MCAL Configuration Verification Manual for DSADC

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 Dsadc

Table of contents

Table of contents

Dsadc driver

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

1.1 File: Dsadc_Cfg.h

The file is generated in 'inc' folder.

1.1.1 Macro: DSADC_AR_RELEASE_MAJOR_VERSION

Table 1 DSADC_AR_RELEASE_MAJOR_VERSION

Name	DSADC_AR_RELEASE_MAJOR_VERSION	
Description	Major version number of AUTOSAR release on which the Dsadc implementation is based on.	
Verification method	The macro is generated as 4. *Note: The macro is not user configurable.*	
Example(s)	**Action**	**Generated output**
	Generate Dsadc_Cfg.h file	`#define DSADC_AR_RELEASE_MAJOR_VERSION (4U)`

1.1.2 Macro: DSADC_AR_RELEASE_MINOR_VERSION

Table 2 DSADC_AR_RELEASE_MINOR_VERSION

Name	DSADC_AR_RELEASE_MINOR_VERSION	
Description	Minor version number of AUTOSAR release on which the Dsadc implementation is based on.	
Verification method	The macro is generated as 2. *Note: The macro is not user configurable.*	
Example(s)	**Action**	**Generated output**
	Generate Dsadc_Cfg.h file	`#define DSADC_AR_RELEASE_MINOR_VERSION (2U)`

1.1.3 Macro: DSADC_AR_RELEASE_REVISION_VERSION

Table 3 DSADC_AR_RELEASE_REVISION_VERSION

Name	DSADC_AR_RELEASE_REVISION_VERSION
Description	Revision version number of AUTOSAR release on which the Dsadc implementation is based on.
Verification	The macro is generated as 2.

method	*Note:* *The macro is not user configurable.*	
Example(s)	**Action**	**Generated output**
	Generate Dsadc_Cfg.h file	`#define DSADC_AR_RELEASE_REVISION_VERSION (2U)`

1.1.4 Macro: DSADC_SW_MAJOR_VERSION

Table 4 DSADC_SW_MAJOR_VERSION

Name	DSADC_SW_MAJOR_VERSION	
Description	Major version number of the Dsadc module.	
Verification method	The macro is generated as 10. *Note:* *The macro is not user configurable.*	
Example(s)	**Action**	**Generated output**
	Generate Dsadc_Cfg.h file	`#define DSADC_SW_MAJOR_VERSION (10U)`

1.1.5 Macro: DSADC_SW_MINOR_VERSION

Table 5 DSADC_SW_MINOR_VERSION

Name	DSADC_SW_MINOR_VERSION	
Description	Minor version number of the Dsadc module.	
Verification method	The macro is generated as 30. *Note:* *The macro is not user configurable.*	
Example(s)	**Action**	**Generated output**
	Generate Dsadc_Cfg.h file	`#define DSADC_SW_MINOR_VERSION (30U)`

1.1.6 Macro: DSADC_SW_PATCH_VERSION

Table 6 DSADC_SW_PATCH_VERSION

Name	DSADC_SW_PATCH_VERSION	
Description	Patch level version number of the Dsadc module.	
Verification method	The macro is generated as 0. *Note:* *The macro is not user configurable.*	
Example(s)	**Action**	**Generated output**
	Generate Dsadc_Cfg.h file	`#define DSADC_SW_PATCH_VERSION (0U)`

1.1.7 Macro: DSADC_SAFETY_ENABLE

Table 7 DSADC_SAFETY_ENABLE

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

1.1.8 Macro: DSADC_DISABLE_DEM_REPORT

Table 8 DSADC_DISABLE_DEM_REPORT

Name	DSADC_DISABLE_DEM_REPORT	
Description	Disables the Production Error reporting. *Note: The macro is not user configurable.*	
Verification method	The macro is always generated with value '0'.	
Example(s)	**Action**	**Generated output**
	Generate 'Dsadc_Cfg.h'	`#define DSADC_DISABLE_DEM_REPORT (0U)`

1.1.9 Macro: DSADC_ENABLE_DEM_REPORT

Table 9 DSADC_ENABLE_DEM_REPORT

Name	DSADC_ENABLE_DEM_REPORT	
Description	Enables the Production Error reporting. *Note: The macro is not user configurable.*	
Verification method	The macro is always generated with value '1'.	
Example(s)	**Action**	**Generated output**
	Generate 'Dsadc_Cfg.h'	`#define DSADC_ENABLE_DEM_REPORT (1U)`

1.1.10 Macro: DSADC_CLC_FAILURE_DEM_NOTIF

Table 10 DSADC_CLC_FAILURE_DEM_NOTIF

Name	DSADC_CLC_FAILURE_DEM_NOTIF

Description	Enables/disables the reporting of Production Error for the CLC failure	
Verification method	The macro is generated as DSADC_ENABLE_DEM_REPORT if DsadcDemEventParameterRefs/DsadcClcFailureNotification is configured else the macro is generated as DSADC_DISABLE_DEM_REPORT.	
Example(s)	**Action**	**Generated output**
	DsadcDemEventParameterRefs/ DsadcClcFailureNotification is configured	`#define DSADC_CLC_FAILURE_DEM_NOTIF (DSADC_ENABLE_DEM_REPORT)`
	DsadcDemEventParameterRefs/ DsadcClcFailureNotification is not configured	`#define DSADC_CLC_FAILURE_DEM_NOTIF (DSADC_DISABLE_DEM_REPORT)`

1.1.11 Macro: DSADC_FIFO_FAILURE_DEM_NOTIF

Table 11 DSADC_FIFO_FAILURE_DEM_NOTIF

Name	DSADC_FIFO_FAILURE_DEM_NOTIF	
Description	Enables/disables the reporting of Production Error for the HW FIFO failure	
Verification method	The macro is generated as DSADC_ENABLE_DEM_REPORT if DsadcDemEventParameterRefs/ DsadcFifoFailureNotification is configured else the macro is generated as DSADC_DISABLE_DEM_REPORT.	
Example(s)	**Action**	**Generated output**
	DsadcDemEventParameterRefs/ DsadcFifoFailureNotification is configured	`#define DSADC_FIFO_FAILURE_DEM_NOTIF (DSADC_ENABLE_DEM_REPORT)`
	DsadcDemEventParameterRefs/ DsadcFifoFailureNotification is not configured	`#define DSADC_FIFO_FAILURE_DEM_NOTIF (DSADC_DISABLE_DEM_REPORT)`

1.1.12 Macro: DSADC_E_CLC_FAILURE

Table 12 DSADC_E_CLC_FAILURE

Name	DSADC_E_CLC_FAILURE	
Description	DEM Event information for CLC failure	
Verification method	The macro is generated only when DsadcDemEventParameterRefs/ DsadcClcFailureNotification is configured else the macro is not generated.	
Example(s)	**Action**	**Generated output**
	DsadcDemEventParameterRefs/ DsadcFifoFailureNotification is configured with valid reference “DSADC_E_CLC_FAILURE”.	`#define DSADC_E_CLC_FAILURE (DemConf_DemEventParameter_DSADC_E_CLC_FAILURE)`
	DsadcDemEventParameterRefs/ DsadcFifoFailureNotification is not configured	`The macro is not generated.`

1.1.13 Macro: DSADC_E_FIFO_FAILURE

Table 13 DSADC_E_FIFO_FAILURE

Name	DSADC_E_FIFO_FAILURE	
Description	DEM Event information for HW FIFO failure	
Verification method	The macro is generated only when DsadcDemEventParameterRefs/ DsadcFifoFailureNotification is configured else the macro is not generated.	
Example(s)	**Action**	**Generated output**
	DsadcDemEventParameterRefs/ DsadcFifoFailureNotification is configured with valid reference “DSADC_E_FIFO_FAILURE”.	`#define DSADC_E_CLC_FAILURE (DemConf_DemEventParameter_DSADC_E_FIFO_FAILURE)`
	DsadcDemEventParameterRefs/ DsadcFifoFailureNotification is not configured	`The macro is not generated.`

1.1.14 Macro: DSADC_SUPERVISOR_MODE

Table 14 DSADC_SUPERVISOR_MODE

Name	DSADC_SUPERVISOR_MODE	
Description	Supervisor Mode *Note: The macro is not user configurable.*	
Verification method	The macro is always generated with value ‘0’.	
Example(s)	**Action**	**Generated output**
	Generate ‘Dsadc_Cfg.h’	`#define DSADC_SUPERVISOR_MODE (0U)`

1.1.15 Macro: DSADC_USER1_MODE

Table 15 DSADC_USER1_MODE

Name	DSADC_USER1_MODE	
Description	User Mode *Note: The macro is not user configurable.*	
Verification method	The macro is always generated with value ‘1’.	
Example(s)	**Action**	**Generated output**
	Generate ‘Dsadc_Cfg.h’	`#define DSADC_USER1_MODE (1U)`

1.1.16 Macro: DSADC_RUN_TIME_API_MODE

Table 16 DSADC_RUN_TIME_API_MODE

Name	DSADC_RUN_TIME_API_MODE	
Description	Decides the mode of execution of Run Time API's	
Verification method	The macro is generated as DSADC_USER1_MODE if DsadcRuntimeApiMode configuration parameter is set to 'DSADC_MCAL_USER1' else the macro is generated as DSADC_SUPERVISOR_MODE.	
Example(s)	**Action**	**Generated output**
	DsadcRuntimeApiMode = DSADC_MCAL_USER1	`#define DSADC_RUN_TIME_API_MODE (DSADC_USER1_MODE)`
	DsadcRuntimeApiMode = DSADC_MCAL_SUPERVISOR	`#define DSADC_RUN_TIME_API_MODE (DSADC_SUPERVISOR_MODE)`

1.1.17 Macro: DSADC_INIT_DEINIT_API_MODE

Table 17 DSADC_INIT_DEINIT_API_MODE

Name	DSADC_INIT_DEINIT_API_MODE	
Description	Decides the mode of execution of Init and DeInit API's.	
Verification method	The macro is generated as DSADC_USER1_MODE if DsadcInitDeInitApiMode configuration parameter is set to 'DSADC_MCAL_USER1' else the macro is generated as DSADC_SUPERVISOR_MODE.	
Example(s)	**Action**	**Generated output**
	DsadcInitDeInitApiMode = DSADC_MCAL_USER1	`#define DSADC_INIT_DEINIT_API_MODE (DSADC_USER1_MODE)`
	DsadcInitDeInitApiMode = DSADC_MCAL_SUPERVISOR	`#define DSADC_INIT_DEINIT_API_MODE (DSADC_SUPERVISOR_MODE)`

1.1.18 Macro: DSADC_INITCHECK_API

Table 18 DSADC_INITCHECK_API

Name	DSADC_INITCHECK_API	
Description	Enables/disables Dsadc_InitCheck API	
Verification method	The macro is generated as STD_ON if DsadcInitCheckApi configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
Example(s)	**Action**	**Generated output**
	DsadcInitCheckApi = True	`#define DSADC_INITCHECK_API (STD_ON)`
	DsadcInitCheckApi = False	`#define DSADC_INITCHECK_API (STD_OFF)`

1.1.19 Macro: DSADC_DEINIT_API

Table 19 DSADC_INITCHECK_API

Name	DSADC_DEINIT_API	
Description	Enables/disables Dsadc_DeInit API	
Verification method	The macro is generated as STD_ON if DsadcDeInitApi configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
Example(s)	**Action**	**Generated output**
	DsadcDeInitApi = True	`#define DSADC_DEINIT_API (STD_ON)`
	DsadcDeInitApi = False	`#define DSADC_DEINIT_API (STD_OFF)`

1.1.20 Macro: DSADC_VERSION_INFO_API

Table 20 DSADC_VERSION_INFO_API

Name	DSADC_VERSION_INFO_API	
Description	Enables/disables Dsadc_GetVersionInfo	
Verification method	The macro is generated as STD_ON if DsadcVersionInfoApi configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
Example(s)	**Action**	**Generated output**
	DsadcVersionInfoApi= True	`#define DSADC_VERSION_INFO_API (STD_ON)`
	DsadcVersionInfoApi= False	`#define DSADC_VERSION_INFO_API (STD_OFF)`

1.1.21 Macro: DSADC_DEV_ERROR_DETECT

Table 21 DSADC_DEV_ERROR_DETECT

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

1.1.22 Macro: DSADC_ALL_CH_RESULT_HANDLING_DMA

Table 22 DSADC_ALL_CH_RESULT_HANDLING_DMA

Name	DSADC_ALL_CH_RESULT_HANDLING_DMA	
Description	To determine whether all channels are using the DMA or not.	
Verification method	Macro is generated as STD_ON if all DSADC channel configures the parameter DsadcAccessMode as DSADC_DMA_ACCESS else the macro is generated as STD_OFF.	
Example(s)	**Action**	**Generated output**
	Assume channel 0 and channel 1 are configured. In Dsadc Channel 0: DsadcAccessMode = DSADC_DMA_ACCESS In Dsadc Channel 1: DsadcAccessMode = DSADC_DMA_ACCESS	`#define DSADC_ALL_CH_RESULT_HANDLING_DMA (STD_ON)`
	Assume channel 0 and channel 1 are configured. In Dsadc Channel 0: DsadcAccessMode = DSADC_DMA_ACCESS In Dsadc Channel 1: DsadcAccessMode = DSADC_SINGLE_READ	`#define DSADC_ALL_CH_RESULT_HANDLING_DMA (STD_OFF)`
	Assume channel 0 and channel 1 are configured. In Dsadc Channel 0: DsadcAccessMode = DSADC_STREAM_LINEAR_BUFFER In Dsadc Channel 1: DsadcAccessMode = DSADC_SINGLE_READ	`#define DSADC_ALL_CH_RESULT_HANDLING_DMA (STD_OFF)`

1.1.23 Macro: DSADC_NUM_OF_CHANNELS

Table 23 DSADC_NUM_OF_CHANNELS

Name	DSADC_NUM_OF_CHANNELS	
Description	Indicates the maximum number of channels present in the HW.	
Verification method	The macro is generated as a numeric value which corresponds to the number of elements defined in ‘Dsadc.NoOfChannels’ device specific resource properties file.	
Example(s)	**Action**	**Generated output**
	Generate Dsadc_Cfg.h	`#define DSADC_NUM_OF_CHANNELS (14U)`

1.1.24 Macro: DSADC_GTMTRIGGER_USED

Table 24 DSADC_GTMTRIGGER_USED

Name	DSADC_GTMTRIGGER_USED	
Description	To determine whether any channel is using the GTM as a trigger source.	
Verification method	Macro is generated as STD_ON if any DSADC channel configures the parameter DsadcTriggerSelect as GTM else the macro is generated as STD_OFF.	
Example(s)	**Action**	**Generated output**
	Assume channel 0 and channel 1 are configured. In Dsadc Channel 0: DsadcTriggerSelect = TRIGGER_0_NO_DSADC_TRIG In Dsadc Channel 1: DsadcTriggerSelect = TRIGGER_1_GTM_DSADC_TRIG1	`#define DSADC_GTMTRIGGER_USED (STD_ON)`
	Assume channel 0 and channel 1 are configured. In Dsadc Channel 0: DsadcTriggerSelect = TRIGGER_0_NO_DSADC_TRIG In Dsadc Channel 1: DsadcTriggerSelect = TRIGGER_6_ERU_PDOUT0	`#define DSADC_GTMTRIGGER_USED (STD_OFF)`
	Assume channel 0 and channel 1 are configured. In Dsadc Channel 0: DsadcTriggerSelect = TRIGGER_0_GTM_DSADC_TRIG0 In Dsadc Channel 1: DsadcTriggerSelect = TRIGGER_0_GTM_DSADC_TRIG0	`#define DSADC_GTMTRIGGER_USED (STD_ON)`

1.1.25 Macro: DSADC_ERUTRIGGER_USED

Table 25 DSADC_ERUTRIGGER_USED

Name	DSADC_ERUTRIGGER_USED	
Description	To determine whether any channel is using the ERU as a trigger source.	
Verification method	Macro is generated as STD_ON if any DSADC channel configures the parameter DsadcTriggerSelect as ERU else the macro is generated as STD_OFF.	
Example(s)	**Action**	**Generated output**
	Assume channel 0 and channel 1 are configured. In Dsadc Channel 0:	`#define DSADC_ERUTRIGGER_USED (STD_ON)`

DsadcTriggerSelect = TRIGGER_0_NO_DSADC_TRIG In Dsadc Channel 1: DsadcTriggerSelect = TRIGGER_6_ERU_PDOUT0	
Assume channel 0 and channel 1 are configured. In Dsadc Channel 0: DsadcTriggerSelect = TRIGGER_0_NO_DSADC_TRIG In Dsadc Channel 1: DsadcTriggerSelect = TRIGGER_1_GTM_DSADC_TRIG1	`#define DSADC_ERUTRIGGER_USED (STD_OFF)`
Assume channel 0 and channel 1 are configured. In Dsadc Channel 0: DsadcTriggerSelect = TRIGGER_6_ERU_PDOUT0 In Dsadc Channel 1: DsadcTriggerSelect = TRIGGER_6_ERU_PDOUT2	`#define DSADC_ERUTRIGGER_USED (STD_ON)`

1.1.26 Macro: DSADC_MAX_CHANNELS_CONFIGURED

Table 26 DSADC_MAX_CHANNELS_CONFIGURED

Name	DSADC_MAX_CHANNELS_CONFIGURED	
Description	Indicates the number of DSADC Channel configured.	
Verification method	The macro is generated as a total number of channels configured	
Example(s)	**Action**	**Generated output**
	Configure 3 DSADC channel	`#define DSADC_MAX_CHANNELS_CONFIGURED (3U)`
	Configure 14 DSADC channel	`#define DSADC_MAX_CHANNELS_CONFIGURED (14U)`

1.1.27 Macro: DSADC_MAX_ERS_CHANNELS_CONFIGURED

Table 27 DSADC_MAX_ERS_CHANNELS_CONFIGURED

Name	DSADC_MAX_ERS_CHANNELS_CONFIGURED	
Description	Indicates the number of ERS channels configured for DSADC driver for pattern detection.	
Verification method	The macro is generated as a total number of ERS channels configured	
Example(s)	**Action**	**Generated output**
	Configure 3 ERS channel	`#define`

	`DSADC_MAX_ERS_CHANNELS_CONFIGURED (3U)`
Configure 1 DSADC channel	`#define DSADC_MAX_ERS_CHANNELS_CONFIGURED (1U)`

1.1.28 Macro: DsadcChannel__<DsadcChannelName>

Table 28 DsadcChannel__<DsadcChannelName>

Name	DsadcChannel__<DsadcChannelName>	
Description	Indicates the symbolic name with DsadcChannelId for each configured DsadcChannel.	
Verification method	The macro is generated as a numeric value which is configured in 'DsadcConfigSet/ DsadcChannelConfiguration. < DsadcChannelId> is the name of the DSADC channel's container name.	
Example(s)	**Action**	**Generated output**

	• Configure 4 Dsadc channels.
	• Container for Dsadc Channel ID 0 is named as DsadcChannelConfiguration_0.
	• Container for Dsadc Channel ID 1 is named as DsadcChannelConfiguration_1.
	• Container for Dsadc Channel ID 2 is named as DsadcChannelConfiguration_2
	• Container for Dsadc Channel ID 3 is named as DsadcChannelConfiguration_3
		`#define DsadcChannel_DsadcChannelConfiguration_0 (0U)`
		`#define DsadcChannel_DsadcChannelConfiguration_1 (1U)`
		`#define DsadcChannel_DsadcChannelConfiguration_2 (2U)`
		`#define DsadcChannel_DsadcChannelConfiguration_0 (3U)`

1.1.29 Macro: DSADC_RESTART_INTEGRATOR_API

Table 29 DSADC_RESTART_INTEGRATOR_API

Name	DSADC_RESTART_INTEGRATOR_API	
Description	Enables/disables Dsadc_RestartIntegrator API	
Verification method	The macro is generated as STD_ON if DsadcRestartIntegratorApi configuration parameter is set to 'True' else the macro is generated as STD_OFF.	
Example(s)	**Action**	**Generated output**
	DsadcRestartIntegratorApi = True	`#define DSADC_RESTART_INTEGRATOR_API (STD_ON)`

	DsadcRestartIntegratorApi = False	`#define DSADC_RESTART_INTEGRATOR_API (STD_OFF)`

1.2 File: Dsadc[_<variant>]_PBcfg.c

The file is generated in 'src' folder.

1.2.1 Structure: Dsadc_Config[_<variant>]

Table 30 Dsadc_Config[_<variant>]

Name	Dsadc_Config[_<variant>]	
Type	Dsadc_ConfigType	
Description	Root configuration structure of DSADC driver which will be used during initialization.	
Verification method	The generated structure is present in Dsadc[_<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**

	Configure ERS channels(variant-unaware)	`const Dsadc_ConfigType Dsadc_Config =`
		`{`
		`  /* pointer to DSDAC channel configuration */`
		`  &Dsadc_kChannelConfiguration[0],`
		`  /* pointer to ERU input channel configuration */`
		`  &Dsadc_kErsInputConfiguration[0],`
		`  /* Contents of DSADC Clock control register, CLC */`
		`  0x00000000U,`
		`  /* Contents DSADC Global configuration register GLOBCFG */`
		`  0x00009000U,`
		`  /* Contents DSADC Carrier generator configuration register CGCFG */`
		`  0x00000000U,`
		`  /*Carrier Generator Waveform*/`
		`  DSADC_CARR_SIG_STOPPED`
		`};`

	ERS channels not configured(variant-unaware)	`const Dsadc_ConfigType Dsadc_Config =`

	{
	/* pointer to DSDAC channel configuration */
	&Dsadc_kChannelConfiguration[0],
	/* pointer to ERU input channel configuration */
	NULL_PTR,
	/* Contents of DSADC Clock control register, CLC */
	0x00000000U,
	/* Contents DSADC Global configuration register GLOBCFG */
	0x00009000U,
	/* Contents DSADC Carrier generator configuration register CGCFG */
	0x00000000U,
	/*Carrier Generator Waveform*/
	DSADC_CARR_SIG_STOPPED
	};

Configure ERS channels (variant-aware. Variant name is 'Gasoline')	const Dsadc_ConfigType Dsadc_Config_Gasoline =
	{
	/* pointer to DSDAC channel configuration */
	&Dsadc_kChannelConfiguration[0],
	/* pointer to ERU input channel configuration */
	&Dsadc_kErsInputConfiguration[0],
	/* Contents of DSADC Clock control register, CLC */
	0x00000000U,
	/* Contents DSADC Global configuration register GLOBCFG */
	0x00009000U,
	/* Contents DSADC Carrier generator configuration register CGCFG */
	0x00000000U,
	/*Carrier Generator Waveform*/
	DSADC_CARR_SIG_STOPPED
	};

	ERS channels not configured(variant-aware. Variant name is 'Gasoline')	const Dsadc_ConfigType Dsadc_Config_Gasoline =
		{
		/* pointer to DSDAC channel configuration */
		&Dsadc_kChannelConfiguration[0],
		/* pointer to ERU input channel configuration */
		NULL_PTR,
		/* Contents of DSADC Clock control register, CLC */
		0x00000000U,
		/* Contents DSADC Global configuration register GLOBCFG */
		0x00009000U,
		/* Contents DSADC Carrier generator configuration register CGCFG */
		0x00000000U,
		/*Carrier Generator Waveform*/
		DSADC_CARR_SIG_STOPPED
		};

1.2.1.1 Member: Dsadc_kChannelConfiguration[_variant] [x]

Table 31 Dsadc_kChannelConfiguration[_variant] [x]

Name	Dsadc_kChannelConfiguration[_variant] [x]	
Type	Dsadc_ChannelConfigType *	
Description	Configuration structure of DSADC driver for an array of channel specific configuration parameter. (x = Maximum DSADC channel configured)	
Verification method	The generated structure member is present in the Dsadc_Config[_<variant>] structure. For a variant-aware configuration, Member name is appended with the <variant> name. For variant-unaware configuration <variant> is ignored.	
Example(s)	**Action**	**Generated output**
	variant-unaware configuration	&Dsadc_kChannelConfiguration[0]
	Variant-aware. Variant name is 'Gasoline'	&Dsadc_kChannelConfiguration_Gasoline[0]

1.2.1.2 Member: Dsadc_kErsInputConfiguration[_variant] [x]

Table 32 Dsadc_kErsInputConfiguration[_variant] [x]

Name	Dsadc_kErsInputConfiguration[_variant] [x]	
Type	Dsadc_EruErsConfigType*	
Description	Configuration structure of DSADC driver for an array of ERU-ERS channel specific configuration parameter. (x = Maximum ERU-ERS channel configured).	
Verification method	The generated structure member is present in the Dsadc_Config[_<variant>] structure. For a variant-aware configuration, Member name is appended with the <variant> name. For variant-unaware configuration <variant> is ignored	
Example(s)	**Action**	**Generated output**
	Variant-aware. Variant name is 'Gasoline'	`&Dsadc_kErsInputConfiguration_Gasoline[0]`
	variant-unaware configuration	`&Dsadc_kErsInputConfiguration[0]`

1.2.1.3 Member: DsadcClcCtrlReg

Table 33 DsadcClcCtrlReg

Name	DsadcClcCtrlReg	
Type	uint32	
Description	Clock control register configuration.	
Verification method	The generated structure member is present in the Dsadc_Config[_<variant>] structure.	
Example(s)	**Action**	**Generated output**
	Configure DsadcSleepMode with SLEEP_ ENABLE	`0x00000000U, /*Configuration value for CLC register */`
	Configure DsadcSleepMode with SLEEP_DISABLE	`0x00000008U, /*Configuration value for CLC register */`

1.2.1.4 Member: GlobalConfigReg

Table 34 GlobalConfigReg

Name	GlobalConfigReg	
Type	uint32	
Description	Global configuration register configuration.	
Verification method	The structure member is generated as a value of global configuration for GLOBCFG register. Bit 8-10 stores value configured in DsadcDitheringTrimValue. Bit 12 stores value configured in DsadcSyncClockGen. Bit 13-14 stores value configured in DsadcSupplyVoltageLevel. All other bits are generated with value 0.	
Example(s)	**Action**	**Generated output**

	• Configure DsadcDitheringTrimValue with DSADC_DITHERING_MIN_50_MILVLT. • Configure DsadcSyncClockGen wikth UNSYNCHRONIZED_MODE • Configure DsadcSupplyVoltageLevel with VOLTAGESUPPLY_AUTO.	`0x00009000U, /*Configuration value for GLOBCFG register */`
	• Configure DsadcDitheringTrimValue with DSADC_DITHERING_HIGH_400_MILVLT. • Configure DsadcSyncClockGen wikth SYNCHRONIZED_MODE • Configure DsadcSupplyVoltageLevel with VOLTAGESUPPLY_3_3V.	`0x0000c700U, /*Configuration value for GLOBCFG register */`

1.2.1.5 Member: CarrierGenConfigReg

Table 35 CarrierGenConfigReg

Name	CarrierGenConfigReg	
Type	uint32	
Description	Carrier generator register configuration.	
Verification method	The structure member is generated as a value of carrier generator configuration for CGCFG register. Bit 2 stores value configured in DsadcPwmGenerationMode. Bit 3 stores value configured in DsadcCarrierSignalPolarity. Bit 4-7 stores value configured in DsadcCarrierFrequencyClockDiv. All other bits are generated with value 0.	
Example(s)	**Action**	**Generated output**
	• Configure DsadcPwmGenerationMode with DSADC_NORMAL_MODE. • Configure DsadcCarrierSignalPolarity wikth DSADC_CARR_SIG_NORMAL • Configure DsadcCarrierFrequencyClockDiv with DSADC_CG_CLOCKDIVIDER_DIV2.	`0x00000000U, /*Configuration value for CGCFG register */`
	• Configure DsadcPwmGenerationMod with	`0x000000fcU, /*Configuration value for CGCFG register */`

	DSADC_BIT_REVERSE_MODE. • Configure DsadcCarrierSignalPolarity with DSADC_CARR_SIG_INVERTED • Configure DsadcCarrierFrequencyClockDiv with DSADC_CG_CLOCKDIVIDER_DIV32.	

1.2.1.6 Member: CGWaveform

Table 36 CGWaveform

Name	CGWaveform	
Type	uint8	
Description	Indicate the carrier generator waveform type to be generated.	
Verification method	The structure member is generated as a waveform type to be generated from the carrier generator.	
Example(s)	**Action**	**Generated output**
	Configure DsadcCarrierSignalType with DSADC_CARR_SIG_SQUAREWAVE.	`DSADC_CARR_SIG_SQUAREWAVE /*Square wave */`
	Configure DsadcCarrierSignalType with DSADC_CARR_SIG_TRIANGLE.	`DSADC_CARR_SIG_TRIANGLE /*Triangular wave */`

1.2.2 Structure: Dsadc_kErsInputConfiguration[_variant] [x]

Table 37 Dsadc_kErsInputConfiguration[_variant] [x]

Name	Dsadc_kErsInputConfiguration[_variant] [x]	
Type	Dsadc_EruErsConfigType	
Description	Configuration structure of DSADC driver for ERU-ERS configuration. (x = Maximum ERS channel configured. X ranges from 0 to maximum ERS channel available in the derivative).	
Verification method	The generated file has this structure if atleast one ERS channel is configured.For a variant aware configuration the structure name is appended with the variant name. For variant unaware configuration <variant> is ignored.	
Example(s)	**Action**	**Generated output**

	Configure 1 ERS channel. (variant-aware. Variant name is ‘Gasoline’)	`static const Dsadc_EruErsConfigType Dsadc_kErsInputConfiguration_Gasoline [DSADC_MAX_ERS_CHANNELS_CONFIGURED] =`
		`{`
		`  /*Configuration of ERS Input channel 0*/`

	{
	/*EICR configuration for the given ERS input channel*/
	0x0500U,
	/*ERS channel number*/
	0x00U
	}
	};

Configure 1 ERS channel (variant-unaware)	static const Dsadc_EruErsConfigType Dsadc_kErsInputConfiguration [DSADC_MAX_ERS_CHANNELS_CONFIGURED] =
	{
	/*Configuration of ERS Input channel 0*/
	{
	/*EICR configuration for the given ERS input channel*/
	0x0500U,
	/*ERS channel number*/
	0x00U
	}
	};

Configure 3 ERS channel. (variant-aware. Variant name is 'Gasoline')	static const Dsadc_EruErsConfigType Dsadc_kErsInputConfiguration_Gasoline [DSADC_MAX_ERS_CHANNELS_CONFIGURED] = { /*Configuration of ERS Input channel 0*/ { /*EICR configuration for the given ERS input channel*/ 0x0500U, /*ERS channel number*/ 0x00U }, /*Configuration of ERS Input channel 1*/ { /*EICR configuration for the given ERS input channel*/ 0x0500U, /*ERS channel number*/

	0x01U
	},
	/*Configuration of ERS Input channel 2*/
	{
	/*EICR configuration for the given ERS input channel*/
	0x0500U,
	/*ERS channel number*/
	0x02U
	}
	};

Configure 3 ERS channel (variant-unaware)	static const Dsadc_EruErsConfigType Dsadc_kErsInputConfiguration_Gasoline [DSADC_MAX_ERS_CHANNELS_CONFIGURED] = { /*Configuration of ERS Input channel 0*/ { /*EICR configuration for the given ERS input channel*/ 0x0500U, /*ERS channel number*/ 0x00U }, /*Configuration of ERS Input channel 1*/ { /*EICR configuration for the given ERS input channel*/ 0x0500U, /*ERS channel number*/ 0x01U }, /*Configuration of ERS Input channel 2*/ { /*EICR configuration for the given ERS input channel*/ 0x0500U, /*ERS channel number*/ 0x02U

		} };

1.2.2.1 Member: EruErsEicr

Table 38 EruErsEicr

Name	EruErsEicr	
Type	uint16	
Description	Indicates the value of EICR register for the configured ERS channel.	
Verification method	This structure member is generated as a value of EICR register. Bits 4-6 stores the value configured in DsadcEruErsInputPin. Bits 8-9 stores the value configured in DsadcEruStatusFlagConfig. Bit 10 always generated with value 1. All other bits are generated with value 0.	
Example(s)	**Action**	**Generated output**
	• Configure DsadcEruErsInputPin with ERS_0_REQ0A_PORTS_P15_4. • Configure DsadcEruStatusFlagConfig with DSADC_ETL_FALLING_EDGE	`0x0500U /*EICR configuration for the given ERS input channel*/`
	• Configure DsadcEruErsInputPin e with ERS_2_REQ0C_PORTS_P10_7. • Configure DsadcEruStatusFlagConfig with DSADC_ETL_RISING_EDGE	`0x0620U /*EICR configuration for the given ERS input channel*/`

1.2.2.2 Member: ErsChannelNo

Table 39 ErsChannelNo

Name	ErsChannelNo
Type	uint8
Description	Indicates the ERS channel number configured.
Verificatio	This structure member is generated as a value of ERU-ERS channel number.

n method	ErsChannelNo stores the suffixed value of /Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuEruAllocationConf_0/McuEruChannelInputLineConf_0 after McuEruChannelInputLineConf_’ .

Example(s)	**Action**	**Generated output**
	Configure DsadcEruErsRef with McuEruChannelInputLineConf_0.	`0x00U /*ERS channel number*/`
	Configure DsadcEruErsRef with McuEruChannelInputLineConf_4.	`0x04U /*ERS channel number*/`

1.2.3 Structure: Dsadc_kChannelConfiguration[_variant] [x]

Table 40 Dsadc_kChannelConfiguration[_variant] [x]

Name	Dsadc_kChannelConfiguration[_variant] [x]	
Type	Dsadc_ChannelConfigType	
Description	Configuration structure of DSADC driver for all configured channels, where 'x' is number of channels configured and ranges from 1 to number of channels available in the hardware derivative.	
Verification method	The generated structure member is present in the Dsadc_Config[_<variant>] structure. For a variant-aware configuration, Member name is appended with the <variant> name. For variant-unaware configuration <variant> is ignored.	
Example(s)	**Action**	**Generated output**

	Configure 2 DSADC channels.(Variant Unaware)	`static const Dsadc_ChannelConfigType Dsadc_kChannelConfiguration [DSADC_MAX_CHANNELS_CONFIGURED] =`
		`{`
		`  /*Configuration of DSADC Channel Id 0*/`
		`  {`
		`  /*Address for the OGU trigger configuration structure*/`
		`  NULL_PTR,`
		`  /* Modulator Configuration Register */`
		`  0x8000800cU,`
		`  /* Common Mode Voltage Configuration Register */`
		`  0x00000000U,`
		`  /* Demodulator Configuration Register */`
		`  0x90408000U,`
		`  /* Filter Configuration Register */`

```
    0x80008008U,
    /* CIC Filter Configuration Register
*/
    0x01ff01ffU,
    /* AUX CIC Filter Configuration
Register */
    0x00000000U,
    /* Timestamp counter Register */
    0x00080000U,
    /* Integrator Window Control
Register */
    0x00000000U,
    /* Result FIFO Control Register */
    0x00000002U,
    /* Offset Compensation Register */
    0x00000000U,
    /* Gain Calibration Register */
    0x61a81170U,
    /* Gain Control Register */
    0x061b1170U,
    /* Gain Correction Register */
    0x001b1170U,
    /* Limit Checking boundary
configuration Register */
    0x00000000U,
    /* Overshoot compensation
configuration Register */
    0x00000000U,
    /* Carrier Generator Synchronization
Register */
    0x00000000U,
    /* Rectification Configuration
Register */
    0x00000000U,
    /* DSADC Channel number */
    0x00U,
    /* DSADC Channel Access Mode */
    DSADC_SINGLE_READ,
    /* DSADC Timestamp*/
    DSADC_TIMESTAMP_ENABLED,
    /* DSADC Channel Trigger Mode */
```

```
    DSADC_TRIGGER_MODE_WINDOW,
    /* DSADC Channel Trigger Source */
    DSADC_TRIGGER_GTM,
    /* DSADC Channel
DsadcGateActiveLevel */
    DSADC_GATE_LOW_LEVEL,
    /* DSADC Channel Interrupt Mode*/
    0x03U,
    /* DSADC Buffer Full Notification */
    NULL_PTR,
    /* DSADC New Result Notification */
    NULL_PTR,
    /* DSADC Window Close Notification
*/
    NULL_PTR
  },
  /*Configuration of DSADC Channel Id
1*/
  {
    /*Address for the OGU trigger
configuration structure*/
    &Dsadc_kOguTriggerConfig1,
    /* Modulator Configuration Register
*/
    0x80008000U,
    /* Common Mode Voltage Configuration
Register */
    0x00000000U,
    /* Demodulator Configuration
Register */
    0x84068000U,
    /* Filter Configuration Register */
    0x80008008U,
    /* CIC Filter Configuration Register
*/
    0x01ff01ffU,
    /* AUX CIC Filter Configuration
Register */
    0x00000000U,
    /* Timestamp counter Register */
    0x00000000U,
    /* Integrator Window Control
```

```
Register */
    0x00000000U,
    /* Result FIFO Control Register */
    0x00000000U,
    /* Offset Compensation Register */
    0x00000000U,
    /* Gain Calibration Register */
    0x61a81170U,
    /* Gain Control Register */
    0x061b1170U,
    /* Gain Correction Register */
    0x001b1170U,
    /* Limit Checking boundary
configuration Register */
    0x00000000U,
    /* Overshoot compensation
configuration Register */
    0x00000000U,
    /* Carrier Generator Synchronization
Register */
    0x00000000U,
    /* Rectification Configuration
Register */
    0x00000000U,
    /* DSADC Channel number */
    0x01U,
    /* DSADC Channel Access Mode */
    DSADC_SINGLE_READ,
    /* DSADC Timestamp*/
    DSADC_TIMESTAMP_DISABLED,
    /* DSADC Channel Trigger Mode */
    DSADC_TRIGGER_MODE_WINDOW,
    /* DSADC Channel Trigger Source */
    DSADC_TRIGGER_ERU,
    /* DSADC Channel
DsadcGateActiveLevel */
    DSADC_GATE_HIGH_LEVEL,
    /* DSADC Channel Interrupt Mode*/
    0x01U,
    /* DSADC Buffer Full Notification */
```

	NULL_PTR, /* DSADC New Result Notification */ NULL_PTR, /* DSADC Window Close Notification */ NULL_PTR }, };
Configure 3 DSADC channels. (variant-aware. Variant name is 'Gasoline')	static const Dsadc_ChannelConfigType Dsadc_kChannelConfiguration_Gasoline [DSADC_MAX_CHANNELS_CONFIGURED] = { /*Configuration of DSADC Channel Id 0*/ { /*Address for the OGU trigger configuration structure*/ NULL_PTR, /* Modulator Configuration Register */ 0x8000800cU, /* Common Mode Voltage Configuration Register */ 0x00000000U, /* Demodulator Configuration Register */ 0x90408000U, /* Filter Configuration Register */ 0x80008008U, /* CIC Filter Configuration Register */ 0x01ff01ffU, /* AUX CIC Filter Configuration Register */ 0x00000000U, /* Timestamp counter Register */ 0x00080000U, /* Integrator Window Control Register */ 0x00000000U, /* Result FIFO Control Register */ 0x00000002U,

```
    /* Offset Compensation Register */
    0x00000000U,
    /* Gain Calibration Register */
    0x61a81170U,
    /* Gain Control Register */
    0x061b1170U,
    /* Gain Correction Register */
    0x001b1170U,
    /* Limit Checking boundary
configuration Register */
    0x00000000U,
    /* Overshoot compensation
configuration Register */
    0x00000000U,
    /* Carrier Generator Synchronization
Register */
    0x00000000U,
    /* Rectification Configuration
Register */
    0x00000000U,
    /* DSADC Channel number */
    0x00U,
    /* DSADC Channel Access Mode */
    DSADC_SINGLE_READ,
    /* DSADC Timestamp*/
    DSADC_TIMESTAMP_ENABLED,
    /* DSADC Channel Trigger Mode */
    DSADC_TRIGGER_MODE_WINDOW,
    /* DSADC Channel Trigger Source */
    DSADC_TRIGGER_GTM,
    /* DSADC Channel
DsadcGateActiveLevel */
    DSADC_GATE_LOW_LEVEL,
    /* DSADC Channel Interrupt Mode*/
    0x03U,
    /* DSADC Buffer Full Notification */
    NULL_PTR,
    /* DSADC New Result Notification */
    NULL_PTR,
    /* DSADC Window Close Notification
*/
```

```
    NULL_PTR
  },
  /*Configuration of DSADC Channel Id
1*/
  {
    /*Address for the OGU trigger
configuration structure*/
    &Dsadc_kOguTriggerConfig1,
    /* Modulator Configuration Register
*/
    0x80008000U,
    /* Common Mode Voltage Configuration
Register */
    0x00000000U,
    /* Demodulator Configuration
Register */
    0x84068000U,
    /* Filter Configuration Register */
    0x80008008U,
    /* CIC Filter Configuration Register
*/
    0x01ff01ffU,
    /* AUX CIC Filter Configuration
Register */
    0x00000000U,
    /* Timestamp counter Register */
    0x00000000U,
    /* Integrator Window Control
Register */
    0x00000000U,
    /* Result FIFO Control Register */
    0x00000000U,
    /* Offset Compensation Register */
    0x00000000U,
    /* Gain Calibration Register */
    0x61a81170U,
    /* Gain Control Register */
    0x061b1170U,
    /* Gain Correction Register */
    0x001b1170U,
    /* Limit Checking boundary
configuration Register */
```

```
    0x00000000U,
    /* Overshoot compensation
configuration Register */
    0x00000000U,
    /* Carrier Generator Synchronization
Register */
    0x00000000U,
    /* Rectification Configuration
Register */
    0x00000000U,
    /* DSADC Channel number */
    0x01U,
    /* DSADC Channel Access Mode */
    DSADC_SINGLE_READ,
    /* DSADC Timestamp*/
    DSADC_TIMESTAMP_DISABLED,
    /* DSADC Channel Trigger Mode */
    DSADC_TRIGGER_MODE_WINDOW,
    /* DSADC Channel Trigger Source */
    DSADC_TRIGGER_ERU,
    /* DSADC Channel
DsadcGateActiveLevel */
    DSADC_GATE_HIGH_LEVEL,
    /* DSADC Channel Interrupt Mode*/
    0x01U,
    /* DSADC Buffer Full Notification */
    NULL_PTR,
    /* DSADC New Result Notification */
    NULL_PTR,
    /* DSADC Window Close Notification
*/
    NULL_PTR
  },
  /*Configuration of DSADC Channel Id
2*/
  {
    /*Address for the OGU trigger
configuration structure*/
    NULL_PTR,
    /* Modulator Configuration Register
*/
```

```
    0x80008000U,
    /* Common Mode Voltage Configuration
Register */
    0x00000000U,
    /* Demodulator Configuration
Register */
    0x84008000U,
    /* Filter Configuration Register */
    0x80008008U,
    /* CIC Filter Configuration Register
*/
    0x01ff01ffU,
    /* AUX CIC Filter Configuration
Register */
    0x00000000U,
    /* Timestamp counter Register */
    0x00000000U,
    /* Integrator Window Control
Register */
    0x00000000U,
    /* Result FIFO Control Register */
    0x00000000U,
    /* Offset Compensation Register */
    0x00000000U,
    /* Gain Calibration Register */
    0x61a81170U,
    /* Gain Control Register */
    0x061b1170U,
    /* Gain Correction Register */
    0x001b1170U,
    /* Limit Checking boundary
configuration Register */
    0x00000000U,
    /* Overshoot compensation
configuration Register */
    0x00000000U,
    /* Carrier Generator Synchronization
Register */
    0x00000000U,
    /* Rectification Configuration
Register */
    0x00000000U,
```

```
    /* DSADC Channel number */
    0x02U,
    /* DSADC Channel Access Mode */
    DSADC_SINGLE_READ,
    /* DSADC Timestamp*/
    DSADC_TIMESTAMP_DISABLED,
    /* DSADC Channel Trigger Mode */
    DSADC_TRIGGER_MODE_NORMAL,
    /* DSADC Channel Trigger Source */
    DSADC_TRIGGER_NONE,
    /* DSADC Channel
DsadcGateActiveLevel */
    DSADC_GATE_HIGH_LEVEL,
    /* DSADC Channel Interrupt Mode*/
    0x03U,
    /* DSADC Buffer Full Notification */
    NULL_PTR,
    /* DSADC New Result Notification */
    NULL_PTR,
    /* DSADC Window Close Notification
*/
    NULL_PTR
  },
};
```

1.2.3.1 Member: Dsadc_EruOguConf

Table 41 Dsadc_EruOguConf

Name	Dsadc_EruOguConf
Type	Dsadc_EruOguConfigType*
Description	Pointer to the ERU-OGU configuration structure.
Verification method	The structure member is generated as an address of ERU-OGU configuration structure for the corresponding DSADC channel. *Note: This parameter is user configurable only when 'DsadcTriggerSelect is configured as ERU resource.*

Example(s)	**Action**	**Generated output**
	Configure DsadcOguConfig container in DSADC channel 3	`&Dsadc_kOguTriggerConfig3, /*Address for the OGU trigger configuration`

		structure */
	Do not configure DsadcOguConfig container.	NULL_PTR, /*Address for the OGU trigger configuration structure*/

1.2.3.2 Member: ModulatorConfigReg

Table 42 ModulatorConfigReg

Name	ModulatorConfigReg	
Type	uint32	
Description	Indicates the value for modulator configuration register.	
Verification method	The structure member is generated as a value of modulator configuration for MODCFGx register. Bits 0-1 stores the value configured in DsadcPositiveInputLine. Bits 2-3 stores the value configured in DsadcNegativeInputLine. Bits 4-7 stores the value configured in DsadcInputGain. Bits 8-9 stores the value configured in DsadcInputPinSelection. Bits 12-13 stores the value configured in DsadcInputMuxControlMode. Bits 14 stores the value configured in DsadcInputMuxActionMode. Bits 16-18 stores the value configured in DsadcClockDivider. Bits 20-22 stores the value configured in DsadcAnalogClockSyncDelay. Bits 26 stores the value configured in DsadcDitheringEnable. Bits 27 stores the value configured in DsadcIntegratorResetEnable. All other bits are generated with value 0.	
Example(s)	**Action**	**Generated output**
	• Configure DsadcAnalogClockSyncDelay with 0. • Configure DsadcClockDivider with DSADC_CLOCKDIVIDER_DIV4. • Configure DsadcDitheringEnable with false. • Configure DsadcIntegratorResetEnable with false. • Configure DsadcInputGain with DSADC_INPUT_GAIN_FACTOR_1 • Configure DsadcInputMuxActionMode with DSADC_INPUTMUX_PRESET_MODE • Configure DsadcInputMuxControlMode with DSADC_INMUX_SOFTWARE_CONTROL. • Configure DsadcNegativeInputLine with DSADC_NEG_INPUT_PIN. • Configure DsadcPositiveInputLine with DSADC_POS_INPUT_PIN. • Configure DsadcInputPinSelection with	0x80008000U /* Modulator Configuration Register */

	INPUT_PIN_0_AN2_AN3.	
	• Configure DsadcAnalogClockSyncDelay with 7. • Configure DsadcClockDivider with DSADC_CLOCKDIVIDER_DIV18. • Configure DsadcDitheringEnable with true. • Configure DsadcIntegratorResetEnable with true. • Configure DsadcInputGain with DSADC_INPUT_GAIN_FACTOR_4. • Configure DsadcInputMuxActionMode with DSADC_INPUTMUX_SINGLE_STEP_MODE. • Configure DsadcInputMuxControlMode with DSADC_INMUX_TRIG_EVENT_BOTH_EDGES. • Configure DsadcNegativeInputLine with DSADC_NEG_IN_REFERENCE_GROUND. • Configure DsadcPositiveInputLine with DSADC_POS_IN_COMMON_MODE_VOLT • Configure DsadcInputPinSelection with INPUT_PIN_1_AN12_AN13.	`0x8c77b12eU /* Modulator Configuration Register */`

1.2.3.3 Member: CommonModeVoltConfigReg

Table 43 CommonModeVoltConfigReg

Name	CommonModeVoltConfigReg	
Type	uint32	
Description	Indicates the value for common mode voltage configuration register.	
Verification method	The structure member is generated as a value of common mode voltage configuration for VCMx register. Bits 0-1 stores the value configured in DsadcCommonModeVoltageSelect. Bits 2 stores the value configured in DsadcCommonModeVoltageEnable. Bits 16 stores the value configured in DsadcComModeVoltPosAEnable. Bits 17 stores the value configured in DsadcComModeVoltPosBEnable. Bits 18 stores the value configured in DsadcComModeVoltPosCEnable. Bits 19 stores the value configured in DsadcComModeVoltPosDEnable. Bits 20 stores the value configured in DsadcComModeVoltNegAEnable. Bits 21 stores the value configured in DsadcComModeVoltNegBEnable. Bits 22 stores the value configured in DsadcComModeVoltNegCEnable. Bits 23 stores the value configured in DsadcComModeVoltNegDEnable.	
Example(s)	**Action**	**Generated output**

• Configure DsadcCommonModeVoltageEnable with false. • Configure DsadcCommonModeVoltageSelect with default value. • Configure DsadcComModeVoltPosAEnable with false. • Configure DsadcComModeVoltPosBEnable with false. • Configure DsadcComModeVoltPosCEnable with false. • Configure DsadcComModeVoltPosDEnable with false. • Configure DsadcComModeVoltNegAEnable with false. • Configure DsadcComModeVoltNegBEnable with false. • Configure DsadcComModeVoltNegCEnable with false. • Configure DsadcComModeVoltNegDEnable with false.	`0x00000000U /* Common Mode Voltage Configuration Register */`
• Configure DsadcCommonModeVoltageEnable with true. • Configure DsadcCommonModeVoltageSelect with DSADC_VCM_VREFX_16. • Configure DsadcComModeVoltPosAEnable with true. • Configure DsadcComModeVoltPosBEnable with true. • Configure DsadcComModeVoltPosCEnable with false. • Configure	`0x00330007U /* Common Mode Voltage Configuration Register */`

	DsadcComModeVoltPosDEnable with false. • Configure DsadcComModeVoltNegAEnable with true. • Configure DsadcComModeVoltNegBEnable with true. • Configure DsadcComModeVoltNegCEnable with false. • Configure DsadcComModeVoltNegDEnable with false.	

1.2.3.4 Member: DemodulatorConfigReg

Table 44 DemodulatorConfigReg

Name	DemodulatorConfigReg	
Type	uint32	
Description	Indicates the value for demodulator configuration register.	
Verification method	The structure member is generated as a value of demodulator configuration for DICFGx register. Bits 16-19 stores the value configured in DsadcTriggerSelect. Bits 20-21 stores the value configured in DsadcIntegratorTriggerMode. Bits 22-23 stores the value for timestamp trigger mode. This value is derived from DsadcTimestampFeature and DsadcTriggerMode and DsadcGateActiveLevel. Bits 26-27 stores the value for data read mode. This value is derived from DsadcTimestampFeature and DsadcTriggerMode. Bits 28 stores the value configured in DsadcTimestampFeature. Bits 29 stores the value configured in DsadcResultDisplayMode. All other bits are generated with value 0.	
Example(s)	**Action**	**Generated output**
	• Configure DsadcIntegratorTriggerMode with DSADC_INTR_RISING_EDGE. • Configure DsadcTriggerSelect with TRIGGER_0_GTM_DSADC_TRIG0. • Configure DsadcResultDisplayMode with DSADC_RES_SIGNED_MODE. • Configure DsadcTimestampFeature with DSADC_TIMESTAMP_DISABLED. • Configure DsadcTriggerMode with DSADC_INPUT_GAIN_FACTOR_1 • Configure DsadcGateActiveLevel with DSADC_GATE_HIGH_LEVEL.	`0x84208000U /* Demodulator Configuration Register */`

	• Configure DsadcIntegratorTriggerMode with DSADC_INTR_FALLING_EDGE. • Configure DsadcTriggerSelect with TRIGGER_13_GTM_DSADC_TRIG3. • Configure DsadcResultDisplayMode with DSADC_RES_UNSIGNED_MODE. • Configure DsadcTimestampFeature with DSADC_TIMESTAMP_ENABLED. • Configure DsadcTriggerMode with DSADC_TRIGGER_MODE_WINDOW • Configure DsadcGateActiveLevel with DSADC_GATE_LOW_LEVEL.	`0xb05d8000U /* Demodulator Configuration Register */`

1.2.3.5 Member: FilterConfigReg

Table 45 FilterConfigReg

Name	FilterConfigReg	
Type	uint32	
Description	Indicates the value for Main Filter configuration register.	
Verification method	The structure member is generated as a value of Main filter configuration for FCFGMx register. Bits 0 stores the value configured in DsadcFIR0FilterEnable. Bits 1 stores the value configured in DsadcFIR1FilterEnable. Bits 2 stores the value configured in DsadcOvershootCompensationEn. Bits 3 stores the value configured in DsadcFIR1FilterDecimationEnable. Bits 5 stores the value configured in DsadcPreFilterEnable. Bits 8-10 stores the value configured in DsadcOffsetCompFilterEnable. Bits 11 stores the value configured in DsadcOffsetCompValueProtect. Bits 20-21 stores the value configured in DsadcAlternateServiceReq. Bits 22-23 stores the value configured in DsadcComparatorEventSelect. All other bits are generated with value 0.	
Example(s)	**Action**	**Generated output**
	• Configure DsadcFIR0FilterEnable with false. • Configure DsadcFIR1FilterEnable with false. • Configure DsadcOvershootCompensationEn with false. • Configure DsadcFIR1FilterDecimationEnable with false. • Configure DsadcPreFilterEnable with false • Configure DsadcAlternateServiceReq with DSADC_ALT_SERVICE_DISABLE.	`0x80008008U /* Filter Configuration Register */`

	• Configure DsadcOffsetCompFilterEnable with false. • Configure DsadcOffsetCompValueProtect with false. • Configure DsadcComparatorEventSelect with DSADC_RESULT_ALWAYS.	
	• Configure DsadcFIR0FilterEnable with true. • Configure DsadcFIR1FilterEnable with true. • Configure DsadcOvershootCompensationEn with true. • Configure DsadcFIR1FilterDecimationEnable with true. • Configure DsadcPreFilterEnable with true • Configure DsadcAlternateServiceReq with DSADC_COMPARATOR_EVENT. • Configure DsadcOffsetCompFilterEnable with DSADC_OFFCOMP_FILTER_RATE_2. • Configure DsadcOffsetCompValueProtect with true. • Configure DsadcComparatorEventSelect with DSADC_RESULT_INSIDE_RANGE.	`0x80108a27U /* Filter Configuration Register */`

1.2.3.6 Member: CICFilterConfigReg

Table 46 CICFilterConfigReg

Name	CICFilterConfigReg	
Type	uint32	
Description	Indicates the value for CIC Filter configuration register.	
Verification method	The structure member is generated as a value of CIC filter configuration for FCFGCx register. Bits 0-8 stores the value configured in DsadcCICFilterDecimationFactor. Bits 16-24 stores the value configured in DsadcCICFilterStartValue.	
Example(s)	**Action**	**Generated output**
	• Configure DsadcCICFilterDecimationFactor with 512. • Configure DsadcCICFilterStartValue with 512.	`0x01ff01ffU /* CIC Filter Configuration Register */`

	• Configure DsadcCICFilterDecimationFactor with 3. • Configure DsadcCICFilterStartValue with 3.	`0x00030003U /* CIC Filter Configuration Register */`

1.2.3.7 Member: AuxCICFilterConfigReg

Table 47 AuxCICFilterConfigReg

Name	AuxCICFilterConfigReg	
Type	uint32	
Description	Indicates the value for Aux CIC Filter configuration register.	
Verification method	The structure member is generated as a value of Aux CIC filter configuration for FCFGAx register. Bits 0 stores the value configured in DsadcAuxCicFilterEnable. Bits 1 stores the value configured in DsadcAuxFilterCicDecimationFactor.	
Example(s)	**Action**	**Generated output**
	• Configure DsadcAuxCicFilterEnable with false. • Configure DsadcAuxFilterCicDecimationFactor with default value.	`0x00000000U /* AUX CIC Filter Configuration Register */`
	• Configure DsadcAuxCicFilterEnable with true. • Configure DsadcAuxFilterCicDecimationFactor with DSADC_AUXCIC_OSR_32.	`0x00000003U /* CIC Filter Configuration Register */`

1.2.3.8 Member: TimeStampConfigReg

Table 48 TimeStampConfigReg

Name	TimeStampConfigReg	
Type	uint32	
Description	Indicates the value for timestamp configuration register.	
Verification method	The structure member is generated as a value of timestamp configuration for TSCNTx register. Bits 16-17 stores the value configured in DsadcTimestampCounterClockSel. Bits 19 stores the value configured in DsadcTimestampFeature. Bits 20 stores the value configured in DsadcInputMuxSetCopyEnable.	
Example(s)	**Action**	**Generated output**
	• Configure DsadcTimestampCounterClockSel with CLOCKDIVIDER_DIV8. • Configure DsadcInputMuxSetCopyEnable with	`0x001b0000U /* Timestamp counter Register */`

	true. • Configure DsadcTimestampFeature with DSADC_TIMESTAMP_ENABLED	
	• Configure DsadcTimestampCounterClockSel with default. • Configure DsadcInputMuxSetCopyEnable with false. • Configure DsadcTimestampFeature with DSADC_TIMESTAMP_DISABLED	`0x00000000U /* Timestamp counter Register */`

1.2.3.9 Member: IntegratorConfigReg

Table 49 IntegratorConfigReg

Name	IntegratorConfigReg	
Type	uint32	
Description	Indicates the value for integrator configuration register.	
Verification method	The structure member is generated as a value of integrator configuration for IWCTRx register. Bit 0-2 stores the value depends on the parameter DsadcIntegrationCount. Refer hardware user manual for more details. Bits 4 stores the value for Integrator window control and it is derived from DsadcTriggerMode. Bit 5 always generated with value 1. Bits 16-21 stores the value configured in DsadcDiscardCount. Bits 24-29 stores the value configured in DsadcIntegrationCount. All other bits are generated with value 0.	
Example(s)	**Action**	**Generated output**
	• Configure DsadcDiscardCount with 5. • Configure DsadcIntegrationCount with 15. • Configure DsadcTriggerMode with DSADC_TRIGGER_MODE_WINDOW	`0x0e050033U /* Integrator Window Control Register */`
	• Configure DsadcDiscardCount with 50. • Configure DsadcIntegrationCount with 59. • Configure DsadcTriggerMode with DSADC_TRIGGER_MODE_NORMAL	`0x3b320025U /* Integrator Window Control Register */`

1.2.3.10 Member: ResultFifoConfigReg

Table 50 ResultFifoConfigReg

Name	ResultFifoConfigReg
Type	uint32
Description	Indicates the value for result FIFO configuration register.

Verification method	The structure member is generated as a value of result FIFO configuration for RFCx register. Bits 0-1 stores the service request FIFO level which will be drivered from DsadcTimestampFeature and DsadcTriggerMode. All other bits are generated with value 0.	
Example(s)	**Action**	**Generated output**
	• Configure DsadcTimestampFeature with DSADC_TIMESTAMP_ENABLED. • Configure DsadcTriggerMode with DSADC_TRIGGER_MODE_NORMAL	`0x00000000U /* Result FIFO Control Register */`
	• Configure DsadcTimestampFeature with DSADC_TIMESTAMP_ENABLED. • Configure DsadcTriggerMode with DSADC_TRIGGER_MODE_WINDOW	`0x00000002U /* Result FIFO Control Register */`

1.2.3.11 Member: OffsetCompConfigReg

Table 51 OffsetCompConfigReg

Name	OffsetCompConfigReg	
Type	uint32	
Description	Indicates the value for offset compensation configuration register.	
Verification method	The structure member is generated as a value of offset compensation configuration for OFFCOMPx register. Bits 0-15 stores the value configured in DsadcOffsetCompValue.	
Example(s)	**Action**	**Generated output**
	• Configure DsadcOffsetCompValue with 600.	`0x00000258U /* Offset Compensation Register */`
	• Configure DsadcOffsetCompValue with 65535.	`0x0000FFFFU /* Offset Compensation Register */`

1.2.3.12 Member: GainCalibConfigReg

Table 52 GainCalibConfigReg

Name	GainCalibConfigReg	
Type	uint32	
Description	Indicates the value for gain calibration configuration register.	
Verification method	The structure member is generated as a value of gain calibration configuration for GAINCALx register. Bits 0-12 stores the value configured in DsadcGainCalibMulFactor. Bits 16-30 stores the value configured in DsadcCalibAlgoTargetValue.	
Example(s)	**Action**	**Generated output**
	• Configure DsadcGainCalibMulFactor	`0x61a81170U /* Gain Calibration`

	with 1.0899135. • Configure DsadcCalibAlgoTargetValue with 22756	Register */

1.2.3.13 Member: GainControlConfigReg

Table 53 GainControlConfigReg

Name	GainControlConfigReg	
Type	uint32	
Description	Indicates the value for gain control configuration register.	
Verification method	The structure member is generated as a value of gain control configuration for GAINCTRx register. Bits 0-12 stores the value configured in DsadcCalibGainCorrMulFactor. Bits 16-20 stores the value configured in DsadcCalibCICFilterOutputShiftPos. Bits 24-26 stores the value configured in DsadcCICDecimationRate.	
Example(s)	**Action**	**Generated output**
	• Configure DsadcCalibCICFilterOutputShiftPos with BITS_6_TO_22. • Configure DsadcCalibGainCorrMulFactor with 1.0899135 • Configure DsadcCICDecimationRate with DSADC_CIC_DECIMATION_RATE_512	0x061b1170U /* Gain Control Register */

1.2.3.14 Member: GainCorrConfigReg

Table 54 GainCorrConfigReg

Name	GainCorrConfigReg	
Type	uint32	
Description	Indicates the value for gain correction configuration register.	
Verification method	The structure member is generated as a value of gain correction configuration for GAINCORRx register. Bits 0-12 stores the value configured in DsadcGainCorrMulFactor. Bits 16-20 stores the value configured in DsadcCICFilterOutputShiftPos.	
Example(s)	**Action**	**Generated output**
	• Configure DsadcCICFilterOutputShiftPos with BITS_6_TO_22. • Configure DsadcGainCorrMulFactor with 1.0899135	0x001b1170U /* Gain Correction Register */

1.2.3.15 Member: LimitCheckingConfigReg

Table 55 LimitCheckingConfigReg

Name	LimitCheckingConfigReg	
Type	uint32	
Description	Indicates the value for limit checking configuration register.	
Verification method	The structure member is generated as a limit checking configuration for BOUNDSELx register. Bits 0-15 stores the value configured in DsadcLowerBoundaryValue. Bits 16-31 stores the value configured in DsadcUpperBoundaryValue.	
Example(s)	**Action**	**Generated output**
	• Configure DsadcLowerBoundaryValue with BITS_6_TO_22. • Configure DsadcUpperBoundaryValue with 1.0899135	`0x223f01f4U /* Limit Checking boundary configuration Register */`

1.2.3.16 Member: OvershootCompenconfigReg

Table 56 OvershootCompenconfigReg

Name	OvershootCompenconfigReg	
Type	uint32	
Description	Indicates the value for overshoot compensation filter configuration register.	
Verification method	The structure member is generated as a overshoot compensation filter configuration for OVSCFGx register. Bits 0-1 stores the value configured in DsadcSlewRateFilterStrength. Bits 2-3 stores the value configured in DsadcSlewRateFilterRunTime. Bits 4 stores the value configured in DsadcStepDetectionMode. Bits 16-26 stores the value configured in DsadcStepDetectionThreshold	
Example(s)	**Action**	**Generated output**
	• Configure DsadcSlewRateFilterStrength with DSADC_MEDIUM_FILTER_EFFECT. • Configure DsadcSlewRateFilterRunTime with DSADC_SLEWRATE_FILTR_RUNTIME_8 • Configure DsadcStepDetectionMode with DSADC_STEP_DETECT_CMP_LAST • Configure DsadcStepDetectionThreshold with 675	`0x02a3000aU /* Overshoot compensation configuration Register */`

1.2.3.17 Member: CarrierGenSyncConfigReg

Table 57 CarrierGenSyncConfigReg

Name	CarrierGenSyncConfigReg
Type	uint32

Description	Indicates the value for carrier generator synchronization configuration register.	
Verification method	The structure member is generated as a carrier generator synchronization configuration for CGSYNCx register. Bits 16-23 stores the value configured in DsadcPosSignDelayValue. Bits 24-31 stores the value configured in DsadcNegSignDelayValue.	
Example(s)	**Action**	**Generated output**
	• Configure DsadcPosSignDelayValue with 155. • Configure DsadcNegSignDelayValue with 155.	`0x9b9b0000U /* Carrier Generator Synchronization Register */`

1.2.3.18 Member: RectificationConfigReg

Table 58 RectificationConfigReg

Name	RectificationConfigReg	
Type	uint32	
Description	Indicates the value for rectification configuration register.	
Verification method	The structure member is generated as a rectification configuration for RECTCFGx register. Bits 0 stores the value configured in DsadcRectificationEnable. Bits 4-5 stores the value configured in DsadcSignSignalSource. Bits 8-11 stores the value configured in DsadcSignSignalChannel.	
Example(s)	**Action**	**Generated output**
	• Configure DsadcRectificationEnable with true. • Configure DsadcSignSignalSource with SRC_0_ON_CHIP_CARRIER_GENERATOR. • Configure DsadcSignSignalChannel with DSADC_CHANNEL_0.	`0x00000001U /* Rectification Configuration Register */`
	• Configure DsadcRectificationEnable with false. • Configure DsadcSignSignalSource with SRC_0_ON_CHIP_CARRIER_GENERATOR. • Configure DsadcSignSignalChannel with DSADC_CHANNEL_0.	`0x00000000U /* Rectification Configuration Register */`

1.2.3.19 Member: HwAssignedChannelNum

Table 59 HwAssignedChannelNum

Name	HwAssignedChannelNum
Type	uint8

Description	Indicates the Hardware channel number configured.	
Verification method	The structure member is generated as a value for the hardware channel ID configured in DsadcHwChannelNum. .	
Example(s)	**Action**	**Generated output**
	Configure DsadcHwChannelNum with DSADC_CHANNEL_0.	`0x00U /* DSADC Channel number */`
	Configure DsadcHwChannelNum with DSADC_CHANNEL_13.	`0x0DU /* DSADC Channel number */`

1.2.3.20 Member: AccessMode

Table 60 AccessMode

Name	AccessMode	
Type	uint8	
Description	Indicates the access mode configured for the channel.	
Verification method	The structure member is generated as a access mode configured in DsadcAccessMode	
Example(s)	**Action**	**Generated output**
	Configure DsadcAccessMode with DSADC_CIRCULAR_BUFFER.	`DSADC_CIRCULAR_BUFFER /*circular buffer */`
	Configure DsadcAccessMode with DSADC_SINGLE_READ.	`DSADC_SINGLE_READ /*Single read */`

1.2.3.21 Member: TimestampMode

Table 61 TimestampMode

Name	TimestampMode	
Type	uint8	
Description	Indicates the timestamp enable/disable.	
Verification method	The structure member is generated as a timestamp mode configured in DsadcTimestampFeature	
Example(s)	**Action**	**Generated output**
	Configure DsadcTimestampFeature with DSADC_TIMESTAMP_ENABLED.	`DSADC_TIMESTAMP_ENABLED /*Timestamp enabled */`
	Configure DsadcTimestampFeature with DSADC_TIMESTAMP_DISABLED.	`DSADC_TIMESTAMP_DISABLED /*Timestamp disabled*/`

1.2.3.22 Member: TriggerMode

Table 62 TriggerMode

Name	TriggerMode
Type	uint8
Description	Indicates the trigger mode configured for DSADC channel.

Verification method	The structure member is generated for the trigger mode configured in DsadcTriggerMode	
Example(s)	**Action**	**Generated output**
	Configure DsadcTriggerMode with DSADC_TRIGGER_MODE_WINDOW.	`DSADC_TRIGGER_MODE_WINDOW /*Trigger mode window */`
	Configure DsadcTriggerMode with DSADC_TRIGGER_MODE_NORMAL.	`DSADC_TRIGGER_MODE_NORMAL /*Trigger mode normal*/`

1.2.3.23 Member: TriggerSource

Table 63 TriggerSource

Name	TriggerSource	
Type	uint8	
Description	Indicates the trigger source configured for DSADC channel.	
Verification method	The structure member is generated for the trigger source configured in DsadcTriggerSelect	
Example(s)	**Action**	**Generated output**
	Configure DsadcTriggerSelect with TRIGGER_0_GTM_DSADC_TRIG0.	`DSADC_TRIGGER_GTM /*Trigger source is configured as GTM */`
	Configure DsadcTriggerSelect with TRIGGER_6_ERU_PDOUT0.	`DSADC_TRIGGER_ERU /*Trigger source is configured as ERU */`

1.2.3.24 Member: GateActiveLevel

Table 64 GateActiveLevel

Name	GateActiveLevel	
Type	uint8	
Description	Indicates the gate active level configured for DSADC channel.	
Verification method	The structure member is generated for the gate active level configured in DsadcGateActiveLevel	
Example(s)	**Action**	**Generated output**
	Configure DsadcGateActiveLevel with DSADC_GATE_LOW_LEVEL.	`DSADC_GATE_LOW_LEVEL /*Gate active level is configured as low */`
	Configure DsadcGateActiveLevel with DSADC_GATE_HIGH_LEVEL.	`DSADC_GATE_HIGH_LEVEL /*Gate active level is configured as high */`

1.2.3.25 Member: ChannelIntMode

Table 65 ChannelIntMode

Name	ChannelIntMode
Type	uint8

Description	Indicates the Interrupt mode intented for the DSADC channel.	
Verification method	The structure member is generated for the interrupt mode based on the DsadcTriggerMode , DsadcGateActiveLevel and DsadcTimestampFeature.configuration parameter.	
Example(s)	**Action**	**Generated output**
	• Configure DsadcTriggerMode with DSADC_TRIGGER_MODE_WINDOW. • Configure DsadcGateActiveLevel with DSADC_GATE_LOW_LEVEL. • Configure DsadcTimestampFeature with DSADC_TIMESTAMP_ENABLED.	`0x03U /* DSADC Channel Interrupt Mode */`
	• Configure DsadcTriggerMode with DSADC_TRIGGER_MODE_WINDOW. • Configure DsadcGateActiveLevel with DSADC_GATE_LOW_LEVEL. • Configure DsadcTimestampFeature with DSADC_TIMESTAMP_DISABLED.	`0x02U /* DSADC Channel Interrupt Mode */`
	• Configure DsadcTriggerMode with DSADC_TRIGGER_MODE_WINDOW. • Configure DsadcGateActiveLevel with DSADC_GATE_HIGH_LEVEL. • Configure DsadcTimestampFeature with DSADC_TIMESTAMP_DISABLED.	`0x01U /* DSADC Channel Interrupt Mode */`

1.2.3.26 Member: BufferFullNotifyPtr

Table 66 BufferFullNotifyPtr

Name	BufferFullNotifyPtr	
Type	Dsadc_NotifyFnPtrType	
Description	Indicates the address of application notification call back for the channel buffer full notification	
Verification method	The structure member is generated as an address of application notification call back for the channel buffer full notification configured in DsadcBufferFullNotification.	
Example(s)	**Action**	**Generated output**
	Configure DsadcBufferFullNotification as IoHwAb_DsadcNotificationbufferfull1	`/* Notification Function Address */ IoHwAb_DsadcNotificationbufferfull1,`
	DsadcBufferFullNotification is not configured	`/* Notification Function Address */ NULL_PTR,`

1.2.3.27 Member: NewResultNotifyPtr

Table 67 NewResultNotifyPtr

Name	NewResultNotifyPtr	
Type	Dsadc_NotifyFnPtrType	
Description	Indicates the address of application notification call back for the channel new result notification	
Verification method	The structure member is generated as an address of application notification call back for the new result notification configured in DsadcNewResultNotification.	
Example(s)	**Action**	**Generated output**
	Configure DsadcNewResultNotification as IoHwAb_DsadcNotificationNewResult1	`/* Notification Function Address */ IoHwAb_ DsadcNotificationNewResult1,`
	DsadcNewResultNotification is not configured	`/* Notification Function Address */ NULL_PTR,`

1.2.3.28 Member: WindowCloseNotifyPtr

Table 68 WindowCloseNotifyPtr

Name	WindowCloseNotifyPtr	
Type	Dsadc_NotifyFnPtrType	
Description	Indicates the address of application notification call back for the channel window close notification	
Verification method	The structure member is generated as an address of application notification call back for the window close notification.configured in DsadcWindowCloseNotification.	
Example(s)	**Action**	**Generated output**
	Configure DsadcWindowCloseNotification as IoHwAb_DsadcNotificationwindow1	`/* Notification Function Address */ IoHwAb_ DsadcNotificationwindow1,`
	DsadcWindowCloseNotification is not configured	`/* Notification Function Address */ NULL_PTR,`

1.2.4 Structure: Dsadc_kOguTriggerConfig[_variant]

Table 69 Dsadc_kOguTriggerConfig[_variant]

Name	Dsadc_kOguTriggerConfig[_variant]
Type	Dsadc_EruOguConfigType
Description	Configuration structure of DSADC driver for ERU-OGU configuration.
Verification method	The generated structure member is present in the Dsadc_kChannelConfiguration[_variant] [x] structure in which ERU-OGU is configured as a trigger source.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 ERU-OGU channel for DSADC channel.0 (variant-aware. Variant name is 'Gasoline')	`static const Dsadc_EruOguConfigType Dsadc_kOguTriggerConfig0_Gasoline =`
		`{`
		`  /*IGCR configuration for the given OGU channel*/`
		`  0x6007U,`
		`  /*OGU channel number */`
		`  0x01U`
		`};`

	Configure ERU-OGU channel for DSADC channel.5 (variant-aware. Variant name is 'Gasoline')	`static const Dsadc_EruOguConfigType Dsadc_kOguTriggerConfig5_Gasoline =`
		`{`
		`  /*IGCR configuration for the given OGU channel*/`
		`  0x6007U,`
		`  /*OGU channel number */`
		`  0x01U`
		`};`

	Configure ERU-OGU channel for DSADC channel.0 (variant-unaware)	`static const Dsadc_EruOguConfigType Dsadc_kOguTriggerConfig0 =`
		`{`
		`  /*IGCR configuration for the given OGU channel*/`
		`  0x6007U,`
		`  /*OGU channel number */`
		`  0x01U`
		`};`

	Configure ERU-OGU channel for DSADC channel.5 (variant-unaware)	`static const Dsadc_EruOguConfigType Dsadc_kOguTriggerConfig5 =`
		`{`
		`  /*IGCR configuration for the given OGU channel*/`
		`  0x6007U,`
		`  /*OGU channel number */`
		`  0x01U`
		`};`

1.2.4.1 Member: EruOguIgcr

Table 70 EruOguIgcr

Name	EruOguIgcr	
Type	uint16	
Description	Indicates the value of IGCR register for the configured ERU-OGU channel.	
Verification method	This structure member is generated as a value of IGCR register. Bit 0 stores the value configured in DsadcEruErsCh0PatternFlagEnable. Bit 1 stores the value configured in DsadcEruErsCh1PatternFlagEnable. Bit 2 stores the value configured in DsadcEruErsCh2PatternFlagEnable. Bit 3 stores the value configured in DsadcEruErsCh3PatternFlagEnable. Bit 4 stores the value configured in DsadcEruErsCh4PatternFlagEnable. Bit 5 stores the value configured in DsadcEruErsCh5PatternFlagEnable. Bit 6 stores the value configured in DsadcEruErsCh6PatternFlagEnable. Bit 7 stores the value configured in DsadcEruErsCh7PatternFlagEnable. Bit 13 always generated with value 1. Bit 14-15 always generated with value 1.	
Example(s)	**Action**	**Generated output**
	• Configure DsadcEruErsCh0PatternFlagEnable with true. • Configure DsadcEruErsCh1PatternFlagEnable with true. • Configure DsadcEruErsCh2PatternFlagEnable with true. • Configure DsadcEruErsCh3PatternFlagEnable with true. • Configure DsadcEruErsCh4PatternFlagEnable with true. • Configure DsadcEruErsCh5PatternFlagEnable with true. • Configure DsadcEruErsCh6PatternFlagEnable with true. • Configure DsadcEruErsCh7PatternFlagEnable with true.	`0x60FFU /*IGCR configuration for the given OGU output channel*/`
	• Configure DsadcEruErsCh0PatternFlagEnable with true. • Configure	`0x6007U /*IGCR configuration for the given OGU output channel*/`

	DsadcEruErsCh1PatternFlagEnable with true. • Configure DsadcEruErsCh2PatternFlagEnable with true. • Configure DsadcEruErsCh3PatternFlagEnable with false. • Configure DsadcEruErsCh4PatternFlagEnable with false. • Configure DsadcEruErsCh5PatternFlagEnable with false. • Configure DsadcEruErsCh6PatternFlagEnable with false. • Configure DsadcEruErsCh7PatternFlagEnable with false.	

1.2.4.2 Member: OguChannelNo

Table 71 OguChannelNo

Name	OguChannelNo	
Type	uint8	
Description	Indicates the OGU channel number configured.	
Verification method	This structure member is generated as a value of ERU-OGU channel number. OguChannelNo stores the suffixed value of '/Mcu/Mcu/McuHardwareResourceAllocationConf_0/McuEruAllocationConf_0/McuEruChannelOutputUnitConf_x' after McuEruChannelOutputUnitConf_ configured in DsadcEruOguRef .	
Example(s)	**Action**	**Generated output**
	Configure DsadcEruOguRef with McuEruChannelOutputUnitConf_0.	`0x00U /*OGU channel number*/`
	Configure DsadcEruOguRef with McuEruChannelOutputUnitConf_3.	`0x03U /*OGU channel number*/`

1.2.5 Function declaration: Dsadc_NotifyFnPtrType

Table 72 Dsadc_NotifyFnPtrType

Name	Dsadc_NotifyFnPtrType	
Type	Dsadc_NotifyFnPtrType	
Description	The extern declaration of the user defined notification function which would be invoked during New result, Buffer full and Window open events	
Verification method	The function configured in 'DsadcNewResultNotification, DsadcBufferFullNotification and DsadcWindowCloseNotification' would be populated as a prototype with extern qualifier.	
Example(s)	**Action**	**Generated output**
	Configure 'IoHwAb_DsadcNotification1' Notify function in 'DsadcNewResultNotification' parameter.	`extern void IoHwAb_DsadcNotification1 (void);`
	Configure 'IoHwAb_DsadcNotification2' Notify function in 'DsadcBufferFullNotification' parameter.	`extern void IoHwAb_DsadcNotification2 (void);`
	Configure 'IoHwAb_DsadcNotification3' Notify function in 'DsadcWindowCloseNotification' parameter.	`extern void IoHwAb_DsadcNotification3 (void);`

1.3 File: Dsadc[_<variant>]_PBcfg.h

The generated header file contains the declaration of the root configuration structure. Post-build time configuration mechanism allows configurable functionality of DSADC driver that is deployed as object code. The file is generated in ‘inc’ folder.

1.3.1 Structure: Dsadc_Config[_<variant>]

Table 73 Dsadc_Config[_<varaint>]

Name	Dsadc_Config[_<variant>]	
Type	Dsadc_ConfigType	
Description	Extern declaration of root configuration structure of DSADC driver which will be used during initialization.	
Verification method	The generated structure is present in Dsadc[_<variant>]_PBcfg.h 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**
	Configure the required DSADC channel. (variant unaware)	`/* Extern declaration of DSADC Config Root */` `extern const Dsadc_ConfigType Dsadc_Config;`
	Configure the required DSADC channel. (variant-aware. Variant name is 'Gasoline')	`/* Extern declaration of DSADC Config Root */` `extern const Dsadc_ConfigType Dsadc_Config_Gasoline;`

Revision history

Major changes since the last revision

Date	Version	Description
2023-05-23	V4.0	Document released.
2023-05-19	V3.1	Changed DEM to Production Error wherever applicable.
2021-11-30	V3.0	Document released.
2021-09-30	V2.1	Added section 1.1.29 for Macro DSADC_RESTART_INTEGRATOR_API.
2020-12-01	V2.0	Document released.
2020-12-01	V1.1	Dsadc driver chapter moved from MC-ISAR_TC3xx_Config_Verification_Manual_CD.pdf to this document.
2019-07-24	V1.0	Review comments are incorporated. Document is released.
2019-07-22	V0.1	Initial Version

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

Edition 2023-05-23
Published by
Infineon Technologies AG
81726 Munich, Germany

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.