

User Manual

for S32K1 GPT Driver

Document Number: UM2GPTASR4.4 Rev0000R1.0.1 Rev. 1.0

1 Revision History	2
2 Introduction	3
2.1 Supported Derivatives	3
2.2 Overview	4
2.3 About This Manual	5
2.4 Acronyms and Definitions	6
2.5 Reference List	6
3 Driver	7
3.1 Requirements	7
3.2 Driver Design Summary	7
3.3 Hardware Resources	8
3.4 Deviations from Requirements	8
3.5 Driver Limitations	10
3.6 Driver usage and configuration tips	10
3.7 Runtime errors	10
3.8 Symbolic Names Disclaimer	11
4 Tresos Configuration Plug-in	12
4.1 Module Gpt	15
4.2 Container GptChannelConfigSet	15
4.3 Container GptChannelConfiguration	16
4.4 Parameter GptChannelId	16
4.5 Parameter GptHwIp	16
4.6 Parameter GptChannelMode	17
4.7 Parameter GptChannelTickFrequency	17
4.8 Parameter GptChannelTickValueMax	18
4.9 Parameter GptEnableWakeup	18
4.10 Parameter GptNotification	19
4.11 Reference GptChannelEcucPartitionRef	19
4.12 Reference GptModuleRef	20
4.13 Reference GptChannelClkSrcRef	20
4.14 Container GptWakeupConfiguration	21
4.15 Reference GptWakeupSourceRef	21
4.16 Container GptFtm	22
4.17 Parameter GptFtmModule	22
4.18 Parameter GptFtmPrescaler	23
4.19 Parameter GptFtmAlternatePrescaler	23
4.20 Parameter GptFtmChannelClkSrc	24
4.21 Parameter FtmFreezeEnable	25

4.22 Parameter GptFtmCountingMode	25
4.23 Container GptFtmChannels	25
4.24 Parameter GptFtmChannel	26
4.25 Parameter FtmAbsoluteCounting	26
4.26 Container GptSRtc	27
4.27 Parameter GptSRtcModule	27
4.28 Parameter GptSRtcChannelClkSrc	28
4.29 Container GptLptmr	28
4.30 Parameter GptLptmrModule	28
4.31 Parameter GptLptmrChannelClkSrc	29
4.32 Parameter GptLptmrPrescaler	30
4.33 Parameter GptLptmrAlternatePrescaler	31
4.34 Parameter PrescalerEnable	32
4.35 Container GptLpit	32
4.36 Parameter GptLpitModule	32
4.37 Parameter LpitFreezeEnable	33
4.38 Parameter LpitDozeEnable	33
4.39 Container GptLpitChannels	34
4.40 Parameter GptLpitChannel	34
4.41 Parameter LPitExternalTrigger	35
4.42 Parameter LPitReloadOnTrigger	35
4.43 Parameter LPitStopOnInterrupt	36
4.44 Parameter LPitStartOnTrigger	36
4.45 Parameter ChainMode	36
4.46 Parameter LPitTriggerChannels	37
4.47 Container GptHwConfiguration	38
4.48 Parameter GptIsrHwId	38
4.49 Parameter GptIsrEnable	39
4.50 Parameter GptChannelsUsed	39
4.51 Container GptConfigurationOfOptApiServices	40
4.52 Parameter GptDeinitApi	40
4.53 Parameter GptEnableDisableNotificationApi	41
4.54 Parameter GptTimeElapsedApi	41
4.55 Parameter GptTimeRemainingApi	42
4.56 Parameter GptVersionInfoApi	42
4.57 Parameter GptWakeupFunctionalityApi	42
4.58 Parameter GptPredefTimerFunctionalityApi	43
4.59 Container GptAutosarExt	43
4.60 Parameter GptEnableDualClockMode	44
4.61 Parameter GptChangeNextTimeoutValueApi	44

4.62	Parameter	GptEnableUserModeSupport	45
4.63	Parameter	ChainModeApi	45
4.64	Parameter	GptStandbyWakeupSupport	45
4.65	Parameter	GptEnableTriggers	46
4.66	Container	GptDriverConfiguration	46
4.67	Parameter	GptDevErrorDetect	47
4.68	Parameter	GptPredefTimer100us32bitEnable	47
4.69	Parameter	GptMulticoreSupport	48
4.70	Parameter	GptPredefTimer1usEnablingGrade	48
4.71	Parameter	GptTimeoutMethod	48
4.72	Parameter	GptTimeoutDuration	49
4.73	Parameter	GptReportWakeupSource	50
4.74	Reference	GptEcucPartitionRef	50
4.75	Reference	GptKernelEcucPartitionRef	51
4.76	Container	GptClockReferencePoint	51
4.77	Reference	GptClockReference	51
4.78	Container	GptPredefTimerConfiguration	52
4.79	Container	GptPredefTimer_1us_16Bit	52
4.80	Parameter	GptHwChannel	53
4.81	Parameter	GptFtmChannelClkSrc	53
4.82	Parameter	GptChannelPrescaler	54
4.83	Parameter	GptFreezeEnable	54
4.84	Reference	GptChannelClkSrcRef	55
4.85	Container	GptPredefTimer_1us_24Bit	55
4.86	Parameter	GptHwChannel	56
4.87	Parameter	GptChannelPrescaler	56
4.88	Parameter	GptFreezeEnable	57
4.89	Reference	GptChannelClkSrcRef	57
4.90	Container	GptPredefTimer_1us_32Bit	58
4.91	Parameter	GptHwChannel	58
4.92	Parameter	GptChannelPrescaler	58
4.93	Parameter	GptFreezeEnable	59
4.94	Reference	GptChannelClkSrcRef	59
4.95	Container	GptPredefTimer_100us_32Bit	60
4.96	Parameter	GptHwChannel	60
4.97	Parameter	GptChannelPrescaler	61
4.98	Parameter	GptFreezeEnable	61
4.99	Reference	GptChannelClkSrcRef	62
4.100	Container	CommonPublishedInformation	62
4.101	Parameter	ArReleaseMajorVersion	62

4.102 Parameter ArReleaseMinorVersion	63
4.103 Parameter ArReleaseRevisionVersion	63
4.104 Parameter ModuleId	64
4.105 Parameter SwMajorVersion	64
4.106 Parameter SwMinorVersion	65
4.107 Parameter SwPatchVersion	65
4.108 Parameter VendorApiInfix	66
4.109 Parameter VendorId	66
5 Module Index	68
5.1 Software Specification	68
6 Module Documentation	69
6.1 FTM IPL	69
6.1.1 Detailed Description	69
6.1.2 Data Structure Documentation	70
6.1.3 Types Reference	71
6.1.4 Enum Reference	71
6.2 Gpt Driver	75
6.2.1 Detailed Description	75
6.2.2 Macro Definition Documentation	76
6.2.3 Types Reference	79
6.2.4 Enum Reference	80
6.2.5 Function Reference	80
6.3 LPit IPL	85
6.3.1 Detailed Description	85
6.3.2 Data Structure Documentation	85
6.3.3 Types Reference	86
6.3.4 Enum Reference	87
6.4 Lptmr IPL	88
6.4.1 Detailed Description	88
6.4.2 Data Structure Documentation	88
6.4.3 Macro Definition Documentation	89
6.4.4 Types Reference	89
6.4.5 Enum Reference	90
6.5 Rtc IPL	92
6.5.1 Detailed Description	92
6.5.2 Data Structure Documentation	92
6.5.3 Types Reference	95
6.5.4 Enum Reference	95



Chapter 1

Revision History

Revision	Date	Author	Description
1.0	24.02.2022	NXP RTD Team	Prepared for release RTD S32K1 Version 1.0.1

Chapter 2

Introduction

- [Supported Derivatives](#)
- [Overview](#)
- [About This Manual](#)
- [Acronyms and Definitions](#)
- [Reference List](#)

This User Manual describes NXP Semiconductor AUTOSAR GPT for S32K1xx. AUTOSAR GPT driver configuration parameters and deviations from the specification are described in GPT Driver chapter of this document. AUTOSAR GPT driver requirements and APIs are described in the AUTOSAR GPT 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:

- s32k116_qfn32
- s32k116_lqfp48
- s32k118_lqfp48
- s32k118_lqfp64
- s32k142_lqfp48
- s32k142_lqfp64
- s32k142_lqfp100
- s32k142w_lqfp48
- s32k142w_lqfp64
- s32k144_lqfp48

- s32k144_lqfp64
- s32k144_lqfp100
- s32k144_mapbga100
- s32k144w_lqfp48
- s32k144w_lqfp64
- s32k146_lqfp64
- s32k146_lqfp100
- s32k146_mapbga100
- s32k146_lqfp144
- s32k148_lqfp100
- s32k148_mapbga100
- s32k148_lqfp144
- s32k148_lqfp176

All of the above microcontroller devices are collectively named as S32K1.

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
GPT	General Purpose Timer
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 Significant Bit
MCU	Micro Controller Unit
MIDE	Multi Integrated Development Environment
MSB	Most Significant Bit
N/A	Not Applicable
RAM	Random Access Memory
SIU	Systems Integration Unit
SWS	Software Specification
VLE	Variable Length Encoding
XML	Extensible Markup Language

2.5 Reference List

#	Title	Version
1	Specification of GPT Driver	AUTOSAR Release 4.↵ 4.0
2	S32K1xx Series Reference Manual	Rev. 14, 09/2021
3	S32K1xx Data Sheet	Rev. 14, 08/2021
4	Errata S32K116_0N96V	Rev. 22/OCT/2021
5	Errata S32K118_0N97V	Rev. 22/OCT/2021
6	Errata S32K142_0N33V	Rev. 22/OCT/2021
7	Errata S32K144_0N57U	Rev. 22/OCT/2021
8	Errata S32K144W_0P64A	Rev. 22/OCT/2021
9	Errata S32K146_0N73V	Rev. 22/OCT/2021
10	Errata S32K148_0N20V	Rev. 22/OCT/2021

Chapter 3

Driver

- [Requirements](#)
- [Driver Design Summary](#)
- [Hardware Resources](#)
- [Deviations from Requirements](#)
- [Driver Limitations](#)
- [Driver usage and configuration tips](#)
- [Runtime errors](#)
- [Symbolic Names Disclaimer](#)

3.1 Requirements

- Requirements for this driver are detailed in the RTD GPT AUTOSAR Release 4.4.0.
- Driver Software Specification document (See S32K1xx Series Reference Manual, Rev. 14, 09/2021).

3.2 Driver Design Summary

The RTD driver assures reentrancy (single core execution) for the APIs based on the following assumptions:

- The "called-again" API is for a different resource (hardware/logic channel);
- Common variables/registers accessed with "rmw" are guarded by Exclusive Areas which need to be correctly implemented in RTE on user side;

The GPT Driver implements the following channels on S32K1xx peripherals.

The table provides information regarding the Timer channels available for the various derivatives across different packages in S32K1xx family. This table lists only the supported packages by GPT driver.

RTC module features :

- 32-bit counter
- SRTC interrupt with interrupt enable.
- Selectable counter clock sources
- Counter runs in all modes of operation.

LPIT timer module features :

- 32-bit counters per module
- Four channel.
- external triggers (triggers from outside the LPIT module)
- or internal triggers (triggers from other timer channels inside the LPIT module).
- Independent timeout periods for each timer
- Independent interrupt source.

LPTMR timer module features :

- One channel 16-bit time counter.
- Configurable clock source for prescaler/glitch filter
- Configurable input source for pulse counter

FTM timer module features :

- One 16-bit up counter with 16-bit
- Prescaler divide-by 1, 2, 4, 8, 16, 32, 64, or 128
- Independent interrupt source for each channel.

3.3 Hardware Resources

#	Hardware IP	Description
1	LPTMR	Low Power Timer
2	LPIT	Low Power Interrupt Timer
3	SRTC	Real Time Clock
4	FTM	FlexTimer

3.4 Deviations from Requirements

The driver deviates from the AUTOSAR GPT Driver software specification in some places.

There are also some additional requirements (on top of requirements detailed in AUTOSAR GPT Driver software specification) which need to be satisfied for correct operation.

Deviations Status Column Description

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

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

Requirement	Status	Description	Notes
SWS_Gpt_00261	N/S	Gpt_Irq.c shall include Gpt.h for the prototype declaration of the notification functions.	Rejection reason: Gpt_Irq.c is not needed. Autosar specific interrupt behaviour is implemented using a normal function placed in the Gpt.c file.
SWS_Gpt_00278	N/S	Module - Header File - Imported Type - EcuM_flex - EcuM.h - EcuM_WakeupSourceType - Std_Types - StandardTypes.h - Std_ReturnType - StandardTypes.h - Std_VersionInfoType -	Rejection reason: No production errors needed for current development.
SWS_Gpt_00381	N/S	These requirements are not applicable to this specification.	Not a requirement
ECUC_Gpt_00235	N/S	Container Name - GptWakeupConfiguration - Description - Function pointer to callback function (for wakeup notification). - Configuration Parameters -	Rejection reason: Wrong Description: Function pointer to callback function (for non-wakeup notification). It shall relate to wakeup configuration.
SWS_Gpt_CONSTR_00001	N/S	DRAFT: The ECUC partitions referenced by GptKernelEcucPartitionRef shall be a subset of the ECUC partitions referenced by GptEcucPartitionRef.()	Type IV Autosar multicore not implemented for current module. AAI-445; Agree that each module can reject the Autosar Standard requirement

Requirement	Status	Description	Notes
ECUC_Gpt_00338	N/S	<p>Name - GptKernelEcuc↔ PartitionRef - Parent Container - GptDriverConfiguration - Description - Maps the GPT kernel to zero or one ECUC partitions to assign the driver kernel to a certain core. The ECUC partition referenced is a subset of the ECUC partitions where the GPT driver is mapped to. Note: The kernel reference shall not be set in case the GPT driver is implemented without a kernel (refer to definition of GptEcucPartitionRef).Tags: atp.Status=draft - Multiplicity - 0..1 - Type - Reference to [EcucPartition] - Post-Build Variant Multiplicity - true - Post-Build Variant Value - true - Multiplicity Configuration Class - Pre-compile time - X - All Variants - Link time - - - - Post-build time - - - - Value Configuration Class - Pre-compile time - X - All Variants - Link time - - - - Post-build time - - - - Scope / Dependency - scope: ECU -</p>	Type IV Autosar multicore not implemented for current module. AAI-445; Agree that each module can reject the Autosar Standard requirement.

3.5 Driver Limitations

The GPT driver software have some following limitations for RTD S32K1xx :

- Multicore support is not available.
- Predefined Timer functionality is supported only with FTM Ip and because the FTM time reference is a 16-bit counter only GptPredefTimer_1us_16Bit is available.

3.6 Driver usage and configuration tips

In this chapter, the extra features from our drivers that are not described in the AutoSAR standard are detailed.

3.7 Runtime errors

The driver generates the following DEM errors at runtime.

Function	Error Code	Condition triggering the error
Gpt_ValidateChannelStatus()	GPT_E_BUSY	API service called when timer channel is still busy (running)
Gpt_ValidateMode()	GPT_E_MODE	API service called when driver is in wrong mode

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 [Gpt](#)
 - Container [GptChannelConfigSet](#)
 - * Container [GptChannelConfiguration](#)
 - Parameter [GptChannelId](#)
 - Parameter [GptHwIp](#)
 - Parameter [GptChannelMode](#)
 - Parameter [GptChannelTickFrequency](#)
 - Parameter [GptChannelTickValueMax](#)
 - Parameter [GptEnableWakeup](#)
 - Parameter [GptNotification](#)
 - Reference [GptChannelEcucPartitionRef](#)
 - Reference [GptModuleRef](#)
 - Reference [GptChannelClkSrcRef](#)
 - Container [GptWakeupConfiguration](#)
 - Reference [GptWakeupSourceRef](#)
 - * Container [GptFtm](#)
 - Parameter [GptFtmModule](#)
 - Parameter [GptFtmPrescaler](#)
 - Parameter [GptFtmAlternatePrescaler](#)
 - Parameter [GptFtmChannelClkSrc](#)
 - Parameter [FtmFreezeEnable](#)
 - Parameter [GptFtmCountingMode](#)
 - Container [GptFtmChannels](#)
 - Parameter [GptFtmChannel](#)
 - Parameter [FtmAbsoluteCounting](#)
 - * Container [GptSRtc](#)
 - Parameter [GptSRtcModule](#)
 - Parameter [GptSRtcChannelClkSrc](#)
 - * Container [GptLptmr](#)
 - Parameter [GptLptmrModule](#)

- Parameter [GptLptmrChannelClkSrc](#)
- Parameter [GptLptmrPrescaler](#)
- Parameter [GptLptmrAlternatePrescaler](#)
- Parameter [PrescalerEnable](#)
- * Container [GptLpit](#)
 - Parameter [GptLpitModule](#)
 - Parameter [LpitFreezeEnable](#)
 - Parameter [LpitDozeEnable](#)
 - Container [GptLpitChannels](#)
 - Parameter [GptLpitChannel](#)
 - Parameter [LPitExternalTrigger](#)
 - Parameter [LPitReloadOnTrigger](#)
 - Parameter [LPitStopOnInterrupt](#)
 - Parameter [LPitStartOnTrigger](#)
 - Parameter [ChainMode](#)
 - Parameter [LPitTriggerChannels](#)
- Container [GptHwConfiguration](#)
 - * Parameter [GptIsrHwId](#)
 - * Parameter [GptIsrEnable](#)
 - * Parameter [GptChannelsUsed](#)
- Container [GptConfigurationOfOptApiServices](#)
 - * Parameter [GptDeinitApi](#)
 - * Parameter [GptEnableDisableNotificationApi](#)
 - * Parameter [GptTimeElapsedApi](#)
 - * Parameter [GptTimeRemainingApi](#)
 - * Parameter [GptVersionInfoApi](#)
 - * Parameter [GptWakeupFunctionalityApi](#)
 - * Parameter [GptPredefTimerFunctionalityApi](#)
- Container [GptAutosarExt](#)
 - * Parameter [GptEnableDualClockMode](#)
 - * Parameter [GptChangeNextTimeoutValueApi](#)
 - * Parameter [GptEnableUserModeSupport](#)
 - * Parameter [ChainModeApi](#)
 - * Parameter [GptStandbyWakeupSupport](#)
 - * Parameter [GptEnableTriggers](#)
- Container [GptDriverConfiguration](#)
 - * Parameter [GptDevErrorDetect](#)
 - * Parameter [GptPredefTimer100us32bitEnable](#)
 - * Parameter [GptMulticoreSupport](#)
 - * Parameter [GptPredefTimer1usEnablingGrade](#)

- * Parameter [GptTimeoutMethod](#)
- * Parameter [GptTimeoutDuration](#)
- * Parameter [GptReportWakeupSource](#)
- * Reference [GptEcucPartitionRef](#)
- * Reference [GptKernelEcucPartitionRef](#)
- * Container [GptClockReferencePoint](#)
 - Reference [GptClockReference](#)
- Container [GptPredefTimerConfiguration](#)
 - * Container [GptPredefTimer_1us_16Bit](#)
 - Parameter [GptHwChannel](#)
 - Parameter [GptFtmChannelClkSrc](#)
 - Parameter [GptChannelPrescaler](#)
 - Parameter [GptFreezeEnable](#)
 - Reference [GptChannelClkSrcRef](#)
 - * Container [GptPredefTimer_1us_24Bit](#)
 - Parameter [GptHwChannel](#)
 - Parameter [GptChannelPrescaler](#)
 - Parameter [GptFreezeEnable](#)
 - Reference [GptChannelClkSrcRef](#)
 - * Container [GptPredefTimer_1us_32Bit](#)
 - Parameter [GptHwChannel](#)
 - Parameter [GptChannelPrescaler](#)
 - Parameter [GptFreezeEnable](#)
 - Reference [GptChannelClkSrcRef](#)
 - * Container [GptPredefTimer_100us_32Bit](#)
 - Parameter [GptHwChannel](#)
 - Parameter [GptChannelPrescaler](#)
 - Parameter [GptFreezeEnable](#)
 - Reference [GptChannelClkSrcRef](#)
- Container [CommonPublishedInformation](#)
 - * Parameter [ArReleaseMajorVersion](#)
 - * Parameter [ArReleaseMinorVersion](#)
 - * Parameter [ArReleaseRevisionVersion](#)
 - * Parameter [ModuleId](#)
 - * Parameter [SwMajorVersion](#)
 - * Parameter [SwMinorVersion](#)
 - * Parameter [SwPatchVersion](#)
 - * Parameter [VendorApiInfix](#)
 - * Parameter [VendorId](#)

4.1 Module Gpt

Configuration of the Gpt (General Purpose Timer) module.

Included containers:

- [GptChannelConfigSet](#)
- [GptHwConfiguration](#)
- [GptConfigurationOfOptApiServices](#)
- [GptAutosarExt](#)
- [GptDriverConfiguration](#)
- [GptPredefTimerConfiguration](#)
- [CommonPublishedInformation](#)

Property	Value
type	ECUC-MODULE-DEF
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantSupport	true
supportedConfigVariants	VARIANT-POST-BUILD, VARIANT-PRE-COMPILE

4.2 Container GptChannelConfigSet

This container is the base of an Configuration Set which contains the configured GPT channels.

This way, different configuration sets can be defined for post-build process.

Included subcontainers:

- [GptChannelConfiguration](#)
- [GptFtm](#)
- [GptSRtc](#)
- [GptLptmr](#)
- [GptLpit](#)

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

4.3 Container GptChannelConfiguration

This container contains the channel-wide configuration (parameters) of the GPT Driver

Included subcontainers:

- [GptWakeupConfiguration](#)

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

4.4 Parameter GptChannelId

Channel Id of the GPT channel. This value will be assigned to the symbolic name derived of the GptChannelConfiguration container short name.

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

4.5 Parameter GptHwIp

Vendor specific: Selects the physical GPT Channel.

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
default Value	FTM
literals	['FTM', 'LPIT', 'LPTMR', 'SRTC']

4.6 Parameter GptChannelMode

Specifies the behaviour of the timerchannel after the timeout has expired

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

4.7 Parameter GptChannelTickFrequency

EN: Specifies the tick frequency of the timer channel in Hz.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	AUTOSAR_ECUC
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	0.0
max	1.6E8
min	0.0

4.8 Parameter GptChannelTickValueMax

Maximum value in ticks, the timer channel is able to count. With the next tick, the timer rolls over to zero.

It is mandatory to set 65535 for the FTM channels (corresponding to the 16 bits counter resolution)!

It is mandatory to set 65535 for the LPtimer channels (corresponding to the 16 bits counter resolution)!

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	65535
max	4294967295
min	65535

4.9 Parameter GptEnableWakeup

Enables wakeup capability of CPU for a channel.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF

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

4.10 Parameter GptNotification

Function pointer to callback function(for non-wakeup notification).

The field is editable only if the switch GptEnableDisableNotificationApi is true.

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

4.11 Reference GptChannelEcucPartitionRef

Maps a GPT channel to zero or one ECUC partition to limit the access to this channel group. The ECUC partitions referenced are a subset of the ECUC partitions where the GPT driver is mapped to.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0

Property	Value
upperMultiplicity	Infinite
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcuDefs/EcuC/EcuPartitionCollection/EcuPartition

4.12 Reference GptModuleRef

Maps a GPT channel to zero or one ECUC partition to limit the access to this channel group. The ECUC partitions referenced are a subset of the ECUC partitions where the GPT driver is mapped to.

Property	Value
type	ECUC-CHOICE-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
destinations	['/TS_T40D2M10I1R0/Gpt/GptChannelConfigSet/GptFtm/GptFtmChannels', '/TS_T40D2M10I1R0/Gpt/GptChannelConfigSet/GptSRtc', '/TS_T40D2M10I1R0/Gpt/GptChannelConfigSet/GptLptmr', '/TS_T40D2M10I1R0/Gpt/GptChannelConfigSet/GptLpit/GptLpitChannels']

4.13 Reference GptChannelClkSrcRef

Reference to the GptClockReferencePoint from which the channel clock is derived.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	1
upperMultiplicity	1

Property	Value
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcuDefs/Gpt/GptDriverConfiguration/GptClockReferencePoint

4.14 Container GptWakeupConfiguration

This container defines the wakeup source codes reported to Ecu State Manager.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.15 Reference GptWakeupSourceRef

In case the wakeup-capability is true this value is transmitted to the Ecu State Manager.

Implementation Type: reference to EcuM_WakeupSourceType

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE

Property	Value
	VARIANT-POST-BUILD: POST-BUILD
requiresSymbolicNameValue	true
destination	/AUTOSAR/EcuDefs/EcuM/EcuMConfiguration/EcuMCommon↵ Configuration/EcuMWakeupSource

4.16 Container GptFtm

Configuration of a Ftm module available on the platform.

Included subcontainers:

- [GptFtmChannels](#)

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	8
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.17 Parameter GptFtmModule

Select the physical Ftm Module.

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	FTM_0
literals	['FTM_0', 'FTM_1', 'FTM_2', 'FTM_3', 'FTM_4', 'FTM_5', 'FTM_6', 'F↵ TM_7']

4.18 Parameter GptFtmPrescaler

Vendor specific: The GPT module specific clock prescaler value.

Note with FTM:

- FTM prescaler should be 1, 2, 4, 8, 16, 32, 64 or 128.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1
max	128
min	1

4.19 Parameter GptFtmAlternatePrescaler

Vendor specific: The GPT module specific clock prescaler value.

Selects one of 8 division factors for the clock source selected by GptFtmChannelClkSrcRef. The new prescaler factor affects

the clock source on the next system clock cycle after the new value is updated into the register bits.

- 1 - Value written in register: 0
- 2 - Value written in register: 1
- 4 - Value written in register: 2
- 8 - Value written in register: 3
- 16 - Value written in register: 4
- 32 - Value written in register: 5
- 64 - Value written in register: 6
- 128 - Value written in register: 7

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1
max	128
min	1

4.20 Parameter GptFtmChannelClkSrc

Vendor specific: The GPT module specific clock input for the timer unit can statically be configured and allows to select different clock sources per module.

Select the clock source for the FlexTimer module for this platform.

FTM_GPT_IP_CLOCK_SOURCE_NONENone use clock for FTM?

FTM_GPT_IP_CLOCK_SOURCE_SYSTEMCLKSystem clock

FTM_GPT_IP_CLOCK_SOURCE_FIXED_FREQUENCYFixed frequency clock

FTM_GPT_IP_CLOCK_SOURCE_EXTERNALCLKExternal clock

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	FTM_GPT_IP_CLOCK_SOURCE_NONE
literals	['FTM_GPT_IP_CLOCK_SOURCE_NONE', 'FTM_GPT_IP_CLOCK_SOURCE_SYSTEMCLK', 'FTM_GPT_IP_CLOCK_SOURCE_FIXED_FREQUENCY', 'FTM_GPT_IP_CLOCK_SOURCE_EXTERNALCLK']

4.21 Parameter FtmFreezeEnable

Enables/Disables freeze bit.

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

4.22 Parameter GptFtmCountingMode

This parameter indicates the count direction for the whole GPT driver.

Note This feature is not supported in current implementation. Up-counting is always used as default

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	FTM_GPT_IP_MODE_UP_TIMER
literals	['FTM_GPT_IP_MODE_DOWN_TIMER', 'FTM_GPT_IP_MODE_UP_TIMER']

4.23 Container GptFtmChannels

Ftm hw channels

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	8
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.24 Parameter GptFtmChannel

Selects one of the Ftm channels available on the platform.

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	CH_0
literals	['CH_0', 'CH_1', 'CH_2', 'CH_3', 'CH_4', 'CH_5', 'CH_6', 'CH_7']

4.25 Parameter FtmAbsoluteCounting

Enables/Disables absolute compare value.

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

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

4.26 Container GptSRtc

Configuration of a sRtc module available on the platform.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	2
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.27 Parameter GptSRtcModule

Select the physical sRtc Module.

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	SRTC_0_CH_0
literals	['SRTC_0_CH_0']

4.28 Parameter GptSRtcChannelClkSrc

Selectable counter clock sources

? CLK_SRC_OSC_32KHZ

? CLK_SRC_LPO_1KHZ

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	SRTC_IP_CLK_SRC_OSC_32KHZ
literals	['SRTC_IP_CLK_SRC_LPO_1KHZ', 'SRTC_IP_CLK_SRC_OSC_32KHZ']

4.29 Container GptLptmr

Configuration of a LPTimer module available on the platform.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	2
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.30 Parameter GptLptmrModule

Select the physical LPTimer Module.

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	LPTMR_0_CH_0
literals	['LPTMR_0_CH_0']

4.31 Parameter GptLptmrChannelClkSrc

Select the clock source for the Low Power Timer module for this platform.

? SIRCDIV2_CLK

? LPO1K

? RTC_CLK

? PCC_LPTMR0

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	LPTMR_GPT_IP_CLOCK_SIRCDIV2
literals	['LPTMR_GPT_IP_CLOCK_SIRCDIV2', 'LPTMR_GPT_IP_CLOCK_LPO1K', 'LPTMR_GPT_IP_CLOCK_RTC_CLK', 'LPTMR_GPT_IP_CLOCK_PCC_LPTMR0']

4.32 Parameter GptLptmrPrescaler

Vendor specific: The GPT module specific clock prescaler value.

Configures the Prescaler in Time Counter mode, from one of 16 divisions.

2 - Value written in register is 0

4 - Value written in register is 1

8 - Value written in register is 2

16 - Value written in register is 3

32 - Value written in register is 4

64 - Value written in register is 5

128 - Value written in register is 6

256 - Value written in register is 7

512 - Value written in register is 8

1024 - Value written in register is 9

2048 - Value written in register is 10

4096 - Value written in register is 11

8192 - Value written in register is 12

16384 - Value written in register is 13

32768 - Value written in register is 14

65536 - Value written in register is 15

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	2
max	65536
min	1

4.33 Parameter GptLptmrAlternatePrescaler

Vendor specific: The GPT module specific clock prescaler value.

Configures the Prescaler in Time Counter mode, from one of 16 divisions.

- 2 - Value written in register is 0
- 4 - Value written in register is 1
- 8 - Value written in register is 2
- 16 - Value written in register is 3
- 32 - Value written in register is 4
- 64 - Value written in register is 5
- 128 - Value written in register is 6
- 256 - Value written in register is 7
- 512 - Value written in register is 8
- 1024 - Value written in register is 9
- 2048 - Value written in register is 10
- 4096 - Value written in register is 11
- 8192 - Value written in register is 12
- 16384 - Value written in register is 13
- 32768 - Value written in register is 14
- 65536 - Value written in register is 15

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE VARIANT-POST-BUILD: POST-BUILD
defaultValue	2
max	65536
min	1

S32K1 GPT Driver

4.34 Parameter PrescalerEnable

When PBYP is set, the selected prescaler clock in Time Counter mode or selected input source in Pulse Counter mode directly clocks the CNR.

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

4.35 Container GptLpit

Configuration of a Lpit module available on the platform.

Included subcontainers:

- [GptLpitChannels](#)

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	2
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.36 Parameter GptLpitModule

Select the physical Lpit Module.

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
default Value	LPIT_0
literals	['LPIT_0']

4.37 Parameter LpitFreezeEnable

Enables/Disables freeze bit.

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
default Value	false

4.38 Parameter LpitDozeEnable

Enables/Disables DOZE Mode.

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

4.39 Container GptLpitChannels

Lpit hw channels.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	4
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.40 Parameter GptLpitChannel

Selects one of the Lpit channels available on the platform.

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

Property	Value
defaultValue	CH_0
literals	['CH_0', 'CH_1', 'CH_2', 'CH_3']

4.41 Parameter LPitExternalTrigger

Select between external/internal trigger sources.

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

4.42 Parameter LPitReloadOnTrigger

LPIT Timer channel will Reload Timer at each rising edge of trigger source.

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

4.43 Parameter LPitStopOnInterrupt

LPIT Timer Channel will stop when its interrupt occurs.

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

4.44 Parameter LPitStartOnTrigger

LPIT Timer Channel will Start at each rising edge of Trigger Signal.

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

4.45 Parameter ChainMode

Enables/Disables chain mode

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF

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

4.46 Parameter LPitTriggerChannels

Vendor specific: LPIT External Trigger Channels Select.

Trigger Channels can be selected from 16 external sources.

0 Timer channel 0 trigger source is selectd.

1 Timer channel 1 trigger source is selectd.

..

14 Timer channel 14 trigger source is selectd.

15 Timer channel 15 trigger source is selectd.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE VARIANT-POST-BUILD: POST-BUILD
defaultValue	Channel_0_Trigger_Source
literals	['Channel_0_Trigger_Source', 'Channel_1_Trigger_Source', 'Channel_2_Trigger_Source', 'Channel_3_Trigger_Source', 'Channel_4_Trigger_Source', 'Channel_5_Trigger_Source', 'Channel_6_Trigger_Source', 'Channel_7_Trigger_Source', 'Channel_8_Trigger_Source', 'Channel_9_Trigger_Source', 'Channel_10_Trigger_Source', 'Channel_11_Trigger_Source', 'Channel_12_Trigger_Source', 'Channel_13_Trigger_Source', 'Channel_14_Trigger_Source', 'Channel_15_Trigger_Source']

4.47 Container GptHwConfiguration

List of all HW channel resources for GPT module.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	78
upperMultiplicity	78
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.48 Parameter GptIsrHwId

ID of HW interrupt resources.

Property	Value
type	ECUC-ENUMERATION-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	LPIT_0_CH_1

Property	Value
literals	['LPIT_0_CH_0', 'LPIT_0_CH_1', 'LPIT_0_CH_2', 'LPIT_0_CH_3', 'FTM_0_CH_0', 'FTM_0_CH_1', 'FTM_0_CH_2', 'FTM_0_CH_3', 'FTM_0_CH_4', 'FTM_0_CH_5', 'FTM_0_CH_6', 'FTM_0_CH_7', 'FTM_1_CH_0', 'FTM_1_CH_1', 'FTM_1_CH_2', 'FTM_1_CH_3', 'FTM_1_CH_4', 'FTM_1_CH_5', 'FTM_1_CH_6', 'FTM_1_CH_7', 'FTM_2_CH_0', 'FTM_2_CH_1', 'FTM_2_CH_2', 'FTM_2_CH_3', 'FTM_2_CH_4', 'FTM_2_CH_5', 'FTM_2_CH_6', 'FTM_2_CH_7', 'FTM_3_CH_0', 'FTM_3_CH_1', 'FTM_3_CH_2', 'FTM_3_CH_3', 'FTM_3_CH_4', 'FTM_3_CH_5', 'FTM_3_CH_6', 'FTM_3_CH_7', 'FTM_4_CH_0', 'FTM_4_CH_1', 'FTM_4_CH_2', 'FTM_4_CH_3', 'FTM_4_CH_4', 'FTM_4_CH_5', 'FTM_4_CH_6', 'FTM_4_CH_7', 'FTM_5_CH_0', 'FTM_5_CH_1', 'FTM_5_CH_2', 'FTM_5_CH_3', 'FTM_5_CH_4', 'FTM_5_CH_5', 'FTM_5_CH_6', 'FTM_5_CH_7', 'FTM_6_CH_0', 'FTM_6_CH_1', 'FTM_6_CH_2', 'FTM_6_CH_3', 'FTM_6_CH_4', 'FTM_6_CH_5', 'FTM_6_CH_6', 'FTM_6_CH_7', 'FTM_7_CH_0', 'FTM_7_CH_1', 'FTM_7_CH_2', 'FTM_7_CH_3', 'FTM_7_CH_4', 'FTM_7_CH_5', 'FTM_7_CH_6', 'FTM_7_CH_7', 'LPTMR_0_CH_0', 'SRTC_0_CH_0', 'FTM_0_PREDEF', 'FTM_1_PREDEF', 'FTM_2_PREDEF', 'FTM_3_PREDEF', 'FTM_4_PREDEF', 'FTM_5_PREDEF', 'FTM_6_PREDEF', 'FTM_7_PREDEF']

4.49 Parameter GptIsrEnable

Enable/Disable HW channels' Interrupt Sources.

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

4.50 Parameter GptChannelsUsed

This column configures HW channels which are going to be used.

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
default Value	false

4.51 Container GptConfigurationOfOptApiServices

This container contains all configuration switches for configuring optional API services of the GPT driver.

Included subcontainers:

- None

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

4.52 Parameter GptDeinitApi

Adds / removes the service Gpt_DeInit() from the code.

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

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

4.53 Parameter GptEnableDisableNotificationApi

Adds / removes the services Gpt_EnableNotification() and Gpt_DisableNotification from the code.

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

4.54 Parameter GptTimeElapsedApi

Adds / removes the service Gpt_GetTimeElapsed() from the code.

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

4.55 Parameter GptTimeRemainingApi

Adds / removes the service Gpt_GetTimeRemaining() from the code.

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

4.56 Parameter GptVersionInfoApi

Adds / removes the service Gpt_GetVersionInfo() from the code.

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

4.57 Parameter GptWakeupFunctionalityApi

Adds / removes the services Gpt_SetMode(), Gpt_EnableWakeup() Gpt_DisableWakeup() and Gpt_Cbk_CheckWakeup() from the code.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF

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

4.58 Parameter GptPredefTimerFunctionalityApi

Adds / removes the services Gpt_SetMode(), Gpt_EnableWakeup() Gpt_DisableWakeup() and Gpt_Cbk_CheckWakeup() from the code.

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

4.59 Container GptAutosarExt

Enabling the settings of this section will configure the driver in a mode not compliant with AUTOSAR requirements.

Included subcontainers:

- None

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

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

4.60 Parameter GptEnableDualClockMode

Enables prescaler settings at mode transition.true: Enabled.false: Disabled.

Note This feature is not required by Autosar.

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

4.61 Parameter GptChangeNextTimeoutValueApi

Vendor specific: Enables settings for changing the channel counter compare value of a running counter.

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

4.62 Parameter GptEnableUserModeSupport

When this parameter is enabled, the GPT module will adapt to run from User Mode. There is no difference between User mode and Privileged mode in GPT module.

Note: Implementation 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

4.63 Parameter ChainModeApi

Vendor specific: Enable/disable API for Chain Mode support.

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

4.64 Parameter GptStandbyWakeupSupport

The driver shall NOT CLEAR the interrupt flag, the interrupt enable bit and also should not disable the counter, during init (Gpt_SRtc_Init()) the flag is already set.

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
default Value	false

4.65 Parameter GptEnableTriggers

Enable Trigger Mode.true: Enabled.false: Disabled.

Note This feature is not required by Autosar.

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
default Value	false

4.66 Container GptDriverConfiguration

This container contains the module-wide configuration (parameters) of the GPT Driver.

Included subcontainers:

- [GptClockReferencePoint](#)

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

4.67 Parameter GptDevErrorDetect

Enables/Disables development error detection.

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

4.68 Parameter GptPredefTimer100us32bitEnable

Enables/Disables the feature 100us/ tick

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

4.69 Parameter GptMulticoreSupport

Enables/Disables Multicore Support.

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

4.70 Parameter GptPredefTimer1usEnablingGrade

Specifies the grade of enabling the GPT Predef Timers with 1us tick duration.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	GPT_PREDEF_TIMER_1US_16BIT_ENABLED
literals	['GPT_PREDEF_TIMER_1US_16BIT_ENABLED', 'GPT_PREDEF_TIMER_1US_16_24BIT_ENABLED', 'GPT_PREDEF_TIMER_1US_16_24_32BIT_ENABLED', 'GPT_PREDEF_TIMER_1US_DISABLED']

4.71 Parameter GptTimeoutMethod

GptTimeoutMethod: Configures the timeout method.

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

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

Note: Implementation Specific Parameter.

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

4.72 Parameter GptTimeoutDuration

The unit of measurement is given in number of microseconds. This is a timeout value which is used to wait till

- PIT_RTI_LDVAL is synchronized into the RTI clock domain

If the Status is not updated then after this timeout a runtime error will be reported.

This parameter is used for PitRti only

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	800
max	65535
min	1

4.73 Parameter GptReportWakeupSource

Enables/Disables wakeup source reporting.

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

4.74 Reference GptEcucPartitionRef

Maps the GPT driver to zero or multiple ECUC partitions to make the driver API available in the according partition. Depending on the addressed timer resource the interfaces operate as follows:

In case of partition local timer resources (n:1 mapping) the API operates as an independent instance in the according ECUC partition.

In case of global timer resources (1:m mapping) the API operates on the global timer resource either by protected access to the resource or by implementing an according kernel.

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

4.75 Reference GptKernelEcucPartitionRef

Maps the GPT kernel to zero or one ECUC partitions to assign the driver kernel to a certain core. The ECUC partition referenced is a subset of the ECUC partitions where the GPT driver is mapped to.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/EcuC/EcucPartitionCollection/EcucPartition

4.76 Container GptClockReferencePoint

This container contains a parameter, which represents a reference to a container of the type McuClockReferencePoint (defined in module MCU).

Included subcontainers:

- None

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

4.77 Reference GptClockReference

Reference to a container of the type McuClockReferencePoint, to select an input clock.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcuDefs/Mcu/McuModuleConfiguration/McuClockSetting↔ Config/McuClockReferencePoint

4.78 Container GptPredefTimerConfiguration

Container for configuring the Predefined Timer functionality.

Included subcontainers:

- [GptPredefTimer_1us_16Bit](#)
- [GptPredefTimer_1us_24Bit](#)
- [GptPredefTimer_1us_32Bit](#)
- [GptPredefTimer_100us_32Bit](#)

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

4.79 Container GptPredefTimer_1us_16Bit

This container contains the 1U_16BIT predef timer configuration (parameters) of the GPT Driver

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.80 Parameter GptHwChannel

Vendor specific: Selects the physical GPT Channel.

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	FTM_0_PREDEF
literals	['FTM_0_PREDEF', 'FTM_1_PREDEF', 'FTM_2_PREDEF', 'FTM_3_P← REDEF', 'FTM_4_PREDEF', 'FTM_5_PREDEF', 'FTM_6_PREDEF', 'F← TM_7_PREDEF']

4.81 Parameter GptFtmChannelClkSrc

Vendor specific: The GPT module specific clock input for the timer unit can statically be configured and allows to select different clock sources per module.

Select the clock source for the FlexTimer module for this platform.

FTM_GPT_IP_CLOCK_SOURCE_NONENone use clock for FTM?

FTM_GPT_IP_CLOCK_SOURCE_SYSTEMCLKSystem clock

FTM_GPT_IP_CLOCK_SOURCE_FIXED_FREQUENCYFixed frequency clock

FTM_GPT_IP_CLOCK_SOURCE_EXTERNALCLKEExternal clock

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	FTM_GPT_IP_CLOCK_SOURCE_NONE
literals	['FTM_GPT_IP_CLOCK_SOURCE_NONE', 'FTM_GPT_IP_CLOCK_SOURCE_SYSTEMCLK', 'FTM_GPT_IP_CLOCK_SOURCE_FIXED_FREQUENCY', 'FTM_GPT_IP_CLOCK_SOURCE_EXTERNALCLK']

4.82 Parameter GptChannelPrescaler

Vendor specific: The GPT module specific clock prescaler value.

Note with FTM:

- FTM prescaler should be 1, 2, 4, 8, 16, 32, 64 or 128.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1
max	256
min	1

4.83 Parameter GptFreezeEnable

Vendor specific: Select to set Freeze enable for the hw resources.

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
default Value	true

4.84 Reference GptChannelClkSrcRef

Reference to the GptClockReferencePoint from which the channel clock is derived.

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/Gpt/GptDriverConfiguration/GptClockReferencePoint

4.85 Container GptPredefTimer__1us__24Bit

This container contains the 1U_24BIT predef timer configuration (parameters) of the GPT Driver

Included subcontainers:

- None

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

Property	Value
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.86 Parameter GptHwChannel

Vendor specific: Selects the physical GPT Channel.

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	NOT_APPLICABLE
literals	['NOT_APPLICABLE']

4.87 Parameter GptChannelPrescaler

Vendor specific: The GPT module specific clock prescaler value.

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

Property	Value
defaultValue	1
max	256
min	1

4.88 Parameter GptFreezeEnable

Vendor specific: Select to set Freeze enable for the hw resources.

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

4.89 Reference GptChannelClkSrcRef

Reference to the GptClockReferencePoint from which the channel clock is derived.

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/Gpt/GptDriverConfiguration/GptClockReferencePoint

4.90 Container GptPredefTimer_1us_32Bit

This container contains the predef timer configuration (parameters) of the GPT Driver

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.91 Parameter GptHwChannel

Vendor specific: Selects the physical GPT Channel.

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	NOT_APPLICABLE
literals	['NOT_APPLICABLE']

4.92 Parameter GptChannelPrescaler

Vendor specific: The GPT module specific clock prescaler value.

Property	Value
type	ECUC-INTEGER-PARAM-DEF

Property	Value
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1
max	256
min	1

4.93 Parameter GptFreezeEnable

Vendor specific: Select to set Freeze enable for the hw resources.

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

4.94 Reference GptChannelClkSrcRef

Reference to the GptClockReferencePoint from which the channel clock is derived.

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP
lowerMultiplicity	1
upperMultiplicity	1

Property	Value
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcuDefs/Gpt/GptDriverConfiguration/GptClockReferencePoint

4.95 Container GptPredefTimer_100us_32Bit

This container contains the channel-wide configuration (parameters) of the GPT Driver

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.96 Parameter GptHwChannel

Vendor specific: Selects the physical GPT Channel.

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

Property	Value
default Value	NOT_APPLICABLE
literals	['NOT_APPLICABLE']

4.97 Parameter GptChannelPrescaler

Vendor specific: The GPT module specific clock prescaler value.

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
default Value	1
max	256
min	1

4.98 Parameter GptFreezeEnable

Vendor specific: Select to set Freeze enable for the hw resources.

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
default Value	true

4.99 Reference GptChannelClkSrcRef

Reference to the GptClockReferencePoint from which the channel clock is derived.

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/EcuDefs/Gpt/GptDriverConfiguration/GptClockReferencePoint

4.100 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

4.101 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

Property	Value
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

4.102 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

4.103 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

Property	Value
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

4.104 Parameter ModuleId

Module ID of this module from Module List.

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	100
max	100
min	100

4.105 Parameter SwMajorVersion

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

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	1
max	1
min	1

4.106 Parameter SwMinorVersion

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

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

4.107 Parameter SwPatchVersion

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

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

Property	Value
postBuildVariantValue	false
valueConfigClasses	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	1
max	1
min	1

4.108 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.

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

4.109 Parameter VendorId

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

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



Chapter 5

Module Index

5.1 Software Specification

Here is a list of all modules:

FTM IPL	69
Gpt Driver	75
LPit IPL	85
Lptmr IPL	88
Rtc IPL	92

Chapter 6

Module Documentation

6.1 FTM IPL

6.1.1 Detailed Description

Data Structures

- struct [Ftm_Gpt_Ip_InstanceConfigType](#)
Structure to configure the FTM instance. [More...](#)
- struct [Ftm_Gpt_Ip_ChannelConfigType](#)
Structure to configure the FTM channels. [More...](#)
- struct [Ftm_Gpt_Ip_ChState](#)
internal context structure [More...](#)
- struct [Ftm_Gpt_Ip_InstanceState](#)
internal context structure [More...](#)

Types Reference

- typedef void(* [Ftm_Gpt_Ip_CallbackType](#)) (uint8 callbackParam)
Callback type for each channel.

Enum Reference

- enum [Ftm_Gpt_Ip_CountingMode](#)
Unit options for counting mode.
- enum [Ftm_Gpt_Ip_ClockSource](#)
Enum containing the FTM module clock sources.
- enum [Ftm_Gpt_Ip_ChannelModeType](#)
Prescaler type. Indicates of whether the clock channel mode is "NORMAL" or "ALTERNATE".

6.1.2 Data Structure Documentation

6.1.2.1 struct Ftm_Gpt_Ip_InstanceConfigType

Structure to configure the FTM instance.

This structure holds the configuration settings for InstanceConfigType

Implements [Ftm_Gpt_Ip_InstanceConfigType](#)

Definition at line 158 of file Ftm_Gpt_Ip_Types.h.

Data Fields

Type	Name	Description
boolean	freezeBit	Enable/Disable freezeBit.
Ftm_Gpt_Ip_ClockSource	clocksource	Select FTM clocksource.
uint8	clockPrescaler	Select prescalerValue.
Ftm_Gpt_Ip_CountingMode	mode	Select mode.

6.1.2.2 struct Ftm_Gpt_Ip_ChannelConfigType

Structure to configure the FTM channels.

This structure holds the configuration settings for the ChannelConfigType

Implements [Ftm_Gpt_Ip_ChannelConfigType](#)

Definition at line 175 of file Ftm_Gpt_Ip_Types.h.

Data Fields

Type	Name	Description
uint8	hwChannel	hwChannel
Ftm_Gpt_Ip_CallbackType	callback	callback
uint8	callbackParam	callbackParam
Ftm_Gpt_Ip_ChannelModeType	channelMode	channelMode

6.1.2.3 struct Ftm_Gpt_Ip_ChState

internal context structure

This structure is used by the IPL driver for internal logic. The content is populated on InitChannel

Definition at line 189 of file Ftm_Gpt_Ip_Types.h.

Data Fields

Type	Name	Description
boolean	chInit	chInit
Ftm_Gpt_Ip_CallbackType	callback	callback
uint8	callbackParam	callbackParam
Ftm_Gpt_Ip_ChannelModeType	channelMode	channelMode

6.1.2.4 struct Ftm_Gpt_Ip_InstanceState

internal context structure

This structure is used by the IPL driver for internal logic. The content is populated on Init

Definition at line 203 of file Ftm_Gpt_Ip_Types.h.

Data Fields

Type	Name	Description
uint8	clockPrescaler	Clock divide value for the NormalPrescaler.
uint8	clockAlternatePrescaler	Clock divide value for the AlternatePrescaler.

6.1.3 Types Reference

6.1.3.1 Ftm_Gpt_Ip_CallbackType

```
typedef void(* Ftm_Gpt_Ip_CallbackType) (uint8 callbackParam)
```

Callback type for each channel.

[Ftm_Gpt_Ip_CallbackType](#)

Definition at line 150 of file Ftm_Gpt_Ip_Types.h.

6.1.4 Enum Reference

6.1.4.1 Ftm_Gpt_Ip_CountingMode

enum `Ftm_Gpt_Ip_CountingMode`

Unit options for counting mode.

This is used to choose how the timer is counting

Enumerator

FTM_GPT_IP_MODE_UP_TIMER	Timer with up counter.
FTM_GPT_IP_MODE_UP_DOWN_TIMER	timer with up-down counter

Definition at line 101 of file Ftm_Gpt_Ip_Types.h.

6.1.4.2 Ftm_Gpt_Ip_ClockSource

```
enum Ftm_Gpt_Ip_ClockSource
```

Enum containing the FTM module clock sources.

This is used to choose the FTM clock sources.

Enumerator

FTM_GPT_IP_CLOCK_SOURCE_NONE	None use clock for FTM
FTM_GPT_IP_CLOCK_SOURCE_SYSTEMCLK	System clock
FTM_GPT_IP_CLOCK_SOURCE_FIXED_FREQUENCY	Fixed frequency
FTM_GPT_IP_CLOCK_SOURCE_EXTERNALCLK	External clock

Definition at line 112 of file Ftm_Gpt_Ip_Types.h.

6.1.4.3 Ftm_Gpt_Ip_ChannelModeType

```
enum Ftm_Gpt_Ip_ChannelModeType
```

Prescaler type. Indicates of whether the clock channel mode is "NORMAL" or "ALTERNATE".


This enumeration specifies the possible types of prescalers used to configure base-clock timers

Channel mode type. Indicates of whether the channel mode is "CONTINUOUS" or "ONE SHOT".

ChannelModeType of channel.

Enumerator

FTM_GPT_IP_CH_MODE_CONTINUOUS	channel mode - continuous mode
FTM_GPT_IP_CH_MODE_ONESHOT	channel mode - one-shot mode.



Module Documentation

Definition at line 137 of file Ftm_Gpt_Ip_Types.h.

6.2 Gpt Driver

6.2.1 Detailed Description

Macros

- `#define GPT_E_PARAM_CHANNEL`
Function Gpt_StartTimer is called when the driver is in sleep mode for a channel which is not wakeup enabled.
- `#define GPT_E_BUSY`
Function called with parameter value out of range.
- `#define GPT_E_TIMEOUT`
Function called when a timeout is occurred.
- `#define GPT_E_PARAM_CONFIG`
Function called with invalid the parameter in function Gpt_Init.
- `#define GPT_STARTTIMER_ID`
API service ID for Gpt_GetVersionInfo function.
- `#define GPT_PROCESSCOMMONINTERRUPT_ID`
API service ID for Gpt_StopTimer function.
- `#define GPT_INSTANCE_ID`
API service ID for Gpt_ChangeNextTimeoutValue function.
- `#define GPT_VALIDATE_GLOBAL_CALL`
GPT_VALIDATE_GLOBAL_CALL.
- `#define GPT_VALIDATE_CHANNEL_CALL`
GPT_VALIDATE_CHANNEL_CALL.
- `#define GPT_VALIDATE_STATE`
GPT_VALIDATE_STATE.
- `#define GPT_VALIDATE_PARAM`
GPT_VALIDATE_PARAM.

Types Reference

- `typedef uint8 Gpt_ChannelType`
Prescaler type. Indicates of whether the clock channel mode is "GPT_NORMAL" or "GPT_ALTERNATE".

Enum Reference

- `enum Gpt_ModeType`
This enumerated type allows the selection of different power modes.
- `enum Gpt_ChannelModeType`
Gpt channel mode type. Indicates of whether the channel mode is "CONTINUOUS" or "ONE SHOT".

Function Reference

- void [Gpt_Init](#) (const Gpt_ConfigType *configPtr)
GPT driver initialization function.
- void [Gpt_StartTimer](#) ([Gpt_ChannelType](#) channel, Gpt_ValueType value)
GPT driver function for starting a timer channel.
- void [Gpt_StopTimer](#) ([Gpt_ChannelType](#) channel)
GPT driver function for stopping a timer channel.
- void [Gpt_ProcessCommonInterrupt](#) (uint8 channel)
Gpt common handler to implements generic part of the ISR.

6.2.2 Macro Definition Documentation

6.2.2.1 GPT_E_PARAM_CHANNEL

```
#define GPT_E_PARAM_CHANNEL
```

Function [Gpt_StartTimer](#) is called when the driver is in sleep mode for a channel which is not wakeup enabled.

Errors and exceptions that will be detected by the GPT driver.

Function called without module initialization.

Errors and exceptions that will be detected by the GPT driver.

Initialization called when already initialized.

Errors and exceptions that will be detected by the GPT driver.

Function called for invalid channel.

Errors and exceptions that will be detected by the GPT driver.

Definition at line 197 of file Gpt.h.

6.2.2.2 GPT_E_BUSY

```
#define GPT_E_BUSY
```

Function called with parameter value out of range.

Errors and exceptions that will be detected by the GPT driver

Function called when timer channel is still running.

Errors and exceptions that will be detected by the GPT driver.

Definition at line 236 of file Gpt.h.

6.2.2.3 GPT_E_TIMEOUT

```
#define GPT_E_TIMEOUT
```

Function called when a timeout is occurred.

Errors and exceptions that will be detected by the GPT driver.

Definition at line 253 of file Gpt.h.

6.2.2.4 GPT_E_PARAM_CONFIG

```
#define GPT_E_PARAM_CONFIG
```

Function called with invalid the parameter in function Gpt_Init.

Errors and exceptions that will be detected by the GPT driver

API Gpt_SetClockMode service called with wrong parameter.

Parameters used when raising an error/exception

Function called with invalid mode param.

Errors and exceptions that will be detected by the GPT driver

function called for invalid channel on the current core

Errors and exceptions that will be detected by the GPT driver

Definition at line 289 of file Gpt.h.

6.2.2.5 GPT_STARTTIMER_ID

```
#define GPT_STARTTIMER_ID
```

API service ID for Gpt_GetVersionInfo function.

API SERVICE IDs

Parameters used when raising an error/exception

API service ID for Gpt_Init function

Parameters used when raising an error/exception

API service ID for Gpt_DeInit function

Parameters used when raising an error/exception

API service ID for Gpt_GetTimeElapsed function

Parameters used when raising an error/exception

API service ID for Gpt_GetTimeRemaining function

Parameters used when raising an error/exception

API service ID for Gpt_StartTimer function

Parameters used when raising an error/exception

Definition at line 336 of file Gpt.h.

6.2.2.6 GPT_PROCESSCOMMONINTERRUPT_ID

```
#define GPT_PROCESSCOMMONINTERRUPT_ID
```

API service ID for Gpt_StopTimer function.

Parameters used when raising an error/exception

API service ID for Gpt_SetMode function

Parameters used when raising an error/exception

API service ID for Gpt_ProcessCommonInterrupt generic ISR handler

Parameters used when raising an error/exception

Definition at line 397 of file Gpt.h.

6.2.2.7 GPT_INSTANCE_ID

```
#define GPT_INSTANCE_ID
```

API service ID for Gpt_ChangeNextTimeoutValue function.

Parameters used when raising an error/exception

API service ID for Gpt_SetClockMode function

Parameters used when raising an error/exception

API service ID for Gpt_GetPredefTimerValue function

Parameters used when raising an error/exception

API service ID for Gpt_Channel_EnableChainMode function

Parameters used when raising an error/exception

Instance ID of this GPT driver.

Definition at line 436 of file Gpt.h.

6.2.2.8 GPT_VALIDATE_GLOBAL_CALL

```
#define GPT_VALIDATE_GLOBAL_CALL
```

```
GPT_VALIDATE_GLOBAL_CALL.
```

Validates the global call uses all the channels - Gpt_Init, Gpt_DeInit, Gpt_SetMode.

Definition at line 73 of file Gpt_EnvCfg.h.

6.2.2.9 GPT_VALIDATE_CHANNEL_CALL

```
#define GPT_VALIDATE_CHANNEL_CALL
```

```
GPT_VALIDATE_CHANNEL_CALL.
```

Validates the call for a specific channel.

Definition at line 79 of file Gpt_EnvCfg.h.

6.2.2.10 GPT_VALIDATE_STATE

```
#define GPT_VALIDATE_STATE
```

```
GPT_VALIDATE_STATE.
```

Validates the channel status.

Definition at line 85 of file Gpt_EnvCfg.h.

6.2.2.11 GPT_VALIDATE_PARAM

```
#define GPT_VALIDATE_PARAM
```

```
GPT_VALIDATE_PARAM.
```

Validates the time value parameter.

Definition at line 91 of file Gpt_EnvCfg.h.

6.2.3 Types Reference

6.2.3.1 Gpt_ChannelType

```
typedef uint8 Gpt_ChannelType
```

Prescaler type. Indicates of whether the clock channel mode is "GPT_NORMAL" or "GPT_ALTERNATE".

This enumeration specifies the possible types of prescalers used to configure base-clock timers

Definition at line 484 of file Gpt.h.

6.2.4 Enum Reference

6.2.4.1 Gpt_ModeType

```
enum Gpt_ModeType
```

This enumerated type allows the selection of different power modes.

Modes of the GPT driver.

Enumerator

GPT_MODE_NORMAL	GPT Normal operation mode of the GPT.
GPT_MODE_SLEEP	GPT Sleep mode.

Definition at line 447 of file Gpt.h.

6.2.4.2 Gpt_ChannelModeType

```
enum Gpt_ChannelModeType
```

Gpt channel mode type. Indicates of whether the channel mode is "CONTINUOUS" or "ONE SHOT".

ChannelModeType of channel.

Enumerator

GPT_CH_MODE_CONTINUOUS	GPT channel mode - continuous mode.
GPT_CH_MODE_ONESHOT	GPT channel mode - one-shot mode.

Definition at line 457 of file Gpt.h.

6.2.5 Function Reference

6.2.5.1 Gpt_Init()

```
void Gpt_Init (
    const Gpt_ConfigType * configPtr )
```

GPT driver initialization function.

This service is a non reentrant function used for driver initialization. The Initialization function shall initialize all relevant registers of the configured hardware with the values of the structure referenced by the parameter ConfigPtr. All time units used within the API services of the GPT driver shall be of the unit ticks. This function shall only initialize the configured resources. Resources that are not configured in the configuration file shall not be touched. The following rules regarding initialization of controller registers shall apply to the GPT Driver implementation: [1] If the hardware allows for only one usage of the register, the driver module implementing that functionality is responsible for initializing the register [2] If the register can affect several hardware modules and if it is an IO register it shall be initialized by the PORT driver [3] If the register can affect several hardware modules and if it is not an IO register it shall be initialized by the MCU driver [4] One-time writable registers that require initialization directly after reset shall be initialized by the startup code [5] All other registers shall be initialized by the startup code

Parameters

in	<i>configPtr</i>	Pointer to a selected configuration structure
----	------------------	---

Returns

void

Precondition

The data structure including the configuration set required for initializing the GPT driver..

6.2.5.2 Gpt_StartTimer()

```
void Gpt_StartTimer (
    Gpt_ChannelType channel,
    Gpt_ValueType value )
```

GPT driver function for starting a timer channel.

The function Gpt_StartTimer shall start the selected timer channel with a defined time-out period. The function Gpt_StartTimer shall invoke the configured notification for that channel (see also GPT292) after the time-out period referenced via the parameter value (if enabled). All time units used within the API services of the GPT driver shall be of the unit ticks. In production mode no error is generated. The rational is that it adds no additional functionality to the driver. In this case the timer will be restarted with the time-out value, given as a parameter to the service. Usage of re-entrant capability is only allowed if the callers take care that there is no simultaneous usage of the same channel. To get times out of register values it is necessary to know the oscillator frequency, pre-scalers and so on. Since these settings are made in MCU and(or) in other modules it is not possible to calculate such times. Hence the conversions between time and ticks shall be part of an upper layer. The driver needs to be initialized before calling [Gpt_StartTimer\(\)](#). Otherwise, the function Gpt_StartTimer shall raise the development error GPT_E_UNINIT.

Parameters

in	<i>channel</i>	channel id
in	<i>value</i>	time-out period (in number of ticks) after a notification or a wakeup event shall occur.

Returns

void

Precondition

The driver needs to be initialized.

6.2.5.3 Gpt_StopTimer()

```
void Gpt_StopTimer (
    Gpt_ChannelType channel )
```

GPT driver function for stopping a timer channel.

Service for stopping the selected timer channel Stopping a timer channel, not been started before will not return a development error Timer channels configured in one shot mode are stopped automatically, when the time-out period has expired. Usage of re-entrant capability is only allowed if the callers take care that there is no simultaneous usage of the same channel. The driver needs to be initialized before calling [Gpt_StopTimer\(\)](#). Otherwise, the function shall raise the development error GPT_E_UNINIT.

Parameters

in	<i>channel</i>	channel id
----	----------------	------------

Returns

void

Precondition

The driver needs to be initialized. Gpt_StartTimer must be called before.

6.2.5.4 Gpt_ProcessCommonInterrupt()

```
void Gpt_ProcessCommonInterrupt (
    uint8 channel )
```

Gpt common handler to implements generic part of the ISR.

Generic function used by all interrupt service routines to call notification functions and wakeup the EcuM

Module Documentation

Parameters

in	<i>channel</i>	logic channel number
----	----------------	----------------------

Returns

void

Precondition

The driver needs to be initialized.

6.3 LPit IPL

6.3.1 Detailed Description

Data Structures

- struct [Lpit_Gpt_Ip_InstanceConfigType](#)
Structure to configure the LPIT. [More...](#)
- struct [Lpit_Gpt_Ip_ChannelConfigType](#)
Structure to configure the LPIT timer channel. [More...](#)
- struct [Lpit_Gpt_Ip_State](#)
internal context structure [More...](#)

Types Reference

- typedef void(* [Lpit_Gpt_Ip_CallbackType](#)) (uint8 callbackParam)
Callback type for each channel.

Enum Reference

- enum [Lpit_Gpt_Ip_StatusType](#)
LPit Status error.
- enum [Lpit_Gpt_Ip_ChannelModeType](#)
Channel mode type. Indicates of whether the channel mode is "CONTINUOUS" or "ONE SHOT".

6.3.2 Data Structure Documentation

6.3.2.1 struct Lpit_Gpt_Ip_InstanceConfigType

Structure to configure the LPIT.

This structure holds the configuration settings for the LPIT Implements :

Definition at line 127 of file LPit_Gpt_Ip_Types.h.

Data Fields

Type	Name	Description
boolean	stopRunInDebug	Stop timer running in debug mode.

6.3.2.2 struct Lpit_Gpt_Ip_ChannelConfigType

Structure to configure the LPIT timer channel.

This structure holds the configuration settings for the LPIT timer channel Implements :

Definition at line 141 of file LPit_Gpt_Ip_Types.h.

Data Fields

	Type	Name	Description
	uint8	hwChannel	Timer channel number
Lpit_Gpt_Ip_CallbackType		callback	callback
	uint8	callbackParam	callbackParam
Lpit_Gpt_Ip_ChannelModeType		channelMode	channelMode

6.3.2.3 struct Lpit_Gpt_Ip_State

internal context structure

This structure is used by the IPL driver for internal logic. The content is populated on Init

Definition at line 158 of file LPit_Gpt_Ip_Types.h.

Data Fields

	Type	Name	Description
	boolean	chInit	chInit
Lpit_Gpt_Ip_CallbackType		callback	callback
	uint8	callbackParam	callbackParam
Lpit_Gpt_Ip_ChannelModeType		channelMode	channelMode

6.3.3 Types Reference

6.3.3.1 Lpit_Gpt_Ip_CallbackType

```
typedef void(* Lpit_Gpt_Ip_CallbackType) (uint8 callbackParam)
```

Callback type for each channel.

Pit_Ip_CallbackType

Definition at line 119 of file LPit_Gpt_Ip_Types.h.

6.3.4 Enum Reference

6.3.4.1 Lpit_Gpt_Ip_StatusType

enum `Lpit_Gpt_Ip_StatusType`

LPit Status error.

Status error

Enumerator

LPIT_GPT_IP_SUCCESS	Status value is SUCCESS.
LPIT_GPT_IP_ERROR	Status value is ERROR.

Definition at line 96 of file LPit_Gpt_Ip_Types.h.

6.3.4.2 Lpit_Gpt_Ip_ChannelModeType

enum `Lpit_Gpt_Ip_ChannelModeType`

Channel mode type. Indicates of whether the channel mode is "CONTINUOUS" or "ONE SHOT".

ChannelModeType of channel.

Enumerator

LPIT_GPT_IP_CH_MODE_CONTINUOUS	channel mode - continuous mode
LPIT_GPT_IP_CH_MODE_ONESHOT	channel mode - one-shot mode.

Definition at line 106 of file LPit_Gpt_Ip_Types.h.

6.4 Lptmr IPL

6.4.1 Detailed Description

Data Structures

- struct [Lptmr_Gpt_Ip_State](#)
Internal context structure [Lptmr_Gpt_Ip_State](#). [More...](#)
- struct [Lptmr_Gpt_Ip_InstanceState](#)
Prescaler type. Indicates of whether the clock channel mode is "NORMAL" or "ALTERNATE". [More...](#)

Macros

- `#define E_TIMEOUT`
LPTMR [E_TIMEOUT](#).

Types Reference

- typedef void(* [Lptmr_Gpt_Ip_CallbackType](#)) (uint8 callbackParam)
Callback type for each channel.

Enum Reference

- enum [Lptmr_Gpt_Ip_ClockSelectType](#)
Enum containing the LPTMR module clock sources.
- enum [Lptmr_Gpt_Ip_StatusType](#)
LPTMR Status error.
- enum [Lptmr_Gpt_Ip_ChannelModeType](#)
Channel mode type. Indicates of whether the channel mode is "CONTINUOUS" or "ONE SHOT".

6.4.2 Data Structure Documentation

6.4.2.1 struct Lptmr_Gpt_Ip_State

Internal context structure [Lptmr_Gpt_Ip_State](#).

This structure is used by the IPL driver for internal logic. The content is populated on Init.

Definition at line 161 of file [Lptmr_Gpt_Ip_Types.h](#).

Data Fields

Type	Name	Description
boolean	chInit	chInit
Lptmr_Gpt_Ip_CallbackType	callback	callback
uint8	callbackParam	callbackParam
Lptmr_Gpt_Ip_ChannelModeType	channelMode	channelMode

6.4.2.2 struct Lptmr_Gpt_Ip_InstanceState

Prescaler type. Indicates of whether the clock channel mode is "NORMAL" or "ALTERNATE".

This enumeration specifies the possible types of prescalers used to configure base-clock timers

internal context structure

This structure is used by the IPL driver for internal logic. The content is populated on Init

Definition at line 188 of file Lptmr_Gpt_Ip_Types.h.

Data Fields

Type	Name	Description
uint8	clockPrescaler	Clock divide value for the NormalPrescaler.
uint8	clockAlternatePrescaler	Clock divide value for the AlternatePrescaler.

6.4.3 Macro Definition Documentation

6.4.3.1 E_TIMEOUT

```
#define E_TIMEOUT
```

LPTMR E_TIMEOUT.

Definition at line 89 of file Lptmr_Gpt_Ip_Types.h.

6.4.4 Types Reference

6.4.4.1 Lptmr_Gpt_Ip_CallbackType

```
typedef void(* Lptmr_Gpt_Ip_CallbackType) (uint8 callbackParam)
```

Callback type for each channel.

Lptmr_Gpt_Ip_CallbackType

Definition at line 136 of file Lptmr_Gpt_Ip_Types.h.

6.4.5 Enum Reference

6.4.5.1 Lptmr_Gpt_Ip_ClockSelectType

```
enum Lptmr_Gpt_Ip_ClockSelectType
```

Enum containing the LPTMR module clock sources.

Lptmr_Gpt_Ip_ClockSelectType

Enumerator

LPTMR_GPT_IP_CLOCK_SIRCDIV2	LPTMR clock source SIRCDIV2_CLK.
LPTMR_GPT_IP_CLOCK_LPO1K	LPTMR clock source LPO1K.
LPTMR_GPT_IP_CLOCK_RTC_CLK	LPTMR clock source RTC_CLK.
LPTMR_GPT_IP_CLOCK_PCC_LPTMR0	LPTMR clock source PCC_LPTMR0.

Definition at line 100 of file Lptmr_Gpt_Ip_Types.h.

6.4.5.2 Lptmr_Gpt_Ip_StatusType

```
enum Lptmr_Gpt_Ip_StatusType
```

LPTMR Status error.

Status error

Enumerator

LPTMR_GPT_IP_SUCCESS	Status value is SUCCESS.
LPTMR_GPT_IP_ERROR	Status value is ERROR.
LPTMR_GPT_IP_TIMEOUT	Status value is TIMEOUT.

Definition at line 112 of file Lptmr_Gpt_Ip_Types.h.

6.4.5.3 Lptmr_Gpt_Ip_ChannelModeType

enum [Lptmr_Gpt_Ip_ChannelModeType](#)

Channel mode type. Indicates of whether the channel mode is "CONTINUOUS" or "ONE SHOT".

ChannelModeType of channel.

Enumerator

LPTMR_GPT_IP_CH_MODE_CONTINUOUS	channel mode - continuous mode
LPTMR_GPT_IP_CH_MODE_ONESHOT	channel mode - one-shot mode.

Definition at line 123 of file Lptmr_Gpt_Ip_Types.h.

6.5 Rtc IPL

6.5.1 Detailed Description

Data Structures

- struct [Srtc_Ip_ConfigType](#)
Structure to configure the SRTC. [More...](#)
- struct [Srtc_Ip_TimedateType](#)
SRTC Time Date structure. [More...](#)
- struct [Srtc_Ip_AlarmConfigType](#)
SRTC alarm configuration. [More...](#)
- struct [Srtc_Ip_State](#)
Internal context structure [Srtc_Ip_State](#). [More...](#)

Types Reference

- typedef void(* [Srtc_Ip_CallbackType](#)) (uint8 callbackParam)
Callback type for each channel.

Enum Reference

- enum [Srtc_Ip_ClockSelectType](#)
Enum containing the SRTC module clock sources.
- enum [Srtc_Ip_ClockOutType](#)
SRTC CLKOUT pin configuration.
- enum [Srtc_Ip_InterruptType](#)
Enum containing SRTC interrupt flags.
- enum [Srtc_Ip_StatusType](#)
SRTC Status error.
- enum [Srtc_Ip_SecIntFreqType](#)
SRTC Seconds interrupt configuration.
- enum [Srtc_Ip_ChannelModeType](#)
Channel mode type. Indicates of whether the channel mode is "CONTINUOUS" or "ONE SHOT".

6.5.2 Data Structure Documentation

6.5.2.1 struct Srtc_Ip_ConfigType

Structure to configure the SRTC.

This structure holds the configuration settings for the SRTC

Definition at line 197 of file [SRTC_Ip_Types.h](#).

Data Fields

Type	Name	Description
Srtc_Ip_ClockSelectType	clockSelect	SRTC Clock Select
Srtc_Ip_ClockOutType	clockOutSelect	SRTC Clock Pin select - RTC_CLKOUT
Srtc_Ip_CallbackType	callback	Periodic interrupt callback
uint8	callbackParam	Pointer to callback parameters
Srtc_Ip_ChannelModeType	channelMode	channelMode

6.5.2.2 struct Srtc_Ip_TimedateType

SRTC Time Date structure.

[Srtc_Ip_TimedateType](#)

Definition at line 214 of file SRTC_Ip_Types.h.

Data Fields

Type	Name	Description
uint16	year	Year
uint16	month	Month
uint16	day	Day
uint16	hour	Hour
uint16	minutes	Minutes
uint8	seconds	Seconds

6.5.2.3 struct Srtc_Ip_AlarmConfigType

SRTC alarm configuration.

[Srtc_Ip_AlarmConfigType](#)

Definition at line 228 of file SRTC_Ip_Types.h.

Data Fields

Type	Name	Description
Srtc_Ip_TimedateType	alarmTime	Alarm time
uint32	repetitionInterval	Interval of repetition in seconds
uint32	numberOfRepeats	Number of alarm repeats
boolean	repeatForever	Repeat forever if set, discard number of repeats.
boolean	alarmIntEnable	Enable alarm interrupt
Srtc_Ip_CallbackType	alarmCallback	Pointer to the user callback method.
uint8	callbackParams	Pointer to the callback parameters.

6.5.2.4 struct Srtc_Ip_State

Internal context structure [Srtc_Ip_State](#).

This structure is used by the IPL driver for internal logic. The content is populated on Init.

Definition at line 256 of file [SRtc_Ip_Types.h](#).

Data Fields

Type	Name	Description
boolean	alarmInit	alarmInit
uint32	repetitionInterval	Interval of repetition in seconds
volatile uint32	numberOfRepeats	Number of alarm repeats
boolean	repeatForever	Repeat forever if set, discard number of repeats.
boolean	alarmIntEnable	Enable alarm interrupt
Srtc_Ip_CallbackType	alarmCallback	Pointer to the user callback method.
uint8	callbackParams	Pointer to the callback parameters.
volatile boolean	isAlarmTimeNew	Check if there is a new alarm
boolean	chInit	chInit
Srtc_Ip_CallbackType	callback	callback
uint8	callbackParam	callbackParam
Srtc_Ip_ChannelModeType	channelMode	channelMode

6.5.3 Types Reference

6.5.3.1 Srtc_Ip_CallbackType

```
typedef void(* Srtc_Ip_CallbackType) (uint8 callbackParam)
```

Callback type for each channel.

Srtc_Ip_CallbackType

Definition at line 191 of file SRtc_Ip_Types.h.

6.5.4 Enum Reference

6.5.4.1 Srtc_Ip_ClockSelectType

```
enum Srtc_Ip_ClockSelectType
```

Enum containing the SRTC module clock sources.

Srtc_Ip_ClockSelectType

Enumerator

SRTC_IP_CLK_SRC_OSC_32KHZ	SRTC Prescaler increments using 32 KHz crystal
SRTC_IP_CLK_SRC_LPO_1KHZ	SRTC Prescaler increments using 1KHz LPO

Definition at line 108 of file SRtc_Ip_Types.h.

6.5.4.2 Srtc_Ip_ClockOutType

```
enum Srtc_Ip_ClockOutType
```

SRTC CLKOUT pin configuration.

Enumerator

SRTC_IP_CLKOUT_DISABLED	Clock out pin is disabled
SRTC_IP_CLKOUT_SRC_TSIC	Output on RTC_CLKOUT as configured on Time seconds interrupt
SRTC_IP_CLKOUT_SRC_32KHZ	Output on RTC_CLKOUT of the 32KHz clock

Definition at line 117 of file SRtc_Ip_Types.h.

6.5.4.3 Srtc_Ip_InterruptType

enum [Srtc_Ip_InterruptType](#)

Enum containing SRTC interrupt flags.

SRTC_Ip_InterruptType

Enumerator

SRTC_IP_INVALID_INTERRUPT	RTC_TIME_INVALID_INTERRUPT
SRTC_IP_OVERFLOW_INTERRUPT	RTC_TIME_OVERFLOW_INTERRUPT.
SRTC_IP_ALARM_INTERRUPT	RTC_TIME_ALARM_INTERRUPT
SRTC_IP_SECONDS_INTERRUPT	RTC_TIME_SECONDS_INTERRUPT

Definition at line 128 of file SRtc_Ip_Types.h.

6.5.4.4 Srtc_Ip_StatusType

enum [Srtc_Ip_StatusType](#)

SRTC Status error.

Status error

Enumerator

SRTC_IP_SUCCESS	Status value is SUCCESS.
SRTC_IP_ERROR	Status value is ERROR

Definition at line 140 of file SRtc_Ip_Types.h.

6.5.4.5 Srtc_Ip_SecIntFreqType

enum [Srtc_Ip_SecIntFreqType](#)

SRTC Seconds interrupt configuration.

Enumerator

SRTC_IP_INT_1HZ	SRTC seconds interrupt occurs at 1 Hz
SRTC_IP_INT_2HZ	SRTC seconds interrupt occurs at 2 Hz
SRTC_IP_INT_4HZ	SRTC seconds interrupt occurs at 4 Hz
SRTC_IP_INT_8HZ	SRTC seconds interrupt occurs at 8 Hz
SRTC_IP_INT_16HZ	SRTC seconds interrupt occurs at 16 Hz
SRTC_IP_INT_32HZ	SRTC seconds interrupt occurs at 32 Hz
SRTC_IP_INT_64HZ	SRTC seconds interrupt occurs at 64 Hz
SRTC_IP_INT_128HZ	SRTC seconds interrupt occurs at 128 Hz

Definition at line 149 of file SRtc_Ip_Types.h.

6.5.4.6 Srtc_Ip_ChannelModeType

enum `Srtc_Ip_ChannelModeType`

Channel mode type. Indicates of whether the channel mode is "CONTINUOUS" or "ONE SHOT".

ChannelModeType of channel.

Enumerator

SRTC_IP_CH_MODE_CONTINUOUS	channel mode - continuous mode
SRTC_IP_CH_MODE_ONESHOT	channel mode - one-shot mode.

Definition at line 178 of file SRtc_Ip_Types.h.

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.

NXP, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, COOLFLUX, EMBRACE, GREENCHIP, HITAG, I2C BUS, ICODE, JCOP, LIFE VIBES, MIFARE, MIFARE CLASSIC, MIFARE DESFire, MIFARE PLUS, MIFARE FLEX, MANTIS, MIFARE ULTRALIGHT, MIFARE4MOBILE, MIGLO, NTAG, ROADLINK, SMARTLX, SMARTMX, STARPLUG, TOPFET, TRENCHMOS, UCODE, Freescale, the Freescale logo, Altivec, C-5, CodeTEST, CodeWarrior, ColdFire, ColdFire+, C-Ware, the Energy Efficient Solutions logo, Kinetis, Layerscape, MagniV, mobileGT, PEG, PowerQUICC, Processor Expert, QorIQ, QorIQ Qonverge, Ready Play, SafeAssure, the SafeAssure logo, StarCore, Symphony, VortiQa, Vybrid, Airfast, BeeKit, BeeStack, CoreNet, Flexis, MXC, Platform in a Package, QUICC Engine, SMARTMOS, Tower, TurboLink, and UMEMS are trademarks of NXP B.V. All other product or service names are the property of their respective owners. ARM, AMBA, ARM Powered, Artisan, Cortex, Jazelle, Keil, SecurCore, Thumb, TrustZone, and Vision are registered trademarks of ARM Limited (or its subsidiaries) in the EU and/or elsewhere. ARM7, ARM9, ARM11, big.LITTLE, CoreLink, CoreSight, DesignStart, Mali, mbed, NEON, POP, Sensinode, Socrates, ULINK and Versatile are trademarks of ARM Limited (or its subsidiaries) in the EU and/or elsewhere. All rights reserved. Oracle and Java are registered trademarks of Oracle and/or its affiliates. The Power Architecture and Power.org word marks and the Power and Power.org logos and related marks are trademarks and service marks licensed by Power.org.

© 2022 NXP B.V.

