FS0B_AND_RSTB_ASSERT']

4.379 Parameter PmicVddioUndervoltageImpact

Configures the Fail-Safe reaction on RSTB and/or FS0B when an under-voltage fault is detected on VDDIO (VDDIO_UV_FS

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	FS0B_ASSERT
literals	['DISABLED', 'FS0B_ASSERT', 'FS0B_AND_RSTB_ASSERT']

4.380 Parameter PmicVmon4OvervoltageImpact

Configures the Fail-Safe reaction on RSTB and/or FS0B when an over-voltage fault is detected on VMON4 (VMON4_OV_FS

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	FS0B_AND_RSTB_ASSERT
literals	['DISABLED', 'FS0B_ASSERT', 'FS0B_AND_RSTB_ASSERT']

4.381 Parameter PmicVmon4UndervoltageImpact

Configures the Fail-Safe reaction on RSTB and/or FS0B when an under-voltage fault is detected on VMON4 (VMON4_UV_FS_IMPACT[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FS0B_ASSERT
literals	['DISABLED', 'FS0B_ASSERT', 'FS0B_AND_RSTB_ASSERT']

4.382 Parameter PmicVmon3OvervoltageImpact

Configures the Fail-Safe reaction on RSTB and/or FS0B when an over-voltage fault is detected on VMON3 (VMON3_OV_FS_

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	FS0B_AND_RSTB_ASSERT
literals	['DISABLED', 'FS0B_ASSERT', 'FS0B_AND_RSTB_ASSERT']

4.383 Parameter PmicVmon3UndervoltageImpact

Configures the Fail-Safe reaction on RSTB and/or FS0B when an under-voltage fault is detected on VMON3 (VMON3_UV_FS_IMPACT[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FS0B_ASSERT
literals	['DISABLED', 'FS0B_ASSERT', 'FS0B_AND_RSTB_ASSERT']

4.384 Parameter PmicVmon2OvervoltageImpact

Configures the Fail-Safe reaction on RSTB and/or FS0B when an over-voltage fault is detected on VMON2 (VMON2_OV_FS_

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	FS0B_AND_RSTB_ASSERT
literals	['DISABLED', 'FS0B_ASSERT', 'FS0B_AND_RSTB_ASSERT']

4.385 Parameter PmicVmon2UndervoltageImpact

Configures the Fail-Safe reaction on RSTB and/or FS0B when an under-voltage fault is detected on VMON2 (VMON2_UV_FS_IMPACT[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FS0B_ASSERT
literals	['DISABLED', 'FS0B_ASSERT', 'FS0B_AND_RSTB_ASSERT']

4.386 Parameter PmicVmon1OvervoltageImpact

Configures the Fail-Safe reaction on RSTB and/or FS0B when an over-voltage fault is detected on VMON1 (VMON1_OV_FS_

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	FS0B_AND_RSTB_ASSERT
literals	['DISABLED', 'FS0B_ASSERT', 'FS0B_AND_RSTB_ASSERT']

4.387 Parameter PmicVmon1UndervoltageImpact

Configures the Fail-Safe reaction on RSTB and/or FS0B when an under-voltage fault is detected on VMON1 (VMON1_UV_FS_IMPACT[1:0]).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	FS0B_ASSERT
literals	['DISABLED', 'FS0B_ASSERT', 'FS0B_AND_RSTB_ASSERT']

4.388 Container PmicWatchdogConfiguration

This container contains the initial configuration parameters (set in the INIT_FS state) for the Watchdog.

Included subcontainers:

- PmicWatchdogSettingConf

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

4.389 Parameter PmicWatchdogApi

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

#define PMIC_WATCHDOG_API (STD_ON)/(STD_OFF) will be generated in Pmic_Cfg.h file.

This API cannot be enabled unless the Watchdog Monitoring is enabled by OTP

(PmicOtpFailSafeUnitConfiguration/PmicOtpSafetyIOConfiguration/PmicWatchdogEnable = 'true').

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

4.390 Parameter PmicWatchdogErrorCounterLimit

Controls the Watchdog Error Counter Limit (FS_I_WD_CFG[WD_ERR_LIMIT]).

The Watchdog Error Counter is incremented by 2 after each bad Watchdog

refresh and decremented by 1 after each good Watchdog refresh.

When the Watchdog Error Counter reaches this limit, the Fault Error

Counter will be incremented by 1 and will trigger a Fail-Safe reaction

configured by FS_I_WD_CFG[WD_FS_IMPACT] (PmicWatchdogErrorImpact).

Note: This field is editable only when the Watchdog Monitoring is enabled

by OTP (i.e. PmicOtpFailSafeUnitConfiguration/PmicOtpSafetyIOConfiguration/

PmicWatchdogEnable = 'true')

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	MAX_6
literals	['MAX_8', 'MAX_6', 'MAX_4', 'MAX_2']

4.391 Parameter PmicWatchdogRefreshCounterLimit

Controls the Watchdog Refresh Counter Limit (FS_I_WD_CFG[WD_RFR_LIMIT]).

The Watchdog Refresh Counter is incremented by 1 after each good Watchdog

refresh, but is reset to 0 every time there is a bad Watchdog refresh.

When the Watchdog Refresh Counter reaches this limit, and if the next

Watchdog refresh is also good, the Fault Error Counter is decremented

by 1.

Note: This field is editable only when the Watchdog Monitoring is enabled

by OTP (i.e. PmicOtpFailSafeUnitConfiguration/PmicOtpSafetyIOConfiguration/

PmicWatchdogEnable = 'true')

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	MAX_6
literals	['MAX_6', 'MAX_4', 'MAX_2', 'MAX_1']

4.392 Parameter PmicWatchdogErrorImpact

Configures the Fail-Safe reaction on RSTB and/or FS0B when the Watchdog

Error Counter reaches its maximum value (FS_I_WD_CFG[WD_FS_IMPACT]).

Note: This field is editable only when the Watchdog Monitoring is enabled

by OTP (i.e. PmicOtpFailSafeUnitConfiguration/PmicOtpSafetyIOConfiguration/

PmicWatchdogEnable = 'true')

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	FS0B_AND_RSTB_ASSERT
literals	['DISABLED', 'FS0B_ASSERT', 'FS0B_AND_RSTB_ASSERT']

4.393 Parameter PmicWatchdogWindowPeriodEnable

This field is used to enable/disable the Watchdog Window Period (FS_WD_WINDOW_DUR[WD_WINDOW] = DISABLED).

The Watchdog Window Period is derived from the Fail-Safe oscillator with a +/-5% accuracy.

0 - Watchdog Window Period is Disabled

1 - Watchdog Window Period is Enabled

Note: This field is editable only when the Watchdog Monitoring is enabled

by OTP (i.e. PmicOtpFailSafeUnitConfiguration/PmicOtpSafetyIOConfiguration/

PmicWatchdogEnable = 'true')

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

4.394 Parameter PmicWatchdogWindowPeriod_Setting

Controls the Watchdog Window Period (FS_WD_WINDOW_DUR[WD_WINDOW]).

This new watchdog window period is effective after the next

watchdog refresh.

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	TIME_3MS
literals	['TIME_1MS', 'TIME_2MS', 'TIME_3MS', 'TIME_4MS', 'TIME_6MS', 'TIME_8MS', 'TIME_12MS', 'TIME_16MS', 'TIME_24MS', 'TIME_32MS', 'TIME_64MS', 'TIME_128MS', 'TIME_256MS', 'TIME_512MS', 'TIME_1024MS']

4.395 Parameter PmicWatchdogClosedWindowDutyCycle_Setting

Controls the Watchdog Window Duty Cycle (FS_WD_WINDOW_DUR[WDW_DC]).

This new duty cycle is effective after the next watchdog refresh.

The first half of the window is said CLOSED and the second half is

said OPEN. A good watchdog refresh is a good watchdog answer during

the OPEN window. A bad watchdog refresh is a bad watchdog answer

during the OPEN window, no watchdog refresh during the OPEN window

or a good watchdog answer during the CLOSED window. After a good

or bad watchdog refresh, a new window period starts immediately

for the MCU to keep the synchronization with the windowed watchdog.

Note: The OPEN window duty cycle is equal to (100 - PmicWatchdogClosedWindowDutyCycle)%.

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

4.396 Parameter PmicWatchdogRecoveryWindowPeriod

Controls the Watchdog Window Duration when the device is in Fault Recovery Strategy (FS_WD_WINDOW_DUR[WDW_R

When a fault is triggered by the MCU via its FCCU pins, the FS0B is asserted by the device and the watchdog window duration becomes automatically an open window (no more duty cycle), whose duration is given by this field.

The transition from the normal watchdog window duration to this new open window duration happens when the FCCU pin indicates an error and FS0B is asserted. If the MCU sends a good watchdog refresh before the end of the open window duration, the device switches bask to the normal watchdog window duration and associated duty cycle if the FCCU pins do not indicate an error anymore. Otherwise, a new open window period is started. If the MCU does not send a good watchdog refresh before the end of this second open window duration, then a reset pulse is generated and the Fail-Safe State Machine moves back to INIT_FS.

Note: This field is editable only when the Watchdog Monitoring is enabled by OTP (i.e. PmicOtpFailSafeUnitConfiguration/PmicOtpSafetyIOConfiguration/ PmicWatchdogEnable = 'true')

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	TIME_64MS
literals	['DISABLE', 'TIME_1MS', 'TIME_2MS', 'TIME_3MS', 'TIME_4MS', 'TIM↩ E_6MS', 'TIME_8MS', 'TIME_12MS', 'TIME_16MS', 'TIME_24MS', 'TIME↩ _32MS', 'TIME_64MS', 'TIME_128MS', 'TIME_256MS', 'TIME_512MS', 'T↩ IME_1024MS']

4.397 Parameter PmicWatchdogSeed

Configures the Watchdog Seed (FS_WD_SEED[WD_SEED]).

This value is used to calculate the value to be written in the WD_ANSWER

register during the OPEN watchdog window.

Note: This field is editable only when the Watchdog Monitoring is enabled

by OTP (i.e. PmicOtpFailSafeUnitConfiguration/PmicOtpSafetyIOConfiguration/

PmicWatchdogEnable = 'true')

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	23218
max	65534
min	1

4.398 Container PmicWatchdogSettingConf

This container contains the configuration for the Watchdog settings of the PMIC.

If the Watchdog Monitoring is enabled by OTP, the list must have at least one element.

Otherwise, the list must be empty.

Note: Implementation Specific Parameter.

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

4.399 Parameter PmicWatchdogSettingId

This parameter shall represent the ID of the Watchdog configuration.

Note: Implementation Specific Parameter.

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	255
min	0

4.400 Parameter PmicWatchdogWindowPeriod

Controls the Watchdog Window Period (FS_WD_WINDOW_DUR[WD_WINDOW]).

This new watchdog window period is effective after the next

watchdog refresh.

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	TIME_3MS
literals	['TIME_1MS', 'TIME_2MS', 'TIME_3MS', 'TIME_4MS', 'TIME_6MS', 'TIME_8MS', 'TIME_12MS', 'TIME_16MS', 'TIME_24MS', 'TIME_32MS', 'TIME_64MS', 'TIME_128MS', 'TIME_256MS', 'TIME_512MS', 'TIME_1024MS']

4.401 Parameter PmicWatchdogClosedWindowDutyCycle

Controls the Watchdog Window Duty Cycle (FS_WD_WINDOW_DUR[WDW_DC]).

This new duty cycle is effective after the next watchdog refresh.

The first half of the window is said CLOSED and the second half is

said OPEN. A good watchdog refresh is a good watchdog answer during

the OPEN window. A bad watchdog refresh is a bad watchdog answer

during the OPEN window, no watchdog refresh during the OPEN window

or a good watchdog answer during the CLOSED window. After a good

or bad watchdog refresh, a new window period starts immediately

for the MCU to keep the synchronization with the windowed watchdog.

Note: The OPEN window duty cycle is equal to (100 - PmicWatchdogClosedWindowDutyCycle)%.

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

4.402 Container PmicABIST2Configuration

This container contains the initial configuration parameters (set in the INIT_FS state) for the ABIST2 checks.

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

4.403 Parameter PmicHVLdoMonitorABIST2Enable

Configures whether HVLDOMON is checked by ABIST2 or not (FS_I_ABIST2_CTRL[HVLDO_VMON_ABIST2]).

The Fail-Safe State Machine includes two ABISTs to verify the correct functionality

of the safety analog monitoring. ABIST2 is executed on demand by I2C request after

INIT_FS closure and before releasing the FS0B pin.

It is recommended to verify latent faults on VCOREMON, HVLDOMON, and VDDIOMON during

ABIST1, while the latent faults on VMONx can be verified either during ABIST1 or

ABIST2.

In case of ABIST2 fail, the FS0B pin remains stuck low and cannot be released.

0 - HVLDOMON check by ABIST2 is Disabled

1 - HVLDOMON check by ABIST2 is Enabled

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

4.404 Parameter PmicVcoreMonitorABIST2Enable

Configures whether VCOREMON is checked by ABIST2 or not (FS_I_ABIST2_CTRL[VCORE_ABIST2]).

The Fail-Safe State Machine includes two ABISTs to verify the correct functionality

of the safety analog monitoring. ABIST2 is executed on demand by I2C request after

INIT_FS closure and before releasing the FS0B pin.

It is recommended to verify latent faults on VCOREMON, HVLDOMON, and VDDIOMON during

ABIST1, while the latent faults on VMONx can be verified either during ABIST1 or

ABIST2.

In case of ABIST2 fail, the FS0B pin remains stuck low and cannot be released.

0 - VCOREMON check by ABIST2 is Disabled

1 - VCOREMON check by ABIST2 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

4.405 Parameter PmicVddioMonitorABIST2Enable

Configures whether VDDIOMON is checked by ABIST2 or not (FS_I_ABIST2_CTRL[VDDIO_ABIST2]).

The Fail-Safe State Machine includes two ABISTs to verify the correct functionality

of the safety analog monitoring. ABIST2 is executed on demand by I2C request after

INIT_FS closure and before releasing the FS0B pin.

It is recommended to verify latent faults on VCOREMON, HVLDOMON, and VDDIOMON during

ABIST1, while the latent faults on VMONx can be verified either during ABIST1 or

ABIST2.

In case of ABIST2 fail, the FS0B pin remains stuck low and cannot be released.

0 - VDDIOMON check by ABIST2 is Disabled

1 - VDDIOMON check by ABIST2 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

4.406 Parameter PmicVmon1MonitorABIST2Enable

Configures whether VMON1 is checked by ABIST2 or not (FS_I_ABIST2_CTRL[VMON1_ABIST2]).

The Fail-Safe State Machine includes two ABISTs to verify the correct functionality

of the safety analog monitoring. ABIST2 is executed on demand by I2C request after

INIT_FS closure and before releasing the FS0B pin.

It is recommended to verify latent faults on VCOREMON, HVLDOMON, and VDDIOMON during

ABIST1, while the latent faults on VMONx can be verified either during ABIST1 or

ABIST2.

In case of ABIST2 fail, the FS0B pin remains stuck low and cannot be released.

0 - VMON1 check by ABIST2 is Disabled

1 - VMON1 check by ABIST2 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	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.407 Parameter PmicVmon2MonitorABIST2Enable

Configures whether VMON2 is checked by ABIST2 or not (FS_I_ABIST2_CTRL[VMON2_ABIST2]).

The Fail-Safe State Machine includes two ABISTs to verify the correct functionality

of the safety analog monitoring. ABIST2 is executed on demand by I2C request after

INIT_FS closure and before releasing the FS0B pin.

It is recommended to verify latent faults on VCOREMON, HVLDOMON, and VDDIOMON during

ABIST1, while the latent faults on VMONx can be verified either during ABIST1 or

ABIST2.

In case of ABIST2 fail, the FS0B pin remains stuck low and cannot be released.

0 - VMON2 check by ABIST2 is Disabled

1 - VMON2 check by ABIST2 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

4.408 Parameter PmicVmon3MonitorABIST2Enable

Configures whether VMON3 is checked by ABIST2 or not (FS_I_ABIST2_CTRL[VMON3_ABIST2]).

The Fail-Safe State Machine includes two ABISTs to verify the correct functionality

of the safety analog monitoring. ABIST2 is executed on demand by I2C request after

INIT_FS closure and before releasing the FS0B pin.

It is recommended to verify latent faults on VCOREMON, HVLDOMON, and VDDIOMON during

ABIST1, while the latent faults on VMONx can be verified either during ABIST1 or

ABIST2.

In case of ABIST2 fail, the FS0B pin remains stuck low and cannot be released.

0 - VMON3 check by ABIST2 is Disabled

1 - VMON3 check by ABIST2 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

4.409 Parameter PmicVmon4MonitorABIST2Enable

Configures whether VMON4 is checked by ABIST2 or not (FS_I_ABIST2_CTRL[VMON4_ABIST2]).

The Fail-Safe State Machine includes two ABISTs to verify the correct functionality

of the safety analog monitoring. ABIST2 is executed on demand by I2C request after

INIT_FS closure and before releasing the FS0B pin.

It is recommended to verify latent faults on VCOREMON, HVLDOMON, and VDDIOMON during

ABIST1, while the latent faults on VMONx can be verified either during ABIST1 or

ABIST2.

In case of ABIST2 fail, the FS0B pin remains stuck low and cannot be released.

0 - VMON4 check by ABIST2 is Disabled

1 - VMON4 check by ABIST2 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

4.410 Container PmicSafeInputsConfiguration

This container contains the initial configuration parameters (set in the INIT_FS state) for the safe inputs (FCCU + Standby Timing Window).

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

4.411 Parameter PmicSafetyStandbyWindowDuration

Configures the safety standby window duration (FS_I_SAFE_INPUTS[TIMING_WINDOW_STBY]).

There are two possible transitioning paths to Standby Mode (selected via OTP):

- Standard path using only STBY pin transition.

- Safety path using an I2C request + STBY pin transition.

If the safety path is used, this field defines the maximum timeout between the

I2C request and the STBY pin transition.

This field is editable only if the safety standby window is enabled by OTP

(i.e. PmicOtpFailSafeUnitConfiguration/PmicOtpSafetyIOConfiguration/PmicSafetyStandbyWindowEnable = 'true')

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	TIME_1MS
literals	['DISABLED', 'TIME_60US', 'TIME_80US', 'TIME_100US', 'TIME_200US', 'TIME_300US', 'TIME_500US', 'TIME_1MS', 'TIME_2MS', 'TIME_3MS', 'TIME_5MS', 'TIME_8MS', 'TIME_10MS']

4.412 Parameter PmicFccuMonitoringConfiguration

Configures the operating mode of the FCCU monitoring (FS_I_SAFE_INPUTS[FCCU_CFG]).

The FCCU input pins are in charge of monitoring the HW failure from MCU. The FCCU

monitoring is active as soon as the INIT_FS is losed by the first good watchdog refresh.

The FCCU input pins can be configured by pair or as single independent inputs.

When FCCU1 and/or FCCU2 are used independently, the FCCU inputs can monitor 2 different

and independent error signals.

Note: This field is editable only when the FCCU Monitoring is enabled by OTP

(i.e. PmicOtpFailSafeUnitConfiguration/PmicOtpSafetyIOConfiguration/PmicFccuEnable = 'true')

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	FCCU1_AND_FCCU2_PAIR
literals	['DISABLED', 'FCCU1_AND_FCCU2_PAIR', 'FCCU1_OR_FCCU2_IND↩EPENDENT', 'FCCU1_ONLY']

4.413 Parameter PmicFccu12FaultPolarity

Configures the fault polarity of the FCCU12 signal (FS_I_SAFE_INPUTS[FCCU12_FLT_POL]).

When FCCU12 are used by pair, the bi-stable protocol is employed.

FCCU1_LOW_OR_FCCU2_HIGH_IS_FAULT - External pull-up resistor to VDDIO on FCCU2 and pull-down

resistor to GND on FCCU1 are required to provide passive error state if the MCU does

not drive its FCCU output pins.

FCCU1_HIGH_OR_FCCU2_LOW_IS_FAULT - External pull-up resistor to VDDIO on FCCU1 and pull-down

resistor to GND on FCCU2 are required to provide passive error state if the MCU does

not drive its FCCU output pins.

The pull-down resistor value must be at least 4 times bigger than the pull-up resistor in order

to detect FCCU1-short-to-FCCU2 failure mode, whatever the VDDIO voltage is (1.8V or 3.3V).

Note: This field is editable only when the FCCU Monitoring is enabled by OTP

(i.e. PmicOtpFailSafeUnitConfiguration/PmicOtpSafetyIOConfiguration/PmicFccuEnable = 'true')

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	FCCU1_LOW_OR_FCCU2_HIGH_IS_FAULT
literals	['FCCU1_LOW_OR_FCCU2_HIGH_IS_FAULT', 'FCCU1_HIGH_OR_F↩ CCU2_LOW_IS_FAULT']

4.414 Parameter PmicFccu12FaultImpact

Configures the Fail-Safe reaction on RSTB and/or FS0B when a FCCU12

fault is detected (FS_I_SAFE_INPUTS[FCCU12_FS_IMPACT]).

Note: This field is editable only when the FCCU Monitoring is enabled by OTP

(i.e. PmicOtpFailSafeUnitConfiguration/PmicOtpSafetyIOConfiguration/PmicFccuEnable = 'true')

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	FS0B_AND_RSTB_ASSERT
literals	['FS0B_ASSERT', 'FS0B_AND_RSTB_ASSERT']

4.415 Parameter PmicFccu1FaultPolarity

Configures the fault polarity of the FCCU1 signal (FS_I_SAFE_INPUTS[FCCU1_FLT_POL]).

Note: This field is editable only when the FCCU Monitoring is enabled by OTP

(i.e. PmicOtpFailSafeUnitConfiguration/PmicOtpSafetyIOConfiguration/PmicFccuEnable = 'true')

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	LOW_LEVEL_IS_FAULT
literals	['LOW_LEVEL_IS_FAULT', 'HIGH_LEVEL_IS_FAULT']

4.416 Parameter PmicFccu1FaultImpact

Configures the Fail-Safe reaction on RSTB and/or FS0B when a FCCU1

fault is detected (FS_I_SAFE_INPUTS[FCCU1_FS_IMPACT]).

Note: This field is editable only when the FCCU Monitoring is enabled by OTP

(i.e. PmicOtpFailSafeUnitConfiguration/PmicOtpSafetyIOConfiguration/PmicFccuEnable = 'true')

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	FS0B_AND_RSTB_ASSERT
literals	['FS0B_ASSERT', 'FS0B_AND_RSTB_ASSERT']

4.417 Parameter PmicFccu2FaultPolarity

Configures the fault polarity of the FCCU2 signal (FS_I_SAFE_INPUTS[FCCU2_FLT_POL]).

Note: This field is editable only when the FCCU Monitoring is enabled by OTP

(i.e. PmicOtpFailSafeUnitConfiguration/PmicOtpSafetyIOConfiguration/PmicFccuEnable = 'true')

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	LOW_LEVEL_IS_FAULT
literals	['LOW_LEVEL_IS_FAULT', 'HIGH_LEVEL_IS_FAULT']

4.418 Parameter PmicFccu2FaultImpact

Configures the Fail-Safe reaction on RSTB and/or FS0B when a FCCU2

fault is detected (FS_I_SAFE_INPUTS[FCCU2_FS_IMPACT]).

Note: This field is editable only when the FCCU Monitoring is enabled by OTP

(i.e. PmicOtpFailSafeUnitConfiguration/PmicOtpSafetyIOConfiguration/PmicFccuEnable = 'true')

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	FS0B_AND_RSTB_ASSERT
literals	['FS0B_ASSERT', 'FS0B_AND_RSTB_ASSERT']

4.419 Container PmicStateMachineConfiguration

This container contains the initial configuration parameters (set in the INIT_FS state) for the state machine functionality of the Fail-Safe Unit.

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

4.420 Parameter PmicFaultErrorCounterLimit

Controls the Fault Error Counter Limit (FS_I_FSSM[FLT_ERR_CNT_LIMIT]).

The Fault Error Counter counts the number of faults occuring in the system

(related to the device itself and also caused by external events).

The Fault Error Counter is incremented by 1 each time the RSTB and/or FS0B

pin is asserted and each time the Watchdog Error Counter Limit is reached.

It is decremented by 1 each time the Watchdog Refresh Counter Limit is reached.

The Fault Error Counter has two output values: Intermediate and Final. The

intermediate value is calculated as FEC_Limit / 2 (i.e. the value of this

field divided by 2).

This maximum limit of the Fault Error Counter is used to transition in

Deep Fail-Safe mode.

After each power-up or wake-up from Standby mode, the Fault Error Counter

starts at level 1 to ensure the safe state. It is recommended to read

this counter and decrement it to an appropriate value by several

consecutive good Watchdog refreshes before a reset request by I2C.

The number of consecutive good Watchdog refreshes needed (N) to

decrement the Fault Error Counter to 0 is equal to:

N = FS_I_FSSM[FLT_ERR_CNT] * (FS_I_WD_CFG[WD_RFR_LIMIT] + 1).

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	MAX_6
literals	['MAX_2', 'MAX_6', 'MAX_8', 'MAX_12']

4.421 Parameter PmicFaultErrorCounterImpact

Configures the Fail-Safe reaction on RSTB and/or FS0B when the Fault

Error Counter reaches its intermediate value (FS_I_FSSM[FLT_ERR_IMPACT]).

The intermediate value of the Fault Error Counter is equal to:

INTERMEDIATE_VALUE = PmicFaultErrorCounterLimit / 2.

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	FS0B_AND_RSTB_ASSERT
literals	['DISABLED', 'FS0B_ASSERT', 'FS0B_AND_RSTB_ASSERT']

4.422 Parameter PmicResetDuration

Configures how long the RSTB signal is kept asserted (FS_I_FSSM[RSTB_DUR]).

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	TIME_10MS
literals	['TIME_10MS', 'TIME_1MS']

4.423 Parameter PmicBackupSafetyPath

Configures whether the RSTB pin is asserted when a short-to-high failure is

detected on the FS0B pin (FS_I_FSSM[BACKUP_SAFETY_PATH]).

A FS0B short-to-high failure is detected after a duration of maximum 815us.

The RSTB pin assertion acts as a redundant path to reset the MCU and assert

its FCCU pin.

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	RSTB_ASSERT
literals	['DISABLED', 'RSTB_ASSERT']

4.424 Parameter PmicClockMonitoringEnable

This field is used to enable/disable the IRCOSC/FIN Clock Monitoring (FS_I_FSSM[CLK_MON_DIS]).

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

4.425 Parameter PmicRSTBTimerEnable

This field is used to enable/disable the 8s RSTB timer (FS_I_FSSM[DIS8S]).

If enabled, the 8s RSTB timer starts each time the RSTB pin is asserted low.

When the LBIST is done, the 8s timer monitoring the RSTB pin starts and

the ABIST1 is automatically executed when all the regulators assigned to

ABIST1 have passed their UV and OV checks. When the ABIST1 is done,

RSTB and PGOOD pins are released, and the initialization of the device

is opened for 256ms. ABIST1 fail does not prevent RSTB and PGOOD release.

If RSTB is not released before the RSTB timer expires (i.e. within

8 seconds), the device will go into Deep Fail-Safe Mode.

0 - RSTB Timer is Disabled

1 - RSTB Timer is Enabled

Note: This field is editable only when the 8s RSTB Timer is enabled

by OTP (i.e. PmicOtpFailSafeUnitConfiguration/PmicOtpSafetyConfiguration/

PmicRSTBTimerEnable = 'true')

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

4.426 Parameter PmicLowPowerClockMonitoringEnable

This field is used to enable/disable the Low Power OSC Clock Monitoring (FS_I_FSSM[LPCLK_MON_DIS]).

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

4.427 Container PmicDemEventParameterRefs

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

4.428 Reference PMIC_E_ACESS_FAILURE

Reference to the DemEventParameter which shall be issued when the device is not accesible.

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

4.429 Reference PMIC_E_INTEGRITY_CORRUPTED

Reference to the DemEventParameter which shall be issued when the device reports LBIST, ABIST or OTP failures.

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

4.430 Reference PMIC_E_SIGNAL_SHORTED

Reference to the DemEventParameter which shall be issued when the device does not successfully pass the safety signals check.

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP

Property	Value
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

4.431 Reference PMIC_E_CLOCK_FAILURE

Reference to the DemEventParameter which shall be issued when the device reports clock failures (e.g. oscillator drifts, unlocked PLLs, etc).

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

4.432 Reference PMIC_E_TIMEOUT_FAILURE

Reference to the DemEventParameter which shall be issued when the error "Timeout caused by hardware error" has occurred.

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP
lowerMultiplicity	0
upperMultiplicity	1

Property	Value
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

4.433 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

4.434 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

Property	Value
valueConfigClasses	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	4
max	4
min	4

4.435 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	4
max	4
min	4

4.436 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

Property	Value
defaultValue	0
max	0
min	0

4.437 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	255
max	255
min	255

4.438 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

Property	Value
valueConfigClasses	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	4
max	4
min	4

4.439 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

4.440 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

Property	Value
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	0
max	0
min	0

4.441 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	

4.442 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

This chapter describes the Tresos configuration plug-in for the Pmic 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 Pmic HLD

6.1.1 Detailed Description

Data Structures

- struct Pmic_RawFaultType
  The structure containing the raw fault events for a PMIC device. More...
- struct Pmic_DeviceInfoType
  The device information structure for a PMIC device. More...
- struct Pmic_InterfaceUnitConfigType
  The interface configuration structure for a PMIC device. More...
- struct Pmic_DeviceConfigType
  Configuration structure for a PMIC device. More...
- struct Pmic_DemConfigType
  DEM error reporting configuration. More...
- struct Pmic_ConfigType
  Configuration structure of the driver. More...

Macros

- #define PMIC_E_OK
  Internal return codes.

Types Reference

- typedef void(* Pmic_WdgTaskNotificationType) (const Pmic_DeviceIndexType DeviceId, const uint8 u8↩ WdWindowValue, const uint8 u8WdDcValue)
  This gives the call-out type for watchdog task notifications.

Enum Reference

- enum Pmic_ApiServiceIdType
  API service IDs.
- enum Pmic_ErrorIdType
  Error IDs.
- enum Pmic_InterfaceType
  Interface selection.
- enum Pmic_InterfaceDstType
  Interface destination selection.

Function Reference

- void Pmic_Init (const Pmic_ConfigType *pConfigPtr)
  Pmic_Init.
- Std_ReturnType Pmic_InitDevice (const Pmic_DeviceIndexType DeviceId)
  Pmic_InitDevice.
- Std_ReturnType Pmic_EmulateDeviceOTP (const Pmic_DeviceIndexType DeviceId)
  Pmic_EmulateDeviceOTP.
- Std_ReturnType Pmic_ReadRegister (const Pmic_DeviceIndexType DeviceId, const Pmic_InterfaceDstType eDestination, const uint8 u8RegAddr, uint16 *pu16ValueOut)
  Pmic_ReadRegister.
- Std_ReturnType Pmic_WriteRegister (const Pmic_DeviceIndexType DeviceId, const Pmic_InterfaceDstType eDestination, const uint8 u8RegAddr, const uint16 u16WriteValue)
  Pmic_WriteRegister.
- Std_ReturnType Pmic_InitClock (const Pmic_DeviceIndexType DeviceId, const Pmic_ClockSettingIndex↩ Type ClockSettingId)
  Pmic_InitClock.
- Std_ReturnType Pmic_SetMode (const Pmic_DeviceIndexType DeviceId, const Pmic_ModeIndexType ModeSettingId)
  Pmic_SetMode.
- Std_ReturnType Pmic_SetAnalogMux (const Pmic_DeviceIndexType DeviceId, const Pmic_Amux↩ ChannelIndexType AmuxChannelId)
  Pmic_SetAnalogMux.
- Std_ReturnType Pmic_SwitchSVS (const Pmic_DeviceIndexType DeviceId, const Pmic_SvsSettingIndex↩ Type SvsSettingID)
  Pmic_SwitchSVS.
- Std_ReturnType Pmic_ConfigureWatchdog (const Pmic_DeviceIndexType DeviceId, const Pmic_↩ WatchdogSettingIndexType WatchdogSettingId)
  Pmic_ConfigureWatchdog.
- void Pmic_TriggerWatchdog (const Pmic_DeviceIndexType DeviceId)
  Pmic_TriggerWatchdog.
- Std_ReturnType Pmic_DisableWatchdog (const Pmic_DeviceIndexType DeviceId)
  Pmic_DisableWatchdog.
- Std_ReturnType Pmic_ReleaseFs0b (const Pmic_DeviceIndexType DeviceId)
  PMIC executes the release safety FS0B pin.
- Std_ReturnType Pmic_GotoInitFS (const Pmic_DeviceIndexType DeviceId)

Pmic_GotoInitFS.

- Std_ReturnType Pmic_GetRawFaultEvents (const Pmic_DeviceIndexType DeviceId, Pmic_RawFaultType *pRawFaultEventsOut)

  Pmic_GetRawFaultEvents.
- Std_ReturnType Pmic_SetReactions (const Pmic_DeviceIndexType DeviceId, const Pmic_ReactionSettingIndexType ReactionSettingId)

  Pmic_SetReactions.
- Std_ReturnType Pmic_GetDeviceInfo (const Pmic_DeviceIndexType DeviceId, Pmic_DeviceInfoType *pDeviceInfoOut)

  Pmic_GetDeviceInfo.
- void Pmic_GetVersionInfo (Std_VersionInfoType *pVersionInfoOut)

  Pmic_GetVersionInfo.
- void Pmic_Dem_Check_Device (const Pmic_ReturnType eStatusCode)

  API checks status and report the device run time error.
- Std_ReturnType Pmic_HLDChecksEntry (const Pmic_ApiServiceIdType ePmicServiceID)

  API checks the Pmic_Init executed successfully before.
- void Pmic_HLDChecksExit (const Std_ReturnType eRetStatus, const Pmic_ApiServiceIdType ePmicServiceID)

  API checks the Pmic driver status in Pmic_Init API.
- Std_ReturnType Pmic_HLDDeviceChecksEntry (const Pmic_DeviceIndexType u32DeviceId, const Pmic_ApiServiceIdType ePmicServiceID)

  API checks the PMIC device already initialized before call another API.
- void Pmic_HLDDeviceChecksExit (const Std_ReturnType eRetStatus, const Pmic_DeviceIndexType u32DeviceId, const Pmic_ApiServiceIdType ePmicServiceID)

  API checks the Pmic driver status when user executes the Pmic_InitDevice, Pmic_EmulateDeviceOTP API.
- Std_ReturnType Pmic_Check_Init (const Pmic_ConfigType *pConfigPtr)

  API checks the Pmic_Init parameter input valid.
- Std_ReturnType Pmic_Check_ReadRegister (const Pmic_DeviceIndexType u32DeviceId, const uint8 u8RegAddr, const uint16 *pu16ValueOut)

  API checks the Pmic_ReadRegister parameter input valid.
- Std_ReturnType Pmic_Check_WriteRegister (const Pmic_DeviceIndexType u32DeviceId, const uint8 u8RegAddr)

  API checks the Pmic_WriteRegister parameter input valid.
- Std_ReturnType Pmic_Check_SetMode (const Pmic_DeviceIndexType u32DeviceId, const Pmic_ModeIndexType u32ModeSettingId)

  API checks the Pmic_Setmode parameter input valid.
- Std_ReturnType Pmic_Check_InitClock (const Pmic_DeviceIndexType u32DeviceId, const Pmic_ClockSettingIndexType u32ClockSettingID)

  API checks the Pmic_InitClock parameter input valid.
- Std_ReturnType Pmic_Check_SetAnalogMux (const Pmic_DeviceIndexType u32DeviceId, const Pmic_AmuxChannelIndexType u32AmuxChannelId)

  API checks the Pmic_SetAnalogMux parameter input valid.
- Std_ReturnType Pmic_Check_SwitchSVS (const Pmic_DeviceIndexType u32DeviceId, const Pmic_SvsSettingIndexType u32SvsSettingID)

  API checks the Pmic_SwitchSVS parameter input valid.
- Std_ReturnType Pmic_Check_ConfigureWatchdog (const Pmic_DeviceIndexType u32DeviceId, const Pmic_WatchdogSettingIndexType u32WatchdogSettingId)

  API checks the Pmic_ConfigureWatchdog parameter input valid.
- Std_ReturnType Pmic_Check_TriggerWatchdog (const Pmic_DeviceIndexType u32DeviceId)

API checks the Pmic_TriggerWatchdog parameter input valid.

- Std_ReturnType Pmic_Check_GotoInitFS (const Pmic_DeviceIndexType u32DeviceId)

  API checks the Pmic_GotoInitFS parameter input valid.
- Std_ReturnType Pmic_Check_GetRawFaultEvents (const Pmic_RawFaultType *pRawFaultEventsOut)

  API checks the Pmic_GetRawFaultEvents parameter input valid.
- Std_ReturnType Pmic_Check_SetReactions (const Pmic_DeviceIndexType u32DeviceId, const Pmic_↩ ReactionSettingIndexType u32ReactionSettingId)

  API checks the Pmic_SetReactions parameter input valid.

  .
- Std_ReturnType Pmic_Check_GetDeviceInfo (const Pmic_DeviceInfoType *pDeviceInfoOut)

  API checks the Pmic_GetDeviceInfo parameter input valid.
- Std_ReturnType Pmic_Check_GetVersionInfo (const Std_VersionInfoType *pVersionInfo)

  API checks the Pmic_GetVersionInfo parameter input valid.
- void Pmic_Det_Check_ReportErrors (const Pmic_ReturnType eStatusCode, const Pmic_ApiServiceIdType eServiceId)

  API checks status and report the run time error.

Variables

- boolean Pmic_abTestMode [PMIC_MAX_DEVICECONFIGS]

  Array used to indicate whether the indexed device is in test mode.
- boolean Pmic_abErrorDetected [PMIC_MAX_DEVICECONFIGS]

  Array used to indicate whether the indexed device encountered an error.
- boolean Pmic_abGateWdgTrigger [PMIC_MAX_DEVICECONFIGS]

  Array used to indicate watchdog trigger will be forwarded to the device.
- boolean Pmic_abIsWdgTrigger [PMIC_MAX_DEVICECONFIGS]

  Array used to check watchdog trigger in progress or not. When Wdg_trigger is in-progress, variable will set true.

6.1.2 Data Structure Documentation

6.1.2.1 struct Pmic_RawFaultType

The structure containing the raw fault events for a PMIC device.

Contains various error flags of the Main-Unit and Fail-Safe Unit.

Definition at line 255 of file Pmic_Types.h.

6.1.2.2 struct Pmic_DeviceInfoType

The device information structure for a PMIC device.

It contains the family id and the device id of the PMIC device.

Definition at line 267 of file Pmic_Types.h.

6.1.2.3 struct Pmic_InterfaceUnitConfigType

The interface configuration structure for a PMIC device.

It contains the type of the interface, of the communication, and the device address.

Definition at line 290 of file Pmic_Types.h.

Data Fields

Type	Name	Description
Pmic_InterfaceType	eInterfaceType	The type of the interface.
uint8	u8ChannelId	The channel id of the interface.
uint8	u8MainUnitAddress	The address used to communicate with the Main Unit.
uint8	u8FailSafeUnitAddress	The address used to communicate with the Fail-Safe Unit.
uint32	u32I2cSclPinIndex	The scl pin index.
uint8	u8I2cSclPinInitialMode	The device's i2c scl pin - initial mode.
uint8	u8I2cSclPinMode	The device's i2c scl pin - mode.
uint32	u32CPUFreq	The device's operating frequency.
uint32	u32I2cFreq	The device's i2c channel frequency.
uint16	u16I2cSclDioChannelId	The device's dio channel id.

6.1.2.4 struct Pmic_DeviceConfigType

Configuration structure for a PMIC device.

It contains the configuration that a particular PMIC device is using.

Definition at line 312 of file Pmic_Types.h.

Data Fields

- uint8 u8DeviceUnitId
  Numeric instance value of the PMIC device.
- uint8 u8NrOfModeConfigs
  Total number of mode configurations.
- uint8 u8NrOfClockConfigs
  Total number of clock configurations.
- uint8 u8NrOfReactionConfigs
  Total number of reaction configurations.
- uint8 u8NrOfAmuxChannelConfigs

Total number of AMUX channel configurations.

- uint8 u8NrOfSVSConfigs

  Total number of SVS configurations.
- uint8 u8NrOfWatchdogConfigs

  Total number of watchdog configurations.
- const Pmic_Ipw_ModeConfigType(* paModeConfig)[]

  Mode data configurations.
- const Pmic_Ipw_ClockConfigType(* paClockConfig)[]

  Clock data configurations.
- const Pmic_Ipw_ReactionConfigType(* paReactionConfig)[]

  Reaction data configurations.
- const Pmic_Ipw_AmuxChannelConfigType(* paAmuxChannelConfig)[]

  AMUX channel configurations.
- const Pmic_Ipw_SVSConfigType(* paSVSConfig)[]

  SVS configurations.
- const Pmic_Ipw_WatchdogConfigType(* paWatchdogConfig)[]

  Watchdog data configurations.
- Pmic_InterfaceUnitConfigType * pInterfaceConfig

  The configuration of the interface used to communicate with the PMIC device.
- const Pmic_Ipw_HwDeviceConfigType * pHwDeviceConfig

  The hardware configuration of the PMIC device.

6.1.2.4.1 Field Documentation

6.1.2.4.1.1 u8DeviceUnitId `uint8 u8DeviceUnitId`

Numeric instance value of the PMIC device.

Definition at line 314 of file Pmic_Types.h.

6.1.2.4.1.2 u8NrOfModeConfigs `uint8 u8NrOfModeConfigs`

Total number of mode configurations.

Definition at line 315 of file Pmic_Types.h.

6.1.2.4.1.3 u8NrOfClockConfigs `uint8 u8NrOfClockConfigs`

Total number of clock configurations.

Definition at line 316 of file Pmic_Types.h.

6.1.2.4.1.4 u8NrOfReactionConfigs `uint8 u8NrOfReactionConfigs`

Total number of reaction configurations.

Definition at line 317 of file Pmic_Types.h.

6.1.2.4.1.5 u8NrOfAmuxChannelConfigs `uint8 u8NrOfAmuxChannelConfigs`

Total number of AMUX channel configurations.

Definition at line 319 of file Pmic_Types.h.

6.1.2.4.1.6 u8NrOfSVSConfigs `uint8 u8NrOfSVSConfigs`

Total number of SVS configurations.

Definition at line 322 of file Pmic_Types.h.

6.1.2.4.1.7 u8NrOfWatchdogConfigs `uint8 u8NrOfWatchdogConfigs`

Total number of watchdog configurations.

Definition at line 325 of file Pmic_Types.h.

6.1.2.4.1.8 paModeConfig `const Pmic_Ipw_ModeConfigType(* paModeConfig)[]`

Mode data configurations.

Definition at line 327 of file Pmic_Types.h.

6.1.2.4.1.9 paClockConfig `const Pmic_Ipw_ClockConfigType(* paClockConfig)[]`

Clock data configurations.

Definition at line 328 of file Pmic_Types.h.

6.1.2.4.1.10 paReactionConfig `const Pmic_Ipw_ReactionConfigType(* paReactionConfig)[]`

Reaction data configurations.

Definition at line 329 of file Pmic_Types.h.

6.1.2.4.1.11 paAmuxChannelConfig `const Pmic_Ipw_AmuxChannelConfigType(* paAmuxChannelConfig)[]`

AMUX channel configurations.

Definition at line 331 of file Pmic_Types.h.

6.1.2.4.1.12 paSVSConfig `const Pmic_Ipw_SVSConfigType(* paSVSConfig)[]`

SVS configurations.

Definition at line 334 of file Pmic_Types.h.

6.1.2.4.1.13 paWatchdogConfig `const Pmic_Ipw_WatchdogConfigType(* paWatchdogConfig)[]`

Watchdog data configurations.

Definition at line 337 of file Pmic_Types.h.

6.1.2.4.1.14 pInterfaceConfig `Pmic_InterfaceUnitConfigType* pInterfaceConfig`

The configuration of the interface used to communicate with the PMIC device.

Definition at line 339 of file Pmic_Types.h.

6.1.2.4.1.15 pHwDeviceConfig `const Pmic_Ipw_HwDeviceConfigType* pHwDeviceConfig`

The hardware configuration of the PMIC device.

Definition at line 340 of file Pmic_Types.h.

6.1.2.5 struct Pmic_DemConfigType

DEM error reporting configuration.

This structure contains information DEM error reporting.

Definition at line 351 of file Pmic_Types.h.

6.1.2.6 struct Pmic_ConfigType

Configuration structure of the driver.

It contains the configuration used for initializing the driver.

Definition at line 368 of file Pmic_Types.h.

Data Fields

- uint32 Pmic_u32DiePreocessOcotpIndex
  Index of Ocotp channel read OCOTP_BANK0_WORD6.
- uint8 Pmic_u8NrOfDevices
  Total number of configured PMIC devices.
- const Pmic_WdgTaskNotificationType Pmic_pWdgTaskNotification
  Pointer to callout configured by user for watchdog task notifications.
- const Pmic_DeviceConfigType(* Pmic_paDeviceConfig)[PMIC_MAX_DEVICECONFIGS]
  Pointer to list of PMIC device configurations.
- const Pmic_DemConfigType * Pmic_pDemConfig
  DEM error reporting configuration.

6.1.2.6.1 Field Documentation

6.1.2.6.1.1 Pmic_u32DiePreocessOcotpIndex `uint32 Pmic_u32DiePreocessOcotpIndex`

Index of Ocotp channel read OCOTP_BANK0_WORD6.

Definition at line 371 of file Pmic_Types.h.

6.1.2.6.1.2 Pmic_u8NrOfDevices `uint8 Pmic_u8NrOfDevices`

Total number of configured PMIC devices.

Definition at line 373 of file Pmic_Types.h.

6.1.2.6.1.3 Pmic_pWdgTaskNotification

`const Pmic_WdgTaskNotificationType Pmic_pWdgTaskNotification`

Pointer to callout configured by user for watchdog task notifications.

Definition at line 375 of file Pmic_Types.h.

6.1.2.6.1.4 Pmic_paDeviceConfig

`const Pmic_DeviceConfigType(* Pmic_paDeviceConfig)[PMIC_MAX_DEVICECONFIGS]`

Pointer to list of PMIC device configurations.

Definition at line 377 of file Pmic_Types.h.

6.1.2.6.1.5 Pmic_pDemConfig

`const Pmic_DemConfigType* Pmic_pDemConfig`

DEM error reporting configuration.

Definition at line 379 of file Pmic_Types.h.

6.1.3 Macro Definition Documentation

6.1.3.1 PMIC_E_OK

```
#define PMIC_E_OK
```

Internal return codes.

The enumeration with all the internal return codes interpreted by the driver

Definition at line 104 of file Pmic_Internals.h.

6.1.4 Types Reference

6.1.4.1 Pmic_WdgTaskNotificationType

```
typedef void(* Pmic_WdgTaskNotificationType) (const Pmic_DeviceIndexType DeviceId, const uint8 u8WdWindowValue, const uint8 u8WdDcValue)
```

This gives the call-out type for watchdog task notifications.

Definition at line 247 of file Pmic_Types.h.

6.1.5 Enum Reference

6.1.5.1 Pmic_ApiServiceIdType

`enum Pmic_ApiServiceIdType`

API service IDs.

Service IDs of the PMIC API.

Enumerator

PMIC_INIT_ID	Pmic_Init() ID.
PMIC_INIT_DEVICE_ID	Pmic_InitDevice() ID.
PMIC_EMULATE_DEVICE_OTP_ID	Pmic_EmulateDeviceOTP() ID.
PMIC_READ_REGISTER_ID	Pmic_ReadRegister() ID.
PMIC_WRITE_REGISTER_ID	Pmic_WriteRegister() ID.
PMIC_INIT_CLOCK_ID	Pmic_InitClock() ID.
PMIC_SET_MODE_ID	Pmic_SetMode() ID.
PMIC_SET_ANALOG_MUX_ID	Pmic_SetAnalogMux() ID.
PMIC_CONFIGURE_WATCHDOG_ID	Pmic_ConfigureWatchdog() ID.
PMIC_TRIGGER_WATCHDOG_ID	Pmic_TriggerWatchdog() ID.
PMIC_GET_RAW_FAULT_EVENTS_ID	Pmic_GetRawFaultEvents() ID.
PMIC_SET_REACTIONS_ID	Pmic_SetReactions() ID.
PMIC_GET_DEVICE_INFO_ID	Pmic_GetDeviceInfo() ID.
PMIC_GET_VERSION_INFO_ID	Pmic_GetVersionInfo() ID.
PMIC_DISABLE_WATCHDOG_ID	Pmic_DisableWatchdog() ID
PMIC_SWITCH_SVS_ID	Pmic_SwitchSVS() ID.
PMIC_GOTO_INITFS_ID	Pmic_GotoInitFS() ID.
PMIC_RELEASE_FS0B_ID	Pmic_ReleaseFs0b() ID.

Definition at line 144 of file Pmic_Types.h.

6.1.5.2 Pmic_ErrorIdType

```
enum Pmic_ErrorIdType
```

Error IDs.

Error IDs of the PMIC API.

Enumerator

PMIC_E_UNINIT	Driver uninitialized.
PMIC_E_INIT_FAILED	Invalid configuration pointer.
PMIC_E_DEVICE_UNINIT	Device uninitialized.
PMIC_E_DEVICE_OTP_UNEMU	Un-emulated OTP on blank device.
PMIC_E_PARAM_POINTER	Invalid pointer argument.
PMIC_E_PARAM_DEVICE	Invalid device ID.
PMIC_E_PARAM_MODE	Invalid mode configuration ID.
PMIC_E_PARAM_CLOCK	Invalid clock configuration ID.
PMIC_E_PARAM_WDG	Invalid watchdog configuration ID.
PMIC_E_PARAM_AMUX	Invalid AMUX channel ID
PMIC_E_PARAM_REACTION	Invalid reaction configuration ID.
PMIC_E_PARAM_REG_ADDR	Invalid register address.
PMIC_E_COMM_FAILURE	Underlying communication failed
PMIC_E_INVALID_DATA	CRC Verification Failed
PMIC_E_PARAM_SVS	Invalid SVS ID

Definition at line 186 of file Pmic_Types.h.

6.1.5.3 Pmic_InterfaceType

```
enum Pmic_InterfaceType
```

Interface selection.

The enumeration with interface possibilities

Enumerator

PMIC_INTERFACE_I2C	Device communicates through I2C.
PMIC_INTERFACE_SPI	Device communicates through SPI.

Definition at line 221 of file Pmic_Types.h.

6.1.5.4 Pmic_InterfaceDstType

```
enum Pmic_InterfaceDstType
```

Interface destination selection.

The enumeration with interface destination possibilities.

Enumerator

PMIC_MAIN_UNIT	Interface with the Main Unit.
PMIC_FAIL_SAFE_UNIT	Interface with the Fail-Safe Unit.

Definition at line 232 of file Pmic_Types.h.

6.1.6 Function Reference

6.1.6.1 Pmic_Init()

```
void Pmic_Init (
            const Pmic_ConfigType * pConfigPtr )
```

Pmic_Init.

Parameters

in	*pConfigPtr.*	

Returns

void.

6.1.6.2 Pmic_InitDevice()

```
Std_ReturnType Pmic_InitDevice (
            const Pmic_DeviceIndexType DeviceId )
```

Pmic_InitDevice.

Parameters

in	*DeviceId.*	

Returns

Std_ReturnType.

6.1.6.3 Pmic_EmulateDeviceOTP()

```
Std_ReturnType Pmic_EmulateDeviceOTP (
            const Pmic_DeviceIndexType DeviceId )
```

Pmic_EmulateDeviceOTP.

Parameters

in	*DeviceId.*	

Returns

Std_ReturnType.

6.1.6.4 Pmic_ReadRegister()

```
Std_ReturnType Pmic_ReadRegister (
            const Pmic_DeviceIndexType DeviceId,
            const Pmic_InterfaceDstType eDestination,
            const uint8 u8RegAddr,
            uint16 * pu16ValueOut )
```

Pmic_ReadRegister.

Parameters

in	*DeviceId.*	
in	*eDestination.*	
in	*u8RegAddr.*	
in	*pu16ValueOut.*	

Returns

Std_ReturnType.

6.1.6.5 Pmic_WriteRegister()

```
Std_ReturnType Pmic_WriteRegister (
            const Pmic_DeviceIndexType DeviceId,
            const Pmic_InterfaceDstType eDestination,
            const uint8 u8RegAddr,
            const uint16 u16WriteValue )
```

Pmic_WriteRegister.

Parameters

in	*DeviceId.*	
in	*eDestination.*	
in	*u8RegAddr.*	
in	*u16WriteValue.*	

Returns

Std_ReturnType.

6.1.6.6 Pmic_InitClock()

```
Std_ReturnType Pmic_InitClock (
            const Pmic_DeviceIndexType DeviceId,
            const Pmic_ClockSettingIndexType ClockSettingId )
```

Pmic_InitClock.

Parameters

in	*DeviceId.*	
in	*ClockSettingId.*	

Returns

Std_ReturnType.

6.1.6.7 Pmic_SetMode()

```
Std_ReturnType Pmic_SetMode (
            const Pmic_DeviceIndexType DeviceId,
            const Pmic_ModeIndexType ModeSettingId )
```

Pmic_SetMode.

Parameters

in	*DeviceId.*	
in	*ModeSettingId.*	

Returns

Std_ReturnType.

6.1.6.8 Pmic_SetAnalogMux()

```
Std_ReturnType Pmic_SetAnalogMux (
            const Pmic_DeviceIndexType DeviceId,
            const Pmic_AmuxChannelIndexType AmuxChannelId )
```

Pmic_SetAnalogMux.

Parameters

in	*DeviceId.*	
in	*AmuxChannelId.*	

Returns

Std_ReturnType.

6.1.6.9 Pmic_SwitchSVS()

```
Std_ReturnType Pmic_SwitchSVS (
            const Pmic_DeviceIndexType DeviceId,
            const Pmic_SvsSettingIndexType SvsSettingID )
```

Pmic_SwitchSVS.

Parameters

in	*DeviceId.*	
in	*SvsSettingID.*	

Returns

Std_ReturnType.

6.1.6.10 Pmic_ConfigureWatchdog()

```
Std_ReturnType Pmic_ConfigureWatchdog (
            const Pmic_DeviceIndexType DeviceId,
            const Pmic_WatchdogSettingIndexType WatchdogSettingId )
```

Pmic_ConfigureWatchdog.

Parameters

in	*DeviceId.*	
in	*WatchdogSettingId.*	

Returns

Std_ReturnType.

6.1.6.11 Pmic_TriggerWatchdog()

```
void Pmic_TriggerWatchdog (
            const Pmic_DeviceIndexType DeviceId )
```

Pmic_TriggerWatchdog.

Parameters

in	*DeviceId.*	

Returns

void.

6.1.6.12 Pmic_DisableWatchdog()

```
Std_ReturnType Pmic_DisableWatchdog (
            const Pmic_DeviceIndexType DeviceId )
```

Pmic_DisableWatchdog.

Parameters

in	*DeviceId.*	

Returns

Std_ReturnType.

6.1.6.13 Pmic_ReleaseFs0b()

```
Std_ReturnType Pmic_ReleaseFs0b (
            const Pmic_DeviceIndexType DeviceId )
```

PMIC executes the release safety FS0B pin.

This function is used to release the safety FS0B pin. The API will not release the FS0B pin and return the E_↩ NOT_OK if the PMIC device is not in the ASSERT_FS0B state. This API will report the E_NOT_OK if the Pmic_Init and Pmic_InitDevice APIs were not called successfully before. This function has powerful in case PMIC reacted to assert the FS0B pin and is stuck in the ASSERT_FS0B state. If users want PMIC to release FS0B and switch to the NORMAL_FS state.

Parameters

in	*Device↩ Id*	The ID of device.

Returns

E_OK: Release FS0B pin successfully E_NOT_OK: Release FS0B pin unsuccessfully

6.1.6.14 Pmic_GotoInitFS()

```
Std_ReturnType Pmic_GotoInitFS (
            const Pmic_DeviceIndexType DeviceId )
```

Pmic_GotoInitFS.

Parameters

in	*DeviceId.*	

Returns

Std_ReturnType.

6.1.6.15 Pmic_GetRawFaultEvents()

```
Std_ReturnType Pmic_GetRawFaultEvents (
            const Pmic_DeviceIndexType DeviceId,
            Pmic_RawFaultType * pRawFaultEventsOut )
```

Pmic_GetRawFaultEvents.

Parameters

in	*DeviceId.*	
out	*pRawFaultEventsOut.*	

Returns

Std_ReturnType.

6.1.6.16 Pmic_SetReactions()

```
Std_ReturnType Pmic_SetReactions (
            const Pmic_DeviceIndexType DeviceId,
            const Pmic_ReactionSettingIndexType ReactionSettingId )
```

Pmic_SetReactions.

Parameters

in	*DeviceId.*	
in	*ReactionSettingId.*	

Returns

Std_ReturnType.

6.1.6.17 Pmic_GetDeviceInfo()

```
Std_ReturnType Pmic_GetDeviceInfo (
            const Pmic_DeviceIndexType DeviceId,
            Pmic_DeviceInfoType * pDeviceInfoOut )
```

Pmic_GetDeviceInfo.

Parameters

in	*DeviceId.*	
in	*pDeviceInfoOut.*	

Returns

Std_ReturnType.

6.1.6.18 Pmic_GetVersionInfo()

```
void Pmic_GetVersionInfo (
            Std_VersionInfoType * pVersionInfoOut )
```

Pmic_GetVersionInfo.

Parameters

in	*pVersionInfoOut.*	

Returns

void.

6.1.6.19 Pmic_Dem_Check_Device()

```
void Pmic_Dem_Check_Device (
            const Pmic_ReturnType eStatusCode )
```

API checks status and report the device run time error.

The function will report when users cannot access the device, time out by hardware checking, the wrong hardware behavior, clock setting failure, safety pin cannot release.

Parameters

in	*eStatusCode*	The status of device need to report runtime.

6.1.6.20 Pmic_HLDChecksEntry()

```
Std_ReturnType Pmic_HLDChecksEntry (
            const Pmic_ApiServiceIdType ePmicServiceID )
```

API checks the Pmic_Init executed successfully before.

The API will report the det error if Pmic_Init did not execute successfully before.

Parameters

in	*ePmicServiceID*	The ID of PMIC API service.

Returns

E_OK: Check is successful E_NOT_OK: Check is unsuccessful

6.1.6.21 Pmic_HLDChecksExit()

```
void Pmic_HLDChecksExit (
            const Std_ReturnType eRetStatus,
            const Pmic_ApiServiceIdType ePmicServiceID )
```

API checks the Pmic driver status in Pmic_Init API.

Parameters

in	*eRetStatus*	the status of process before entrance this function. This will use to set the driver status.
in	*ePmicServiceID*	The ID of PMIC API service.

6.1.6.22 Pmic_HLDDeviceChecksEntry()

```
Std_ReturnType Pmic_HLDDeviceChecksEntry (
            const Pmic_DeviceIndexType u32DeviceId,
            const Pmic_ApiServiceIdType ePmicServiceID )
```

API checks the PMIC device already initialized before call another API.

The API will report the det error if u32DeviceId is out of range. The det error also reports if PMIC device uninitialized.

Parameters

in	*u32DeviceId*	The ID of PMIC device
in	*ePmicServiceID*	The ID of PMIC API service.

Returns

E_OK: Check is successful E_NOT_OK: Check is unsuccessful

6.1.6.23 Pmic_HLDDeviceChecksExit()

```
void Pmic_HLDDeviceChecksExit (
            const Std_ReturnType eRetStatus,
            const Pmic_DeviceIndexType u32DeviceId,
            const Pmic_ApiServiceIdType ePmicServiceID )
```

API checks the Pmic driver status when user executes the Pmic_InitDevice, Pmic_EmulateDeviceOTP API.

Parameters

in	*eRetStatus*	the status of process before entrance this function.
in	*u32DeviceId*	The ID of PMIC device
in	*ePmicServiceID*	The ID of PMIC API service.

6.1.6.24 Pmic_Check_Init()

```
Std_ReturnType Pmic_Check_Init (
            const Pmic_ConfigType * pConfigPtr )
```

API checks the Pmic_Init parameter input valid.

Parameters

in	*pConfigPtr*	The pointer driver configuration.

Returns

E_OK: Check is successful E_NOT_OK: Check is unsuccessful

6.1.6.25 Pmic_Check_ReadRegister()

```
Std_ReturnType Pmic_Check_ReadRegister (
            const Pmic_DeviceIndexType u32DeviceId,
            const uint8 u8RegAddr,
            const uint16 * pu16ValueOut )
```

API checks the Pmic_ReadRegister parameter input valid.

Parameters

in	*u32DeviceId*	The ID of PMIC device
in	*u8RegAddr*	The PMIC register address
in	*pu16ValueOut*	The pointer stores the reading value.

Returns

E_OK: Check is successful E_NOT_OK: Check is unsuccessful

6.1.6.26 Pmic_Check_WriteRegister()

```
Std_ReturnType Pmic_Check_WriteRegister (
            const Pmic_DeviceIndexType u32DeviceId,
            const uint8 u8RegAddr )
```

API checks the Pmic_WriteRegister parameter input valid.

Parameters

in	*u32↩ DeviceId*	The ID of PMIC device
in	*u8RegAddr*	The PMIC register address

Returns

E_OK: Check is successful E_NOT_OK: Check is unsuccessful

6.1.6.27 Pmic_Check_SetMode()

```
Std_ReturnType Pmic_Check_SetMode (
            const Pmic_DeviceIndexType u32DeviceId,
            const Pmic_ModeIndexType u32ModeSettingId )
```

API checks the Pmic_Setmode parameter input valid.

Parameters

in	*u32DeviceId*	The ID of PMIC device
in	*u32Mode↩ SettingId*	The mode setting index

Returns

E_OK: Check is successful E_NOT_OK: Check is unsuccessful

6.1.6.28 Pmic_Check_InitClock()

```
Std_ReturnType Pmic_Check_InitClock (
            const Pmic_DeviceIndexType u32DeviceId,
            const Pmic_ClockSettingIndexType u32ClockSettingID )
```

API checks the Pmic_InitClock parameter input valid.

Parameters

in	*u32DeviceId*	The ID of PMIC device
in	*u32Clock↩ SettingId*	The clock setting index

Returns

E_OK: Check is successful E_NOT_OK: Check is unsuccessful

6.1.6.29 Pmic_Check_SetAnalogMux()

```
Std_ReturnType Pmic_Check_SetAnalogMux (
            const Pmic_DeviceIndexType u32DeviceId,
            const Pmic_AmuxChannelIndexType u32AmuxChannelId )
```

API checks the Pmic_SetAnalogMux parameter input valid.

Parameters

in	*u32DeviceId*	The ID of PMIC device
in	*u32Amux↩ ChannelId*	The amux setting index

Returns

E_OK: Check is successful E_NOT_OK: Check is unsuccessful

6.1.6.30 Pmic_Check_SwitchSVS()

```
Std_ReturnType Pmic_Check_SwitchSVS (
            const Pmic_DeviceIndexType u32DeviceId,
            const Pmic_SvsSettingIndexType u32SvsSettingID )
```

API checks the Pmic_SwitchSVS parameter input valid.

Parameters

in	*u32DeviceId*	The ID of PMIC device
in	*u32SvsSettingID*	The SVS setting index

Returns

E_OK: Check is successful E_NOT_OK: Check is unsuccessful

6.1.6.31 Pmic_Check_ConfigureWatchdog()

```
Std_ReturnType Pmic_Check_ConfigureWatchdog (
            const Pmic_DeviceIndexType u32DeviceId,
            const Pmic_WatchdogSettingIndexType u32WatchdogSettingId )
```

API checks the Pmic_ConfigureWatchdog parameter input valid.

Parameters

in	*u32DeviceId*	The ID of PMIC device
in	*u32Watchdog↩ SettingId*	The watchdog setting index

Returns

E_OK: Check is successful E_NOT_OK: Check is unsuccessful

6.1.6.32 Pmic_Check_TriggerWatchdog()

```
Std_ReturnType Pmic_Check_TriggerWatchdog (
            const Pmic_DeviceIndexType u32DeviceId )
```

API checks the Pmic_TriggerWatchdog parameter input valid.

Parameters

in	*u32↩ DeviceId*	The ID of PMIC device

Returns

E_OK: Check is successful E_NOT_OK: Check is unsuccessful

6.1.6.33 Pmic_Check_GotoInitFS()

```
Std_ReturnType Pmic_Check_GotoInitFS (
            const Pmic_DeviceIndexType u32DeviceId )
```

API checks the Pmic_GotoInitFS parameter input valid.

Parameters

in	*u32↩ DeviceId*	The ID of PMIC device

Returns

E_OK: Check is successful E_NOT_OK: Check is unsuccessful

6.1.6.34 Pmic_Check_GetRawFaultEvents()

```
Std_ReturnType Pmic_Check_GetRawFaultEvents (
            const Pmic_RawFaultType * pRawFaultEventsOut )
```

API checks the Pmic_GetRawFaultEvents parameter input valid.

Parameters

in	*pRawFaultEventsOut*	The pointer stores the fault error counter.

Returns

E_OK: Check is successful E_NOT_OK: Check is unsuccessful

6.1.6.35 Pmic_Check_SetReactions()

```
Std_ReturnType Pmic_Check_SetReactions (
            const Pmic_DeviceIndexType u32DeviceId,
            const Pmic_ReactionSettingIndexType u32ReactionSettingId )
```

API checks the Pmic_SetReactions parameter input valid.

.

Parameters

in	*u32DeviceId*	The ID of PMIC device
in	*u32Reaction↩ SettingId*	The reaction setting index

Returns

E_OK: Check is successful E_NOT_OK: Check is unsuccessful

6.1.6.36 Pmic_Check_GetDeviceInfo()

```
Std_ReturnType Pmic_Check_GetDeviceInfo (
            const Pmic_DeviceInfoType * pDeviceInfoOut )
```

API checks the Pmic_GetDeviceInfo parameter input valid.

Parameters

in	*pDeviceInfoOut*	The pointer gets the device information.

Returns

E_OK: Check is successful E_NOT_OK: Check is unsuccessful

6.1.6.37 Pmic_Check_GetVersionInfo()

```
Std_ReturnType Pmic_Check_GetVersionInfo (
            const Std_VersionInfoType * pVersionInfo )
```

API checks the Pmic_GetVersionInfo parameter input valid.

Parameters

in	*pVersionInfo*	The pointer gets the driver version information.

Returns

E_OK: Check is successful E_NOT_OK: Check is unsuccessful

6.1.6.38 Pmic_Det_Check_ReportErrors()

```
void Pmic_Det_Check_ReportErrors (
            const Pmic_ReturnType eStatusCode,
            const Pmic_ApiServiceIdType eServiceId )
```

API checks status and report the run time error.

Parameters

in	*eStatusCode*	The status of driver
in	*eServiceId*	The API service id

Returns

E_OK: Check is successful E_NOT_OK: Check is unsuccessful

6.1.7 Variable Documentation

6.1.7.1 Pmic_abTestMode

```
boolean Pmic_abTestMode[PMIC_MAX_DEVICECONFIGS]  [extern]
```

Array used to indicate whether the indexed device is in test mode.

6.1.7.2 Pmic_abErrorDetected

```
boolean Pmic_abErrorDetected[PMIC_MAX_DEVICECONFIGS]  [extern]
```

Array used to indicate whether the indexed device encountered an error.

6.1.7.3 Pmic_abGateWdgTrigger

```
boolean Pmic_abGateWdgTrigger[PMIC_MAX_DEVICECONFIGS]  [extern]
```

Array used to indicate watchdog trigger will be forwarded to the device.

6.1.7.4 Pmic_abIsWdgTrigger

```
boolean Pmic_abIsWdgTrigger[PMIC_MAX_DEVICECONFIGS]  [extern]
```

Array used to check watchdog trigger in progress or not. When Wdg_trigger is in-progress, variable will set true.

6.2 Pmic Ipw driver

6.2.1 Detailed Description

Data Structures

- struct Pmic_Ipw_ModeConfigType
  The structure contains the mode configuration type. More...
- struct Pmic_Ipw_ClockConfigType
  The structure contains the clock configuration type. More...
- struct Pmic_Ipw_ReactionConfigType
  The structure contains the reaction configuration type. More...
- struct Pmic_Ipw_HwDeviceConfigType
  The structure contains the hardware configuration type. More...

6.2.2 Data Structure Documentation

6.2.2.1 struct Pmic_Ipw_ModeConfigType

The structure contains the mode configuration type.

Definition at line 115 of file Pmic_Ipw_Types.h.

6.2.2.2 struct Pmic_Ipw_ClockConfigType

The structure contains the clock configuration type.

Definition at line 125 of file Pmic_Ipw_Types.h.

6.2.2.3 struct Pmic_Ipw_ReactionConfigType

The structure contains the reaction configuration type.

Definition at line 135 of file Pmic_Ipw_Types.h.

6.2.2.4 struct Pmic_Ipw_HwDeviceConfigType

The structure contains the hardware configuration type.

Definition at line 178 of file Pmic_Ipw_Types.h.

6.3 Pmic VR5510 driver

6.3.1 Detailed Description

Data Structures

- struct Pmic_VR55XX_ModeConfigType

  The structure contains the mode configuration type for VR55XX. More...
- struct Pmic_VR55XX_ClockConfigType

  The structure contains the clock configuration type for VR55XX. More...
- struct Pmic_VR55XX_ReactionConfigType

  The structure contains the reaction configuration type for VR55XX. More...
- struct Pmic_VR55XX_AmuxChannelConfigType

  The structure contains the AMUX channel configuration type for VR55XX. More...
- struct Pmic_VR55XX_SVSConfigType

  The structure contains the SVS setting configuration type for VR55XX. More...
- struct Pmic_VR55XX_WatchdogConfigType

  The structure contains the watchdog configuration type for VR55XX. More...
- struct Pmic_VR55XX_DeviceInfoType

  The structure contains the device information type for VR55XX. More...
- struct Pmic_VR55XX_RawFaultType

  The structure contains the raw fault events statuses for VR55XX. More...

Enum Reference

- enum Pmic_VR55XX_PowerModeType

  Power Modes Encoding for VR55XX.

Function Reference

- Pmic_ReturnType Pmic_VR55XX_I2C_ReadRegister (const Pmic_DeviceIndexType u32DeviceId, const Pmic_InterfaceDstType eDestination, const uint8 u8RegAddr, uint16 *pu16ValueOut)

  Pmic_VR55XX_I2C_ReadRegister.
- Pmic_ReturnType Pmic_VR55XX_I2C_WriteRegister (const Pmic_DeviceIndexType u32DeviceId, const Pmic_InterfaceDstType eDestination, const uint8 u8RegAddr, const uint16 u16WriteValue)

  Pmic_VR55XX_I2C_WriteRegister.
- Pmic_ReturnType Pmic_VR55XX_InitDevice (const Pmic_DeviceIndexType u32DeviceId)

  Pmic_VR55XX_InitDevice.
- Pmic_ReturnType Pmic_VR55XX_EmulateDeviceOTP (const Pmic_DeviceIndexType u32DeviceId)

  Pmic_VR55XX_EmulateDeviceOTP.
- Pmic_ReturnType Pmic_VR55XX_InitClock (const Pmic_DeviceIndexType u32DeviceId, const Pmic_↩ ClockSettingIndexType u32ClockSettingID)

  Pmic_VR55XX_InitClock.

- Pmic_ReturnType Pmic_VR55XX_SetMode (const Pmic_DeviceIndexType u32DeviceId, const Pmic_↩ ModeIndexType u32ModeSettingID)

  Pmic_VR55XX_SetMode.
- Pmic_ReturnType Pmic_VR55XX_SetAnalogMux (const Pmic_DeviceIndexType u32DeviceId, const Pmic_AmuxChannelIndexType u32AmuxChannelID)

  Pmic_VR55XX_SetAnalogMux.
- Pmic_ReturnType Pmic_VR55XX_SwitchSVS (const Pmic_DeviceIndexType u32DeviceId, const Pmic_↩ SvsSettingIndexType u32SvsSettingID)

  Pmic_VR55XX_SwitchSVS.
- Pmic_ReturnType Pmic_VR55XX_ConfigureWatchdog (const Pmic_DeviceIndexType u32DeviceId, const Pmic_WatchdogSettingIndexType u32WatchdogSettingID)

  Pmic_VR55XX_ConfigureWatchdog.
- Pmic_ReturnType Pmic_VR55XX_TriggerWatchdog (const Pmic_DeviceIndexType u32DeviceId)

  Pmic_VR55XX_TriggerWatchdog.
- Pmic_ReturnType Pmic_VR55XX_DisableWatchdog (const Pmic_DeviceIndexType u32DeviceId)

  Pmic_VR55XX_DisableWatchdog.
- Pmic_ReturnType Pmic_VR55XX_ReleaseFs0b (const Pmic_DeviceIndexType u32DeviceId)

  Pmic_VR55XX_ReleaseFs0b.
- Pmic_ReturnType Pmic_VR55XX_GetRawFaultEvents (const Pmic_DeviceIndexType u32DeviceId, Pmic_VR55XX_RawFaultType *pRawFaultEventsOut_VR55XX)

  Pmic_VR55XX_GetRawFaultEvents.
- Pmic_ReturnType Pmic_VR55XX_SetReactions (const Pmic_DeviceIndexType u32DeviceId, const Pmic↩ _ReactionSettingIndexType u32ReactionSettingID)

  Pmic_VR55XX_SetReactions.
- Pmic_ReturnType Pmic_VR55XX_GetDeviceInfo (const Pmic_DeviceIndexType u32DeviceId, Pmic_VR55XX_DeviceIn *pDeviceInfoOut)

  Pmic_VR55XX_GetDeviceInfo.
- Pmic_ReturnType Pmic_VR55XX_GotoInitFS (const Pmic_DeviceIndexType u32DeviceId)

  Pmic_VR55XX_GotoInitFS.

6.3.2 Data Structure Documentation

6.3.2.1 struct Pmic_VR55XX_ModeConfigType

The structure contains the mode configuration type for VR55XX.

Definition at line 168 of file Pmic_VR55XX_Types.h.

6.3.2.2 struct Pmic_VR55XX_ClockConfigType

The structure contains the clock configuration type for VR55XX.

Definition at line 185 of file Pmic_VR55XX_Types.h.

6.3.2.3 struct Pmic_VR55XX_ReactionConfigType

The structure contains the reaction configuration type for VR55XX.

Definition at line 198 of file Pmic_VR55XX_Types.h.

6.3.2.4 struct Pmic_VR55XX_AmuxChannelConfigType

The structure contains the AMUX channel configuration type for VR55XX.

Definition at line 215 of file Pmic_VR55XX_Types.h.

6.3.2.5 struct Pmic_VR55XX_SVSConfigType

The structure contains the SVS setting configuration type for VR55XX.

Definition at line 227 of file Pmic_VR55XX_Types.h.

6.3.2.6 struct Pmic_VR55XX_WatchdogConfigType

The structure contains the watchdog configuration type for VR55XX.

Definition at line 238 of file Pmic_VR55XX_Types.h.

6.3.2.7 struct Pmic_VR55XX_DeviceInfoType

The structure contains the device information type for VR55XX.

Definition at line 250 of file Pmic_VR55XX_Types.h.

6.3.2.8 struct Pmic_VR55XX_RawFaultType

The structure contains the raw fault events statuses for VR55XX.

Definition at line 260 of file Pmic_VR55XX_Types.h.

6.3.3 Enum Reference

6.3.3.1 Pmic_VR55XX_PowerModeType

```
enum Pmic_VR55XX_PowerModeType
```

Power Modes Encoding for VR55XX.

Enumerator

PMIC_NORMAL_MODE	Normal Mode.
PMIC_RESET_MODE	Reset Mode.
PMIC_SHUTDOWN_MODE	Shutdown Mode.
PMIC_STANDBY_MODE	Standby Mode.
PMIC_DEEPSLEEP_MODE	Deep Sleep Mode.

Definition at line 155 of file Pmic_VR55XX_Types.h.

6.3.4 Function Reference

6.3.4.1 Pmic_VR55XX_I2C_ReadRegister()

```
Pmic_ReturnType Pmic_VR55XX_I2C_ReadRegister (
            const Pmic_DeviceIndexType u32DeviceId,
            const Pmic_InterfaceDstType eDestination,
            const uint8 u8RegAddr,
            uint16 * pu16ValueOut )
```

Pmic_VR55XX_I2C_ReadRegister.

Parameters

in	*u32DeviceId.*	
in	*eDestination.*	
in	*u8RegAddr.*	
in	*pu16ValueOut.*	

Returns

Pmic_ReturnType.

6.3.4.2 Pmic_VR55XX_I2C_WriteRegister()

```
Pmic_ReturnType Pmic_VR55XX_I2C_WriteRegister (
            const Pmic_DeviceIndexType u32DeviceId,
            const Pmic_InterfaceDstType eDestination,
            const uint8 u8RegAddr,
            const uint16 u16WriteValue )
```

Pmic_VR55XX_I2C_WriteRegister.

Parameters

in	*u32DeviceId.*	
in	*eDestination.*	
in	*u8RegAddr.*	
in	*u16WriteValue.*	

Returns

Pmic_ReturnType.

6.3.4.3 Pmic_VR55XX_InitDevice()

```
Pmic_ReturnType Pmic_VR55XX_InitDevice (
            const Pmic_DeviceIndexType u32DeviceId )
```

Pmic_VR55XX_InitDevice.

Parameters

in	*u32DeviceId.*	

Returns

Pmic_ReturnType.

6.3.4.4 Pmic_VR55XX_EmulateDeviceOTP()

```
Pmic_ReturnType Pmic_VR55XX_EmulateDeviceOTP (
            const Pmic_DeviceIndexType u32DeviceId )
```

Pmic_VR55XX_EmulateDeviceOTP.

Parameters

in	*u32DeviceId.*	

Returns

Pmic_ReturnType.

6.3.4.5 Pmic_VR55XX_InitClock()

```
Pmic_ReturnType Pmic_VR55XX_InitClock (
            const Pmic_DeviceIndexType u32DeviceId,
            const Pmic_ClockSettingIndexType u32ClockSettingID )
```

Pmic_VR55XX_InitClock.

Parameters

in	*u32DeviceId.*	
in	*ClockSettingID.*	

Returns

Pmic_ReturnType.

6.3.4.6 Pmic_VR55XX_SetMode()

```
Pmic_ReturnType Pmic_VR55XX_SetMode (
            const Pmic_DeviceIndexType u32DeviceId,
            const Pmic_ModeIndexType u32ModeSettingID )
```

Pmic_VR55XX_SetMode.

Parameters

in	*u32DeviceId.*	
in	*u32ModeSettingID.*	

Returns

Pmic_ReturnType.

6.3.4.7 Pmic_VR55XX_SetAnalogMux()

```
Pmic_ReturnType Pmic_VR55XX_SetAnalogMux (
            const Pmic_DeviceIndexType u32DeviceId,
            const Pmic_AmuxChannelIndexType u32AmuxChannelID )
```

Pmic_VR55XX_SetAnalogMux.

Parameters

in	*u32DeviceId.*	
in	*u32AmuxChannelID.*	

Returns

Pmic_ReturnType.

6.3.4.8 Pmic_VR55XX_SwitchSVS()

```
Pmic_ReturnType Pmic_VR55XX_SwitchSVS (
            const Pmic_DeviceIndexType u32DeviceId,
            const Pmic_SvsSettingIndexType u32SvsSettingID )
```

Pmic_VR55XX_SwitchSVS.

Parameters

in	*u32DeviceId.*	
in	*u32SvsSettingID.*	

Returns

Pmic_ReturnType.

6.3.4.9 Pmic_VR55XX_ConfigureWatchdog()

```
Pmic_ReturnType Pmic_VR55XX_ConfigureWatchdog (
            const Pmic_DeviceIndexType u32DeviceId,
            const Pmic_WatchdogSettingIndexType u32WatchdogSettingID )
```

Pmic_VR55XX_ConfigureWatchdog.

Parameters

in	*u32DeviceId.*	
in	*u32WatchdogSettingID.*	

Returns

Pmic_ReturnType.

6.3.4.10 Pmic_VR55XX_TriggerWatchdog()

```
Pmic_ReturnType Pmic_VR55XX_TriggerWatchdog (
            const Pmic_DeviceIndexType u32DeviceId )
```

Pmic_VR55XX_TriggerWatchdog.

Parameters

in	*u32DeviceId.*	

Returns

Pmic_ReturnType.

6.3.4.11 Pmic_VR55XX_DisableWatchdog()

```
Pmic_ReturnType Pmic_VR55XX_DisableWatchdog (
            const Pmic_DeviceIndexType u32DeviceId )
```

Pmic_VR55XX_DisableWatchdog.

Parameters

in	*u32DeviceId.*	

Returns

Pmic_ReturnType.

6.3.4.12 Pmic_VR55XX_ReleaseFs0b()

```
Pmic_ReturnType Pmic_VR55XX_ReleaseFs0b (
            const Pmic_DeviceIndexType u32DeviceId )
```

Pmic_VR55XX_ReleaseFs0b.

Parameters

in	*u32DeviceId.*	

Returns

Pmic_ReturnType.

6.3.4.13 Pmic_VR55XX_GetRawFaultEvents()

```
Pmic_ReturnType Pmic_VR55XX_GetRawFaultEvents (
            const Pmic_DeviceIndexType u32DeviceId,
            Pmic_VR55XX_RawFaultType * pRawFaultEventsOut_VR55XX )
```

Pmic_VR55XX_GetRawFaultEvents.

Parameters

in	*u32DeviceId.*	
in	*pRawFaultEventsOut_VR55XX.*	

Returns

Pmic_ReturnType.

6.3.4.14 Pmic_VR55XX_SetReactions()

```
Pmic_ReturnType Pmic_VR55XX_SetReactions (
            const Pmic_DeviceIndexType u32DeviceId,
            const Pmic_ReactionSettingIndexType u32ReactionSettingID )
```

Pmic_VR55XX_SetReactions.

Parameters

in	*u32DeviceId.*	
in	*u32ReactionSettingID.*	

Returns

Pmic_ReturnType.

6.3.4.15 Pmic_VR55XX_GetDeviceInfo()

```
Pmic_ReturnType Pmic_VR55XX_GetDeviceInfo (
            const Pmic_DeviceIndexType u32DeviceId,
            Pmic_VR55XX_DeviceInfoType * pDeviceInfoOut )
```

Pmic_VR55XX_GetDeviceInfo.

Parameters

in	*u32DeviceId.*	
in	*pDeviceInfoOut.*	

Returns

Pmic_ReturnType.

6.3.4.16 Pmic_VR55XX_GotoInitFS()

```
Pmic_ReturnType Pmic_VR55XX_GotoInitFS (
            const Pmic_DeviceIndexType u32DeviceId )
```

Pmic_VR55XX_GotoInitFS.

Parameters

in	*u32DeviceId.*	

Returns

Pmic_ReturnType.

6.4 I2c_external_access

6.4.1 Detailed Description

Function Reference

- Std_ReturnType Pmic_VR55XX_I2cTransferExternal (const Pmic_VR55XX_I2cSetupType *pI2cSetup↩ TypePtr, uint8 *pTxBufferPtr, uint8 *pRxBufferPtr)

  I2c transfer.

6.4.2 Function Reference

6.4.2.1 Pmic_VR55XX_I2cTransferExternal()

```
Std_ReturnType Pmic_VR55XX_I2cTransferExternal (
            const Pmic_VR55XX_I2cSetupType * pI2cSetupTypePtr,
            uint8 * pTxBufferPtr,
            uint8 * pRxBufferPtr )
```

I2c transfer.

This function transfers data via I2C.

Parameters

in	*I2cSetupTypePtr*	pointer setup I2c interface
in	*TxBufferPtr*	I2C Tx buffer
	[in/out]	RxBufferPtr I2C Rx buffer

6.5 Ocotp_external_access

6.5.1 Detailed Description

Function Reference

- Std_ReturnType Pmic_VR55XX_Ocotp_ReadShadowRegister (const uint32 u32OcotpInstance, const uint32 u32OcotpIndex, uint32 *u32Data)

  Ocotp read shadow register.

6.5.2 Function Reference

6.5.2.1 Pmic_VR55XX_Ocotp_ReadShadowRegister()

```
Std_ReturnType Pmic_VR55XX_Ocotp_ReadShadowRegister (
            const uint32 u32OcotpInstance,
            const uint32 u32OcotpIndex,
            uint32 * u32Data )
```

Ocotp read shadow register.

This function transfers data via OCOTP.

Parameters

in	*u32OcotpInstance*	OCOTP instance
in	*u32OcotpIndex*	OCOTP register index
out	*u32Data*	OCOTP raw register data.

6.6 Pin_external_access

6.6.1 Detailed Description

Function Reference

- void Pmic_VR55XX_Port_SetPinMode (const uint8 u8Siul2Instance, const uint32 u32Pin, const uint8 u8↩ Mode)

  Pmic_VR55XX_Port_SetPinMode sets Pin operation mode.
- void Pmic_VR55XX_Dio_FlipChannel (const uint8 u8Siul2Instance, const uint16 u16ChannelId)

  Pmic_VR55XX_Dio_FlipChannel inverts the level of a pin.

6.6.2 Function Reference

6.6.2.1 Pmic_VR55XX_Port_SetPinMode()

```
void Pmic_VR55XX_Port_SetPinMode (
            const uint8 u8Siul2Instance,
            const uint32 u32Pin,
            const uint8 u8Mode )
```

Pmic_VR55XX_Port_SetPinMode sets Pin operation mode.

This function controls the pin mode.

Parameters

in	*u8Siul2Instance*	Instance SIUL (non-AUTOSAR) (not use with AUTOSAR)
in	*u32Pin*	Pin number (non-AUTOSAR) / Pin index setting (AUTOSAR)
in	*u8Mode*	Pin operation mode

6.6.2.2 Pmic_VR55XX_Dio_FlipChannel()

```
void Pmic_VR55XX_Dio_FlipChannel (
            const uint8 u8Siul2Instance,
            const uint16 u16ChannelId )
```

Pmic_VR55XX_Dio_FlipChannel inverts the level of a pin.

This function control the signal of pin

Parameters

in	*u8Siul2Instance*	Instance SIUL (non-AUTOSAR) (not use with AUTOSAR)
in	*u16ChannelId*	channel number (non-AUTOSAR) / channel index setting (AUTOSAR)

How to Reach Us:

Home Page:
nxp.com

Web Support:
nxp.com/support

Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including "typicals," must be validated for each customer application by customer's technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.