/IPB/20_Software/SCDD Library/LVDC DSP_C

SCDD_Gtm

Software Component Detailed Design

Version: 0.1 (C0_RfR)
Printed by: I-Ritesh.K
Printed on: Tuesday, July 23, 2024

Generated from DOORS 9.5.1.2

1	**Software Component Design Description**	**1**
1.1	Introduction	1
2	**Attributes**	**2**
3	**Views**	**3**
4	**Gtm**	**4**
4.1	External Interfaces	4
4.2	Internal design	4
4.3	Requirements	5
4.3.1	ASIL	5
4.3.2	Hardware Interface	5
4.3.3	Gtm Function	7
4.3.3.1	ASIL Function	7
4.3.3.1.1	SET_PWMSR0(n,sr0)	7
4.3.3.1.2	SET_PWMSR1(n,sr1)	7
4.3.3.1.3	SET_PWM_PWMEN(n,en)	7
4.3.3.1.4	SET_SAFETY_PWMEN(n,en)	7
4.3.3.1.5	GET_PWMSTATUS()	7
4.3.3.1.6	SET_SAFETY_WDIDIS(dis)	7
4.3.3.1.7	void Gtm_CheckPwmStatus(void)	8
4.3.3.1.8	void Gtm_StopWDI(void)	8
4.3.3.1.9	void Gtm_PwmEnDis_Calc(void)	8
4.3.3.1.10	void Gtm_EnDisReg(bit en, u16 type)	9
4.3.3.1.11	void Gtm_Pri_EnDis(bit enable)	9
4.3.3.1.12	void Gtm_Sr_EnDis(bit enable)	9
4.3.3.1.13	void Gtm_Clamp_EnDis(bit enable)	9
4.3.3.1.14	void Gtm_EnDisDuty(bit en, u16 type)	9
4.3.3.1.15	void Gtm_Pri_EnDisDuty(bit enable)	9
4.3.3.1.16	void Gtm_Sr_EnDisDuty(bit enable)	9
4.3.3.1.17	void Gtm_Clamp_EnDisDuty(bit enable)	10
4.3.3.1.18	void Gtm_ResetDuty(u16 type)	10
4.3.3.1.19	void Gtm_Pri_ResetDuty(void)	10
4.3.3.1.20	void Gtm_Sr_ResetDuty(void)	10
4.3.3.1.21	void Gtm_Clamp_ResetDuty(void)	10

4.3.3.1.22	void Gtm_DebugPwm(f32 duty)	10
4.3.3.1.23	void Gtm_DebugPwmScope(f32 real,f32 real_div,f32 debug_div)	10
4.3.3.2	QM Function	10
4.3.3.2.1	void Gtm_Init(void)	10
4.3.3.2.2	void Gtm_vTom_cfg(void)	11
4.3.3.2.3	void Gtm_vTomCHcfg(void)	11
4.3.3.2.4	void Gtm_vTomCHcfg_HwVer1(void)	11
4.3.3.2.5	void Gtm_vTomCHcfg_HwVer0(void)	12
4.3.3.2.6	void Gtm_vIsr_cfg(void)	12
4.3.3.2.7	void Gtm_vTim_cfg(void)	12
4.3.3.2.8	void Gtm_Pwm_Pt_Init(void)	12
4.3.3.2.9	void Gtm_PwmEnDis_Init(void)	12

ID	Software Component Detailed Design
SCDD_Gtm_1	1 **Software Component Design Description**
SCDD_Gtm_2	1.1 **Introduction**
SCDD_Gtm_3	This document describes the needed requirements for a SWC or BSWM.
SCDD_Gtm_4	This is module is the Software Component Detail Description. It contains each SW component of each SW architecture. It is always structured in: External Interface Internal Design Requirements

ID	Software Component Detailed Design
SCDD_Gtm_5	[2] **Attributes**
SCDD_Gtm_6	Agreed attributes for SWE.3 (ENG.6)

ID	Software Component Detailed Design
SCDD_Gtm_10	[3] **Views**
SCDD_Gtm_11	**SwConstructionView**: This view is used for the sw construction process.
SCDD_Gtm_12	**SCDD_EditView**: This view is used for creating the content of SCDD

SCDD_Gtm_13

4 Gtm

SCDD_Gtm_14

4.1 External Interfaces

SCDD_Gtm_134

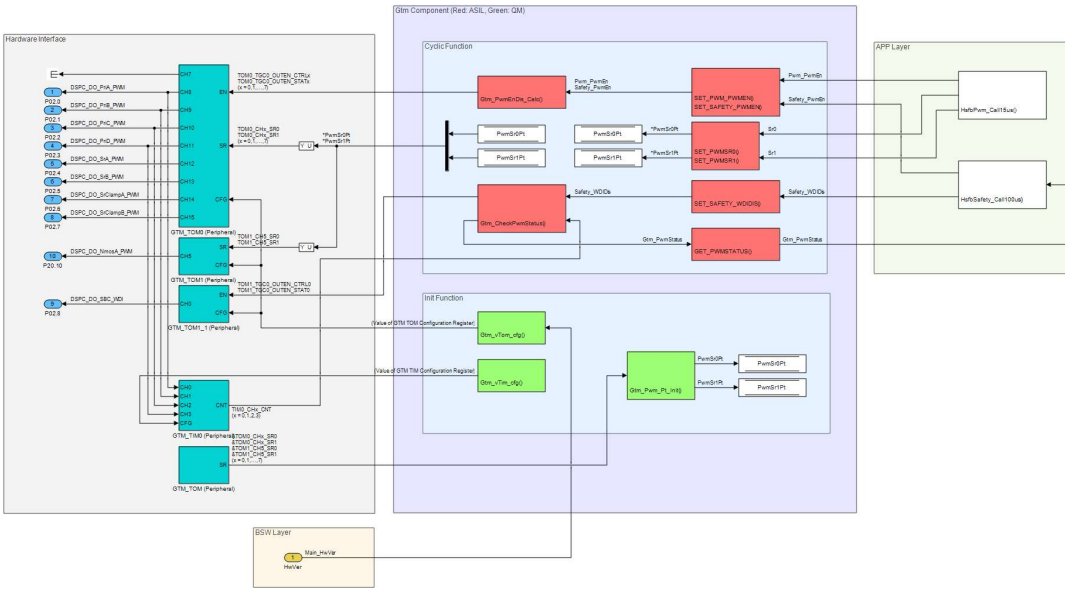
The function interface of this component are as following:

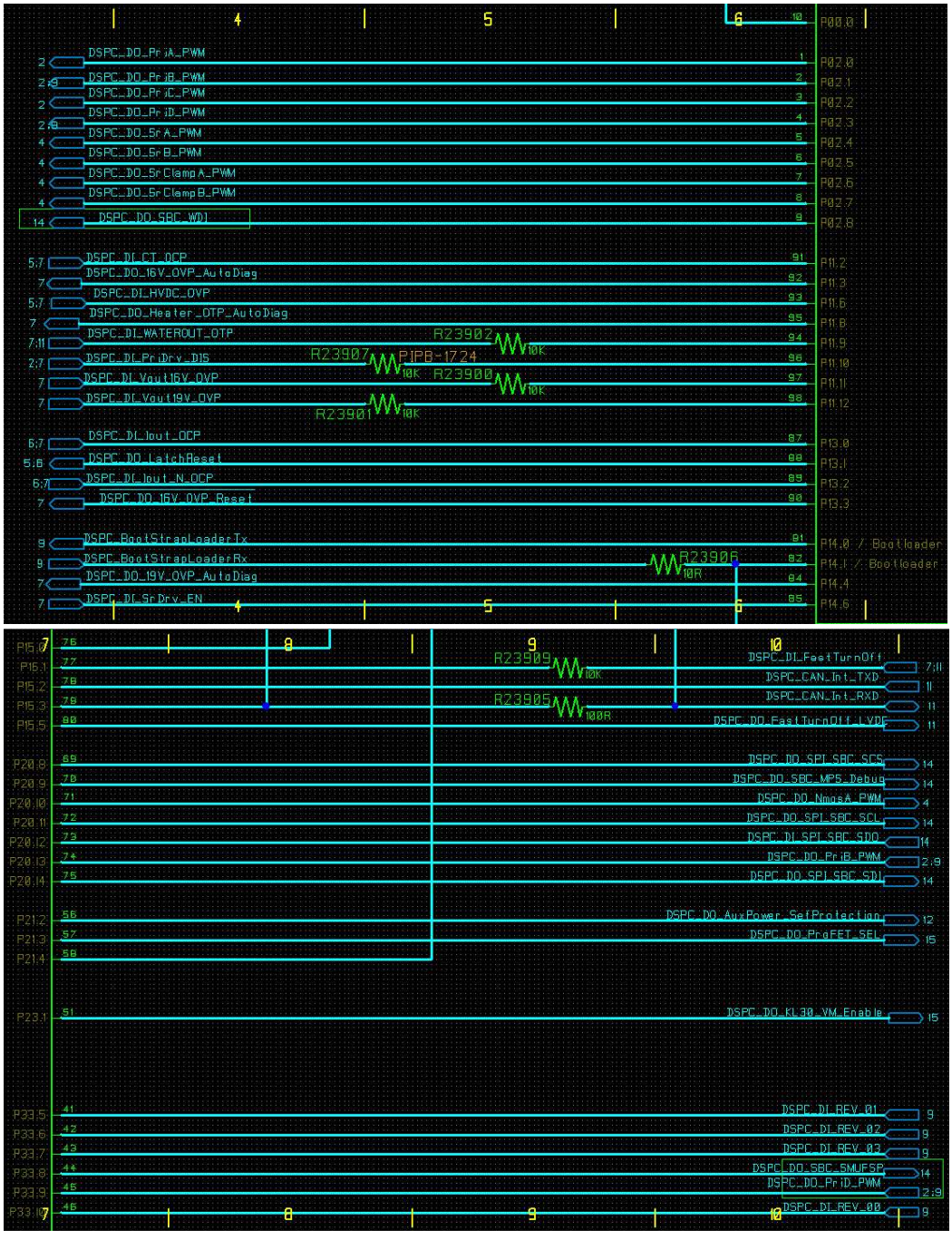
Function	Signal Name	Data Type	Direction
Gtm_Init	N/A	N/A	N/A
Gtm_Pwm_Pt_Init	N/A	N/A	N/A
Gtm_PwmEnDis_Init	N/A	N/A	N/A
Gtm_StopWDI	N/A	N/A	N/A
Gtm_CheckPwmStatus	N/A	N/A	N/A
Gtm_PwmEnDis_Calc	N/A	N/A	N/A
Gtm_EnDisReg	en	boolean	Input
Gtm_EnDisReg	type	uint16	Input
Gtm_EnDisDuty	en	boolean	Input
Gtm_EnDisDuty	type	uint16	Input
Gtm_ResetDuty	type	uint16	Input
Gtm_Pri_EnDis	en	boolean	Input
Gtm_Sr_EnDis	en	boolean	Input
Gtm_Clamp_EnDis	en	boolean	Input
Gtm_Pri_EnDisDuty	en	boolean	Input
Gtm_Sr_EnDisDuty	en	boolean	Input
Gtm_Clamp_EnDisDuty	en	boolean	Input
Gtm_Pri_ResetDuty	N/A	N/A	N/A
Gtm_Sr_ResetDuty	N/A	N/A	N/A
Gtm_Clamp_ResetDuty	N/A	N/A	N/A
Gtm_DebugPwm	duty	float32	Input
Gtm_DebugPwmScope	real	float32	Input
Gtm_DebugPwmScope	real_div	float32	Input
Gtm_DebugPwmScope	debug_div	float32	Input
Gtm_vTom_cfg	N/A	N/A	N/A
Gtm_vTomCHcfg	N/A	N/A	N/A
Gtm_vIsr_cfg	N/A	N/A	N/A
Gtm_vTomCHcfg_HwVer1	N/A	N/A	N/A
Gtm_vTomCHcfg_HwVer0	N/A	N/A	N/A
Gtm_vTim_cfg	N/A	N/A	N/A

SCDD_Gtm_16

4.2 Internal design

SCDD_Gtm_23

ID	Software Component Detailed Design
SCDD_Gtm_23	
SCDD_Gtm_18	4.3 **Requirements**
SCDD_Gtm_19	4.3.1 **ASIL**
SCDD_Gtm_24	The Gtm component is ASIL-B level.
SCDD_Gtm_20	4.3.2 **Hardware Interface**
SCDD_Gtm_25	The C code of this component can be found in following link: https://desoeap16.delta.corp/svn/IPB_PPE_auto_porsche/trunk/20_Design/23_Software/2304_Implementation/10_APPL/40_DcDcController/4010_HSFB_LVDC_B1_MBD/30_Bsw/Mcal/Gtm
SCDD_Gtm_27	The HSI of LVDC controller can be found in following link: https://desoeap16.delta.corp/svn/IPB_PPE_auto_porsche/trunk/20_Design/20_System_Design/2020_System_Archicture/50_HSI/HSI_all_C0.xlsx
SCDD_Gtm_26	The hardware digital interface of LVDC controller in schematic is shown as following:

ID	Software Component Detailed Design
SCDD_Gtm_26	

SCDD_Gtm_28

The hardware digital PWM signals of LVDC controller are as following:

Signal Name	Pin No. / Name	Description
DSPC_DO_PriA_PWM	P02.0	PriA PWM Driver
DSPC_DO_PriB_PWM	P02.1	PriB PWM Driver
DSPC_DO_PriC_PWM	P02.2	PriC PWM Driver
DSPC_DO_PriD_PWM	P02.3	PriD PWM Driver
DSPC_DO_SrA_PWM	P02.4	SrA PWM Driver
DSPC_DO_SrB_PWM	P02.5	SrB PWM Driver
DSPC_DO_SrClampA_PWM	P02.6	ClampA PWM Driver
DSPC_DO_SrClampB_PWM	P02.7	ClampB PWM Driver
DSPC_DO_NmosA_PWM	P20.10	NmosA PWM Driver

SCDD_Gtm_21

4.3.3 Gtm Function

SCDD_Gtm_29

4.3.3.1 ASIL Function

SCDD_Gtm_56

4.3.3.1.1 SET_PWMSR0(n,sr0)

SCDD_Gtm_57

This function is a macro function, which will write SR0 value to TOMx_CHy_SR0 register. The address of this register is stored in PwmSr0Pt. The input of this function is the index of PwmSr0Pt and SR0 value.

SCDD_Gtm_58

4.3.3.1.2 SET_PWMSR1(n,sr1)

SCDD_Gtm_59

This function is a macro function, which will write SR1 value to TOMx_CHy_SR1 register. The address of this register is stored in PwmSr1Pt. The input of this function is the index of PwmSr1Pt and SR1 value.

SCDD_Gtm_60

4.3.3.1.3 SET_PWM_PWMEN(n,en)

SCDD_Gtm_61

This function is a macro function, which will write enable value to Pwm_PwmEn. The input of this function is the index of Pwm_PwmEn and enable value.

SCDD_Gtm_62

4.3.3.1.4 SET_SAFETY_PWMEN(n,en)

SCDD_Gtm_63

This function is a macro function, which will write enable value to Safety_PwmEn. The input of this function is the index of Safety_PwmEn and enable value.

SCDD_Gtm_64

4.3.3.1.5 GET_PWMSTATUS()

SCDD_Gtm_65

This function is a macro function, which will return the value of Gtm_PwmStatus.

SCDD_Gtm_67

4.3.3.1.6 SET_SAFETY_WDIDIS(dis)

SCDD_Gtm_68

This function is a macro function, which will write disable value to Safety_WDIDis. The input of this function is disable value.

ID	Software Component Detailed Design
SCDD_Gtm_71	4.3.3.1.7 **void Gtm_CheckPwmStatus(void)**
SCDD_Gtm_72	This function has a cycle time of 100us. This function check the value in TIMx_CHy_CNT, if the last value of TIMx_CHy_CNT is not equal to the current value TIMx_CHy_CNT, then it means this PWM driver is turned on. The bit0 of Gtm_PwmStatus will be set to 1 if PriA is turned on (same logic with bit1, bit2, bit3 for PriB, PriC, PriD).
SCDD_Gtm_73	If Safety_PwmEn[0] is false, Gtm_PwmStatus is not 0, and SafetyWDIDis is true, then it means the Primary PWM driver is not turned off by the safety mechanism. In this case, the SBC WDI will be stopped by writing disable value to TOMx_TGCy_OUTEN register.
SCDD_Gtm_135	This function will call following function: Gtm_StopWDI()
SCDD_Gtm_136	4.3.3.1.8 **void Gtm_StopWDI(void)**
SCDD_Gtm_137	This function will stop the SBC WDI by writing disable value to TOMx_TGCy_OUTEN register. This function has no input and no output.
SCDD_Gtm_45	4.3.3.1.9 **void Gtm_PwmEnDis_Calc(void)**
SCDD_Gtm_47	This function has a cycle time of 30us. This function calculate the enable status for Primary, SR and Clamp PWM driver. The enable command is Gtm_PwmEn[i] (i = 0, 1, 2), which is calculated by Pwm_PwmEn[i] and Safety_PwmEn[i]. Only when both Pwm_PwmEn[i] and Safety_PwmEn[i] are 1 will Gtm_PwmEn[i] be set to 1.
SCDD_Gtm_49	In this function, the Gtm_PwmEnState[i] will be calculated, which has 4 states and will write value to enable register and SR register of TOM in different state.
SCDD_Gtm_50	**State GTM_E_DisReg** State GTM_E_DisReg will be actived after DSP reset. In this state, the PWM driver will be disabled by writing disable value to enable register. **State Transition GTM_E_DisReg -> GTM_E_WaitEn** After execute all the function, the state GTM_E_DisReg will be changed to state GTM_E_WaitEn.
SCDD_Gtm_53	**State GTM_E_WaitEn** State GTM_E_WaitEn is used for waiting enable command. No task will be done during this state. **State Transition GTM_E_WaitEn -> GTM_E_EnReg** If Gtm_PwmEn[i] is 1, the state GTM_E_WaitEn will be changed to state GTM_E_EnReg.
SCDD_Gtm_52	**State GTM_E_EnReg** In this state, the PWM driver will be enabled by writing enable value to enable register. **State Transition GTM_E_EnReg -> GTM_E_WaitDis** After execute all the function, the state GTM_E_EnReg will be changed to state GTM_E_WaitDis.
SCDD_Gtm_51	**State GTM_E_WaitDis** State GTM_E_WaitDis is used for waiting disable command. No task will be done during this state. After disable command is available, it will write disable duty once. **State Transition GTM_E_WaitDis -> GTM_E_DisReg** If Gtm_PwmEn[i] is 0, the state GTM_E_WaitDis will be changed to state GTM_E_DisReg.

ID	Software Component Detailed Design
SCDD_Gtm_138	This function will call following function: Gtm_EnDisReg() Gtm_EnDisDuty() Gtm_ResetDuty()
SCDD_Gtm_78	4.3.3.1.10 **void Gtm_EnDisReg(bit en, u16 type)**
SCDD_Gtm_79	This function enable/disable control of Primary, Sr and Clamp PWM driver. The type are 0, 1, 2, which stand for Pri, Sr, Clamp PWM. The input of this function are enable value and type, and no output.
SCDD_Gtm_83	This function will call following function: Gtm_Pri_EnDis() Gtm_Sr_EnDis() Gtm_Clamp_EnDis()
SCDD_Gtm_81	4.3.3.1.11 **void Gtm_Pri_EnDis(bit enable)**
SCDD_Gtm_82	This function enable/disable control of Primary PWM driver by writing enable/disable value to TOMx_TGCy_OUTEN register. The input of this function are enable value and no output.
SCDD_Gtm_93	4.3.3.1.12 **void Gtm_Sr_EnDis(bit enable)**
SCDD_Gtm_94	This function enable/disable control of Sr PWM driver by writing enable/disable value to TOMx_TGCy_OUTEN register. The input of this function are enable value and no output.
SCDD_Gtm_95	4.3.3.1.13 **void Gtm_Clamp_EnDis(bit enable)**
SCDD_Gtm_96	This function enable/disable control of Clamp PWM driver by writing enable/disable value to TOMx_TGCy_OUTEN register. The input of this function are enable value and no output.
SCDD_Gtm_84	4.3.3.1.14 **void Gtm_EnDisDuty(bit en, u16 type)**
SCDD_Gtm_85	This function enable/disable duty control of Primary, Sr and Clamp PWM driver. The type are 0, 1, 2, which stand for Pri, SR, Clamp PWM. The input of this function are enable value and type, and no output.
SCDD_Gtm_86	This function will call following function: Gtm_Pri_EnDisDuty() Gtm_Sr_EnDisDuty(bit enable) Gtm_Clamp_EnDisDuty(bit enable)
SCDD_Gtm_87	4.3.3.1.15 **void Gtm_Pri_EnDisDuty(bit enable)**
SCDD_Gtm_88	Nothing will be done during this function.
SCDD_Gtm_97	4.3.3.1.16 **void Gtm_Sr_EnDisDuty(bit enable)**

ID	Software Component Detailed Design
SCDD_Gtm_98	This function control duty of SrA/B PWM by write SR0 and SR1 value to TOMx_CHy_SR0/1 register.When enable is set to be 1, nothing will be done. The input of this function is enable value and no output.
SCDD_Gtm_99	4.3.3.1.17 **void Gtm_Clamp_EnDisDuty(bit enable)**
SCDD_Gtm_100	This function control duty of ClampA/B PWM by write SR0 and SR1 value to TOMx_CHy_SR0/1 register.When enable is set to be 1, nothing will be done. The input of this function is enable value and no output.
SCDD_Gtm_89	4.3.3.1.18 **void Gtm_ResetDuty(u16 type)**
SCDD_Gtm_90	This function reset duty control of Primary, SR and Clamp PWM. The type are 0, 1, 2, which stand for Pri, SR, Clamp PWM. The input of this function is type and no output.
SCDD_Gtm_91	This function will call following function: Gtm_Pri_ResetDuty() Gtm_Sr_ResetDuty() Gtm_Clamp_ResetDuty()
SCDD_Gtm_92	4.3.3.1.19 **void Gtm_Pri_ResetDuty(void)**
SCDD_Gtm_101	This function disable PWM drive of PrimaryA/B/C/D all the time by writing PWM_SR_MAX to TOMx_CHy_SR0 and 0 to TOMx_CHy_SR1. The function has no input and no output.
SCDD_Gtm_102	4.3.3.1.20 **void Gtm_Sr_ResetDuty(void)**
SCDD_Gtm_103	This function disable PWM drive of SrA/B all the time by writing PWM_SR_MAX to TOMx_CHy_SR0 and 0 to TOMx_CHy_SR1. The function has no input and no output.
SCDD_Gtm_105	4.3.3.1.21 **void Gtm_Clamp_ResetDuty(void)**
SCDD_Gtm_106	This function disable PWM drive of ClampA/B all the time by writing PWM_SR_MAX to TOMx_CHy_SR0 and 0 to TOMx_CHy_SR1. The function has no input and no output.
SCDD_Gtm_107	4.3.3.1.22 **void Gtm_DebugPwm(f32 duty)**
SCDD_Gtm_108	This function limit the duty value between 0 and 1, and will write DEBUG_PERIOD*duty to GTM_TOM0_CH6_SR1. The input of function is duty and no output.
SCDD_Gtm_109	4.3.3.1.23 **void Gtm_DebugPwmScope(f32 real,f32 real_div,f32 debug_div)**
SCDD_Gtm_110	This function calculate the duty value of real and 3.3v for Measurement debug, and will write DEBUG_PERIOD*duty to GTM_TOM1_CH13_SR1, the duty between 0 and 1. The input of function are real, real_div, debug_div and no output.
SCDD_Gtm_36	4.3.3.2 **QM Function**
SCDD_Gtm_111	4.3.3.2.1 **void Gtm_Init(void)**

<table>
<tr><th>ID</th><th>Software Component Detailed Design</th></tr>
<tr><td>SCDD_Gtm_112</td><td>This function is the initialization function of Gtm component. This function will intialize the configuration of Gtm_vTom, Gtm_vTim, the value of PwmSr0Pt/PwmSr1Pt and the enable status for Primary, SR and Clamp PWM driver.
This function has no input and no output.</td></tr>
<tr><td>SCDD_Gtm_113</td><td>This function will call following function:
Gtm_vTom_cfg()
Gtm_vTim_cfg()
Gtm_Pwm_Pt_Init()
Gtm_PwmEnDis_Init()</td></tr>
<tr><td>SCDD_Gtm_114</td><td>4.3.3.2.2 void Gtm_vTom_cfg(void)</td></tr>
<tr><td>SCDD_Gtm_115</td><td>This function is used for configuration of Gtm TOM module. The function will configure the TGC0/1_OUTEN_CTRL, TGC0/1_ENDIS_CTRL, TGC0/1_GLB_CTRL, TGC0/1_FUPD_CTRL register, and also configure the Tom channel, TOM Interrupt and so on.The function also enable TOM channel output and TOM trigger signal.
This function has no input and no output.</td></tr>
<tr><td>SCDD_Gtm_116</td><td>This function will call following function:
Gtm_vTomCHcfg()
Gtm_vIsr_cfg()</td></tr>
<tr><td>SCDD_Gtm_117</td><td>4.3.3.2.3 void Gtm_vTomCHcfg(void)</td></tr>
<tr><td>SCDD_Gtm_118</td><td>This function is used for configuration of Gtm TOM channel, such as CN0, SR0, SR1, CTRL register.Due to the interface difference, the configuration of TOMO_14 and TOMO_15 will be determined by HW version. And also this function will connect the gtm_adc trigger to ADC0 and ADC1 and connect the TOM_OUT to the Port.
This function has no input and no output.</td></tr>
<tr><td>SCDD_Gtm_119</td><td>The GTM TOM channel configuration is shown as following:
<table>
<tr><th>Signal Name</th><th>Pin No. / Name</th><th>Channel</th></tr>
<tr><td>DSPC_DO_PriA_PWM</td><td>P02.0</td><td>TOM0_CH8</td></tr>
<tr><td>DSPC_DO_PriB_PWM</td><td>P02.1</td><td>TOM0_CH9</td></tr>
<tr><td>DSPC_DO_PriC_PWM</td><td>P02.2</td><td>TOM0_CH10</td></tr>
<tr><td>DSPC_DO_PriD_PWM</td><td>P02.3</td><td>TOM0_CH11</td></tr>
<tr><td>DSPC_DO_SrA_PWM</td><td>P02.4</td><td>TOM0_CH12</td></tr>
<tr><td>DSPC_DO_SrB_PWM</td><td>P02.5</td><td>TOM0_CH13</td></tr>
<tr><td>DSPC_DO_SrClampA_PWM</td><td>P02.6</td><td>TOM0_CH14</td></tr>
<tr><td>DSPC_DO_SrClampB_PWM</td><td>P02.7</td><td>TOM0_CH15</td></tr>
<tr><td>DSPC_DO_NmosA_PWM</td><td>P20.10</td><td>TOM1_CH5</td></tr>
<tr><td>DSPC_DO_SBC_WDI</td><td>P20.10</td><td>TOM1_CH0</td></tr>
<tr><td>DSPC_DO_SBC_MPS_Debug</td><td>P20.9</td><td>TOM1_CH13</td></tr>
</table></td></tr>
<tr><td>SCDD_Gtm_120</td><td>This function will call following function:
Gtm_vTomCHcfg_HwVer1()
Gtm_vTomCHcfg_HwVer0()</td></tr>
<tr><td>SCDD_Gtm_121</td><td>4.3.3.2.4 void Gtm_vTomCHcfg_HwVer1(void)</td></tr>
</table>

ID	Software Component Detailed Design
SCDD_Gtm_122	This function is uesd for configuration of TOMO_14 and TOMO_15, based on B1/C0 HW version.The function will configurate the CN0, SR0, SR1, CTRL register. This function has no input and no output.
SCDD_Gtm_123	4.3.3.2.5 **void Gtm_vTomCHcfg_HwVer0(void)**
SCDD_Gtm_124	This function is uesd for configuration of TOMO_14 and TOMO_15, based on B01/B02 HW version.The function will configurate the CN0, SR0, SR1, CTRL register. This function has no input and no output.
SCDD_Gtm_125	4.3.3.2.6 **void Gtm_vIsr_cfg(void)**
SCDD_Gtm_126	This function is uesd for TOM interrupt configuration.The function will configurate the IRQ_NOTIFY, IRQ_EN, IRQ_MODE register. This function has no input and no output.
SCDD_Gtm_127	4.3.3.2.7 **void Gtm_vTim_cfg(void)**
SCDD_Gtm_128	This function is uesd for GTM TIM module configuration.The function will configurate the TIM_MODE, CICTRL, GPR0_SEL, GPR1_SEL, CNTS_SEL, DSL, TIM_EN and so on. This function has no input and no output.
SCDD_Gtm_129	4.3.3.2.8 **void Gtm_Pwm_Pt_Init(void)**
SCDD_Gtm_130	This function is used for initializing the value of PwmSr0Pt and PwmSr1Pt. The address of TOMi_CHx_SR0 register will be stored in PwmSr0Pt[i] and the address of TOMi_CHx_SR1 register will be stored in PwmSr1Pt[i]. This function has no input and no output.
SCDD_Gtm_131	4.3.3.2.9 **void Gtm_PwmEnDis_Init(void)**
SCDD_Gtm_132	This function is used for initializing the enable status for Primary, SR and Clamp PWM driver. And the Pwm_PwmEn[i], Safety_PwmEn[i] and Gtm_PwmEn[i] of PWM driver will be set to 0. And the Gtm_PwmEnState[i] will be set to GTM_E_DisReg. This function has no input and no output.