

# MCAL Configuration Verification Manual for Pwm\_17\_GtmCcu6

**32-bit TriCore™ AURIX™ TC3xx microcontroller family**

## About this document

### Scope and purpose

This Configuration Data Reference document is applicable to all TC3xx devices in the TriCore™ AURIX™ family of 32-bit microcontrollers.

The purpose of this document is to facilitate the integrator to verify the generated code based on the input configuration parameters. This document describes details of structures, defines, macros and variables generated from the configuration parameters.

### Intended audience

This document is intended for integrators who need to understand the logic of the generated configuration code of AURIX™ AUTOSAR MCAL.

### Reference documents

This document should be read in conjunction with the following documents:

- AURIX™ TC3xx MCAL User Manual Pwm\_17\_GtmCcu6

## Table of contents

About this document.....	1
Table of contents.....	2
<b>1 Pwm_17_GtmCcu6 driver.....</b>	<b>4</b>
1.1 File: Pwm_17_GtmCcu6_Cfg.h .....	4
1.1.1 Macro: PWM_17_GTMCCU6_AR_RELEASE_MAJOR_VERSION.....	4
1.1.2 Macro: PWM_17_GTMCCU6_AR_RELEASE_MINOR_VERSION .....	4
1.1.3 Macro: PWM_17_GTMCCU6_AR_RELEASE_REVISION_VERSION.....	4
1.1.4 Macro: PWM_17_GTMCCU6_SW_MAJOR_VERSION.....	5
1.1.5 Macro: PWM_17_GTMCCU6_SW_MINOR_VERSION .....	5
1.1.6 Macro: PWM_17_GTMCCU6_SW_PATCH_VERSION .....	5
1.1.7 Macro: PWM_17_GTMCCU6_SAFETY_ENABLE .....	6
1.1.8 Macro: PWM_17_GTMCCU6_INITCHECK_API.....	6
1.1.9 Macro: PWM_17_GTMCCU6_DE_INIT_API.....	6
1.1.10 Macro: PWM_17_GTMCCU6_DEV_ERROR_DETECT .....	7
1.1.11 Macro: PWM_17_GTMCCU6_MULTICORE_ERROR_DETECT .....	7
1.1.12 Macro: PWM_17_GTMCCU6_SET_DUTY_CYCLE_API .....	7
1.1.13 Macro: PWM_17_GTMCCU6_GET_OUTPUT_STATE_API .....	8
1.1.14 Macro: PWM_17_GTMCCU6_SET_OUTPUT_TO_IDLE_API .....	8
1.1.15 Macro: PWM_17_GTMCCU6_VERSION_INFO_API .....	9
1.1.16 Macro: PWM_17_GTMCCU6_SET_PERIOD_AND_DUTY_API.....	9
1.1.17 Macro: PWM_17_GTMCCU6_HANDLE_SHIFT_BY_OFFSET.....	9
1.1.18 Macro: PWM_17_GTMCCU6_NOTIFICATION_SUPPORTED .....	10
1.1.19 Macro: PWM_17_GTMCCU6_NOTIF_FOR_100_0_ENABLE .....	10
1.1.20 Macro: PWM_17_GTMCCU6_DUTYCYCLE_UPDATED_ENDPERIOD .....	10
1.1.21 Macro: PWM_17_GTMCCU6_DUTY_PERIOD_UPDATED_ENDPERIOD.....	11
1.1.22 Macro: PWM_17_GTMCCU6_DUTY_SHIFT_IN_TICKS .....	12
1.1.23 Macro: PWM_17_GTMCCU6_INSTANCE_ID .....	12
1.1.24 Macro: PWM_17_GTMCCU6_MAX_CHANNELS_CORE<x> .....	12
1.1.25 Macro: PWM_17_GTMCCU6_MAX_CHANNELS .....	14
1.1.26 Macro: PWM_17_GTMCCU6_MAX_CORES.....	14
1.1.27 Macro: PWM_17_GTMCCU6_GTM_TIMER_USED .....	14
1.1.28 Macro: PWM_17_GTMCCU6_CCU6_TIMER_USED.....	15
1.1.29 Macro: Pwm_17_GtmCcu6Conf_PwmChannel_<channel name> .....	16
1.2 File: Pwm_17_GtmCcu6[_<variant>]_PBcfg.c.....	16
1.2.1 Structure: Pwm_17_GtmCcu6_Config[_<variant>] .....	17
1.2.1.1 Member: PwmCoreAdd[PWM_17_GTMCCU6_MAX_CORES] .....	18
1.2.1.2 Member: Pwm_ChannelIndexMap .....	19
1.2.1.3 Member: PwmCcu6ChIdx[MCU_17_CCU6_NO_OF_KERNELS] .....	19
1.2.2 Structure: Pwm_CoreConfigCore_<x>[_<variant>] .....	21
1.2.2.1 Member: Pwm_ChannelConfigPtr .....	22
1.2.2.2 Member: MaxChannels.....	22
1.2.3 Structure: Pwm_kChannelConfigurationCore <x>[_<variant>] .....	23
1.2.3.1 Member: PwmChannelId .....	24
1.2.3.2 Member: PwmTimerUsed .....	24
1.2.3.3 Member: PwmNotification.....	25
1.2.3.4 Member: PwmChannelInfo .....	25
1.2.3.5 Member: PwmPeriodDefault .....	28

## Table of contents

1.2.3.6	Member: PwmDutycycleDefault.....	28
1.2.3.7	Member: PwmShiftValue .....	29
1.2.3.8	Member: PwmTimerPtr.....	30
1.2.4	Array: Pwm_ChannelIndexMap[_<variant>][ PWM_17_GTMCCU6_MAX_CHANNELS].....	30
1.2.5	Structure: Pwm_kChannelConfigGtm_Core<x> [_<variant>][<y>] .....	32
1.2.5.1	Member: TimerType.....	33
1.2.5.2	Member: TimerId.....	34
1.2.5.3	Member: TimerChCtrlReg .....	35
1.2.5.4	Member: TimerChCN0Reg .....	38
1.2.5.5	Member: TimerChCM0Reg .....	40
1.2.5.6	Member: TimerChCM1Reg .....	42
1.2.5.7	Member: TimerChSR0Reg.....	44
1.2.5.8	Member: TimerChSR1Reg.....	45
1.2.5.9	Member: TimerChIntEnMode .....	45
1.2.6	Structure: Pwm_kChannelConfigCcu6_Core<x> [_<variant>][<y>] .....	46
1.2.6.1	Member: TimerId.....	47
1.2.6.2	Member: TimerCtrlReg0.....	48
1.2.6.3	Member: ModCtrlReg .....	49
1.2.6.4	Member: PasStateLvlReg.....	50
1.2.6.5	Member: TimerCntReg .....	53
1.2.6.6	Member: TimerPeriodReg.....	55
1.2.6.7	Member: Ccu6ShadowReg.....	55
1.2.6.8	Member: TimerModeSelectReg .....	57
1.2.6.9	Member: PortInSelReg0 .....	58
1.2.6.10	Member: IntEnReg.....	58
1.2.6.11	Member: IntNodePointerReg.....	59
1.3	File: Pwm_17_GtmCcu6[_<variant>]_PBcfg.h .....	61
1.3.1	Extern: Pwm_17_GtmCcu6_Config[_<variant>] .....	61
<b>Revision history.....</b>		<b>62</b>

## 1 Pwm\_17\_GtmCcu6 driver

This chapter describes the details of the configuration data generated from the PWM driver.

### 1.1 File: Pwm\_17\_GtmCcu6\_Cfg.h

The generated header file contains all pre-compile configuration parameters. Pre-compile time configuration allows decoupling of the static configuration from implementation. The file is generated in 'inc' folder.

#### 1.1.1 Macro: PWM\_17\_GTMCCU6\_AR\_RELEASE\_MAJOR\_VERSION

**Table 1** PWM\_17\_GTMCCU6\_AR\_RELEASE\_MAJOR\_VERSION

<b>Name</b>	PWM_17_GTMCCU6_AR_RELEASE_MAJOR_VERSION	
<b>Description</b>	Major version number of AUTOSAR release on which the Pwm_17_GtmCcu6 implementation is based on.	
<b>Verification method</b>	The macro is generated with the value present in 'CommonPublishedInformation/ArMajorVersion'.  <i>Note: The macro is not user configurable.</i>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Generate Pwm_17_GtmCcu6_Cfg.h file with ArMajorVersion 4	<pre>#define PWM_17_GTMCCU6_AR_RELEASE_MAJOR_VERSION (4U)</pre>

#### 1.1.2 Macro: PWM\_17\_GTMCCU6\_AR\_RELEASE\_MINOR\_VERSION

**Table 2** PWM\_17\_GTMCCU6\_AR\_RELEASE\_MINOR\_VERSION

<b>Name</b>	PWM_17_GTMCCU6_AR_RELEASE_MINOR_VERSION	
<b>Description</b>	Minor version number of AUTOSAR release on which the Pwm_17_GtmCcu6 implementation is based on.	
<b>Verification method</b>	The macro is generated with the value present in 'CommonPublishedInformation/ArMinorVersion'.  <i>Note: The macro is not user configurable.</i>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Generate Pwm_17_GtmCcu6_Cfg.h file with ArMinorVersion 2	<pre>#define PWM_17_GTMCCU6_AR_RELEASE_MINOR_VERSION (2U)</pre>

#### 1.1.3 Macro: PWM\_17\_GTMCCU6\_AR\_RELEASE\_REVISION\_VERSION

**Table 3** PWM\_17\_GTMCCU6\_AR\_RELEASE\_REVISION\_VERSION

<b>Name</b>	PWM_17_GTMCCU6_AR_RELEASE_REVISION_VERSION
-------------	--

<b>Description</b>	Revision version number of AUTOSAR release on which the Pwm_17_GtmCcu6 implementation is based on.	
<b>Verification method</b>	The macro is generated with the value present in 'CommonPublishedInformation/ArPatchVersion'.  <i>Note: The macro is not user configurable.</i>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Generate Pwm_17_GtmCcu6_Cfg.h file with ArPatchVersion 2	<pre>#define PWM_17_GTMCCU6_AR_RELEASE_REVISION_VERSION (2U)</pre>

#### 1.1.4 Macro: PWM\_17\_GTMCCU6\_SW\_MAJOR\_VERSION

**Table 4 PWM\_17\_GTMCCU6\_SW\_MAJOR\_VERSION**

<b>Name</b>	PWM_17_GTMCCU6_SW_MAJOR_VERSION	
<b>Description</b>	Major version number of the Pwm_17_GtmCcu6 module.	
<b>Verification method</b>	The macro is generated with the value present in 'CommonPublishedInformation/SwMajorVersion'.  <i>Note: The macro is not user configurable.</i>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Generate Pwm_17_GtmCcu6_Cfg.h file with SwMajorVersion 20	<pre>#define PWM_17_GTMCCU6_SW_MAJOR_VERSION (20U)</pre>

#### 1.1.5 Macro: PWM\_17\_GTMCCU6\_SW\_MINOR\_VERSION

**Table 5 PWM\_17\_GTMCCU6\_SW\_MINOR\_VERSION**

<b>Name</b>	PWM_17_GTMCCU6_SW_MINOR_VERSION	
<b>Description</b>	Minor version number of the Pwm_17_GtmCcu6 module.	
<b>Verification method</b>	The macro is generated with the value present in 'CommonPublishedInformation/SwMinorVersion'.  <i>Note: The macro is not user configurable.</i>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Generate Pwm_17_GtmCcu6_Cfg.h file with SwMinorVersion 0	<pre>#define PWM_17_GTMCCU6_SW_MINOR_VERSION (0U)</pre>

#### 1.1.6 Macro: PWM\_17\_GTMCCU6\_SW\_PATCH\_VERSION

**Table 6 PWM\_17\_GTMCCU6\_SW\_PATCH\_VERSION**

<b>Name</b>	PWM_17_GTMCCU6_SW_PATCH_VERSION	
<b>Description</b>	Patch level version number of the Pwm_17_GtmCcu6 module.	
<b>Verification method</b>	<p>The macro is generated with the value present in 'CommonPublishedInformation/SwPatchVersion'.</p> <p><i>Note: The macro is not user configurable.</i></p>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Generate Pwm_17_GtmCcu6_Cfg.h file with SwPatchVersion 0	#define PWM_17_GTMCCU6_SW_PATCH_VERSION (0U)

### 1.1.7 Macro: PWM\_17\_GTMCCU6\_SAFETY\_ENABLE

**Table 7 PWM\_17\_GTMCCU6\_SAFETY\_ENABLE**

<b>Name</b>	PWM_17_GTMCCU6_SAFETY_ENABLE	
<b>Description</b>	Enables/disables safety features	
<b>Verification method</b>	The macro is generated as STD_ON if PwmSafetyEnable configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	PwmSafetyEnable = True	#define PWM_17_GTMCCU6_SAFETY_ENABLE (STD_ON)
	PwmSafetyEnable = False	#define PWM_17_GTMCCU6_SAFETY_ENABLE (STD_OFF)

### 1.1.8 Macro: PWM\_17\_GTMCCU6\_INITCHECK\_API

**Table 8 PWM\_17\_GTMCCU6\_INITCHECK\_API**

<b>Name</b>	PWM_17_GTMCCU6_INITCHECK_API	
<b>Description</b>	Enables/disables Pwm_17_GtmCcu6_InitCheck API	
<b>Verification method</b>	The macro is generated as STD_ON if PwmInitCheckApi configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	PwmInitCheckApi = True	#define PWM_17_GTMCCU6_INITCHECK_API (STD_ON)
	PwmInitCheckApi = False	#define PWM_17_GTMCCU6_INITCHECK_API (STD_OFF)

### 1.1.9 Macro: PWM\_17\_GTMCCU6\_DE\_INIT\_API

**Table 9 PWM\_17\_GTMCCU6\_DE\_INIT\_API**

<b>Name</b>	PWM_17_GTMCCU6_DE_INIT_API	
<b>Description</b>	Decides the mode of execution of Init and Delnit API's.	

<b>Verification method</b>	The macro is generated as STD_ON if PwmDelInitApi configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	PwmDelInitApi = True	#define PWM_17_GTMCCU6_DE_INIT_API (STD_ON)
	PwmDelInitApi = False	#define PWM_17_GTMCCU6_DE_INIT_API (STD_OFF)

### 1.1.10 Macro: PWM\_17\_GTMCCU6\_DEV\_ERROR\_DETECT

**Table 10 PWM\_17\_GTMCCU6\_DEV\_ERROR\_DETECT**

<b>Name</b>	PWM_17_GTMCCU6_DEV_ERROR_DETECT	
<b>Description</b>	Enables/disables the Development Error Detection.	
<b>Verification method</b>	The macro is generated as STD_ON if PwmDevErrorDetect configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	PwmDevErrorDetect = True	#define PWM_17_GTMCCU6_DEV_ERROR_DETECT (STD_ON)
	PwmDevErrorDetect = False	#define PWM_17_GTMCCU6_DEV_ERROR_DETECT (STD_OFF)

### 1.1.11 Macro: PWM\_17\_GTMCCU6\_MULTICORE\_ERROR\_DETECT

**Table 11 PWM\_17\_GTMCCU6\_MULTICORE\_ERROR\_DETECT**

<b>Name</b>	PWM_17_GTMCCU6_MULTICORE_ERROR_DETECT	
<b>Description</b>	Enables/disables MultiCore DET Check	
<b>Verification method</b>	The macro is generated as STD_ON if PwmMultiCoreErrorDetect configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	PwmMultiCoreErrorDetect = True	#define PWM_17_GTMCCU6_MULTICORE_ERROR_DETECT (STD_ON)
	PwmMultiCoreErrorDetect = False	#define PWM_17_GTMCCU6_MULTICORE_ERROR_DETECT (STD_OFF)

### 1.1.12 Macro: PWM\_17\_GTMCCU6\_SET\_DUTY\_CYCLE\_API

**Table 12 PWM\_17\_GTMCCU6\_SET\_DUTY\_CYCLE\_API**

<b>Name</b>	PWM_17_GTMCCU6_SET_DUTY_CYCLE_API	
<b>Description</b>	Enables/disables Pwm_17_GtmCcu6_SetDutyCycle API	

<b>Verification method</b>	The macro is generated as STD_ON if PwmSetDutyCycle configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	PwmSetDutyCycle = True	#define PWM_17_GTMCCU6_SET_DUTY_CYCLE_API (STD_ON)
	PwmSetDutyCycle = False	#define PWM_17_GTMCCU6_SET_DUTY_CYCLE_API (STD_OFF)

### 1.1.13 Macro: PWM\_17\_GTMCCU6\_GET\_OUTPUT\_STATE\_API

**Table 13 PWM\_17\_GTMCCU6\_GET\_OUTPUT\_STATE\_API**

<b>Name</b>	PWM_17_GTMCCU6_GET_OUTPUT_STATE_API	
<b>Description</b>	Enables/disables Pwm_17_GtmCcu6_GetOutputState API	
<b>Verification method</b>	The macro is generated as STD_ON if PwmGetOutputState configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	PwmGetOutputState = True	#define PWM_17_GTMCCU6_GET_OUTPUT_STATE_API (STD_ON)
	PwmGetOutputState = False	#define PWM_17_GTMCCU6_GET_OUTPUT_STATE_API (STD_OFF)

### 1.1.14 Macro: PWM\_17\_GTMCCU6\_SET\_OUTPUT\_TO\_IDLE\_API

**Table 14 PWM\_17\_GTMCCU6\_SET\_OUTPUT\_TO\_IDLE\_API**

<b>Name</b>	PWM_17_GTMCCU6_SET_OUTPUT_TO_IDLE_API	
<b>Description</b>	Enables/disables Pwm_17_GtmCcu6_SetOutputToIdle API	
<b>Verification method</b>	The macro is generated as STD_ON if PwmSetOutputToIdle configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	PwmSetOutputToIdle = True	#define PWM_17_GTMCCU6_SET_OUTPUT_TO_IDLE_API (STD_ON)
	PwmSetOutputToIdle = False	#define PWM_17_GTMCCU6_SET_OUTPUT_TO_IDLE_API (STD_OFF)



### 1.1.15 Macro: PWM\_17\_GTMCCU6\_VERSION\_INFO\_API

**Table 15 PWM\_17\_GTMCCU6\_VERSION\_INFO\_API**

<b>Name</b>	PWM_17_GTMCCU6_VERSION_INFO_API	
<b>Description</b>	Enables/disables Pwm_17_GtmCcu6_GetVersionInfo API	
<b>Verification method</b>	The macro is generated as STD_ON if PwmVersionInfoApi configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	PwmVersionInfoApi = True	#define PWM_17_GTMCCU6_VERSION_INFO_API (STD_ON)
	PwmVersionInfoApi = False	#define PWM_17_GTMCCU6_VERSION_INFO_API (STD_OFF)

### 1.1.16 Macro: PWM\_17\_GTMCCU6\_SET\_PERIOD\_AND\_DUTY\_API

**Table 16 PWM\_17\_GTMCCU6\_SET\_PERIOD\_AND\_DUTY\_API**

<b>Name</b>	PWM_17_GTMCCU6_SET_PERIOD_AND_DUTY_API	
<b>Description</b>	Enables/disables Pwm_17_GtmCcu6_SetPeriodAndDuty API	
<b>Verification method</b>	The macro is generated as STD_ON if PwmSetPeriodAndDuty configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	PwmSetPeriodAndDuty = True	#define PWM_17_GTMCCU6_SET_PERIOD_AND_DUTY_API (STD_ON)
	PwmSetPeriodAndDuty = False	#define PWM_17_GTMCCU6_SET_PERIOD_AND_DUTY_API (STD_OFF)

### 1.1.17 Macro: PWM\_17\_GTMCCU6\_HANDLE\_SHIFT\_BY\_OFFSET

**Table 17 PWM\_17\_GTMCCU6\_HANDLE\_SHIFT\_BY\_OFFSET**

<b>Name</b>	PWM_17_GTMCCU6_HANDLE_SHIFT_BY_OFFSET	
<b>Description</b>	Enable/disable the handling of Shifted channel by offset	
<b>Verification method</b>	The macro is generated as STD_ON if PwmHandleShiftByOffset configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	PwmHandleShiftByOffset = True	#define PWM_17_GTMCCU6_HANDLE_SHIFT_BY_OFFSET (STD_ON)
	PwmHandleShiftByOffset = False	#define PWM_17_GTMCCU6_HANDLE_SHIFT_BY_OFFSET (STD_OFF)

### 1.1.18 Macro: PWM\_17\_GTMCCU6\_NOTIFICATION\_SUPPORTED

**Table 18 PWM\_17\_GTMCCU6\_NOTIFICATION\_SUPPORTED**

<b>Name</b>	PWM_17_GTMCCU6_NOTIFICATION_SUPPORTED	
<b>Description</b>	Enable/disable following APIs: Pwm_17_GtmCcu6_EnableNotification Pwm_17_GtmCcu6_DisableNotification	
<b>Verification method</b>	The macro is generated as STD_ON if PwmNotificationSupported configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	PwmNotificationSupported = True	#define PWM_17_GTMCCU6_NOTIFICATION_SUPPORTED (STD_ON)
	PwmNotificationSupported = False	#define PWM_17_GTMCCU6_NOTIFICATION_SUPPORTED (STD_OFF)

### 1.1.19 Macro: PWM\_17\_GTMCCU6\_NOTIF\_FOR\_100\_0\_ENABLE

**Table 19 PWM\_17\_GTMCCU6\_NOTIF\_FOR\_100\_0\_ENABLE**

<b>Name</b>	PWM_17_GTMCCU6_NOTIF_FOR_100_0_ENABLE	
<b>Description</b>	Enable/disable notification for 0% and 100% duty for variable and fixed period channels	
<b>Verification method</b>	The macro is generated <ul style="list-style-type: none"> <li>As STD_ON, if PwmNotificationSupported configuration parameter is set to 'True' and PwmEnable0Or100DutyNotification configuration parameter is set to 'True'.</li> <li>As STD_OFF, if PwmNotificationSupported configuration parameter is set to 'True' and PwmEnable0Or100DutyNotification configuration parameter is set to 'False'.</li> <li>As STD_OFF, if PwmNotificationSupported configuration parameter is set to 'False'</li> </ul>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	<ul style="list-style-type: none"> <li>PwmNotificationSupported = True</li> <li>PwmEnable0Or100DutyNotification = True</li> </ul>	#define PWM_17_GTMCCU6_NOTIF_FOR_100_0_ENABLE (STD_ON)
	<ul style="list-style-type: none"> <li>PwmNotificationSupported = True</li> <li>PwmEnable0Or100DutyNotification = False</li> </ul>	#define PWM_17_GTMCCU6_NOTIF_FOR_100_0_ENABLE (STD_OFF)

### 1.1.20 Macro: PWM\_17\_GTMCCU6\_DUTYCYCLE\_UPDATED\_ENDPERIOD

**Table 20 PWM\_17\_GTMCCU6\_DUTYCYCLE\_UPDATED\_ENDPERIOD**

<b>Name</b>	PWM_17_GTMCCU6_DUTYCYCLE_UPDATED_ENDPERIOD	
<b>Description</b>	Enables/disables the update of duty at the end period	
<b>Verification method</b>	<p>The macro is generated</p> <ul style="list-style-type: none"> <li>As STD_ON, if PwmChannelCoherentSelection configuration parameter is set to 'False' and PwmDutycycleUpdatedEndperiod configuration parameter is set to 'True'.</li> <li>As STD_OFF, if PwmChannelCoherentSelection configuration parameter is set to 'False' and PwmDutycycleUpdatedEndperiod configuration parameter is set to 'False'.</li> <li>As STD_OFF, if PwmChannelCoherentSelection configuration parameter is set to 'True'</li> </ul>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	<ul style="list-style-type: none"> <li>PwmChannelCoherentSelection = False</li> <li>PwmDutycycleUpdatedEndperiod = True</li> </ul>	<pre>#define PWM_17_GTMCCU6_DUTYCYCLE_UPDATED_ENDPERIOD (STD_ON)</pre>
	<ul style="list-style-type: none"> <li>PwmChannelCoherentSelection = False</li> <li>PwmDutycycleUpdatedEndperiod = False</li> </ul>	<pre>#define PWM_17_GTMCCU6_DUTYCYCLE_UPDATED_ENDPERIOD (STD_OFF)</pre>

### 1.1.21 Macro: PWM\_17\_GTMCCU6\_DUTY\_PERIOD\_UPDATED\_ENDPERIOD

**Table 21 PWM\_17\_GTMCCU6\_DUTY\_PERIOD\_UPDATED\_ENDPERIOD**

<b>Name</b>	PWM_17_GTMCCU6_DUTY_PERIOD_UPDATED_ENDPERIOD	
<b>Description</b>	Enables the update of period and duty at the end period for variable period channel	
<b>Verification method</b>	<p>The macro is generated</p> <ul style="list-style-type: none"> <li>As STD_ON, if PwmChannelCoherentSelection configuration parameter is set to 'False' and PwmPeriodUpdatedEndperiod configuration parameter is set to 'True'.</li> <li>As STD_OFF, if PwmChannelCoherentSelection configuration parameter is set to 'False' and PwmPeriodUpdatedEndperiod configuration parameter is set to 'False'.</li> <li>As STD_OFF, if PwmChannelCoherentSelection configuration parameter is set to 'True'</li> </ul>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	<ul style="list-style-type: none"> <li>PwmChannelCoherentSelection = False</li> <li>PwmPeriodUpdatedEndperiod = True</li> </ul>	<pre>#define PWM_17_GTMCCU6_DUTY_PERIOD_UPDATED_ENDPERIOD (STD_ON)</pre>
	<ul style="list-style-type: none"> <li>PwmChannelCoherentSelection = False</li> <li>PwmPeriodUpdatedEndperiod = False</li> </ul>	<pre>#define PWM_17_GTMCCU6_DUTY_PERIOD_UPDATED_ENDPERIOD (STD_OFF)</pre>

### 1.1.22 Macro: PWM\_17\_GTMCCU6\_DUTY\_SHIFT\_IN\_TICKS

**Table 22 PWM\_17\_GTMCCU6\_DUTY\_SHIFT\_IN\_TICKS**

<b>Name</b>	PWM_17_GTMCCU6_DUTY_SHIFT_IN_TICKS	
<b>Description</b>	Enables the user to enter the duty cycle and shift value in absolute ticks, instead of percentage	
<b>Verification method</b>	The macro is generated as STD_ON if PwmDutyShiftInTicks configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	PwmDutyShiftInTicks = True	#define PWM_17_GTMCCU6_DUTY_SHIFT_IN_TICKS (STD_ON)
	PwmDutyShiftInTicks = False	#define PWM_17_GTMCCU6_DUTY_SHIFT_IN_TICKS (STD_OFF)

### 1.1.23 Macro: PWM\_17\_GTMCCU6\_INSTANCE\_ID

**Table 23 PWM\_17\_GTMCCU6\_INSTANCE\_ID**

<b>Name</b>	PWM_17_GTMCCU6_INSTANCE_ID	
<b>Description</b>	Instance ID of PWM module	
<b>Verification method</b>	The macro is generated as a numeric value set in the configuration parameter 'PwmIndex'	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Set PwmIndex as 0	#define PWM_17_GTMCCU6_INSTANCE_ID (0U)
	Set PwmIndex as 170	#define PWM_17_GTMCCU6_INSTANCE_ID (170U)

### 1.1.24 Macro: PWM\_17\_GTMCCU6\_MAX\_CHANNELS\_CORE<x>

**Table 24 PWM\_17\_GTMCCU6\_MAX\_CHANNELS\_CORE<x>**

<b>Name</b>	PWM_17_GTMCCU6_MAX_CHANNELS_CORE<x>	
<b>Description</b>	PWM_17_GTMCCU6_MAX_CHANNELS_CORE<x> (x ranges from 0 to 5)	
<b>Verification method</b>	Indicates the total number of channels configured for CORE<x>.	
	<i>Note: Channels not assigned to any core are assigned to master core (ResourceMMasterCore).</i>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	<ul style="list-style-type: none"> <li>Configure 9 PWM channels in Pwm/PwmChannelConfigSet/PwmChannel (PwmChannel_0 to</li> </ul>	<pre>#define PWM_17_GTMCCU6_MAX_CHANNELS_CORE0 (0U) #define PWM_17_GTMCCU6_MAX_CHANNELS_CORE1 (0U) #define</pre>

<ul style="list-style-type: none"> <li>PwmChannel_8).</li> <li>ResourceMMasterCore is CORE3 in ResourceM/ResourceMMcal Config/ResourceMMcalConfig_0</li> <li>Assign PwmChannel_0, PwmChannel_3, PwmChannel_6 and PwmChannel_7 with ResourceMCoreID as CORE4 in ResourceM/ResourceMMcal Config/ResourceMMcalConfig_0/ResourceMMcalCore/ResourceMAllocation</li> <li>Do not assign rest of the PWM channels to any core.</li> </ul>	<pre>PWM_17_GTMCCU6_MAX_CHANNELS_CORE2 (0U) #define PWM_17_GTMCCU6_MAX_CHANNELS_CORE3 (5U) #define PWM_17_GTMCCU6_MAX_CHANNELS_CORE4 (4U) #define PWM_17_GTMCCU6_MAX_CHANNELS_CORE5 (0U)</pre>
<ul style="list-style-type: none"> <li>Configure 4 PWM channels in Pwm/PwmChannelConfigSet/PwmChannel (PwmChannel_0 to PwmChannel_3).</li> <li>Set ResourceMMasterCore as CORE1 in ResourceM/ResourceMMcal Config/ResourceMMcalConfig_0</li> </ul> <p>Do not assign PWM channels in any ResourceMAllocation</p>	<pre>#define PWM_17_GTMCCU6_MAX_CHANNELS_CORE0 (0U) #define PWM_17_GTMCCU6_MAX_CHANNELS_CORE1 (4U) #define PWM_17_GTMCCU6_MAX_CHANNELS_CORE2 (0U) #define PWM_17_GTMCCU6_MAX_CHANNELS_CORE3 (0U) #define PWM_17_GTMCCU6_MAX_CHANNELS_CORE4 (0U) #define PWM_17_GTMCCU6_MAX_CHANNELS_CORE5 (0U)</pre>
<ul style="list-style-type: none"> <li>Configure 4 PWM channels in Pwm/PwmChannelConfigSet/PwmChannel (PwmChannel_0 to PwmChannel_3).</li> <li>Assign all the channels with ResourceMCoreID as CORE0 in ResourceM/ResourceMMcal Config/ResourceMMcalConfig_0/ResourceMMcalCore/ResourceMAllocation</li> </ul>	<pre>#define PWM_17_GTMCCU6_MAX_CHANNELS_CORE0 (4U) #define PWM_17_GTMCCU6_MAX_CHANNELS_CORE1 (0U) #define PWM_17_GTMCCU6_MAX_CHANNELS_CORE2 (0U) #define PWM_17_GTMCCU6_MAX_CHANNELS_CORE3 (0U) #define PWM_17_GTMCCU6_MAX_CHANNELS_CORE4 (0U) #define PWM_17_GTMCCU6_MAX_CHANNELS_CORE5 (0U)</pre>

### 1.1.25 Macro: PWM\_17\_GTMCCU6\_MAX\_CHANNELS

**Table 25 PWM\_17\_GTMCCU6\_MAX\_CHANNELS**

<b>Name</b>	PWM_17_GTMCCU6_MAX_CHANNELS	
<b>Description</b>	Indicates the total number of PWM channels configured	
<b>Verification method</b>	The macro is generated as a numeric value which corresponds to the number of elements in the list 'PwmChannelConfigSet/PwmChannel'.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure 3 PWM channels in 'PwmChannelConfigSet' container (PwmChannel_0 to PwmChannel_2)	#define PWM_17_GTMCCU6_MAX_CHANNELS (3U)
	Configure 8 PWM channels in 'PwmChannelConfigSet' container (PwmChannel_0 to PwmChannel_7)	#define PWM_17_GTMCCU6_MAX_CHANNELS (8U)

### 1.1.26 Macro: PWM\_17\_GTMCCU6\_MAX\_CORES

**Table 26 PWM\_17\_GTMCCU6\_MAX\_CORES**

<b>Name</b>	PWM_17_GTMCCU6_MAX_CORES	
<b>Description</b>	Indicates the total number of cores available in the device	
<b>Verification method</b>	The macro is generated as a numeric value which corresponds to the number of cores available in the device	
	<i>Note: The macro is not user configurable.</i>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Number of cores available for the device is 6	#define PWM_17_GTMCCU6_MAX_CORES (6U)
	Number of cores available for the device is 3	#define PWM_17_GTMCCU6_MAX_CORES (3U)

### 1.1.27 Macro: PWM\_17\_GTMCCU6\_GTM\_TIMER\_USED

**Table 27 PWM\_17\_GTMCCU6\_GTM\_TIMER\_USED**

<b>Name</b>	PWM_17_GTMCCU6_GTM_TIMER_USED	
<b>Description</b>	Indicates that GTM channel/s are configured in PWM	
<b>Verification method</b>	The macro is generated as STD_ON if one or more PWM channel have PwmAssignedHwUnit configuration parameter as 'GTM' else the macro is generated as STD_OFF.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	<ul style="list-style-type: none"> <li>Configure 4 PWM channels in Pwm/PwmChannelConfigSet/PwmChannel</li> </ul>	#define PWM_17_GTMCCU6_GTM_TIMER_USED (STD_ON)

(PwmChannel_0 to PwmChannel_3) <ul style="list-style-type: none"> <li>Assign PwmAssignedHwUnit as GTM for PwmChannel_0 and PwmChannel_3</li> <li>Assign PwmAssignedHwUnit as CCU6 for PwmChannel_1 and PwmChannel_2</li> </ul>	
<ul style="list-style-type: none"> <li>Configure 3 PWM channels in Pwm/PwmChannelConfigSet/PwmChannel (PwmChannel_0 to PwmChannel_2)</li> <li>Assign PwmAssignedHwUnit as CCU6 for PwmChannel_0, PwmChannel_1 and PwmChannel_2</li> </ul>	#define PWM_17_GTMCCU6_GTM_TIMER_USED (STD_OFF)

### 1.1.28 Macro: PWM\_17\_GTMCCU6\_CCU6\_TIMER\_USED

**Table 28** PWM\_17\_GTMCCU6\_CCU6\_TIMER\_USED

<b>Name</b>	PWM_17_GTMCCU6_CCU6_TIMER_USED	
<b>Description</b>	Indicates that CCU6 channel/s are configured in PWM	
<b>Verification method</b>	The macro is generated as STD_ON if one or more PWM channel have PwmAssignedHwUnit configuration parameter as 'CCU6' else the macro is generated as STD_OFF.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	<ul style="list-style-type: none"> <li>Configure 4 PWM channels in Pwm/PwmChannelConfigSet/PwmChannel (PwmChannel_0 to PwmChannel_3).</li> <li>Assign PwmAssignedHwUnit as GTM for PwmChannel_0 and PwmChannel_3</li> <li>Assign PwmAssignedHwUnit as CCU6 for PwmChannel_1 and PwmChannel_2</li> </ul>	#define PWM_17_GTMCCU6_CCU6_TIMER_USED (STD_ON)
	<ul style="list-style-type: none"> <li>Configure 3 PWM channels</li> </ul>	#define PWM_17_GTMCCU6_CCU6_TIMER_USED

	in Pwm/PwmChannelConfigSet/PwmChannel (PwmChannel_0 to PwmChannel_2)	(STD_OFF)
	<ul style="list-style-type: none"> <li>Assign            PwmAssignedHwUnit as            GTM for PwmChannel_0,            PwmChannel_1 and            PwmChannel_2</li> </ul>	

### 1.1.29 Macro: Pwm\_17\_GtmCcu6Conf\_PwmChannel\_<channel name>

**Table 29** Pwm\_17\_GtmCcu6Conf\_PwmChannel\_<channel name>

<b>Name</b>	Pwm_17_GtmCcu6Conf_PwmChannel_<channel name>	
<b>Description</b>	The macro is the symbolic name generated for the configuration parameter 'Pwm/PwmChannelConfigSet/PwmChannel'	
<b>Verification method</b>	The macro is generated as a numeric value which is configured in 'Pwm/PwmChannelConfigSet/PwmChannel'. <channel name> is the name of the PWM channel's container name.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	<ul style="list-style-type: none"> <li>Configure 4 PWM channels in Pwm/PwmChannelConfigSet/PwmChannel.</li> <li>Container PwmChannel for Channel ID 0 is named TOM1.</li> <li>Container PwmChannel for Channel ID 1 is named ATOM1.</li> <li>Container PwmChannel for Channel ID 2 is named CCU60_1</li> <li>Container PwmChannel for Channel ID 3 is named TOM2</li> </ul>	<pre>#define Pwm_17_GtmCcu6Conf_PwmChannel_TOM1 (0U) #define Pwm_17_GtmCcu6Conf_PwmChannel_ATOM1 (1U) #define Pwm_17_GtmCcu6Conf_PwmChannel_CCU60_1 (2U) ) #define Pwm_17_GtmCcu6Conf_PwmChannel_TOM2 (3U)</pre>

## 1.2 File: Pwm\_17\_GtmCcu6[\_<variant>]\_PBcfg.c

The generated file contains all post-build configuration parameters. Post-build time configuration mechanism allows configurable functionality of PWM driver that is deployed as object code. The file is generated in 'src' folder.



### 1.2.1 Structure: Pwm\_17\_GtmCcu6\_Config[\_<variant>]

**Table 30** Pwm\_17\_GtmCcu6\_Config[\_<variant>]

Name	Pwm_17_GtmCcu6_Config[_<variant>]	
Type	Pwm_17_GtmCcu6_ConfigType	
Description	Root configuration structure of PWM driver which will be used during initialization.	
Verification method	The generated structure is present in Pwm_17_GtmCcu6[_<variant>]_PBcfg.c file. The <variant> indicates the name of the post-build variant. For a variant-aware configuration the structure name is appended with the variant name. For variant-unaware configuration <variant> is ignored.	
Example(s)	Action	Generated output
	Allocate all PWM channels to Core0 (variant-unaware)	<pre>const Pwm_17_GtmCcu6_ConfigType Pwm_17_GtmCcu6_Config = {     {         (Pwm_17_GtmCcu6_CoreConfigType*)&amp;Pwm_CoreConfigCore0,         NULL_PTR,         NULL_PTR,         NULL_PTR,         NULL_PTR,         NULL_PTR     },     (uint8*)&amp;Pwm_ChannelIndexMap,     {         0xff050403U,         0xfffffffffU     } };</pre>
Example(s)	Allocate all PWM channels to Core0 (variant-aware. Variant name is 'Petrol')	<pre>const Pwm_17_GtmCcu6_ConfigType Pwm_17_GtmCcu6_Config_Petrol = {     {         (Pwm_17_GtmCcu6_CoreConfigType*)&amp;Pwm_CoreConfigCore0_Petrol,         NULL_PTR,         NULL_PTR,         NULL_PTR,         NULL_PTR,         NULL_PTR     },     (uint8*)&amp;Pwm_ChannelIndexMap_Petrol,     {         0xff050403U,         0xfffffffffU     } };</pre>

	<pre>(uint8*)&amp;Pwm_ChannelIndexMap_Petrol, {     0xff050403U,     0xffffffffU } };</pre>
--	---

### 1.2.1.1 Member: PwmCoreAdd[PWM\_17\_GTMCCU6\_MAX\_CORES]

**Table 31** PwmCoreAdd[PWM\_17\_GTMCCU6\_MAX\_CORES]

<b>Name</b>	PwmCoreAdd[PWM_17_GTMCCU6_MAX_CORES]	
<b>Type</b>	Pwm_17_GtmCcu6_CoreConfigType*	
<b>Description</b>	Array of core-specific configuration.	
<b>Verification method</b>	The generated structure member is present in the Pwm_17_GtmCcu6_Config[_<variant>] structure. If a Core<x> is allocated at least one channel, then the element <x> shall be generated as a pointer to Pwm_17_GtmCcu6_CoreConfigType (&Pwm_CoreConfigCore<x>) else 'NULL_PTR' is generated. (x in range 0 to 5).	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	All the PWM channels are allocated to Core 0 (variant-unaware)	<pre>{ (Pwm_17_GtmCcu6_CoreConfigType*)&amp;Pwm_CoreConfigCore0,     NULL_PTR,     NULL_PTR,     NULL_PTR,     NULL_PTR,     NULL_PTR }</pre>
	All the PWM channels are allocated to Core 0 (variant-aware. Variant name is 'Petrol')	<pre>{ (Pwm_17_GtmCcu6_CoreConfigType*)&amp;Pwm_CoreConfigCore0_Petrol,     NULL_PTR,     NULL_PTR,     NULL_PTR,     NULL_PTR,     NULL_PTR }</pre>
	All the PWM channels are	{

split between all cores except Core 0. (variant-unaware)	<pre> NULL_PTR,  (Pwm_17_GtmCcu6_CoreConfigType*) &amp;Pwm_CoreConfigCore1,  (Pwm_17_GtmCcu6_CoreConfigType*) &amp;Pwm_CoreConfigCore2,  (Pwm_17_GtmCcu6_CoreConfigType*) &amp;Pwm_CoreConfigCore3,  (Pwm_17_GtmCcu6_CoreConfigType*) &amp;Pwm_CoreConfigCore4,  (Pwm_17_GtmCcu6_CoreConfigType*) &amp;Pwm_CoreConfigCore5  } </pre>
--	--

### 1.2.1.2 Member: Pwm\_ChannelIndexMap

**Table 32** PwmChannelIdxmap

<b>Name</b>	Pwm_ChannelIndexMap	
<b>Type</b>	uint8*	
<b>Description</b>	Pointer to channel index map.	
<b>Verification method</b>	The generated structure member contains the address of Pwm_ChannelIndexMap.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure one or more PWM channels in Pwm/ELEMENTS/Pwm/PwmChannelConfigSet/PwmChannel	(uint8*) &Pwm_ChannelIndexMap

### 1.2.1.3 Member: PwmCcu6ChIdx[MCU\_17\_CCU6\_NO\_OF\_KERNELS]

**Table 33** PwmCcu6ChIdx[MCU\_17\_CCU6\_NO\_OF\_KERNELS]

<b>Name</b>	PwmCcu6ChIdx[MCU_17_CCU6_NO_OF_KERNELS]	
<b>Type</b>	uint32	
<b>Description</b>	Channel ID of CCU6 channels for a particular kernel.	
<b>Verification method</b>	The generated structure member contains 2 array entries with each entry corresponding to CCU6 kernel. If the PWM channel is not configured '0xFFFFFFFF' is generated, else the channel ID of CCU6 channels are generated. The structure member is generated as a value:	
	1. Value = 0 2. Value = Value   CCU60 Channel ID	

	3. Value = Value   CCU61 Channel ID <= 8 4. Value = Value   CCU62 Channel ID <= 16	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> <li>Assign CCU6 to PwmAssignedHwUnit parameter for PwmChannel_4 with PwmChannelId 3</li> <li>Assign T12 to CCU6TimerUsed parameter and Cc61 to Cc6xChannel parameter in container 'PwmChannel/CCU6CC6C onfiguration'</li> <li>Assign /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuCcu6ModuleAllocationConf_1 to CCU6KernelUsed parameter in container 'PwmChannel/CCU6CC6C onfiguration' for PwmChannel_4</li> </ul>	<pre>{     0xFFFFFFFFFU,     0xFFFF03FFU }</pre>
	<ul style="list-style-type: none"> <li>Assign CCU6 to PwmAssignedHwUnit parameter for PwmChannel_0 with PwmChannelId 1</li> <li>Assign T12 to CCU6TimerUsed parameter and Cc60 to Cc6xChannel parameter in container 'PwmChannel/CCU6CC6C onfiguration' for PwmChannel_0</li> <li>Assign CCU6 to PwmAssignedHwUnit parameter for PwmChannel_4 with PwmChannelId 4</li> <li>Assign T12 to CCU6TimerUsed parameter and Cc61 to Cc6xChannel parameter in container</li> </ul>	<pre>{     0xFFFF0401U,     0xFFFFFFFFFU, }</pre>

<p>‘PwmChannel/CCU6CC6C onfiguration’ for PwmChannel_4</p> <ul style="list-style-type: none"> <li>Assign /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuCcu6ModuleAllocationConf_0 to CCU6KernelUsed parameter in container ‘PwmChannel/CCU6CC6C onfiguration’ for PwmChannel_4 and PwmChannel_0</li> </ul>	
--	--

### 1.2.2 Structure: Pwm\_CoreConfigCore\_<x>[\_<variant>]

Table 34 Pwm\_CoreConfigCore\_<x>[\_<variant>]

Name	Pwm_CoreConfigCore_<x>[_<variant>]	
Type	Pwm_17_GtmCcu6_CoreConfigType	
Description	Configuration structure which holds the channel information for Core <x>, which will be referenced in root configuration structure. (x ranges from 0 to 5)	
Verification method	The generated file has this structure if at least one channel is assigned to Core <x>. <Variant> indicates the name of the post-build variant. For a variant aware configuration the structure name is appended with the variant name. For variant unaware configuration <variant> is ignored.	
Example(s)	Action	Generated output
	Configure PWM channels to Core0 (variant-unaware).	<pre>static const Pwm_17_GtmCcu6_CoreConfigType Pwm_CoreConfigCore0 = {     (Pwm_17_GtmCcu6_ChannelType) 8U,     (const Pwm_17_GtmCcu6_ChannelConfigType*) &amp;Pwm_k ChannelConfigurationCore0 };</pre>
	Configure 4 PWM channels to Core0 and 3 channels to Core1 (variant-aware. Variant name is ‘Petrol’)	<pre>static const Pwm_17_GtmCcu6_CoreConfigType Pwm_CoreConfigCore0_Petrol = {     (Pwm_17_GtmCcu6_ChannelType) 4U,     (const Pwm_17_GtmCcu6_ChannelConfigType*) &amp;Pwm_k ChannelConfigurationCore0_Petrol };  static const</pre>

		<pre> Pwm_17_GtmCcu6_CoreConfigType Pwm_CoreConfigCore1_Petrol = {     (Pwm_17_GtmCcu6_ChannelType) 3U,     (const Pwm_17_GtmCcu6_ChannelConfigType*) &amp;Pwm_k ChannelConfigurationCore1_Petrol }; </pre>
--	--	---

### 1.2.2.1 Member: Pwm\_ChannelConfigPtr

Table 35 Pwm\_ChannelConfigPtr

<b>Name</b>	Pwm_ChannelConfigPtr	
<b>Type</b>	Pwm_17_GtmCcu6_ChannelConfigType *	
<b>Description</b>	Pointer to the base of array which stores the data of each channel configured to Core<x>.	
<b>Verification method</b>	The structure member is generated with base address of array which stores the channel data of Core <x>.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure at least 1 PWM channel to Core0 in ResourceM/ResourceMMcalConfig/ResourceMMcalConfig_0/ResourceMMcalCore/ResourceMMcalCore_0/ResourceM Allocation/ResourceMAllocation_0	<pre> (const Pwm_17_GtmCcu6_ChannelConfigType*) &amp;Pwm_k ChannelConfigurationCore0[0] </pre>
	Configure 6 PWM channels to Core0 in ResourceM/ResourceMMcalConfig/ResourceMMcalConfig_0/ResourceMMcalCore/ResourceMMcalCore_0/ResourceM Allocation/ResourceMAllocation_0	<pre> (const Pwm_17_GtmCcu6_ChannelConfigType*) &amp;Pwm_k ChannelConfigurationCore0[0] </pre>

### 1.2.2.2 Member: MaxChannels

Table 36 MaxChannels

<b>Name</b>	MaxChannels
<b>Type</b>	Pwm_17_GtmCcu6_ChannelType
<b>Description</b>	Indicates the total number of channels assigned to Core <x>.
<b>Verification method</b>	The structure member is generated as total number of channels allocated to CORE<x>.

	<p>Note: Channels not assigned to any core are assigned to master core (ResourceMMasterCore).</p>	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> <li>Configure 4 PWM channels in Pwm/PwmChannelConfig Set/PwmChannel. Allocate 3 channels to Core 0.</li> <li>Allocate 1 channel to Core 1.</li> </ul> <p>Output is shown for Core 0</p>	3
	<ul style="list-style-type: none"> <li>Configure 14 PWM channels in Pwm/PwmChannelConfig Set/PwmChannel. Allocate 3 channels to Core 1.</li> <li>ResourceMMasterCore is CORE0.</li> <li>Do not allocate rest of the channels to any core.</li> </ul> <p>Output is shown for Core 0</p>	11

### 1.2.3 Structure: Pwm\_kChannelConfigurationCore <x>[\_<variant>]

**Table 1** Pwm\_kChannelConfigurationCore <x>[\_<variant>]

<b>Name</b>	Pwm_kChannelConfigurationCore <x>[_<variant>]	
<b>Type</b>	Pwm_17_GtmCcu6_ChannelConfigType	
<b>Description</b>	Configuration structure which holds the channel specific information belonging to Core <x> which will be referenced in core specific configuration structure. (x ranges from 0 to 5)	
<b>Verification method</b>	The generated file has this structure if at least one channel is assigned to Core <x>. <Variant> indicates the name of the post-build variant. For a variant aware configuration the structure name is appended with the variant name. For variant unaware configuration <variant> is ignored.	
Example(s)	Action	Generated output
	Configure 1 PWM channel to Core0 in ResourceM/ResourceMMcalConfig/ResourceMMcalConfig_0/ResourceMMcalCore/ResourceMMcalCore_0/ResourceMAllocation	<pre>static const Pwm_17_GtmCcu6_ChannelConfigType Pwm_kChannelConfigurationCore0[ ] = {     {         0, /* Pwm logical channel ID */</pre>

		<pre> 0, /* Timer used GTM:0/CCU6:1 */ Pwm_1ConfigChannel( 0U, /*Channel reset from other channel or not*/ PWM_17_GTMCCU6_FIXED_PERIOD, /*ChannelCla ss*/  PWM_17_GTMCCU6_COHERENT, /*Coherency*/ PWM_17_GTMCCU6_LOW, /*channel Idle state */ PWM_17_GTMCCU6_HIGH /* channel polarity */  ),  (Pwm_17_GtmCcu6_PeriodType)0xea60, /*Default Period*/  (uint32)0x1999, /*Default Duty Cycle*/  (uint32)0x0, /*Shift Value*/  (const void*)&amp;Pwm_kChannelConfigGtm_Core0[0] } } </pre>
--	--	--

### 1.2.3.1 Member: PwmChannelId

**Table 2 PwmChannelId**

<b>Name</b>	PwmChannelId	
<b>Type</b>	Pwm_17_GtmCcu6_ChannelType	
<b>Description</b>	Indicates the channel ID of a PWM channel	
<b>Verification method</b>	The structure member is generated as a numeric value set in the configuration parameter 'PwmChannelConfigSet/PwmChannel/PwmChannel_<x>/PwmChannelId'. (x is the configured channel number)	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure an PWM channel with PwmChannelId = 7 in PwmChannelConfigSet/PwmChannel/PwmChannel_1/PwmChannelId'	7

### 1.2.3.2 Member: PwmTimerUsed

**Table 3 PwmTimerUsed**

<b>Name</b>	PwmTimerUsed	
<b>Type</b>	uint8	
<b>Description</b>	Hardware type selected for the PWM channel.	
<b>Verification method</b>	The structure member is generated as the hardware type selected in PwmAssignedHwUnit	



Example(s)	Action	Generated output
	Configure a PWM channel with PwmAssignedHwUnit = GTM in Pwm/PwmChannelConfigSet /PwmChannel/PwmChannel_1	0
	Configure a PWM channel with PwmAssignedHwUnit = CCU6 in in Pwm/PwmChannelConfigSet /PwmChannel/PwmChannel_2	1

### 1.2.3.3 Member: PwmNotification

**Table 4 PwmNotification**

<b>Name</b>	PwmNotification	
<b>Type</b>	Pwm_17_GtmCcu6_NotifiPtrType	
<b>Description</b>	Pointer to the callback functions configured by the user.	
<b>Verification method</b>	This structure member holds the address of the notification function.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure notification for Channel 7 with 'NotifyFunction' in Pwm/PwmChannelConfigSet /PwmChannel/PwmChannel_7/PwmNotification  The output will show the address of the notification function 'NotifyFunction'	<pre>#if (PWM_17_GTMCCU6_NOTIFICATION_SUPPORTED == STD_ON) (Pwm_17_GtmCcu6_NotifiPtrType)&amp;NotifyFunction #endif</pre>
	Configure notification for Channel 5 with '0x800012ab' in Pwm/PwmChannelConfigSet /PwmChannel/PwmChannel_5/PwmNotification	<pre>#if (PWM_17_GTMCCU6_NOTIFICATION_SUPPORTED == STD_ON) (Pwm_17_GtmCcu6_NotifiPtrType) 0x800012ab #endif</pre>

### 1.2.3.4 Member: PwmChannelInfo

**Table 5 PwmChannelInfo**

<b>Name</b>	PwmChannelInfo
<b>Type</b>	uint16
<b>Description</b>	The structure member contains the information about Polarity, Idlestate, Coherency, Class, Shift reset and DsadcNotif value for a PWM channel.

<b>Verification method</b>	<p>The structure member is generated as a value:</p> <ol style="list-style-type: none"><li>1. Value = 0</li><li>2. Value = Value   PwmPolarity</li><li>3. Value = Value   PwmIdleState &lt;&lt; 1</li><li>4. Value = Value   PwmCoherentUpdate &lt;&lt; 2</li><li>5. Value = Value   PwmChannelClass &lt;&lt; 3</li><li>6. Value = Value   Shift Reset &lt;&lt; 5</li><li>7. Value = Value   DsadcNotif &lt;&lt; 6<ol style="list-style-type: none"><li>a. Shift Reset = 0 if (PwmHandleShiftByOffset = 'False') &amp;&amp; (PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED or PWM_FIXED_PERIOD_CENTER_ALIGNED)</li><li>b. Shift Reset = 1 if (PwmHandleShiftByOffset = 'False') &amp;&amp; (PwmChannelClass = PWM_FIXED_PERIOD or PWM_VARIABLE_PERIOD)</li><li>c. Shift Reset = 0 if (PwmHandleShiftByOffset = 'True') &amp;&amp; (PwmChannelClass = PWM_FIXED_PERIOD or PWM_VARIABLE_PERIOD)</li><li>d. Shift Reset = 1 if (PwmHandleShiftByOffset = 'True') &amp;&amp; (PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED or PWM_FIXED_PERIOD_CENTER_ALIGNED)</li></ol></li></ol> <p>Note: The configuration parameters PwmPolarity, PwmIdleState, PwmCoherentUpdate and PwmChannelClass are present in container 'PwmChannelConfigSet/PwmChannel/PwmChannel_&lt;x&gt;'. (x is the configured channel number)</p> <p>Note: The configuration parameter PwmCoherentUpdate is available only when configuration parameter PwmChannelCoherentSelection is set to 'True'.</p> <p>Note: DsadcNotif is set/reset by configuration parameter McuAtomChannelEventHandledByDsadc/McuTomChannelEventHandledByDsadc for ATOM/TOM channels respectively. It is available only when configuration parameter PwmNotificationSupported is set to 'True' and Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_x/PwmAssignedHwUnit = GTM</p>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure a PWM channel with PwmPolarity = PWM_HIGH, PwmIdleState = PWM_LOW, PwmCoherentUpdate = 'True', PwmChannelClass = PWM_FIXED_PERIOD, PwmHandleShiftByOffset = 'True' node:ref(/GtmTimerUsed)/McuAtomChannelEvent	Pwm_1ConfigChannel( 1U, /* The notification flag to enable GTM interrupts to trigger DSADC */ 0U, /* Channel reset from other channel or not*/ PWM_17_GTMCCU6_FIXED_PERIOD, /* Channel Class */ PWM_17_GTMCCU6_COHERENT, /* Coherency */ PWM_17_GTMCCU6_LOW, /* channel Idle state */ PWM 17 GTMCCU6 HIGH /* channel polarity

	<p>HandledByDsadc = 'True'</p> <p>PwmAssignedHwUnit = GTM</p>	<pre>*/ )</pre>
	<p>Configure a PWM channel with PwmPolarity = PWM_HIGH, PwmIdleState = PWM_LOW, PwmCoherentUpdate = 'True', PwmChannelClass = PWM_FIXED_PERIOD, PwmHandleShiftByOffset = 'True'</p> <p>PwmAssignedHwUnit = CCU6</p>	<pre>Pwm_1ConfigChannel( 0U, /* The notification flag to enable GTM interrupts to trigger DSADC */ 0U, /* Channel reset from other channel or not*/ PWM_17_GTMCCU6_FIXED_PERIOD, /* Channel Class */ PWM_17_GTMCCU6_COHERENT, /* Coherency */ PWM_17_GTMCCU6_LOW, /* channel Idle state */ PWM_17_GTMCCU6_HIGH /* channel polarity */ )</pre>
	<p>Configure a PWM channel with PwmPolarity = PWM_HIGH, PwmIdleState = PWM_LOW, PwmCoherentUpdate = 'True', PwmChannelClass = PWM_FIXED_PERIOD, PwmHandleShiftByOffset = 'True'</p> <p>PwmAssignedHwUnit = CCU6</p>	<pre>Pwm_1ConfigChannel( 0U, /* The notification flag to enable GTM interrupts to trigger DSADC */ 0U, /* Channel reset from other channel or not*/ PWM_17_GTMCCU6_FIXED_PERIOD, /* Channel Class */ PWM_17_GTMCCU6_COHERENT, /* Coherency */ PWM_17_GTMCCU6_LOW, /* channel Idle state */ PWM_17_GTMCCU6_HIGH /* channel polarity */ )</pre>
	<p>Configure a PWM channel with PwmPolarity = PWM_HIGH, PwmIdleState = PWM_HIGH, PwmCoherentUpdate = 'True', PwmChannelClass = PWM_VARIABLE_PERIOD, PwmHandleShiftByOffset = 'False'</p> <p>node:ref(/GtmTimerUsed)/McuAtomChannelEvent</p> <p>HandledByDsadc = 'False'</p> <p>PwmAssignedHwUnit = GTM</p>	<pre>Pwm_1ConfigChannel( 0U, /* The notification flag to enable GTM interrupts to trigger DSADC */ 0U, /* Channel reset from other channel or not*/ PWM_17_GTMCCU6_VARIABLE_PERIOD, /* Channel Class */ PWM_17_GTMCCU6_COHERENT, /* Coherency */ PWM_17_GTMCCU6_HIGH, /* channel Idle state */ PWM_17_GTMCCU6_HIGH /* channel polarity */ )</pre>

	Configure a PWM channel with PwmPolarity = PWM_LOW, PwmIdleState = PWM_LOW, PwmCoherentUpdate = 'False', PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED, PwmHandleShiftByOffset = 'True' node:ref(/GtmTimerUsed)/McuTomChannelEventHandledByDsadc = 'True' PwmAssignedHwUnit = GTM	<pre> Pwm_1ConfigChannel( 1U, /* The notification flag to enable GTM interrupts to trigger DSADC */ 1U, /* Channel reset from other channel or not*/ PWM_FIXED_PERIOD_SHIFTED, /* Channel Class */ PWM_17_GTMCCU6_NON_COHERENT, /* Coherency */ PWM_17_GTMCCU6_LOW, /* channel Idle state */ PWM_17_GTMCCU6_LOW /* channel polarity */ ) </pre>
--	--	---

### 1.2.3.5 Member: PwmPeriodDefault

**Table 6** PwmPeriodDefault

<b>Name</b>	PwmPeriodDefault	
<b>Type</b>	Pwm_17_GtmCcu6_PeriodType	
<b>Description</b>	Default period for PWM channel	
<b>Verification method</b>	The structure member is generated as a numeric value set in the configuration parameter 'PwmChannelConfigSet/PwmChannel/PwmChannel_<x>/PwmPeriodDefault'. (x is the configured channel number)  <i>Note: PwmPeriodDefault value is fixed permanently for a PWM channel configured with Fixed Period type.</i>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	<ul style="list-style-type: none"> <li>Configure a PWM channel with PwmPeriodDefault = 60000 in 'PwmChannelConfigSet/PwmChannel/PwmChannel_0/PwmPeriodDefault'</li> </ul>	<pre> (Pwm_17_GtmCcu6_PeriodType) 0xea60, /* Default Period */ </pre>

### 1.2.3.6 Member: PwmDutycycleDefault

**Table 7** PwmDutycycleDefault

<b>Name</b>	PwmDutycycleDefault
<b>Type</b>	uint32
<b>Description</b>	Default duty cycle for PWM channel
<b>Verification</b>	The structure member is generated as a numeric value set in the configuration

<b>method</b>	parameter 'PwmChannelConfigSet/PwmChannel/PwmChannel_<x>/PwmDutycycleDefault'. (x is the configured channel number)  <i>Note: Value of PwmDutycycleDefault is entered as a percentage when configuration parameter PwmDutyShiftInTicks is set to 'False', else the value is entered as absolute ticks.</i>	
<b>Example(s)</b>	<b>Action</b> <ul style="list-style-type: none"> <li>Configure a PWM channel with PwmDutycycleDefault = 6553 in PwmChannelConfigSet/PwmChannel/PwmChannel_0/PwmDutycycleDefault'</li> <li>Parameter PwmDutyShiftInTicks = 'True'</li> </ul>	<b>Generated output</b>  (uint32)0x1999, /* Default Duty Cycle */
	<ul style="list-style-type: none"> <li>Configure a PWM channel with PwmDutycycleDefault = 6553 in PwmChannelConfigSet/PwmChannel/PwmChannel_0/PwmDutycycleDefault'</li> <li>Parameter PwmDutyShiftInTicks = 'False'</li> </ul>	(uint32)0x2ede, /* Default Duty Cycle */

### 1.2.3.7 Member: PwmShiftValue

**Table 8 PwmShiftValue**

<b>Name</b>	PwmShiftValue	
<b>Type</b>	uint32	
<b>Description</b>	Shift of PWM_FIXED_PERIOD_SHIFTED channel with respect to reference channel.	
<b>Verification method</b>	The structure member is generated as a numeric value set in the configuration parameter 'PwmChannelConfigSet/PwmChannel/PwmChannel_<x>/ PwmShiftValue'. (x is the configured channel number)  <i>Note: Value of PwmShiftValue is entered as a percentage when configuration parameter PwmDutyShiftInTicks is set to 'False', else the value is entered as absolute ticks.</i>  <i>Note: Configuration parameter PwmShiftValue is available only when PwmChannelClass is PWM_FIXED_PERIOD_SHIFTED.</i>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>

	<ul style="list-style-type: none"> <li>Configure a PWM channel with PwmShiftValue = 20 in PwmChannelConfigSet/PwmChannel/PwmChannel_0/ PwmShiftValue</li> <li>Parameter PwmDutyShiftInTicks = 'True'</li> </ul>	(uint32)0x14, /* Shift Value*/
	<ul style="list-style-type: none"> <li>Configure a PWM channel with PwmShiftValue = 20 in PwmChannelConfigSet/PwmChannel/PwmChannel_0/ PwmShiftValue</li> <li>Parameter PwmDutyShiftInTicks = 'False'</li> </ul>	(uint32)0x24, /* Shift Value*/

### 1.2.3.8 Member: PwmTimerPtr

**Table 9** PwmTimerPtr

<b>Name</b>	PwmTimerPtr	
<b>Type</b>	const void*	
<b>Description</b>	Pointer to GTM/CCU6 timer channel information.	
<b>Verification method</b>	The structure member is generated with pointer to a structure which holds the timer specific information for a channel.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure a PWM channel in Core2 with PwmAssignedHwUnit = GTM in Pwm/PwmChannelConfigSet/PwmChannel	(const void*)&Pwm_kChannelConfigGtm_Core2[2]
	Configure a PWM channel in Core1 with PwmAssignedHwUnit = CCU6 in in Pwm/PwmChannelConfigSet/PwmChannel	(const void*)&Pwm_kChannelConfigCcu6_Core1[0]

### 1.2.4 Array: Pwm\_ChannelIndexMap[\_<variant>][PWM\_17\_GTMCCU6\_MAX\_CHANNELS]

**Table 10** Pwm\_ChannelIndexMap[\_<variant>][ PWM\_17\_GTMCCU6\_MAX\_CHANNELS]

<b>Name</b>	Pwm_ChannelIndexMap[_<variant>][ PWM_17_GTMCCU6_MAX_CHANNELS]
-------------	---

<b>Type</b>	Pwm_17_GtmCcu6_ChannelConfigType	
<b>Description</b>	This array holds index of each configured channel within the allocated core.	
<b>Verification method</b>	Pwm_ChannelIndexMap[<x>] = Index of (PwmChannelId = <x>) in the allocated core. <Variant> indicates the name of the post-build variant. For a variant aware configuration the structure name is appended with the variant name. For variant unaware configuration <variant> is ignored.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	<ul style="list-style-type: none"> <li>Configure 5 PWM channels in Pwm/PwmChannelConfigSet/PwmChannel (PwmChannel_0 to PwmChannel_4).</li> <li>Assign PwmChannel_0, PwmChannel_3 and PwmChannel_4 in CORE4 with PwmChannelId = 0, 3, 4 respectively</li> <li>Assign PwmChannel_1, PwmChannel_2 in CORE2 with PwmChannelId = 1, 2 respectively</li> </ul>	<pre>static const uint8 Pwm_ChannelIndexMap[5] = {     0x0U,     0x0U,     0x1U,     0x1U,     0x2U };</pre>
	<ul style="list-style-type: none"> <li>Configure 5 PWM channels in Pwm/PwmChannelConfigSet/PwmChannel (PwmChannel_0 to PwmChannel_4).</li> <li>Assign PwmChannel_0, PwmChannel_3 and PwmChannel_4 in CORE4 with PwmChannelId = 0, 1, 2 respectively</li> <li>Assign PwmChannel_1, PwmChannel_2 in CORE2 with PwmChannelId = 3, 4 respectively</li> </ul>	<pre>static const uint8 Pwm_ChannelIndexMap[5] = {     0x0U,     0x1U,     0x2U,     0x0U,     0x1U };</pre>
	<ul style="list-style-type: none"> <li>Configure 9 PWM channels in Pwm/PwmChannelConfigSet/PwmChannel (PwmChannel_0 to PwmChannel_8).</li> <li>Assign PwmChannel_0, PwmChannel_1 and PwmChannel_2 in CORE4 with PwmChannelId = 0, 1</li> </ul>	<pre>static const uint8 Pwm_ChannelIndexMap_Petrol[9] = {     0x0U,     0x1U,     0x2U,     0x0U,     0x1U,</pre>

	,2 respectively • Assign PwmChannel_3, PwmChannel_4 in CORE2 with PwmChannelId = 4 ,3 respectively • Assign PwmChannel_5, PwmChannel_6 PwmChannel_7 and PwmChannel_8 in CORE5 with PwmChannelId = 8 ,7,6,5 respectively • This configuration is variant-aware. Variant name is 'Petrol'	0x0U, 0x1U, 0x2U, 0x3U };
--	--	---------------------------------------

### 1.2.5 Structure: Pwm\_kChannelConfigGtm\_Core<x> [\_<variant>][<y>]

**Table 11** Pwm\_kChannelConfigGtm\_Core<x> [\_<variant>][<y>]

<b>Name</b>	Pwm_kChannelConfigGtm_Core<x> [_<variant>][<y>]
<b>Type</b>	Mcu_17_Gtm_TomAtomChConfigType
<b>Description</b>	The configuration data of the all PWM GTM channels belonging to a Core
<b>Verification method</b>	<p>Configuration structure holds the configuration data of the &lt;y&gt; PWM GTM channels belonging to Core &lt;x&gt;.</p> <p>&lt;x&gt; ranges from 0 to 5.</p> <p>&lt;y&gt; is the total number of GTM channels configured for Core&lt;x&gt;.</p> <p>&lt;Variant&gt; indicates the name of the post-build variant. For a variant aware configuration the structure name is appended with the variant name. For variant unaware configuration &lt;variant&gt; is ignored.</p> <p><i>Note: The generated file has this structure if at least one GTM channel is assigned to Core &lt;x&gt;.</i></p>

Example(s)	Action	Generated output
	<ul style="list-style-type: none"> <li>Configure 5 PWM channel to Core0 in ResourceM/ResourceMMcalConfig/ResourceMMcalConfig_0/ResourceMMcalCore_0/ResourceMAllocation</li> <li>Configure 3 PWM channel with PwmAssignedHwUnit = CCU6 in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0</li> <li>Configure 2 PWM channel</li> </ul>	<pre>static const Mcu_17_Gtm_TomAtomChConfigType Pwm_kChannelConfigGtm_Core0_Petrol[2] = {     {         MCU_GTM_TIMER_ATOM, /* Timer Type         (TOM/ATOM) */         /* Bit[15:8] - Module number         Bit[7:0] - Channel number*/         0x4U,         0x103002U, /* Channel Control         Register*/         0x0U, /* CN0 in ticks */     } }</pre>



	with PwmAssignedHwUnit = GTM in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0 <ul style="list-style-type: none"> <li>This configuration is variant-aware. Variant name is 'Petrol'</li> </ul>	<pre> 0x1770U, /* CM0 in ticks */ 0x2710U, /* CM1 in ticks */ 0x1770U, /* SR0 in ticks */ 0x2710U, /* SR1 in ticks */ 0x80U /* Period, Duty Interrupt and mode*/ }, {     MCU_GTM_TIMER_TOM, /* Timer Type (TOM/ATOM) */     /* Bit[15:8] - Module number     Bit[7:0] - Channel number*/     0x50fU,     0x0U, /* Channel Control Register*/     0x0U, /* CN0 in ticks */     0x0U, /* CM0 in ticks */     0x0U, /* CM1 in ticks */     0x0U, /* SR0 in ticks */     0x0U, /* SR1 in ticks */     0x80U /* Period, Duty Interrupt and mode*/ }, }; </pre>
--	---	--

### 1.2.5.1 Member: TimerType

**Table 12** TimerType

<b>Name</b>	TimerType	
<b>Type</b>	Mcu_17_Gtm_TimerOutType	
<b>Description</b>	TOM/ATOM timer type	
<b>Verification method</b>	The structure member is generated as the GtmTimerUsed selected in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_<x>/GtmTimerOutputModuleConfiguration/GtmTimerOutputModuleConfiguration_0	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure a PWM channel with GtmTimerUsed = '/Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuGtmAllocationConf_0/McuGtmAtomAllocationConf_0/McuGtmAtomChannelAllocationConf_1' in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/GtmTimerOutputModuleC	MCU_GTM_TIMER_ATOM

	onfiguration/GtmTimerOutputModuleConfiguration_0	
	Configure a PWM channel with GtmTimerUsed = '/Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuGtmAllocationConf_0/McuGtmTomAllocationConf_3/McuGtmTomChannelAllocationConf_10' in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/GtmTimerOutputModuleConfiguration/GtmTimerOutputModuleConfiguration_0	MCU_GTM_TIMER_TOM

### 1.2.5.2 Member: TimerId

**Table 13** TimerId

<b>Name</b>	TimerId	
<b>Type</b>	Mcu_17_Gtm_TimerChIdentifierType	
<b>Description</b>	Consists of TOM or ATOM module ID and channel ID	
<b>Verification method</b>	The structure member is generated as a value: <ol style="list-style-type: none"> <li>1. Value = 0</li> <li>2. Value = Value   Channel number</li> <li>3. Value = Value   Module number &lt;&lt; 8</li> </ol>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure a PWM channel with GtmTimerUsed = '/Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuGtmAllocationConf_0/McuGtmAtomAllocationConf_0/McuGtmAtomChannelAllocationConf_4' in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/GtmTimerOutputModuleConfiguration/GtmTimerOutputModuleConfiguration_0	0x4U
	Configure a PWM channel with GtmTimerUsed = '/Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuGtmAllocationConf_0/McuGtmTomAllocationConf_3/McuGtmTomChannelAllocationConf_10' in	0x030AU

Pwm/PwmChannelConfigSet  
/PwmChannel/PwmChannel  
\_0/GtmTimerOutputModuleC  
onfiguration/GtmTimerOutp  
utModuleConfiguration\_0

### 1.2.5.3 Member: TimerChCtrlReg

**Table 14** TimerChCtrlReg

<b>Name</b>	TimerChCtrlReg
<b>Type</b>	uint32
<b>Description</b>	TOM/ATOM channel control registers value
<b>Verification method</b>	<p>Channel control register is generated in this structure member as a value:</p> <ol style="list-style-type: none"> <li>Value = 0</li> <li>Value = Value   Mode If GtmTimerUsed = ATOM Mode = 10 (SOMP mode) else Mode = 00</li> <li>Value = Value   Polarity &lt;&lt; 11 If PwmPolarity = 'PWM_HIGH' Polarity = 1 else Polarity = 0</li> <li>Value = Value   CLK_SRC_SR &lt;&lt; 12 The configuration parameter GtmTimerClockSelect value from GTM_CONFIGURABLE_CLOCK_0 to 7 corresponds to 000 to 111 values</li> <li>Value = Value   RST_CCU0 &lt;&lt; 20 If ((PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED &amp;&amp; PwmHandleShiftByOffset = 'False') or (PwmChannelClass = PWM_FIXED_PERIOD_CENTER_ALLIGNED)) RST_CCU0 = 1 Else RST_CCU0 = 0</li> <li>Value = Value   TrigOut &lt;&lt; 24 If (PwmChannelClass of Channel A = PWM_FIXED_PERIOD) &amp;&amp; (PwmHandleShiftByOffset = 'False') &amp;&amp; (PwmChannelClass of Channel B = PWM_FIXED_PERIOD_SHIFTED) &amp;&amp; (PwmReferenceChannel of Channel B = Channel A) TrigOut = 1</li> </ol>

## Pwm\_17\_GtmCcu6 driver

	<p>else if ( PwmChannelClass of Channel A= PWM_FIXED_PERIOD) &amp;&amp;  (PwmChannelClass of Channel B = PWM_FIXED_PERIOD_CENTER_ALLIGNED &amp;&amp;  PwmReferenceChannel of Channel B = Channel A)</p> <p>TrigOut = 1</p> <p>else</p> <p>TrigOut = 0</p>	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> <li>Configure a Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_1/GtmTimerOutputModuleConfiguration/GtmTimerOutputModuleConfiguration_0 with</li> </ul> <ol style="list-style-type: none"> <li>GtmTimerUsed = /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuGtmAllocationConf_0/McuGtmAtomAllocationConf_0/McuGtmAtomChannelAllocationConf_0</li> <li>GtmTimerClockSelect = GTM_CONFIGURABLE_CLOCK_0=1</li> </ol> <ul style="list-style-type: none"> <li>Configure channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_1 with</li> </ul> <ol style="list-style-type: none"> <li>PwmPolarity = PWM_LOW</li> <li>PwmChannelClass = PWM_FIXED_PERIOD</li> </ol> <ul style="list-style-type: none"> <li>Configure parameter PwmHandleShiftByOffset = True</li> </ul>	0x81000002U
	<ul style="list-style-type: none"> <li>Configure a Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/GtmTimerOutputModuleConfiguration/GtmTimerOutputModuleConfiguration_0 with</li> </ul> <ol style="list-style-type: none"> <li>GtmTimerUsed = /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuGtmAllocationConf_0/McuGtmAtomAllocationC</li> </ol>	0x81000802U

	<p>onf_0/McuGtmAtomChannelAllocationConf_0</p> <ol style="list-style-type: none"> <li>2. GtmTimerClockSelect = GTM_CONFIGURABLE_CLOCK_0</li> <li>• Configure channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0 with</li> <li>1. PwmPolarity = PWM_HIGH</li> <li>2. PwmChannelClass = PWM_FIXED_PERIOD</li> <li>• Configure parameter PwmHandleShiftByOffset = True</li> </ol> <p>This channel is used as a reference for the shifted period channel in the below example.</p>	
	<ol style="list-style-type: none"> <li>• Configure a Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_2/GtmTimerOutputModuleConfiguration/GtmTimerOutputModuleConfiguration_0 with</li> <li>3. GtmTimerUsed = /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuGtmAllocationConf_0/McuGtmAtomAllocationConf_0/McuGtmAtomChannelAllocationConf_2</li> <li>4. GtmTimerClockSelect = GTM_CONFIGURABLE_CLOCK_2</li> <li>• Configure channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_2 with</li> <li>5. PwmPolarity = PWM_LOW</li> <li>6. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED</li> <li>• Configure parameter PwmHandleShiftByOffset</li> </ol>	0x80002002U

	<p>= True</p> <p>The Fixed period channel in the example above is taken as a reference channel for this shifted channel.</p>	
	<ul style="list-style-type: none"> <li>Configure a Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_2/GtmTimerOutputModuleConfiguration/GtmTimerOutputModuleConfiguration_0 with               <ol style="list-style-type: none"> <li>GtmTimerUsed = /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuGtmAllocationConf_0/McuGtmAtomAllocationConf_0/McuGtmAtomChannelAllocationConf_2</li> <li>GtmTimerClockSelect = GTM_CONFIGURABLE_CLOCK_2</li> </ol> </li> <li>Configure channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_2 with               <ol style="list-style-type: none"> <li>PwmPolarity = PWM_LOW</li> <li>PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED</li> </ol> </li> <li>Configure parameter PwmHandleShiftByOffset = False</li> </ul> <p>The Fixed period channel in the example above is taken as a reference channel for this shifted channel.</p>	0x80102002U

#### 1.2.5.4 Member: TimerChCN0Reg

**Table 15** TimerChCN0Reg

Name	TimerChCN0Reg
------	---------------

<b>Type</b>	uint32	
<b>Description</b>	TOM/ATOM channel CN0 register value	
<b>Verification method</b>	<p>The structure member holds the counter value of the channel.</p> <pre> if (PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED)&amp;&amp; (PwmHandleShiftByOffset = 'True')     CN0 Value = PwmPeriodDefault - PwmShiftValue if (PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED)&amp;&amp; (PwmHandleShiftByOffset = 'False')     CN0 Value = 0 Else if (PwmChannelClass = PWM_FIXED_PERIOD_CENTER_ALLIGNED)     CN0 Value = 0 Else     CN0 Value = PwmPeriodDefault </pre>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	<ul style="list-style-type: none"> <li>Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0 with <ol style="list-style-type: none"> <li>PwmAssignedHwUnit = GTM</li> <li>PwmPeriodDefault = 60000</li> <li>PwmChannelClass = PWM_FIXED_PERIOD</li> </ol> </li> </ul>	0xea60U
	<ul style="list-style-type: none"> <li>Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with <ol style="list-style-type: none"> <li>PwmAssignedHwUnit = GTM</li> <li>PwmPeriodDefault = 60000(of the referenced channel)</li> <li>PwmShiftValue = 70</li> <li>PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED</li> </ol> </li> <li>PwmHandleShiftByOffset = 'False'</li> </ul>	0x0U
	<ul style="list-style-type: none"> <li>Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with <ol style="list-style-type: none"> <li>PwmAssignedHwUnit = GTM</li> <li>PwmPeriodDefault =</li> </ol> </li> </ul>	0xea1aU

	60000 (of the referenced channel) 7. PwmShiftValue = 70 8. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED • PwmHandleShiftByOffset = 'True'	
--	--	--

### 1.2.5.5 Member: TimerChCM0Reg

**Table 16** TimerChCM0Reg

<b>Name</b>	TimerChCM0Reg	
<b>Type</b>	uint32	
<b>Description</b>	TOM/ATOM channel CM0 register value	
<b>Verification method</b>	<p>The structure member holds the period match value of the channel.</p> <pre> If (PwmChannelClass = PWM_FIXED_PERIOD or PWM_VARIABLE_PERIOD)     CM0 Value = PwmPeriodDefault Else     If((PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED)&amp;&amp; (PwmHandleShiftByOffset = 'False')) or (PwmChannelClass = PWM_FIXED_PERIOD_CENTER_ALLIGNED)         If(PwmDutycycleDefault = 0)             CM0 Value= 0xFFFF (If GtmTimerUsed = TOM)             CM0 Value= 0xFFFF (If GtmTimerUsed = ATOM)         else if (PwmDutycycleDefault = max)             CM0 Value=0         else             CM0 Value= PwmShiftValue         endif     else         CM0 Value= PwmPeriodDefault     Endif Endif </pre>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	<ul style="list-style-type: none"> <li>Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0 with</li> </ul> <ol style="list-style-type: none"> <li>PwmAssignedHwUnit = GTM</li> <li>PwmDutycycleDefault = 6553</li> <li>PwmPeriodDefault = 60000</li> <li>PwmChannelClass =</li> </ol>	0xea60U



	<p>PWM_FIXED_PERIOD</p> <ul style="list-style-type: none"> <li>Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0 with               <ol style="list-style-type: none"> <li>PwmAssignedHwUnit = GTM</li> <li>PwmDutycycleDefault = 1677215</li> <li>PwmPeriodDefault (of the referenced channel)= 60000</li> <li>PwmShiftValue = 70</li> <li>PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED</li> </ol> </li> <li>PwmHandleShiftByOffset = 'False'</li> </ul>	0x0U
	<ul style="list-style-type: none"> <li>Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with               <ol style="list-style-type: none"> <li>PwmAssignedHwUnit = GTM</li> <li>PwmDutycycleDefault = 0</li> <li>PwmPeriodDefault (of the referenced channel)= 60000</li> <li>PwmShiftValue = 70</li> <li>Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2/GtmTimerOutput ModuleConfiguration/GtmTimerOutputModuleConfiguration_0/GtmTimerUsed GtmTimerUsed = /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuGtmAllocationConf_0/McuGtmAtomAllocationConf_0/McuGtmAtomChannelAllocationConf_2</li> <li>PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED</li> </ol> </li> <li>PwmHandleShiftByOffset = 'False'</li> </ul>	0xFFFFFFFFU

	<ul style="list-style-type: none"> <li>Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0 with</li> </ul> <ol style="list-style-type: none"> <li>PwmAssignedHwUnit = GTM</li> <li>PwmDutycycleDefault = 6777</li> <li>PwmPeriodDefault (of the referenced channel)= 60000</li> <li>PwmShiftValue = 70</li> <li>PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED</li> </ol> <ul style="list-style-type: none"> <li>PwmHandleShiftByOffset = 'True'</li> </ul>	0xea60U
--	---	---------

### 1.2.5.6 Member: TimerChCM1Reg

**Table 17** TimerChCM1Reg

<b>Name</b>	TimerChCM1Reg	
<b>Type</b>	uint32	
<b>Description</b>	TOM/ATOM channel CM1 register value	
<b>Verification method</b>	<p>The structure member holds the ScaledDuty as the duty match value of the channel.</p> <pre> If (PwmChannelClass = PWM_FIXED_PERIOD or PWM_VARIABLE_PERIOD)     CM1 Value= ScaledDuty else     If((PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED)&amp;&amp; (PwmHandleShiftByOffset = 'False')) or (PwmChannelClass = PWM_FIXED_PERIOD_CENTER_ALLIGNED)         If(PwmDutycycleDefault = 0)             CM1 Value = 0         else if (PwmDutycycleDefault = max)             CM1 Value= 0xFFFF (If GtmTimerUsed = TOM)             CM1 Value= 0xFFFFF (If GtmTimerUsed = ATOM)         else             CM1 Value= (PwmShiftValue + ScaledDuty ) % PwmDutycycleDefault         endif     else         CM1 Value= ScaledDuty     Endif Endif </pre>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	<ul style="list-style-type: none"> <li>Configure channel in</li> </ul>	0x1999U

	Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0 with 1. PwmAssignedHwUnit = GTM 2. PwmDutycycleDefault = 6553 3. PwmPeriodDefault = 60000 4. PwmChannelClass = PWM_FIXED_PERIOD	
	<ul style="list-style-type: none"> <li>Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0 with</li> </ul> 1. PwmAssignedHwUnit = GTM 2. PwmDutycycleDefault = 1677215 3. PwmPeriodDefault (of the referenced channel)= 60000 4. Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0/GtmTimerOutput ModuleConfiguration/GtmTimerOutputModuleConfiguration_0/GtmTimerUsed GtmTimerUsed = /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuGtmAllocationConf_0/McuGtmAtomAllocationConf_0/McuGtmAtomChannelAllocationConf_2 5. PwmShiftValue = 70 6. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED <ul style="list-style-type: none"> <li>PwmHandleShiftByOffset = 'False'</li> </ul>	0xFFFFFU
	<ul style="list-style-type: none"> <li>Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with</li> </ul> 1. PwmAssignedHwUnit = GTM	0x0U

	<ol style="list-style-type: none"> <li>2. PwmDutycycleDefault = 0</li> <li>3. PwmPeriodDefault (of the referenced channel)= 60000</li> <li>4. PwmShiftValue = 70</li> <li>5. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED</li> </ol> <ul style="list-style-type: none"> <li>• PwmHandleShiftByOffset = 'False'</li> </ul>	
	<ul style="list-style-type: none"> <li>• Configure channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0 with</li> </ul> <ol style="list-style-type: none"> <li>1. PwmAssignedHwUnit = GTM</li> <li>2. PwmDutycycleDefault = 6777</li> <li>3. PwmPeriodDefault (of the referenced channel)= 60000</li> <li>4. PwmShiftValue = 70</li> <li>5. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED</li> </ol> <ul style="list-style-type: none"> <li>• PwmHandleShiftByOffset = 'True'</li> </ul>	0x0U

### 1.2.5.7 Member: TimerChSR0Reg

**Table 18** TimerChSR0Reg

<b>Name</b>	TimerChSR0Reg	
<b>Type</b>	uint32	
<b>Description</b>	TOM/ATOM channel SR0 register value	
<b>Verification method</b>	<p>The structure member is generated as the same value as structure member TimerChCM0Reg.</p> <p><i>Note: This structure member is not configurable by user.</i></p>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure a PWM channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_2 with PwmAssignedHwUnit = GTM for the generated value of TimerChCM0Reg = 0xea60U	0xea60U

### 1.2.5.8 Member: TimerChSR1Reg

**Table 19** TimerChSR1Reg

<b>Name</b>	TimerChSR1Reg	
<b>Type</b>	uint32	
<b>Description</b>	TOM/ATOM channel SR1 register value	
<b>Verification method</b>	<p>The structure member is generated as the same value as structure member TimerChCM1Reg.</p> <p><i>Note: This structure member is not configurable by user.</i></p>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure a PWM channel in Pwm/PwmChannelConfigSet /PwmChannel/PwmChannel_2 with PwmAssignedHwUnit = GTM for the generated value of TimerChCM1Reg = 0x1999U	0x1999U

### 1.2.5.9 Member: TimerChIntEnMode

**Table 20** TimerChIntEnMode

<b>Name</b>	TimerChIntEnMode	
<b>Type</b>	uint8	
<b>Description</b>	This structure member consists of TOM/ATOM channel interrupt mode and enables/disables the CCU0-CCU1 interrupts	
<b>Verification method</b>	<p>The structure member is generated as a value:</p> <ol style="list-style-type: none"> <li>Value = 0</li> <li>Value = Value   CCU0 interrupt enable/disable <ol style="list-style-type: none"> <li>CCU0 interrupt enable = 1</li> <li>CCU0 interrupt disable = 0</li> </ol> </li> <li>Value = Value   CCU1 interrupt enable/disable &lt;&lt; 1 <ol style="list-style-type: none"> <li>CCU1 interrupt enable = 1</li> <li>CCU1 interrupt disable = 0</li> </ol> </li> <li>Value = Value   Interrupt Mode &lt;&lt; 7 <ol style="list-style-type: none"> <li>00-Level Mode</li> <li>01-Pulse Mode</li> <li>10- Pulse Notify Mode</li> <li>11- Single Pulse Mode</li> </ol> </li> </ol> <p><i>Note: This structure member is not user configurable. By default CCU0 and CCU1 interrupts are disabled and the interrupt mode is set to Pulse Notify Mode.</i></p>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>

	Configure a PWM channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_2 with PwmAssignedHwUnit = GTM	0x80U
--	---	-------

### 1.2.6 Structure: Pwm\_kChannelConfigCcu6\_Core<x> [\_<variant>][<y>]

**Table 21** Pwm\_kChannelConfigCcu6\_Core<x>[\_<variant>][<y>]

Name	Pwm_kChannelConfigCcu6_Core<x>[_<variant>][<y>]				
Type	Mcu_17_Ccu6_TimerConfigType				
Description	The configuration data of the all PWM CCU6 channels belonging to a Core				
Verification method	<p>Configuration structure holds the configuration data of the &lt;y&gt; PWM CCU6 channels belonging to Core &lt;x&gt;.</p> <p>&lt;x&gt; ranges from 0 to 5.</p> <p>&lt;y&gt; is the total number of CCU6 channels configured for Core&lt;x&gt;.</p> <p>&lt;Variant&gt; indicates the name of the post-build variant. For a variant aware configuration the structure name is appended with the variant name. For variant unaware configuration &lt;variant&gt; is ignored.</p> <p>Note: The generated file has this structure if at least one CCU6 channel is assigned to Core &lt;x&gt;.</p>				
Example(s)	<table border="1"> <thead> <tr> <th>Action</th><th>Generated output</th></tr> </thead> <tbody> <tr> <td> <ul style="list-style-type: none"> <li>Configure 5 PWM channel to Core2 in ResourceM/ResourceMMcalConfig/ResourceMMcalCore/ResourceMMcalCore_&lt;y&gt;/ResourceMAllocation</li> <li>Configure 1 PWM channel with PwmAssignedHwUnit = CCU6 in in Pwm/PwmChannelConfigSet/PwmChannel</li> <li>Configure 4 PWM channel with PwmAssignedHwUnit = GTM in in Pwm/PwmChannelConfigSet/PwmChannel/</li> <li>This configuration is variant-aware. Variant name is 'Petrol'</li> </ul> </td><td> <pre>static const Mcu_17_Ccu6_TimerConfigType Pwm_kChannelConfigCcu6_Core2_Petrol[1] = {     {         0x10001U, /* Ccu6 Timer ID */         /* T12 - [2-0] - Timer T12 Input         Clock Select                     [3]   - Timer T12 Prescaler         Bit                     [7]   - T12 Operating Mode         T13 - [10-8] - Timer T13 Input         Clock Select                     [11]  - Timer T13         Prescaler Bit */         0xcU,         /* T12 - [1-0] - Timer T12         modulation enable         T13 - [2]   - Enable Compare         Timer T13 output CC63*/         0x1U,         /* T12 - [1-0] - Compare outputs         passive state level</pre> </td></tr> </tbody> </table>	Action	Generated output	<ul style="list-style-type: none"> <li>Configure 5 PWM channel to Core2 in ResourceM/ResourceMMcalConfig/ResourceMMcalCore/ResourceMMcalCore_&lt;y&gt;/ResourceMAllocation</li> <li>Configure 1 PWM channel with PwmAssignedHwUnit = CCU6 in in Pwm/PwmChannelConfigSet/PwmChannel</li> <li>Configure 4 PWM channel with PwmAssignedHwUnit = GTM in in Pwm/PwmChannelConfigSet/PwmChannel/</li> <li>This configuration is variant-aware. Variant name is 'Petrol'</li> </ul>	<pre>static const Mcu_17_Ccu6_TimerConfigType Pwm_kChannelConfigCcu6_Core2_Petrol[1] = {     {         0x10001U, /* Ccu6 Timer ID */         /* T12 - [2-0] - Timer T12 Input         Clock Select                     [3]   - Timer T12 Prescaler         Bit                     [7]   - T12 Operating Mode         T13 - [10-8] - Timer T13 Input         Clock Select                     [11]  - Timer T13         Prescaler Bit */         0xcU,         /* T12 - [1-0] - Timer T12         modulation enable         T13 - [2]   - Enable Compare         Timer T13 output CC63*/         0x1U,         /* T12 - [1-0] - Compare outputs         passive state level</pre>
Action	Generated output				
<ul style="list-style-type: none"> <li>Configure 5 PWM channel to Core2 in ResourceM/ResourceMMcalConfig/ResourceMMcalCore/ResourceMMcalCore_&lt;y&gt;/ResourceMAllocation</li> <li>Configure 1 PWM channel with PwmAssignedHwUnit = CCU6 in in Pwm/PwmChannelConfigSet/PwmChannel</li> <li>Configure 4 PWM channel with PwmAssignedHwUnit = GTM in in Pwm/PwmChannelConfigSet/PwmChannel/</li> <li>This configuration is variant-aware. Variant name is 'Petrol'</li> </ul>	<pre>static const Mcu_17_Ccu6_TimerConfigType Pwm_kChannelConfigCcu6_Core2_Petrol[1] = {     {         0x10001U, /* Ccu6 Timer ID */         /* T12 - [2-0] - Timer T12 Input         Clock Select                     [3]   - Timer T12 Prescaler         Bit                     [7]   - T12 Operating Mode         T13 - [10-8] - Timer T13 Input         Clock Select                     [11]  - Timer T13         Prescaler Bit */         0xcU,         /* T12 - [1-0] - Timer T12         modulation enable         T13 - [2]   - Enable Compare         Timer T13 output CC63*/         0x1U,         /* T12 - [1-0] - Compare outputs         passive state level</pre>				

		<pre> T13 - [2] - Passive state level of output COUT63*/ 0x1U, 0x0U, /* T12/T13 counter value */ 0x0U, /* Period */ 0x0U, /* Duty */ 0x1U, /* Compare mode */ 0x0U, /* Port in select */ 0x0U, /* Interrupt enable */ 0x5U /* Interrupt node Select */  }, } </pre>
--	--	---

### 1.2.6.1 Member: TimerId

**Table 22** TimerId

<b>Name</b>	TimerId	
<b>Type</b>	Mcu_17_Ccu6_TimerChIdentifierType	
<b>Description</b>	CCU6 timer channel user identifier	
<b>Verification method</b>	<p>The structure member is generated as a value:</p> <ol style="list-style-type: none"> <li>Value = 0</li> <li>Value = Value   CCU6TimerUsed If CCU6TimerUsed = T12 , Timer value = 0 If CCU6TimerUsed = T13 , Timer value = 1</li> <li>Value = Value   CCU6KernelUsed &lt;&lt; 8 The value of Kernel can be 0 or 1 depending upon the configuration parameter CCU6KernelUsed</li> <li>Value = Value   Cc6xChannel &lt;&lt; 16 If Cc6xChannel= Cc60 , Comparator value = 0 If Cc6xChannel= Cc61, Comparator value = 1 If Cc6xChannel= Cc62, Comparator value = 2 If CCU6TimerUsed = T13, Comparator value = 3</li> </ol> <p>Cc6xChannel, CCU6KernelUsed and CCU6TimerUsed can be configured in container Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_&lt;x&gt;/CCU6CC6Configuration/CU6CC6Configuration_0.</p>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure a Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CU6CC6Configuration_0/CCU6KernelUsed with <ul style="list-style-type: none"> <li>CCU6TimerUsed = T12</li> </ul>	0x20000

	<ul style="list-style-type: none"> <li>CCU6KernelUsed = /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuCcu6ModuleAllocationConf_0</li> <li>Cc6xChannel = Cc62</li> </ul>	
	Configure a Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0/CCU6KernelUsed with <ul style="list-style-type: none"> <li>CCU6TimerUsed = T13</li> <li>CCU6KernelUsed = /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuCcu6ModuleAllocationConf_1</li> </ul>	0x30101

### 1.2.6.2 Member: TimerCtrlReg0

**Table 23** TimerCtrlReg0

<b>Name</b>	TimerCtrlReg0
<b>Type</b>	uint32
<b>Description</b>	CCU6 Timer channel control register 0 contents
<b>Verification method</b>	<p>The structure member is generated as a value:</p> <ol style="list-style-type: none"> <li>Value = 0</li> <li>Value = Value   Timer T12 Input Clock Select The configuration parameter Ccu6TimerClockSelect value from CCU6_CONFIGURABLE_CLOCK_0 to 7 corresponds to 000 to 111 values.</li> <li>Value = Value   Timer T12 Prescaler Bit &lt;&lt; 3 If CCU6TimerPrescalerEnabled = 'True', Prescaler bit = 1 If CCU6TimerPrescalerEnabled = 'False', Prescaler bit = 0</li> <li>Value = Value   T12 Operating Mode &lt;&lt; 7 Operating mode is not user configurable. It is by default value is set to 0 as timer always works in edge aligned mode.</li> <li>Value = Value   Timer T13 Input Clock Select &lt;&lt; 8 The configuration parameter Ccu6TimerClockSelect value from CCU6_CONFIGURABLE_CLOCK_0 to 7 corresponds to 000 to 111 values.</li> <li>Value = Value   Timer T13 Prescaler Bit &lt;&lt; 11 If CCU6TimerPrescalerEnabled = 'True', Prescaler bit = 1 If CCU6TimerPrescalerEnabled = 'False', Prescaler bit = 0</li> </ol> <p>Ccu6TimerClockSelect and CCU6TimerPrescalerEnabled can be configured in container Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0.</p>



Example(s)	Action	Generated output
	Configure a Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0/CCU6KernelUsed with <ul style="list-style-type: none"> <li>CCU6TimerClockSelect = CCU6_CONFIGURABLE_CLOCK_1</li> <li>CCU6TimerPrescalerEnabled = 'True'</li> <li>CCU6TimerUsed = T12</li> </ul>	0x9
	Configure a Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0/CCU6KernelUsed with <ul style="list-style-type: none"> <li>CCU6TimerClockSelect = CCU6_CONFIGURABLE_CLOCK_6</li> <li>CCU6TimerPrescalerEnabled = 'False'</li> <li>CCU6TimerUsed = T13</li> </ul>	0x606

### 1.2.6.3 Member: ModCtrlReg

**Table 24** ModCtrlReg

<b>Name</b>	ModCtrlReg	
<b>Type</b>	uint32	
<b>Description</b>	Enables/Disables modulation for T12 and T13 timer	
<b>Verification method</b>	The structure member is generated as a value: <ol style="list-style-type: none"> <li>Value = 0</li> <li>Value = Value   Timer T12 modulation value (x=0,1,2)                Modulation enable for CC6x channel= 01                Modulation disable for CC6x channel= 00                Modulation enable for COUT6x and CC6x channel= 11</li> <li>Value = Value   Timer T13 modulation value &lt;&lt; 2                Modulation enable = 01                Modulation disable = 00</li> </ol>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	<ul style="list-style-type: none"> <li>Configure a Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel/Pwm</li> </ul>	0x1U

	Channel_0/CCU6CC6Configuration/CCU6CC6Configuration_0 with CCU6TimerUsed = T12 <ul style="list-style-type: none"> <li>Configure a Pwm channel in            Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0 with            Cout6xChEnable = FALSE</li> </ul>	
	<ul style="list-style-type: none"> <li>Configure a Pwm channel in            Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0 with            CCU6TimerUsed = T12</li> <li>Configure a Pwm channel in            Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0 with            Cout6xChEnable = TRUE</li> </ul>	0x3U
	<ul style="list-style-type: none"> <li>Configure a Pwm channel in            Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0/CCU6KernelUsed with CCU6TimerUsed = T13</li> </ul>	0x4U

#### 1.2.6.4 Member: PasStateLvlReg

**Table 25** PasStateLvlReg

<b>Name</b>	PasStateLvlReg
<b>Type</b>	uint32
<b>Description</b>	This represents the polarity of the PWM signal
<b>Verification method</b>	The structure member is generated as a value: <ol style="list-style-type: none"> <li>Value = 0</li> <li>Value = Value   Timer T12 polarity (x=0,1,2)             <ul style="list-style-type: none"> <li>Polarity of COUT6x channel LOW, CC6x channel LOW = 00</li> <li>Polarity of COUT6x channel LOW, CC6x channel HIGH = 01</li> </ul> </li> </ol>

	Polarity of COUT6x channel HIGH, CC6x channel LOW = 10 Polarity of COUT6x channel HIGH, CC6x channel HIGH = 11  3. Value = Value   Timer T13 polarity << 2 Polarity HIGH = 01 Polarity LOW = 00	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> <li>Configure a Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0 with CCU6TimerUsed = T12</li> <li>Configure COUT6x channel polarity in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0 with Cout6xChPolarity = PWM_LOW</li> <li>Configure CC6x channel polarity in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0 with PwmPolarity = PWM_HIGH</li> </ul>	0x1U
	<ul style="list-style-type: none"> <li>Configure a Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0 with CCU6TimerUsed = T12</li> <li>Configure COUT6x channel polarity in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0 with Cout6xChPolarity = PWM_HIGH</li> </ul>	0x2U

	<ul style="list-style-type: none"> <li>Configure CC6x channel polarity in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0 with PwmPolarity = PWM_LOW</li> </ul>	
	<ul style="list-style-type: none"> <li>Configure a Pwm channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0 with CCU6TimerUsed = T12</li> <li>Configure COUT6x channel polarity in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0 with Cout6xChPolarity = PWM_HIGH</li> <li>Configure CC6x channel polarity in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0 with PwmPolarity = PWM_HIGH</li> </ul>	0x3U
	<ul style="list-style-type: none"> <li>Configure a Pwm channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0/CCU6KernelUsed with CCU6TimerUsed = T13</li> <li>Configure channel polarity in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0 with PwmPolarity = PWM_HIGH</li> </ul>	0x4U
	<ul style="list-style-type: none"> <li>Configure a Pwm channel in</li> </ul>	0x0U

	Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0/CCU6KernelUsed with CCU6TimerUsed = T12 <ul style="list-style-type: none"> <li>• Configure COUT6x channel polarity in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0 with Cout6xChPolarity = PWM_LOW</li> <li>• Configure CC6x channel polarity in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0 with PwmPolarity = PWM_LOW</li> </ul>	
--	--	--

### 1.2.6.5 Member: TimerCntReg

**Table 26** TimerCntReg

<b>Name</b>	TimerCntReg	
<b>Type</b>	uint32	
<b>Description</b>	CCU6 timer channel counter	
<b>Verification method</b>	<p>The structure member is generated as the counter value:</p> <pre> If PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED     If PwmDutyShiftInTicks = 'True'         Value = PwmPeriodDefault - PwmShiftValue     else         Value = PwmPeriodDefault - ScaledShiftValue Else     Value = 0 endif </pre> <p><i>Note: For PWM_FIXED_PERIOD_SHIFTED channels PwmPeriodDefault will be same as the PwmPeriodDefault configured for its reference channel.</i></p>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>

	<ul style="list-style-type: none"> <li>Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with</li> <li>1. PwmAssignedHwUnit = CCU6</li> <li>2. PwmPeriodDefault = 0 (of the referenced channel)</li> <li>3. PwmShiftValue = 0</li> <li>4. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED</li> <li>PwmDutyShiftInTicks = 'True'</li> </ul>	0x0U
	<ul style="list-style-type: none"> <li>Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with</li> <li>1. PwmAssignedHwUnit = CCU6</li> <li>2. PwmPeriodDefault = 11125(of the referenced channel)</li> <li>3. PwmShiftValue = 20</li> <li>4. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED</li> <li>PwmDutyShiftInTicks = 'False'</li> </ul>	0x2b6fU
	<ul style="list-style-type: none"> <li>Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with</li> <li>1. PwmAssignedHwUnit = CCU6</li> <li>2. PwmPeriodDefault = 11125(of the referenced channel)</li> <li>3. PwmShiftValue = 20</li> <li>4. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED</li> <li>PwmDutyShiftInTicks = 'True'</li> </ul>	0x2b61U

### 1.2.6.6 Member: TimerPeriodReg

**Table 27** TimerPeriodReg

<b>Name</b>	TimerPeriodReg	
<b>Type</b>	uint32	
<b>Description</b>	CCU6 timer channel period register contents	
<b>Verification method</b>	<p>The structure member is generated as the period value set in the configuration parameter PwmPeriodDefault.</p> <p><i>Note: For PWM_FIXED_PERIOD_SHIFTED channels PwmPeriodDefault will be same as the PwmPeriodDefault configured for its reference channel.</i></p>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	<ul style="list-style-type: none"> <li>Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with               <ol style="list-style-type: none"> <li>PwmAssignedHwUnit = CCU6</li> <li>PwmPeriodDefault = 11125 (of the referenced channel)</li> <li>PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED</li> </ol> </li> </ul>	0x2b75U
	<ul style="list-style-type: none"> <li>Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with               <ol style="list-style-type: none"> <li>PwmAssignedHwUnit = CCU6</li> <li>PwmPeriodDefault = 60</li> <li>PwmChannelClass = PWM_FIXED_PERIOD</li> </ol> </li> </ul>	0x3CU

### 1.2.6.7 Member: Ccu6ShadowReg

**Table 28** Ccu6ShadowReg

<b>Name</b>	Ccu6ShadowReg
<b>Type</b>	uint32
<b>Description</b>	CCU6 timer channel shadow register contents
<b>Verification method</b>	<p>The structure member holds the scaled duty value:</p> <p style="padding-left: 40px;">if(PwmDutyShiftInTicks = 'False')</p> <p style="padding-left: 80px;">Value = ScaledDuty</p> <p style="padding-left: 40px;">Else</p>

	<pre> Value = PwmDutycycleDefault Endif  If (PwmPeriodDefault == 0)     Value = 0 Endif  If( Value &gt;= PwmPeriodDefault )     Value = PwmPeriodDefault Endif </pre>	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> <li>Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with</li> <li>1. PwmAssignedHwUnit = CCU6</li> <li>2. PwmPeriodDefault = 0 (of the referenced channel)</li> <li>3. PwmDutycycleDefault = 16384</li> <li>4. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED</li> </ul> <p>The output is same for PwmDutyShiftInTicks = 'False' and PwmDutyShiftInTicks = 'True' since period is zero.</p>	0x0U
	<ul style="list-style-type: none"> <li>Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with</li> <li>1. PwmAssignedHwUnit = CCU6</li> <li>2. PwmPeriodDefault = 6000 (of the referenced channel)</li> <li>3. PwmDutycycleDefault = 16384</li> <li>4. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED</li> <li>The output is same for PwmDutyShiftInTicks = 'False'</li> </ul>	0xbb8U
	<ul style="list-style-type: none"> <li>Configure channel in</li> </ul>	0xc8bU



	Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with <ol style="list-style-type: none"> <li>1. PwmAssignedHwUnit = CCU6</li> <li>2. PwmPeriodDefault = 6000 (of the referenced channel)</li> <li>3. PwmDutycycleDefault = 16384</li> <li>4. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED</li> </ol> <ul style="list-style-type: none"> <li>• The output is same for PwmDutyShiftInTicks = 'True'</li> </ul>	
	<ul style="list-style-type: none"> <li>• Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with             <ol style="list-style-type: none"> <li>1. PwmAssignedHwUnit = CCU6</li> <li>2. PwmPeriodDefault = 6000 (of the referenced channel)</li> <li>3. PwmDutycycleDefault = 5515</li> <li>4. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED</li> </ol> </li> <li>• The output is same for PwmDutyShiftInTicks = 'True'</li> </ul>	0x158bU

### 1.2.6.8 Member: TimerModeSelectReg

**Table 29** TimerModeSelectReg

<b>Name</b>	TimerModeSelectReg	
<b>Type</b>	uint8	
<b>Description</b>	CCU6 timer mode select register contents for the input kernel	
<b>Verification method</b>	The structure member holds the value 1 since the timers by default work in compare mode.  <i>Note: This structure member is not user configurable.</i>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>

	Configure channel in Pwm/PwmChannelConfigSet /PwmChannel/PwmChannel_2 with PwmAssignedHwUnit = CCU6	0x1U
--	---	------

### 1.2.6.9 Member: PortInSelReg0

**Table 30** PortInSelReg0

<b>Name</b>	PortInSelReg0	
<b>Type</b>	uint8	
<b>Description</b>	Port Input Select register contents for a kernel	
<b>Verification method</b>	<p>The structure member holds the value 0.</p> <p><i>Note: This structure member is not user configurable.</i></p>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure channel in Pwm/PwmChannelConfigSet /PwmChannel/PwmChannel_2 with PwmAssignedHwUnit = CCU6	0x0U

### 1.2.6.10 Member: IntEnReg

**Table 31** IntEnReg

<b>Name</b>	IntEnReg	
<b>Type</b>	uint8	
<b>Description</b>	CCU6 timer channel interrupt enable register contents	
<b>Verification method</b>	<p>The structure member is generated as a value:</p> <ol style="list-style-type: none"> <li>Value = 0</li> <li>Value = Value   T12 or T13 period match</li> <li>Value = Value   CCU6 rising edge interrupt or T13 compare match &lt;&lt; 1</li> <li>Value = Value   CCU6 falling edge interrupt &lt;&lt; 2</li> </ol> <p><i>Note: This structure member is not user configurable. Since Rising/Falling Edge interrupts are disabled by default after init, Value is set to zero.</i></p>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure a pwm channel in Pwm/PwmChannelConfigSet /PwmChannel/PwmChannel_2 with PwmAssignedHwUnit = CCU6	0x0U, /* Interrupt enable */

### 1.2.6.11 Member: IntNodePointerReg

**Table 32** IntNodePointerReg

<b>Name</b>	IntNodePointerReg	
<b>Type</b>	uint8	
<b>Description</b>	Selects Interrupt Nodes for T12 and T13 Duty and Period match.	
<b>Verification method</b>	<p>The structure member is generated as a value:</p> <ol style="list-style-type: none"> <li>Value = 0</li> <li>Value = Value   Interrupt node for configured T12 Comparator: Duty Match</li> <li>Value = Value   (Interrupt node for T12: Period Match) or (Interrupt node for T13: Compare Match and Period Match) &lt;&lt; 3</li> </ol> <p>If CCU6TimerUsed = T13 Value = 0x0C</p> <p>Else If CCU6TimerUsed = T12</p> <p>If Cc6xChannel = Cc60 &amp;&amp; highest configured comparator in T12 = Cc60 Value = 0x0U</p> <p>Else If Cc6xChannel = Cc60 &amp;&amp; highest configured comparator in T12 = Cc61 Value = 0x04U</p> <p>Else If Cc6xChannel = Cc60 &amp;&amp; highest configured comparator in T12 = Cc62 Value = 0x08U</p> <p>Else If Cc6xChannel = Cc61 &amp;&amp; highest configured comparator in T12 = Cc61 Value = 0x05U</p> <p>Else If Cc6xChannel = Cc61 &amp;&amp; highest configured comparator in T12 = Cc62 Value = 0x09U</p> <p>Else If Cc6xChannel = Cc62 &amp;&amp; highest configured comparator in T12 = Cc62 Value = 0x0AU</p> <p>Endif</p> <p>Endif</p> <p>Cc6xChannel and CCU6TimerUsed can be configured in container Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_&lt;x&gt;/CCU6CC6Configuration/C CU6CC6Configuration_0.</p>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	<ul style="list-style-type: none"> <li>Configure 2 Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel PwmChannel_0 and PwmChannel_1</li> <li>Configure for PwmChannel_0 CCU6TimerUsed = T12 and Cc6xChannel = Cc60 in</li> </ul>	0x4U /* Interrupt node Select */

	<p>Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0</p> <ul style="list-style-type: none"> <li>Configure for PwmChannel_1 CCU6TimerUsed = T12 and Cc6xChannel = Cc61 in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_1/CCU6CC6Configuration/CCU6CC6Configuration_0</li> </ul> <p>The output is for PwmChannel_0</p>	
	<ul style="list-style-type: none"> <li>Configure 3 Pwm channel in Pwm/PwmChannelConfig Set/PwmChannel PwmChannel_0, PwmChannel_1 and PwmChannel_2</li> <li>Configure for PwmChannel_0 CCU6TimerUsed = T12 and Cc6xChannel = Cc60 in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0</li> <li>Configure for PwmChannel_1 CCU6TimerUsed = T12 and Cc6xChannel = Cc61 in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_1/CCU6CC6Configuration/CCU6CC6Configuration_0</li> <li>Configure for PwmChannel_2 CCU6TimerUsed = T12 and Cc6xChannel = Cc62 in</li> </ul>	<p>0x9U /* Interrupt node Select */</p>

	Pwm/PwmChannelConfig Set/PwmChannel/PwmCh annel_2/CCU6CC6Configu ration/CCU6CC6Configura tion_0  The output is for PwmChannel_1	
--	--	--

### 1.3 File: Pwm\_17\_GtmCcu6[\_<variant>]\_PBcfg.h

The generated file contains the extern parameter for the root configuration structure. Post-build time configuration mechanism allows configurable functionality of PWM driver that is deployed as object code. The file is generated in 'inc' folder.

#### 1.3.1 Extern: Pwm\_17\_GtmCcu6\_Config[\_<variant>]

**Table 33** Pwm\_17\_GtmCcu6\_Config[\_<variant>]

<b>Name</b>	Pwm_17_GtmCcu6_Config[_<variant>]	
<b>Type</b>	Pwm_17_GtmCcu6_ConfigType	
<b>Description</b>	Extern for root configuration structure of PWM driver which will be used during initialization.	
<b>Verification method</b>	The generated variable is the extern of the structure which is present in Pwm_17_GtmCcu6[_<variant>]_PBcfg.c file. The <variant> indicates the name of the post-build variant. For a variant-aware configuration the extern variable name is appended with the variant name. For variant-unaware configuration <variant> is ignored.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	PWM is configured and is variant-aware. Variant name is 'Petrol'	extern const Pwm_17_GtmCcu6_ConfigType Pwm_17_GtmCcu6_Config_Petrol;
	PWM is configured and is variant-unaware	extern const Pwm_17_GtmCcu6_ConfigType Pwm_17_GtmCcu6_Config;

## Revision history

## Revision history

## Major changes since the last revision

Date	Version	Description
2021-11-08	V2.0	Document is Released
2021-11-02	V1.1	ModCtrlReg and PasStateLvlReg in the structure Pwm_kChannelConfigCcu6_Core<x> [_<variant>][<y>] are updated with Cout6xChEnable and Cout6xChPolarity to provide COUT feature support
2020-11-05	V1.0	Document is Released
2020-11-04	V0.1	<ul style="list-style-type: none"><li>- Pwm_17_GtmCcu6 driver chapter moved from MC-ISAR_TC3xx_Config_Verification_Manual_BASIC.pdf to this document</li><li>- Structure member TimerChPortOutConfig is deleted from structure Pwm_kChannelConfigGtm_Core&lt;x&gt;</li></ul>

## Trademarks

All referenced product or service names and trademarks are the property of their respective owners.

**Edition 2021-11-08**

**Published by**

**Infineon Technologies AG**

**81726 Munich, Germany**

**© 2021 Infineon Technologies AG.**

**All Rights Reserved.**

**Do you have a question about this document?**

**Email: [erratum@infineon.com](mailto:erratum@infineon.com)**

**Document reference**

**Doc\_Number**

## IMPORTANT NOTICE

The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics ("Beschaffenheitsgarantie").

With respect to any examples, hints or any typical values stated herein and/or any information regarding the application of the product, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights of any third party.

In addition, any information given in this document is subject to customer's compliance with its obligations stated in this document and any applicable legal requirements, norms and standards concerning customer's products and any use of the product of Infineon Technologies in customer's applications.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on the product, technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies office ([www.infineon.com](http://www.infineon.com)).

## WARNINGS

Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.