

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
1 Pwm_17_GtmCcu6 driver.....	4
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	11
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.....	17
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.....	29
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].....	31
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	39
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	51
1.2.6.6	Member: TimerPeriodReg.....	53
1.2.6.7	Member: Ccu6ShadowReg.....	54
1.2.6.8	Member: TimerModeSelectReg	56
1.2.6.9	Member: PortInSelReg0	56
1.2.6.10	Member: IntEnReg.....	56
1.2.6.11	Member: IntNodePointerReg.....	57
1.3	File: Pwm_17_GtmCcu6[_<variant>]_PBcfg.h	59
1.3.1	Extern: Pwm_17_GtmCcu6_Config[_<variant>]	59
Revision history.....		61

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

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

1.1.2 Macro: PWM_17_GTMCCU6_AR_RELEASE_MINOR_VERSION

Table 2 PWM_17_GTMCCU6_AR_RELEASE_MINOR_VERSION

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

1.1.3 Macro: PWM_17_GTMCCU6_AR_RELEASE_REVISION_VERSION

Table 3 PWM_17_GTMCCU6_AR_RELEASE_REVISION_VERSION

Name	PWM_17_GTMCCU6_AR_RELEASE_REVISION_VERSION
-------------	--

Description	Revision version number of AUTOSAR release on which the Pwm_17_GtmCcu6 implementation is based on.	
Verification method	The macro is generated with the value present in 'CommonPublishedInformation/ArPatchVersion'. <i>Note: The macro is not user configurable.</i>	
Example(s)	Action	Generated output
	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

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

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

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

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

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

Name	PWM_17_GTMCCU6_DE_INIT_API
-------------	----------------------------

Description	Decides the mode of execution of Init and Delnit API's.	
Verification method	The macro is generated as STD_ON if PwmDelnitApi configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
Example(s)	Action	Generated output
	PwmDelnitApi = True	#define PWM_17_GTMCCU6_DE_INIT_API (STD_ON)
	PwmDelnitApi = 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

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

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

Name	PWM_17_GTMCCU6_SET_DUTY_CYCLE_API	
Description	Enables/disables Pwm_17_GtmCcu6_SetDutyCycle API	
Verification method	The macro is generated as STD_ON if PwmSetDutyCycle configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
Example(s)	Action	Generated output
	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

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

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

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

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

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

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

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

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

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

iod = False

1.1.22 Macro: PWM_17_GTMCCU6_DUTY_SHIFT_IN_TICKS

Table 22 PWM_17_GTMCCU6_DUTY_SHIFT_IN_TICKS

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

Name	PWM_17_GTMCCU6_INSTANCE_ID	
Description	Instance ID of PWM module	
Verification method	The macro is generated as a numeric value set in the configuration parameter 'PwmIndex'	
Example(s)	Action	Generated output
	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>

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

Pwm_17_GtmCcu6 driver

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

Name	PWM_17_GTMCCU6_MAX_CHANNELS	
Description	Indicates the total number of PWM channels configured	
Verification method	The macro is generated as a numeric value which corresponds to the number of elements in the list 'PwmChannelConfigSet/PwmChannel'.	
Example(s)	Action	Generated output
	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

Name	PWM_17_GTMCCU6_MAX_CORES	
Description	Indicates the total number of cores available in the device	
Verification method	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>	
Example(s)	Action	Generated output
	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

Name	PWM_17_GTMCCU6_GTM_TIMER_USED	
Description	Indicates that GTM channel/s are configured in PWM	
Verification method	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.	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> Configure 4 PWM channels in Pwm/PwmChannelConfigSet/PwmChannel 	#define PWM_17_GTMCCU6_GTM_TIMER_USED (STD_ON)

Pwm_17_GtmCcu6 driver

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

Name	PWM_17_GTMCCU6_CCU6_TIMER_USED	
Description	Indicates that CCU6 channel/s are configured in PWM	
Verification method	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.	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> Configure 4 PWM channels in Pwm/PwmChannelConfigSet/PwmChannel (PwmChannel_0 to PwmChannel_3). Assign PwmAssignedHwUnit as GTM for PwmChannel_0 and PwmChannel_3 Assign PwmAssignedHwUnit as CCU6 for PwmChannel_1 and PwmChannel_2 	#define PWM_17_GTMCCU6_CCU6_TIMER_USED (STD_ON)

Pwm_17_GtmCcu6 driver

<ul style="list-style-type: none"> Configure 3 PWM channels in Pwm/PwmChannelConfigSet/PwmChannel (PwmChannel_0 to PwmChannel_2) Assign PwmAssignedHwUnit as GTM for PwmChannel_0, PwmChannel_1 and PwmChannel_2 	<pre>#define PWM_17_GTMCCU6_CC06_TIMER_USED (STD_OFF)</pre>
--	---

1.1.29 Macro: Pwm_17_GtmCcu6Conf_PwmChannel_<channel name>

Table 29 Pwm_17_GtmCcu6Conf_PwmChannel_<channel name>

Name	Pwm_17_GtmCcu6Conf_PwmChannel_<channel name>	
Description	The macro is the symbolic name generated for the configuration parameter 'Pwm/PwmChannelConfigSet/PwmChannel'	
Verification method	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.	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> Configure 4 PWM channels in Pwm/PwmChannelConfigSet/PwmChannel. Container PwmChannel for Channel ID 0 is named TOM1. Container PwmChannel for Channel ID 1 is named ATOM1. Container PwmChannel for Channel ID 2 is named CCU60_1 Container PwmChannel for Channel ID 3 is named TOM2 	<pre>#define Pwm_17_GtmCcu6Conf_PwmChannel_TOM1 (0U) #define Pwm_17_GtmCcu6Conf_PwmChannel_ATOM1 (1U) #define Pwm_17_GtmCcu6Conf_PwmChannel_CC060_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*)&Pwm_CoreConfigCore0, NULL_PTR, NULL_PTR, NULL_PTR, NULL_PTR, NULL_PTR }, (uint8*)&Pwm_ChannelIndexMap, { 0xff050403U, 0xfffffffffU } };</pre>
	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*)&Pwm_CoreConfigCore0_Petrol, NULL_PTR,</pre>

	<pre> NULL_PTR, NULL_PTR, NULL_PTR, NULL_PTR }, (uint8*)&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]

Name	PwmCoreAdd[PWM_17_GTMCCU6_MAX_CORES]	
Type	Pwm_17_GtmCcu6_CoreConfigType*	
Description	Array of core-specific configuration.	
Verification method	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).	
Example(s)	Action	Generated output
	All the PWM channels are allocated to Core 0 (variant-unaware)	<pre> { (Pwm_17_GtmCcu6_CoreConfigType*)&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*)&Pwm_CoreConfigCore0_Petrol, NULL_PTR, NULL_PTR, </pre>

	<pre> 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*) &Pwm_CoreConfigCore1, (Pwm_17_GtmCcu6_CoreConfigType*) &Pwm_CoreConfigCore2, (Pwm_17_GtmCcu6_CoreConfigType*) &Pwm_CoreConfigCore3, (Pwm_17_GtmCcu6_CoreConfigType*) &Pwm_CoreConfigCore4, (Pwm_17_GtmCcu6_CoreConfigType*) &Pwm_CoreConfigCore5 } </pre>

1.2.1.2 Member: Pwm_ChannelIndexMap

Table 32 PwmChannelIdxmap

Name	Pwm_ChannelIndexMap	
Type	uint8*	
Description	Pointer to channel index map.	
Verification method	The generated structure member contains the address of Pwm_ChannelIndexMap.	
Example(s)	Action	Generated output
	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]

Name	PwmCcu6ChIdx[MCU_17_CCU6_NO_OF_KERNELS]
Type	uint32
Description	Channel ID of CCU6 channels for a particular kernel.
Verification	The generated structure member contains 2 array entries with each entry corresponding

method	<p>to CCU6 kernel. If the PWM channel is not configured '0xFFFFFFFFU' is generated, else the channel ID of CCU6 channels are generated. The structure member is generated as a value:</p> <ol style="list-style-type: none"> 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"> Assign CCU6 to PwmAssignedHwUnit parameter for PwmChannel_4 with PwmChannelId 3 Assign T12 to CCU6TimerUsed parameter and Cc61 to Cc6xChannel parameter in container 'PwmChannel/CCU6CC6C onfiguration' Assign /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuCcu6ModuleAllocationConf_1 to CCU6KernelUsed parameter in container 'PwmChannel/CCU6CC6C onfiguration' for PwmChannel_4 	<pre>{ 0xFFFFFFFFU, 0xFFFF03FFU }</pre>
	<ul style="list-style-type: none"> Assign CCU6 to PwmAssignedHwUnit parameter for PwmChannel_0 with PwmChannelId 1 Assign T12 to CCU6TimerUsed parameter and Cc60 to Cc6xChannel parameter in container 'PwmChannel/CCU6CC6C onfiguration' for PwmChannel_0 Assign CCU6 to PwmAssignedHwUnit parameter for PwmChannel_4 with 	<pre>{ 0xFFFF0401U, 0xFFFFFFFFU, }</pre>

PwmChannelId 4	
<ul style="list-style-type: none"> Assign T12 to CCU6TimerUsed parameter and Cc61 to Cc6xChannel parameter in container 'PwmChannel/CCU6CC6C onfiguration' for PwmChannel_4 Assign /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuCcu6ModuleAllocationConf_0 to CCU6KernelUsed parameter in container 'PwmChannel/CCU6CC6C onfiguration' for PwmChannel_4 and PwmChannel_0 	

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*) &Pwm_k ChannelConfigurationCore0 };</pre>
Example(s)	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 = {</pre>

		<pre> (Pwm_17_GtmCcu6_ChannelType) 4U, (const Pwm_17_GtmCcu6_ChannelConfigType*) &Pwm_k ChannelConfigurationCore0_Petrol }; static const Pwm_17_GtmCcu6_CoreConfigType Pwm_CoreConfigCore1_Petrol = { (Pwm_17_GtmCcu6_ChannelType) 3U, (const Pwm_17_GtmCcu6_ChannelConfigType*) &Pwm_k ChannelConfigurationCore1_Petrol }; </pre>
--	--	--

1.2.2.1 Member: Pwm_ChannelConfigPtr

Table 35 Pwm_ChannelConfigPtr

Name	Pwm_ChannelConfigPtr	
Type	Pwm_17_GtmCcu6_ChannelConfigType *	
Description	Pointer to the base of array which stores the data of each channel configured to Core<x>.	
Verification method	The structure member is generated with base address of array which stores the channel data of Core <x>.	
Example(s)	Action	Generated output
	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*) &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*) &Pwm_k ChannelConfigurationCore0[0] </pre>

1.2.2.2 Member: MaxChannels

Table 36 MaxChannels

Name	MaxChannels	
Type	Pwm_17_GtmCcu6_ChannelType	
Description	Indicates the total number of channels assigned to Core <x>.	
Verification method	<p>The structure member is generated as total number of channels allocated to CORE<x>.</p> <p><i>Note: Channels not assigned to any core are assigned to master core (ResourceMMasterCore).</i></p>	
Example(s)	Action <ul style="list-style-type: none"> Configure 4 PWM channels in Pwm/PwmChannelConfig Set/PwmChannel. Allocate 3 channels to Core 0. Allocate 1 channel to Core 1. <p>Output is shown for Core 0</p>	Generated output 3
	<ul style="list-style-type: none"> Configure 14 PWM channels in Pwm/PwmChannelConfig Set/PwmChannel. Allocate 3 channels to Core 1. ResourceMMasterCore is CORE0. Do not allocate rest of the channels to any core. <p>Output is shown for Core 0</p>	11

1.2.3 Structure: Pwm_kChannelConfigurationCore <x>[_<variant>]

Table 1 Pwm_kChannelConfigurationCore <x>[_<variant>]

Name	Pwm_kChannelConfigurationCore <x>[_<variant>]	
Type	Pwm_17_GtmCcu6_ChannelConfigType	
Description	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)	
Verification method	<p>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.</p>	
Example(s)	Action	Generated output
	Configure 1 PWM channel to Core0 in ResourceM/ResourceMMcalC	<pre>static const Pwm_17_GtmCcu6_ChannelConfigType</pre>

onfig/ResourceMMcalConfig_0/ResourceMMcalCore/ResourceMMcalCore_0/ResourceMAAllocation	<pre> Pwm_kChannelConfigurationCore0[] = { { 0, /* Pwm logical channel ID */ 0, /* Timer used GTM:0/CCU6:1 */ Pwm_lConfigChannel(0U, /*Channel reset from other channel or not*/ PWM_17_GTMCCU6_FIXED_PERIOD, /*ChannelClass*/ 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*)&Pwm_kChannelConfigGtm_Core0[0] } } </pre>
--	---

1.2.3.1 Member: PwmChannelId

Table 2 PwmChannelId

Name	PwmChannelId	
Type	Pwm_17_GtmCcu6_ChannelType	
Description	Indicates the channel ID of a PWM channel	
Verification method	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)	
Example(s)	Action	Generated output
	Configure an PWM channel with PwmChannelId = 7 in PwmChannelConfigSet/PwmChannel/PwmChannel_1/PwmChannelId'	7

1.2.3.2 Member: PwmTimerUsed

Table 3 PwmTimerUsed

Name	PwmTimerUsed
-------------	--------------

Type	uint8	
Description	Hardware type selected for the PWM channel.	
Verification method	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

Name	PwmNotification	
Type	Pwm_17_GtmCcu6_NotifiPtrType	
Description	Pointer to the callback functions configured by the user.	
Verification method	This structure member holds the address of the notification function.	
Example(s)	Action	Generated output
	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) &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

Name	PwmChannelInfo	
Type	uint16	
Description	The structure member contains the information about Polarity, Idlestate, Coherency, Class, Shift reset and DsadcNotif value for a PWM channel.	
Verification method	<p>The structure member is generated as a value:</p> <ol style="list-style-type: none"> 1. Value = 0 2. Value = Value PwmPolarity 3. Value = Value PwmIdleState << 1 4. Value = Value PwmCoherentUpdate << 2 5. Value = Value PwmChannelClass << 3 6. Value = Value Shift Reset << 5 7. Value = Value DsadcNotif << 6 <ol style="list-style-type: none"> a. Shift Reset = 0 if (PwmHandleShiftByOffset = 'False') && (PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED or PWM_FIXED_PERIOD_CENTER_ALIGNED) b. Shift Reset = 1 if (PwmHandleShiftByOffset = 'False') && (PwmChannelClass = PWM_FIXED_PERIOD or PWM_VARIABLE_PERIOD) c. Shift Reset = 0 if (PwmHandleShiftByOffset = 'True') && (PwmChannelClass = PWM_FIXED_PERIOD or PWM_VARIABLE_PERIOD) d. Shift Reset = 1 if (PwmHandleShiftByOffset = 'True') && (PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED or PWM_FIXED_PERIOD_CENTER_ALIGNED) <p>Note: The configuration parameters PwmPolarity, PwmIdleState, PwmCoherentUpdate and PwmChannelClass are present in container 'PwmChannelConfigSet/PwmChannel/PwmChannel_<x>'. (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>	
Example(s)	Action	Generated output
	Configure a PWM channel with PwmPolarity = PWM_HIGH, PwmIdleState = PWM_LOW, PwmCoherentUpdate =	<pre>Pwm_1ConfigChannel(1U, /* The notification flag to enable GTM interrupts to trigger DSADC */ 0U, /* Channel reset from other channel or not*/</pre>

	<p>'True', PwmChannelClass = PWM_FIXED_PERIOD, PwmHandleShiftByOffset = 'True'</p> <p>node:ref(/GtmTimerUsed)/McuAtomChannelEventHandledByDsadc = 'True'</p> <p>PwmAssignedHwUnit = GTM</p>	<pre>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_lConfigChannel(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_lConfigChannel(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</p>	<pre>Pwm_lConfigChannel(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</pre>

	<pre> = 'False' node:ref(./GtmTimerUsed)/McuAtomChannelEvent HandledByDsadc = 'False' PwmAssignedHwUnit = GTM </pre>	<pre> state */ PWM_17_GTMCCU6_HIGH /* channel polarity */) </pre>
	<pre> Configure a PWM channel with PwmPolarity = PWM_LOW, PwmIdleState = PWM_LOW, PwmCoherentUpdate = 'False', PwmChannelClass = PWM_FIXED_PERIOD_SHI FTED, PwmHandleShiftByOffset = 'True' node:ref(./GtmTimerUsed)/McuTomChannelEventH andledByDsadc = 'True' PwmAssignedHwUnit = GTM </pre>	<pre> Pwm_lConfigChannel(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

Name	PwmPeriodDefault	
Type	Pwm_17_GtmCcu6_PeriodType	
Description	Default period for PWM channel	
Verification method	<p>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)</p> <p><i>Note: PwmPeriodDefault value is fixed permanently for a PWM channel configured with Fixed Period type.</i></p>	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> Configure a PWM channel with PwmPeriodDefault = 60000 in 'PwmChannelConfigSet/PwmChannel/PwmChannel_0/PwmPeriodDefault' 	<pre> (Pwm_17_GtmCcu6_PeriodType) 0xea60, /* Default Period */ </pre>

1.2.3.6 Member: PwmDutycycleDefault

Table 7 PwmDutycycleDefault

Name	PwmDutycycleDefault	
Type	uint32	
Description	Default duty cycle for PWM channel	
Verification method	<p>The structure member is generated as a numeric value set in the configuration parameter 'PwmChannelConfigSet/PwmChannel/PwmChannel_<x>/PwmDutycycleDefault'. (x is the configured channel number)</p> <p><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></p>	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> Configure a PWM channel with PwmDutycycleDefault = 6553 in PwmChannelConfigSet/PwmChannel/PwmChannel_0/PwmDutycycleDefault' Parameter PwmDutyShiftInTicks = 'True' 	(uint32)0x1999, /* Default Duty Cycle */
	<ul style="list-style-type: none"> Configure a PWM channel with PwmDutycycleDefault = 6553 in PwmChannelConfigSet/PwmChannel/PwmChannel_0/PwmDutycycleDefault' Parameter PwmDutyShiftInTicks = 'False' 	(uint32)0x2ede, /* Default Duty Cycle */

1.2.3.7 Member: PwmShiftValue

Table 8 PwmShiftValue

Name	PwmShiftValue	
Type	uint32	
Description	Shift of PWM_FIXED_PERIOD_SHIFTED channel with respect to reference channel.	
Verification method	<p>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)</p>	

	<p><i>Note:</i> Value of PwmShiftValue is entered as a percentage when configuration parameter PwmDutyShiftInTicks is set to 'False', else the value is entered as absolute ticks.</p> <p><i>Note:</i> Configuration parameter PwmShiftValue is available only when PwmChannelClass is PWM_FIXED_PERIOD_SHIFTED.</p>	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> Configure a PWM channel with PwmShiftValue = 20 in PwmChannelConfigSet/PwmChannel/PwmChannel_0/ PwmShiftValue Parameter PwmDutyShiftInTicks = 'True' 	(uint32)0x14, /* Shift Value*/
	<ul style="list-style-type: none"> Configure a PWM channel with PwmShiftValue = 20 in PwmChannelConfigSet/PwmChannel/PwmChannel_0/ PwmShiftValue Parameter PwmDutyShiftInTicks = 'False' 	(uint32)0x24, /* Shift Value*/

1.2.3.8 Member: PwmTimerPtr

Table 9 PwmTimerPtr

Name	PwmTimerPtr	
Type	const void*	
Description	Pointer to GTM/CCU6 timer channel information.	
Verification method	The structure member is generated with pointer to a structure which holds the timer specific information for a channel.	
Example(s)	Action	Generated output
	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	(const void*)&Pwm_kChannelConfigCcu6_Core1[0]

Pwm/PwmChannelConfigSet /PwmChannel
--

1.2.4 Array: Pwm_ChannelIndexMap[_<variant>][PWM_17_GTMCCU6_MAX_CHANNELS]

Table 10 Pwm_ChannelIndexMap[_<variant>][PWM_17_GTMCCU6_MAX_CHANNELS]

Name	Pwm_ChannelIndexMap[_<variant>][PWM_17_GTMCCU6_MAX_CHANNELS]	
Type	Pwm_17_GtmCcu6_ChannelConfigType	
Description	This array holds index of each configured channel within the allocated core.	
Verification method	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.	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> Configure 5 PWM channels in Pwm/PwmChannelConfigSet/PwmChannel (PwmChannel_0 to PwmChannel_4). Assign PwmChannel_0, PwmChannel_3 and PwmChannel_4 in CORE4 with PwmChannelId = 0, 3, 4 respectively Assign PwmChannel_1, PwmChannel_2 in CORE2 with PwmChannelId = 1, 2 respectively 	<pre>static const uint8 Pwm_ChannelIndexMap[5] = { 0x0U, 0x0U, 0x1U, 0x1U, 0x2U };</pre>
	<ul style="list-style-type: none"> Configure 5 PWM channels in Pwm/PwmChannelConfigSet/PwmChannel (PwmChannel_0 to PwmChannel_4). Assign PwmChannel_0, PwmChannel_3 and PwmChannel_4 in CORE4 with PwmChannelId = 0, 1, 2 respectively Assign PwmChannel_1, PwmChannel_2 in CORE2 with PwmChannelId = 3, 4 respectively 	<pre>static const uint8 Pwm_ChannelIndexMap[5] = { 0x0U, 0x1U, 0x2U, 0x0U, 0x1U };</pre>
	<ul style="list-style-type: none"> Configure 9 PWM channels 	<pre>static const uint8</pre>

	<p>in Pwm/PwmChannelConfig Set/PwmChannel (PwmChannel_0 to PwmChannel_8).</p> <ul style="list-style-type: none"> Assign PwmChannel_0, PwmChannel_1 and PwmChannel_2 in CORE4 with PwmChannelId = 0, 1, 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' 	<pre>Pwm_ChannelIndexMap_Petrol[9] = { 0x0U, 0x1U, 0x2U, 0x0U, 0x1U, 0x0U, 0x1U, 0x2U, 0x3U };</pre>
--	---	--

1.2.5 Structure: Pwm_kChannelConfigGtm_Core<x> [_<variant>][<y>]

Table 11 Pwm_kChannelConfigGtm_Core<x> [_<variant>][<y>]

Name	Pwm_kChannelConfigGtm_Core<x> [_<variant>][<y>]	
Type	Mcu_17_Gtm_TomAtomChConfigType	
Description	The configuration data of the all PWM GTM channels belonging to a Core	
Verification method	<p>Configuration structure holds the configuration data of the <y> PWM GTM channels belonging to Core <x>.</p> <p><x> ranges from 0 to 5.</p> <p><y> is the total number of GTM channels configured for Core<x>.</p> <p><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.</p> <p><i>Note: The generated file has this structure if at least one GTM channel is assigned to Core <x>.</i></p>	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> Configure 5 PWM channel to Core0 in ResourceM/ResourceMMcalConfig/ResourceMMcalConfig_0/ResourceMMcalC 	<pre>static const Mcu_17_Gtm_TomAtomChConfigType Pwm_kChannelConfigGtm_Core0_Petrol[2] = {</pre>

	<p>ore/ResourceMMcalCore_0/ResourceMAllocation</p> <ul style="list-style-type: none"> • Configure 3 PWM channel with PwmAssignedHwUnit = CCU6 in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0 • Configure 2 PWM channel with PwmAssignedHwUnit = GTM in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0 • This configuration is variant-aware. Variant name is 'Petrol' 	<pre> { 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 */ 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

Name	TimerType	
Type	Mcu_17_Gtm_TimerOutType	
Description	TOM/ATOM timer type	
Verification method	The structure member is generated as the GtmTimerUsed selected in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_<x>/GtmTimerOutputModuleConfiguration/GtmTimerOutputModuleConfiguration_0	
Example(s)	Action	Generated output

	Configure a PWM channel with GtmTimerUsed = '/Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuGtmAllocationConf_0/McuGtmAtomAllocationConf_0/McuGtmAtomChannelAllocationConf_1' in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/GtmTimerOutputModuleConfiguration/GtmTimerOutputModuleConfiguration_0	MCU_GTM_TIMER_ATOM
	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

Name	TimerId	
Type	Mcu_17_Gtm_TimerChIdentifierType	
Description	Consists of TOM or ATOM module ID and channel ID	
Verification method	The structure member is generated as a value: 1. Value = 0 2. Value = Value Channel number 3. Value = Value Module number << 8	
Example(s)	Action	Generated output
	Configure a PWM channel with GtmTimerUsed = '/Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuGtmAllocationConf_0/McuGtmAtomAllocationConf_0/McuGtmAtomChannelAllocationConf_4' in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel	0x4U

	_0/GtmTimerOutputModuleConfiguration/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	0x030AU

1.2.5.3 Member: TimerChCtrlReg

Table 14 TimerChCtrlReg

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

	<pre> RST_CC0= 1 Else RST_CC0 = 0 6. Value = Value TrigOut << 24 If (PwmChannelClass of Channel A= PWM_FIXED_PERIOD) && (PwmHandleShiftByOffset = 'False') && (PwmChannelClass of Channel B = PWM_FIXED_PERIOD_SHIFTED) && (PwmReferenceChannel of Channel B = Channel A) TrigOut = 1 else if (PwmChannelClass of Channel A= PWM_FIXED_PERIOD) && (PwmChannelClass of Channel B = PWM_FIXED_PERIOD_CENTER_ALLIGNED && PwmReferenceChannel of Channel B = Channel A) TrigOut = 1 else TrigOut = 0 </pre>	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> Configure a Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_1/GtmTimerOutputModuleConfiguration/GtmTimerOutputModuleConfiguration_0 with <ol style="list-style-type: none"> GtmTimerUsed = /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuGtmAllocationConf_0/McuGtmAtomAllocationConf_0/McuGtmAtomChannelAllocationConf_0 GtmTimerClockSelect = GTM_CONFIGURABLE_CLOCK_0=1 <ul style="list-style-type: none"> Configure channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_1 with <ol style="list-style-type: none"> PwmPolarity = PWM_LOW PwmChannelClass = PWM_FIXED_PERIOD <ul style="list-style-type: none"> Configure parameter PwmHandleShiftByOffset = True 	0x81000002U

	<ul style="list-style-type: none"> Configure a Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/GtmTimerOutputModuleConfiguration/GtmTimerOutputModuleConfiguration_0 with <ol style="list-style-type: none"> GtmTimerUsed = /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuGtmAllocationConf_0/McuGtmAtomAllocationConf_0/McuGtmAtomChannelAllocationConf_0 GtmTimerClockSelect = GTM_CONFIGURABLE_CLOCK_0 <ul style="list-style-type: none"> Configure channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0 with <ol style="list-style-type: none"> PwmPolarity = PWM_HIGH PwmChannelClass = PWM_FIXED_PERIOD <ul style="list-style-type: none"> Configure parameter PwmHandleShiftByOffset = True <p>This channel is used as a reference for the shifted period channel in the below example.</p>	0x81000802U
	<ul style="list-style-type: none"> Configure a Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_2/GtmTimerOutputModuleConfiguration/GtmTimerOutputModuleConfiguration_0 with <ol style="list-style-type: none"> GtmTimerUsed = /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuGtmAllocationConf_0/McuGtmAtomAllocationConf_0/McuGtmAtomChannelAllocationConf_2 	0x80002002U

	<ol style="list-style-type: none"> 4. GtmTimerClockSelect = GTM_CONFIGURABLE_CLOCK_2 • Configure channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_2 with 5. PwmPolarity = PWM_LOW 6. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED • Configure parameter PwmHandleShiftByOffset = True <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"> • Configure a Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_2/GtmTimerOutputModuleConfiguration/GtmTimerOutputModuleConfiguration_0 with <ol style="list-style-type: none"> 1. GtmTimerUsed = /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuGtmAllocationConf_0/McuGtmAtomAllocationConf_0/McuGtmAtomChannelAllocationConf_2 2. GtmTimerClockSelect = GTM_CONFIGURABLE_CLOCK_2 • Configure channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_2 with 1. PwmPolarity = PWM_LOW 2. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED • Configure parameter PwmHandleShiftByOffset 	0x80102002U

	= False	
	The Fixed period channel in the example above is taken as a reference channel for this shifted channel.	

1.2.5.4 Member: TimerChCN0Reg

Table 15 TimerChCN0Reg

Name	TimerChCN0Reg	
Type	uint32	
Description	TOM/ATOM channel CN0 register value	
Verification method	<p>The structure member holds the counter value of the channel.</p> <pre> if (PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED)&& (PwmHandleShiftByOffset = 'True') CN0 Value = PwmPeriodDefault - PwmShiftValue if(PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED)&& (PwmHandleShiftByOffset = 'False') CN0 Value = 0 Else if(PwmChannelClass = PWM_FIXED_PERIOD_CENTER_ALLIGNED) CN0 Value = 0 Else CN0 Value = PwmPeriodDefault </pre>	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0 with <ol style="list-style-type: none"> PwmAssignedHwUnit = GTM PwmPeriodDefault = 60000 PwmChannelClass = PWM_FIXED_PERIOD 	0xea60U
	<ul style="list-style-type: none"> Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with <ol style="list-style-type: none"> PwmAssignedHwUnit = GTM PwmPeriodDefault = 60000(of the referenced channel) PwmShiftValue = 70 	0x0U

	4. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED • PwmHandleShiftByOffset = 'False'	
	• Configure channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_2 with 5. PwmAssignedHwUnit = GTM 6. PwmPeriodDefault = 60000 (of the referenced channel) 7. PwmShiftValue = 70 8. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED • PwmHandleShiftByOffset = 'True'	0xea1aU

1.2.5.5 Member: TimerChCM0Reg

Table 16 TimerChCM0Reg

Name	TimerChCM0Reg
Type	uint32
Description	TOM/ATOM channel CM0 register value
Verification method	<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)&& (PwmHandleShiftByOffset = 'False')) or (PwmChannelClass = PWM_FIXED_PERIOD_CENTER_ALLIGNED) If(PwmDutycycleDefault = 0) CM0 Value= 0xFFFF (If GtmTimerUsed = TOM) CM0 Value= 0FFFFFFF (If GtmTimerUsed = ATOM) else if (PwmDutycycleDefault = max) CM0 Value=0 else CM0 Value= PwmShiftValue endif else CM0 Value= PwmPeriodDefault Endif Endif </pre>

Example(s)	Action	Generated output
	<ul style="list-style-type: none"> Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0 with <ol style="list-style-type: none"> 1. PwmAssignedHwUnit = GTM 2. PwmDutycycleDefault = 6553 3. PwmPeriodDefault = 60000 4. PwmChannelClass = PWM_FIXED_PERIOD 	0xea60U
	<ul style="list-style-type: none"> Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0 with <ol style="list-style-type: none"> 1. PwmAssignedHwUnit = GTM 2. PwmDutycycleDefault = 1677215 3. PwmPeriodDefault (of the referenced channel)= 60000 4. PwmShiftValue = 70 5. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED <ul style="list-style-type: none"> PwmHandleShiftByOffset = 'False' 	0x0U
	<ul style="list-style-type: none"> Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with <ol style="list-style-type: none"> 1. PwmAssignedHwUnit = GTM 2. PwmDutycycleDefault = 0 3. PwmPeriodDefault (of the referenced channel)= 60000 4. PwmShiftValue = 70 5. Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2/GtmTimerOutput ModuleConfiguration/GtmTimerOutputModuleCo 	0xFFFFFFFFU

	nfiguration_0/GtmTimerUsed GtmTimerUsed = /Mcu/Mcu/McuHardwareResourceAllocationConf_0/ McuGtmAllocationConf_0/ /McuGtmAtomAllocationConf_0/McuGtmAtomChannelAllocationConf_2 6. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED • PwmHandleShiftByOffset = 'False'	
	• Configure channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0 with 1. PwmAssignedHwUnit = GTM 2. PwmDutycycleDefault = 6777 3. PwmPeriodDefault (of the referenced channel)= 60000 4. PwmShiftValue = 70 5. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED • PwmHandleShiftByOffset = 'True'	0xea60U

1.2.5.6 Member: TimerChCM1Reg

Table 17 TimerChCM1Reg

Name	TimerChCM1Reg
Type	uint32
Description	TOM/ATOM channel CM1 register value
Verification method	The structure member holds the ScaledDuty as the duty match value of the channel. If (PwmChannelClass = PWM_FIXED_PERIOD or PWM_VARIABLE_PERIOD) CM1 Value= ScaledDuty else If((PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED)&& (PwmHandleShiftByOffset = 'False')) or (PwmChannelClass = PWM_FIXED_PERIOD_CENTER_ALLIGNED) If(PwmDutycycleDefault = 0) CM1 Value = 0

	<pre> 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>	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0 with <ol style="list-style-type: none"> PwmAssignedHwUnit = GTM PwmDutycycleDefault = 6553 PwmPeriodDefault = 60000 PwmChannelClass = PWM_FIXED_PERIOD 	0x1999U
	<ul style="list-style-type: none"> Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0 with <ol style="list-style-type: none"> PwmAssignedHwUnit = GTM PwmDutycycleDefault = 1677215 PwmPeriodDefault (of the referenced channel)= 60000 Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0/GtmTimerOutput ModuleConfiguration/GtmTimerOutputModuleConfiguration_0/GtmTimerUsed GtmTimerUsed = /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuGtmAllocationConf_0/McuGtmAtomAllocationConf_0/McuGtmAtomChannelAllocationConf_2 	0xFFFFFFFFU

	<ul style="list-style-type: none"> 5. PwmShiftValue = 70 6. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED • PwmHandleShiftByOffset = 'False' 	
	<ul style="list-style-type: none"> • Configure channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_2 with <ul style="list-style-type: none"> 1. PwmAssignedHwUnit = GTM 2. PwmDutycycleDefault = 0 3. PwmPeriodDefault (of the referenced channel)= 60000 4. PwmShiftValue = 70 5. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED • PwmHandleShiftByOffset = 'False' 	0x0U
	<ul style="list-style-type: none"> • Configure channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0 with <ul style="list-style-type: none"> 1. PwmAssignedHwUnit = GTM 2. PwmDutycycleDefault = 6777 3. PwmPeriodDefault (of the referenced channel)= 60000 4. PwmShiftValue = 70 5. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED • PwmHandleShiftByOffset = 'True' 	0x0U

1.2.5.7 Member: TimerChSR0Reg

Table 18 TimerChSR0Reg

Name	TimerChSR0Reg
Type	uint32
Description	TOM/ATOM channel SR0 register value

Verification method	The structure member is generated as the same value as structure member TimerChCM0Reg. <i>Note: This structure member is not configurable by user.</i>	
Example(s)	Action	Generated output
	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

Name	TimerChSR1Reg	
Type	uint32	
Description	TOM/ATOM channel SR1 register value	
Verification method	The structure member is generated as the same value as structure member TimerChCM1Reg. <i>Note: This structure member is not configurable by user.</i>	
Example(s)	Action	Generated output
	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

Name	TimerChIntEnMode	
Type	uint8	
Description	This structure member consists of TOM/ATOM channel interrupt mode and enables/disables the CCU0-CCU1 interrupts	
Verification method	The structure member is generated as a value: 1. Value= 0 2. Value = Value CCU0 interrupt enable/disable a. CCU0 interrupt enable = 1 b. CCU0 interrupt disable = 0 3. Value = Value CCU1 interrupt enable/disable << 1	

	a. CCU1 interrupt enable = 1 b. CCU1 interrupt disable = 0 4. Value = Value Interrupt Mode << 7 a. 00-Level Mode b. 01-Pulse Mode c. 10- Pulse Notify Mode d. 11- Single Pulse Mode <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>	
Example(s)	Action	Generated output
	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	Configuration structure holds the configuration data of the <y> PWM CCU6 channels belonging to Core <x>. <x> ranges from 0 to 5. <y> is the total number of CCU6 channels configured for 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. Note: The generated file has this structure if at least one CCU6 channel is assigned to Core <x>.	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> Configure 5 PWM channel to Core2 in ResourceM/ResourceMMcalConfig/ResourceMMcalCore/ResourceMMcalCore_<y>/ResourceMAllocation Configure 1 PWM channel with PwmAssignedHwUnit = CCU6 in in 	<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</pre>

	<p>Pwm/PwmChannelConfig Set/PwmChannel</p> <ul style="list-style-type: none"> Configure 4 PWM channel with PwmAssignedHwUnit = GTM in in Pwm/PwmChannelConfig Set/PwmChannel/ This configuration is variant-aware. Variant name is 'Petrol' 	<pre> 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 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

Name	TimerId
Type	Mcu_17_Ccu6_TimerChIdentifierType
Description	CCU6 timer channel user identifier
Verification method	<p>The structure member is generated as a value:</p> <ol style="list-style-type: none"> Value = 0 Value = Value CCU6TimerUsed If CCU6TimerUsed = T12 , Timer value = 0 If CCU6TimerUsed = T13 , Timer value = 1 Value = Value CCU6KernelUsed << 8 The value of Kernel can be 0 or 1 depending upon the configuration parameter CCU6KernelUsed Value = Value Cc6xChannel << 16 If Cc6xChannel= Cc60 , Comparator value = 0 If Cc6xChannel= Cc61, Comparator value = 1 If Cc6xChannel= Cc62, Comparator value = 2

	<p>If CCU6TimerUsed = T13, Comparator value = 3</p> <p>Cc6xChannel, CCU6KernelUsed and CCU6TimerUsed can be configured in container Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_<x>/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"> • CCU6TimerUsed = T12 • CCU6KernelUsed = /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuCcu6ModuleAllocationConf_0 • Cc6xChannel = Cc62 	0x20000
	Configure a Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0/CCU6KernelUsed with <ul style="list-style-type: none"> • CCU6TimerUsed = T13 • CCU6KernelUsed = /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuCcu6ModuleAllocationConf_1 	0x30101

1.2.6.2 Member: TimerCtrlReg0

Table 23 TimerCtrlReg0

Name	TimerCtrlReg0
Type	uint32
Description	CCU6 Timer channel control register 0 contents
Verification method	<p>The structure member is generated as a value:</p> <ol style="list-style-type: none"> 1. Value = 0 2. 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. 3. Value = Value Timer T12 Prescaler Bit << 3 If CCU6TimerPrescalerEnabled = 'True', Prescaler bit = 1 If CCU6TimerPrescalerEnabled = 'False', Prescaler bit = 0

	<p>4. Value =Value T12 Operating Mode << 7 Operating mode is not user configurable. It is by default value is set to 0 as timer always works in edge aligned mode.</p> <p>5. Value =Value Timer T13 Input Clock Select << 8 The configuration parameter Ccu6TimerClockSelect value from CCU6_CONFIGURABLE_CLOCK_0 to 7 corresponds to 000 to 111 values.</p> <p>6. Value =Value Timer T13 Prescaler Bit << 11 If CCU6TimerPrescalerEnabled = 'True', Prescaler bit = 1 If CCU6TimerPrescalerEnabled = 'False', Prescaler bit = 0</p> <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"> CCU6TimerClockSelect = CCU6_CONFIGURABLE_CLOCK_1 CCU6TimerPrescalerEnabled = 'True' CCU6TimerUsed = T12 	0x9
	Configure a Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0/CCU6KernelUsed with <ul style="list-style-type: none"> CCU6TimerClockSelect = CCU6_CONFIGURABLE_CLOCK_6 CCU6TimerPrescalerEnabled = 'False' CCU6TimerUsed = T13 	0x606

1.2.6.3 Member: ModCtrlReg

Table 24 ModCtrlReg

Name	ModCtrlReg
Type	uint32
Description	Enables/Disables modulation for T12 and T13 timer
Verification	The structure member is generated as a value:

method	<ol style="list-style-type: none"> Value = 0 Value = Value Timer T12 modulation value Modulation enable = 01 Modulation disable = 00 Value = Value Timer T13 modulation value << 2 Modulation enable = 01 Modulation disable = 00 <p><i>Note: The structure member is not user configurable; by default timer modulation is enabled.</i></p>	
Example(s)	Action	Generated output
	Configure a Pwm channel in Pwm/PwmChannelConfigSet /PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0/CCU6KernelUsed with CCU6TimerUsed = T12	0x1U
	Configure a Pwm channel in Pwm/PwmChannelConfigSet /PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0/CCU6KernelUsed with CCU6TimerUsed = T13	0x4U

1.2.6.4 Member: PasStateLvlReg

Table 25 PasStateLvlReg

Name	PasStateLvlReg	
Type	uint32	
Description	This represents the polarity of the PWM signal	
Verification method	<p>The structure member is generated as a value:</p> <ol style="list-style-type: none"> Value = 0 Value = Value Timer T12 polarity Polarity HIGH= 01 Polarity LOW =00 Value = Value Timer T13 polarity << 2 Polarity HIGH= 01 Polarity LOW =00 	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> Configure a Pwm channel in Pwm/PwmChannelConfig 	0x01U

	Set/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0/CCU6KernelUsed with CCU6TimerUsed = T12 <ul style="list-style-type: none"> Configure channel polarity in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0 with PwmPolarity = PWM_HIGH 	
	<ul style="list-style-type: none"> Configure a Pwm channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0/CCU6KernelUsed with CCU6TimerUsed = T13 Configure channel polarity in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0 with PwmPolarity = PWM_HIGH 	0x4U
	<ul style="list-style-type: none"> Configure a Pwm channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0/CCU6KernelUsed with CCU6TimerUsed = T13 Configure channel polarity in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_0 with PwmPolarity = PWM_LOW 	0x0000U

1.2.6.5 Member: TimerCntReg

Table 26 TimerCntReg

Name	TimerCntReg	
Type	uint32	
Description	CCU6 timer channel counter	
Verification method	<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>	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with <ol style="list-style-type: none"> PwmAssignedHwUnit = CCU6 PwmPeriodDefault = 0 (of the referenced channel) PwmShiftValue = 0 PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED <ul style="list-style-type: none"> PwmDutyShiftInTicks = 'True' 	0x0U
	<ul style="list-style-type: none"> Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with <ol style="list-style-type: none"> PwmAssignedHwUnit = CCU6 PwmPeriodDefault = 11125(of the referenced channel) PwmShiftValue = 20 PwmChannelClass = 	0x2b6fU

	PWM_FIXED_PERIOD_SHIFTED <ul style="list-style-type: none"> PwmDutyShiftInTicks = 'False' 	
	<ul style="list-style-type: none"> Configure channel in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_2 with <ol style="list-style-type: none"> PwmAssignedHwUnit = CCU6 PwmPeriodDefault = 11125(of the referenced channel) PwmShiftValue = 20 PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED PwmDutyShiftInTicks = 'True' 	0x2b61U

1.2.6.6 Member: TimerPeriodReg

Table 27 TimerPeriodReg

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

	<ul style="list-style-type: none"> Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with <ol style="list-style-type: none"> PwmAssignedHwUnit = CCU6 PwmPeriodDefault = 60 PwmChannelClass = PWM_FIXED_PERIOD 	0x3CU
--	--	-------

1.2.6.7 Member: Ccu6ShadowReg

Table 28 Ccu6ShadowReg

Name	Ccu6ShadowReg	
Type	uint32	
Description	CCU6 timer channel shadow register contents	
Verification method	<p>The structure member holds the scaled duty value:</p> <pre> if(PwmDutyShiftInTicks = 'False') Value = ScaledDuty Else Value = PwmDutycycleDefault Endif If (PwmPeriodDefault == 0) Value = 0 Endif If(Value >= PwmPeriodDefault) Value = PwmPeriodDefault Endif </pre>	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with <ol style="list-style-type: none"> PwmAssignedHwUnit = CCU6 PwmPeriodDefault = 0 (of the referenced channel) PwmDutycycleDefault = 16384 PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED <p>The output is same for PwmDutyShiftInTicks = 'False' and</p>	0x0U

	PwmDutyShiftInTicks = 'True' since period is zero.	
	<ul style="list-style-type: none"> Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with <ol style="list-style-type: none"> PwmAssignedHwUnit = CCU6 PwmPeriodDefault = 6000 (of the referenced channel) PwmDutycycleDefault = 16384 PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED <ul style="list-style-type: none"> The output is same for PwmDutyShiftInTicks = 'False' 	0xb8U
	<ul style="list-style-type: none"> Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with <ol style="list-style-type: none"> PwmAssignedHwUnit = CCU6 PwmPeriodDefault = 6000 (of the referenced channel) PwmDutycycleDefault = 16384 PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED <ul style="list-style-type: none"> The output is same for PwmDutyShiftInTicks = 'True' 	0xc8U
	<ul style="list-style-type: none"> Configure channel in Pwm/PwmChannelConfig Set/PwmChannel/PwmChannel_2 with <ol style="list-style-type: none"> PwmAssignedHwUnit = CCU6 PwmPeriodDefault = 6000 (of the referenced channel) PwmDutycycleDefault 	0x158U

	=5515 4. PwmChannelClass = PWM_FIXED_PERIOD_SHIFTED <ul style="list-style-type: none"> The output is same for PwmDutyShiftInTicks = 'True' 	
--	--	--

1.2.6.8 Member: TimerModeSelectReg

Table 29 TimerModeSelectReg

Name	TimerModeSelectReg	
Type	uint8	
Description	CCU6 timer mode select register contents for the input kernel	
Verification method	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>	
Example(s)	Action	Generated output
	Configure channel in Pwm/PwmChannelConfigSet /PwmChannel/PwmChannel_2 with PwmAssignedHwUnit = CCU6	0x1U

1.2.6.9 Member: PortInSelReg0

Table 30 PortInSelReg0

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

1.2.6.10 Member: IntEnReg

Table 31 IntEnReg

Name	IntEnReg	
Type	uint8	
Description	CCU6 timer channel interrupt enable register contents	
Verification method	<p>The structure member is generated as a value:</p> <ol style="list-style-type: none"> 1. Value = 0 2. Value = Value T12 or T13 period match 3. Value = Value CCU6 rising edge interrupt or T13 compare match << 1 4. Value = Value CCU6 falling edge interrupt << 2 <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>	
Example(s)	Action	Generated output
	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

Name	IntNodePointerReg	
Type	uint8	
Description	Selects Interrupt Nodes for T12 and T13 Duty and Period match.	
Verification method	<p>The structure member is generated as a value:</p> <ol style="list-style-type: none"> 1. Value = 0 2. Value = Value Interrupt node for configured T12 Comparator: Duty Match 3. Value = Value (Interrupt node for T12: Period Match) or (Interrupt node for T13: Compare Match and Period Match) << 3 <p>If CCU6TimerUsed = T13 Value = 0x0C</p> <p>Else If CCU6TimerUsed = T12</p> <p>If Cc6xChannel = Cc60 && highest configured comparator in T12 = Cc60 Value = 0x0U</p> <p>Else If Cc6xChannel = Cc60 && highest configured comparator in T12 = Cc61 Value = 0x04U</p> <p>Else If Cc6xChannel = Cc60 && highest configured comparator in T12 = Cc62 Value = 0x08U</p> <p>Else If Cc6xChannel = Cc61 && highest configured comparator in T12 = Cc61 Value = 0x05U</p> <p>Else If Cc6xChannel = Cc61 && highest configured comparator in T12 = Cc62 Value = 0x09U</p>	

	<pre> Else If Cc6xChannel = Cc62 && highest configured comparator in T12 = Cc62 Value = 0x0AU Endif Endif Cc6xChannel and CCU6TimerUsed can be configured in container Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_<x>/CCU6CC6Configuration/C CU6CC6Configuration_0. </pre>	
Example(s)	Action	Generated output
	<ul style="list-style-type: none"> Configure 2 Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel PwmChannel_0 and PwmChannel_1 Configure for PwmChannel_0 CCU6TimerUsed = T12 and Cc6xChannel = Cc60 in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_0/CCU6CC6Configuration/CCU6CC6Configuration_0 Configure for PwmChannel_1 CCU6TimerUsed = T12 and Cc6xChannel = Cc61 in Pwm/PwmChannelConfigSet/PwmChannel/PwmChannel_1/CCU6CC6Configuration/CCU6CC6Configuration_0 <p>The output is for PwmChannel_0</p>	<pre> 0x4U /* Interrupt node Select */ </pre>
	<ul style="list-style-type: none"> Configure 3 Pwm channel in Pwm/PwmChannelConfigSet/PwmChannel PwmChannel_0, PwmChannel_1 and PwmChannel_2 Configure for PwmChannel_0 CCU6TimerUsed = T12 and Cc6xChannel = Cc60 	<pre> 0x9U /* Interrupt node Select */ </pre>

	in Pwm/PwmChannelConfig Set/PwmChannel/PwmCh annel_0/CCU6CC6Configu ration/CCU6CC6Configura tion_0 <ul style="list-style-type: none"> • Configure for PwmChannel_1 CCU6TimerUsed = T12 and Cc6xChannel = Cc61 in Pwm/PwmChannelConfig Set/PwmChannel/PwmCh annel_1/CCU6CC6Configu ration/CCU6CC6Configura tion_0 • Configure for PwmChannel_2 CCU6TimerUsed = T12 and Cc6xChannel = Cc62 in 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>]

Name	Pwm_17_GtmCcu6_Config[_<variant>]
Type	Pwm_17_GtmCcu6_ConfigType
Description	Extern for root configuration structure of PWM driver which will be used during initialization.
Verification method	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.

Example(s)	Action	Generated output
	PWM is configured and is variant-aware. Variant name is 'Petrol'	<pre>extern const Pwm_17_GtmCcu6_ConfigType Pwm_17_GtmCcu6_Config_Petrol;</pre>
	PWM is configured and is variant-unaware	<pre>extern const Pwm_17_GtmCcu6_ConfigType Pwm_17_GtmCcu6_Config;</pre>

Revision history

Revision history

Major changes since the last revision

Date	Version	Description
2020-11-05	V1.0	Document is Released
2020-11-04	V0.1	- Pwm_17_GtmCcu6 driver chapter moved from MC-ISAR_TC3xx_Config_Verification_Manual_BASIC.pdf to this document - Structure member TimerChPortOutConfig is deleted from structure Pwm_kChannelConfigGtm_Core<x>

Trademarks

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

Edition 2020-11-05
Published by

Infineon Technologies AG
81726 Munich, Germany

© 2020 Infineon Technologies AG.
All Rights Reserved.

Do you have a question about this document?

Email: 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).

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.