

# MCAL Configuration Verification Manual for DIO

## 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 Dio

## Table of contents

<b>About this document.....</b>	<b>1</b>
<b>Table of contents.....</b>	<b>2</b>
<b>1 DIO driver.....</b>	<b>3</b>
1.1 File: Dio_Cfg.h.....	3
1.1.1 Macro: DIO_AR_RELEASE_MAJOR_VERSION .....	3
1.1.2 Macro: DIO_AR_RELEASE_MINOR_VERSION .....	3
1.1.3 Macro: DIO_AR_RELEASE_REVISION_VERSION .....	3
1.1.4 Macro: DIO_SW_MAJOR_VERSION .....	4
1.1.5 Macro: DIO_SW_MINOR_VERSION .....	4
1.1.6 Macro: DIO_SW_PATCH_VERSION .....	4
1.1.7 Macro: DIO_SAFETY_ENABLE.....	5
1.1.8 Macro: DIO_FLIP_CHANNEL_API .....	5
1.1.9 Macro: DIO_MASKED_WRITE_PORT_API.....	5
1.1.10 Macro: DIO_VERSION_INFO_API.....	6
1.1.11 Macro: DIO_DEV_ERROR_DETECT .....	6
1.1.12 Macro: DIO_PORTS_AVAILABLE_00_31 .....	6
1.1.13 Macro: DIO_PORTS_AVAILABLE_32_63.....	7
1.1.14 Macro: DIO_PORTS_READONLY_00_31.....	7
1.1.15 Macro: DIO_PORTS_READONLY_32_63.....	8
1.1.16 Macro: DIO_MAX_AVAILABLE_PORT .....	8
1.1.17 Macro: DIO_CHANNELGROUPCOUNT.....	8
1.1.18 Macro: DIO_PORT_<port_number>.....	9
1.1.19 Macro: DioConf_DioPort_<port_name>.....	10
1.1.20 Macro: DioConf_DioChannelGroup_<group_name>.....	10
1.1.21 Macro: DIO_CHANNEL_<port number>_<channel_number> .....	11
1.1.22 Macro: DioConf_DioChannel_<channel name> .....	12
1.1.23 Macro: DIO_PORT_<port number>_MSPIN .....	12
1.1.24 Macro: DIO_MASK_ALL_PINS_PORT<port number> .....	13
1.2 File: Dio_Lcfg.c.....	14
1.2.1 Structure: Dio_Config.....	14
1.2.1.1 Member: Dio_kPortChannelConfig .....	15
1.2.1.2 Member: Dio_kChannelGroupConfig .....	15
1.2.1.3 Member: Dio_ChannelGroupConfigSize .....	15
1.2.2 Structure: Dio_kPortChannelConfig.....	16
1.2.2.1 Member: Dio_PortIdConfig .....	17
1.2.2.2 Member: Dio_ChannelIdConfig .....	17
1.2.3 Structure: Dio_kChannelGroupConfig .....	18
1.2.3.1 Member: Mask .....	19
1.2.3.2 Member: Offset.....	20
1.2.3.3 Member: Port.....	21
<b>Revision history.....</b>	<b>22</b>

## 1 DIO driver

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

### 1.1 File: Dio\_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 the 'inc' folder.

#### 1.1.1 Macro: DIO\_AR\_RELEASE\_MAJOR\_VERSION

Table 1 DIO\_AR\_RELEASE\_MAJOR\_VERSION

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

#### 1.1.2 Macro: DIO\_AR\_RELEASE\_MINOR\_VERSION

Table 2 DIO\_AR\_RELEASE\_MINOR\_VERSION

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

#### 1.1.3 Macro: DIO\_AR\_RELEASE\_REVISION\_VERSION

Table 3 DIO\_AR\_RELEASE\_REVISION\_VERSION

<b>Name</b>	DIO_AR_RELEASE_REVISION_VERSION
-------------	---------------------------------

## DIO driver

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

## 1.1.4 Macro: DIO\_SW\_MAJOR\_VERSION

Table 4 DIO\_SW\_MAJOR\_VERSION

<b>Name</b>	DIO_SW_MAJOR_VERSION	
<b>Description</b>	Major version number of the Dio module.	
<b>Verification method</b>	The macro is generated with the value present in 'CommonPublishedInformation/SwMajorVersion'.  <i>Note: The macro is not user configurable.</i>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Generate Dio_Cfg.h file with SwMajorVersion 10	#define DIO_SW_MAJOR_VERSION (10U)

## 1.1.5 Macro: DIO\_SW\_MINOR\_VERSION

Table 5 DIO\_SW\_MINOR\_VERSION

<b>Name</b>	DIO_SW_MINOR_VERSION	
<b>Description</b>	Minor version number of the Dio module.	
<b>Verification method</b>	The macro is generated with the value present in 'CommonPublishedInformation/SwMinorVersion'.  <i>Note: The macro is not user configurable.</i>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Generate Dio_Cfg.h file with SwMinorVersion 10	#define DIO_SW_MINOR_VERSION (10U)

## 1.1.6 Macro: DIO\_SW\_PATCH\_VERSION

Table 6 DIO\_SW\_PATCH\_VERSION

<b>Name</b>	DIO_SW_PATCH_VERSION	
<b>Description</b>	Patch level version number of the Dio module.	

## DIO driver

<b>Verification method</b>	The macro is generated with the value present in 'CommonPublishedInformation/SwPatchVersion'.	
	<i>Note: The macro is not user configurable.</i>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Generate Dio_Cfg.h file with SwPatchVersion 0	#define DIO_SW_PATCH_VERSION (0U)

## 1.1.7 Macro: DIO\_SAFETY\_ENABLE

Table 7 DIO\_SAFETY\_ENABLE

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

## 1.1.8 Macro: DIO\_FLIP\_CHANNEL\_API

Table 8 DIO\_FLIP\_CHANNEL\_API

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

## 1.1.9 Macro: DIO\_MASKED\_WRITE\_PORT\_API

Table 9 DIO\_MASKED\_WRITE\_PORT\_API

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

## DIO driver

DioMaskedWritePortApi = True	#define DIO_MASKED_WRITE_PORT_API (STD_ON)
DioMaskedWritePortApi = False	#define DIO_MASKED_WRITE_PORT_API (STD_OFF)

## 1.1.10 Macro: DIO\_VERSION\_INFO\_API

Table 10 DIO\_VERSION\_INFO\_API

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

## 1.1.11 Macro: DIO\_DEV\_ERROR\_DETECT

Table 11 DIO\_DEV\_ERROR\_DETECT

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

## 1.1.12 Macro: DIO\_PORTS\_AVAILABLE\_00\_31

Table 12 DIO\_PORTS\_AVAILABLE\_00\_31

<b>Name</b>	DIO_PORTS_AVAILABLE_00_31	
<b>Description</b>	Specifies the ports between 0-31 which are available on the microcontroller.	
	<i>Note: The macro is not configurable by the user.</i>	

## DIO driver

<b>Verification method</b>	The macro is generated as a bit-wise numeric representation of ports (Port0 to Port31) available in hardware. Bit0 corresponds to Port 0 and Bit31 corresponds to Port31. 1 represents a port is available and 0 represents as not available.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Ports Available = 0,1,2,10,11,12,13,14,15,20,21,22,23,24,25,26	#define DIO_PORTS_AVAILABLE_00_31 (0x07F0FC07U)
	Ports Available = 3,4,7,9,16,22,26,27,29	#define DIO_PORTS_AVAILABLE_00_31 (0x2C410298U)

## 1.1.13 Macro: DIO\_PORTS\_AVAILABLE\_32\_63

Table 13 DIO\_PORTS\_AVAILABLE\_32\_63

<b>Name</b>	DIO_PORTS_AVAILABLE_32_63	
<b>Description</b>	Specifies the ports available on microcontroller between ports 32 to 63.	
	<i>Note: The macro is not configurable by the user.</i>	
<b>Verification method</b>	The macro is generated as a bit-wise numeric representation of ports (Port32 to Port63) available in hardware. Bit0 corresponds to Port32 and Bit31 corresponds to Port63. 1 represents a port is available and 0 represents as not available.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Ports Available = 32,33,34,40	#define DIO_PORTS_AVAILABLE_32_63 (0x00000107U)
	Ports Available = 35,36,37,41	#define DIO_PORTS_AVAILABLE_32_63 (0x00000338U)

## 1.1.14 Macro: DIO\_PORTS\_READONLY\_00\_31

Table 14 DIO\_PORTS\_READONLY\_00\_31

<b>Name</b>	DIO_PORTS_READONLY_00_31	
<b>Description</b>	Specifies the ports between 0-31 which have read-only access on the microcontroller.	
	<i>Note: The macro is not configurable by the user.</i>	
<b>Verification method</b>	The macro is generated as a bit-wise numeric representation of read-only ports (Port0 to Port31) available in hardware. Bit0 corresponds to Port0 and Bit31 corresponds to Port31. 1 represents a port is available and 0 represents as not available.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Ports Read-only = 0,2,4,6,8,10,12,27,29,31	#define DIO_PORTS_READONLY_00_31 (0xA8001555U)

## DIO driver

Ports Read-only = 1,3,5,7,11,21,28,30	#define DIO_PORTS_READONLY_00_31 (0x502008AAU)
--	---

## 1.1.15 Macro: DIO\_PORTS\_READONLY\_32\_63

Table 15 DIO\_PORTS\_READONLY\_32\_63

<b>Name</b>	DIO_PORTS_READONLY_32_63	
<b>Description</b>	Specifies the ports which are read-only on the microcontroller between ports 32-63.  <i>Note:</i> The macro is not configurable by the user.	
<b>Verification method</b>	The macro is generated as a bit-wise numeric representation of read-only ports (Port32 to Port63) available in hardware. Bit0 corresponds to Port32 and Bit31 corresponds to Port63. 1 represents a port is available and 0 represents as not available.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Ports Read-only = 32,33,40	#define DIO_PORTS_READONLY_32_63 (0x00000103U)
	Ports Read-only = 40,41	#define DIO_PORTS_READONLY_32_63 (0x00000300U)

## 1.1.16 Macro: DIO\_MAX\_AVAILABLE\_PORT

Table 16 DIO\_MAX\_AVAILABLE\_PORT

<b>Name</b>	DIO_MAX_AVAILABLE_PORT	
<b>Description</b>	Specifies the maximum available port.  <i>Note:</i> The macro is not configurable by the user.	
<b>Verification method</b>	The macro is generated as a numeric value based on the maximum port which is available in the hardware.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Maximum port available = 24	#define DIO_MAX_AVAILABLE_PORT (24U)
	Maximum port available = 41	#define DIO_MAX_AVAILABLE_PORT (41U)

## 1.1.17 Macro: DIO\_CHANNELGROUPCOUNT

Table 17 DIO\_CHANNELGROUPCOUNT

<b>Name</b>	DIO_CHANNELGROUPCOUNT	
<b>Description</b>	Specifies the number of channel groups configured	
<b>Verification method</b>	The macro is generated as a numeric value based on the number of channel groups configured in DioConfig/DioPort/DioPort_x/DioChannelGroup	



## DIO driver

Example(s)	Action	Generated output
	Set Channel group configuration at DioConfig/DioPort/DioPort_x/DioChannelGroup as DioConf_DioChannelGroup_DioChannelGroup_0 DioConf_DioChannelGroup_DioChannelGroup_1	#define DIO_CHANNELGROUPCOUNT (2U)
	Set Channel group configuration at DioConfig/DioPort/DioPort_x/DioChannelGroup as DioConf_DioChannelGroup_DioChannelGroup_1 DioConf_DioChannelGroup_DioChannelGroup_2 DioConf_DioChannelGroup_DioChannelGroup_3 DioConf_DioChannelGroup_DioChannelGroup_4	#define DIO_CHANNELGROUPCOUNT (4U)
	No Channel Group configured	#define DIO_CHANNELGROUPCOUNT (0U)

## 1.1.18 Macro: DIO\_PORT\_&lt;port\_number&gt;

Table 18 DIO\_PORT\_&lt;port\_number&gt;

Name	DIO_PORT_<port_number>	
Description	Specifies the list of ports based on availability in hardware  <i>Note: The macro is not configurable by the user.</i>	
Verification method	The macros will be generated for each available port as DIO_PORT_<port_number> where <port_number> lists all the available ports.	
Example(s)	Action	Generated output
	Available ports: 0, 1, 22, 41	#define DIO_PORT_0 ((Dio_PortType)0) #define DIO_PORT_1 ((Dio_PortType)1) #define DIO_PORT_22 ((Dio_PortType)22) #define DIO_PORT_41 ((Dio_PortType)41)
	Available ports:2,6,15,24,40	#define DIO_PORT_2 ((Dio_PortType)2) #define DIO_PORT_6 ((Dio_PortType)6)

	<pre>#define DIO_PORT_15 ((Dio_PortType) 15)  #define DIO_PORT_24 ((Dio_PortType) 24)  #define DIO_PORT_40 ((Dio_PortType) 40)</pre>
--	--

### 1.1.19 Macro: DioConf\_DioPort\_<port\_name>

**Table 19** DioConf\_DioPort\_<port name>

<b>Name</b>	DioConf_DioPort_<port_name>	
<b>Description</b>	The macro is the user-defined the symbolic name of the container present in the configuration container “DioConfig/DioPort” which corresponds to a port.	
<b>Verification method</b>	<p>The macro is generated based on the container name present in ‘DioConfig/DioPort’ and the value corresponds to the port number associated.</p> <p>The macros will be generated for all available containers configured in “DioConfig/DioPort”</p>	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Set Container which corresponds to Port 0 in container ‘DioConfig/DioPort’ as Motor_Control	<pre>#define DioConf_DioPort_Motor_Control (DIO_PORT_0)</pre>
	Set Container which corresponds to Port 12 in container ‘DioConfig/DioPort’ as Spi_TRX	<pre>#define DioConf_DioPort_Spi_TRX (DIO_PORT_12)</pre>
	Set Container which corresponds to Port 41 ‘DioConfig/DioPort’ as CAN_RX	<pre>#define DioConf_DioPort_CAN_RX (DIO_PORT_41)</pre>

### 1.1.20 Macro: DioConf\_DioChannelGroup\_<group\_name>

**Table 20** DioConf\_DioChannelGroup\_<group\_name>

<b>Name</b>	DioConf_DioChannelGroup_<group_name>	
<b>Description</b>	The macro is the user-defined symbolic name generated for the configuration parameter “DioConfig/DioPort/DioPort/DioChannelGroup”	
<b>Verification method</b>	The macro is generated as a pointer to structure Dio_kChannelGroupConfig corresponding to the channel group and the port in which it is configured.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Available Ports: 0,21,33 Configure DioChannelGroup_0 as Ethernet at container	<pre>#define DioConf_DioChannelGroup_Ethernet</pre>

## DIO driver

'DioConfig/DioPort/DioPort_0/DioChannelGroup'	(&Dio_Config.Dio_ChannelGroupConfigPtr[0])
Configure DioChannelGroup_1 as LinChannel at container DioConfig/DioPort/DioPort_21/DioChannelGroup	#define DioConf_DioChannelGroup_LinChannel (&Dio_Config.Dio_ChannelGroupConfigPtr[1])
Configure DioChannelGroup_2 as BrkCntrl at container DioConfig/DioPort/DioPort_33/DioChannelGroup	#define DioConf_DioChannelGroup_BrkCntrl (&Dio_Config.Dio_ChannelGroupConfigPtr[2])

## 1.1.21 Macro: DIO\_CHANNEL\_&lt;port number&gt;\_&lt;channel number&gt;

Table 21 DIO\_CHANNEL\_&lt;port number&gt;\_&lt;channel number&gt;

<b>Name</b>	DIO_CHANNEL_<port number>_<channel number>	
<b>Description</b>	Specifies the list of channels based on availability in hardware  <i>Note: The macro is not configurable by the user.</i>	
<b>Verification method</b>	The macros will be generated for each available port pin as DIO_CHANNEL_<port number>_<channel number> where <port number> lists all the available ports and <channel number> list all the available channels. The value for each port channel is generated as (<port number> * 16 + <channel number>).	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Available Channels: Port 0 Pin0 Port 22 Pin 3 Port 41 Pin 15	#define DIO_CHANNEL_0_0 ((Dio_ChannelTypeType)0x0000)  #define DIO_CHANNEL_22_3 ((Dio_ChannelType)0x0163)  #define DIO_CHANNEL_41_15 ((Dio_ChannelType)0x29F)
	Available Channels: Port 10 Pin 7 Port 24 Pin 13 Port 31 Pin 5 Port 40 Pin 8	#define DIO_CHANNEL_10_7 ((Dio_ChannelType)0x00a7) #define DIO_CHANNEL_24_13 ((Dio_ChannelType)0x018d) #define DIO_CHANNEL_31_5 ((Dio_ChannelType)0x01f5) #define DIO_CHANNEL_40_8 ((Dio_ChannelType)0x0288)

### 1.1.22 Macro: DioConf\_DioChannel\_<channel name>

**Table 22** DioConf\_DioChannel\_<channel name>

Name	DioConf_DioChannel_<channel name>	
Description	Specifies the symbolic name generated for the configuration parameter “DioConfig/DioPort/DioPort/DioChannel/DioChannel”	
Verification method	The macro is generated for each container present in “DioConfig/DioPort/DioPort/DioChannel/DioChannel”. The value is the macro generated based on the port and channel associated with the container. Refer Section 1.1.21 for the macro.	
Example(s)	Action	Generated output
	Configure Container which corresponds to Port 0 pin 0 at ‘DioConfig/DioPort/DioPort_0/DioChannel’ as MotorControl	#define DioConf_DioChannel_MotorControl (DIO_CHANNEL_0_0)
	Configure Container which corresponds to Port 22 pin 4 at ‘DioConfig/DioPort/DioPort_22/DioChannel’ as SpiChannel	#define DioConf_DioChannel_SpiChannel (DIO_CHANNEL_22_4)
	Configure Container which corresponds to Port 41 pin 15 at ‘DioConfig/DioPort/DioPort/DioChannel’ as CanTRX	#define DioConf_DioChannel_CanTRX (DIO_CHANNEL_41_15)
	Configure Container which corresponds to Port 0 pin 0 at ‘DioConfig/DioPort/DioPort_0/DioChannel’ as MotorControl	#define DioConf_DioChannel_MotorControl (DIO_CHANNEL_0_0)
	Configure Container which corresponds to Port 22 pin 4 at ‘DioConfig/DioPort/DioPort_22/DioChannel’ as SpiChannel	#define DioConf_DioChannel_SpiChannel (DIO_CHANNEL_22_4)
	Configure Container which corresponds to Port 41 pin 15 at ‘DioConfig/DioPort/DioPort/DioChannel’ as CanTRX	#define DioConf_DioChannel_CanTRX (DIO_CHANNEL_41_15)

### 1.1.23 Macro: DIO\_PORT\_<port number>\_MSPIN

**Table 23** DIO\_PORT\_<port number>\_MSPIN

Name	DIO_PORT_<port number>_MSPIN
Description	Indicates the most significant pin of the port.

	<i>Note:</i> The macro is not configurable by the user.	
<b>Verification method</b>	The macro is generated for each available port and the value corresponds to the most significant pin available for that port.  <i>Note:</i> In case of no pins on a port, this value is kept zero.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	For port 0 the most significant pin is 12	#define DIO_PORT_0_MSPIN (12U)
	For port 23 the most significant pin is 15	#define DIO_PORT_23_MSPIN (15U)
	For port 26 the most significant pin is 0	#define DIO_PORT_26_MSPIN (0U)

### 1.1.24 Macro: DIO\_MASK\_ALL\_PINS\_PORT<port number>

**Table 24** DIO\_MASK\_ALL\_PINS\_PORT<port number>

<b>Name</b>	DIO_MASK_ALL_PINS_PORT<port number>	
<b>Description</b>	Indicates the bitwise representation to mask the valid pins within a port.  <i>Note:</i> The macro is not configurable by the user.	
<b>Verification method</b>	The macro is generated for each available port and the value is generated as a bitwise numeric representation of valid pins for a port. 1 indicates pin is available else 0.  <i>Note:</i> In case of undefined pins on a port, this value is kept zero.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Port 0 has all pins available 0-15	#define DIO_MASK_ALL_PINS_PORT0 (0xFFFFU)
	Port 10 has all pins except 3,8,9	#define DIO_MASK_ALL_PINS_PORT10 (0xFCF7U)
	Port 16 has no pins available	#define DIO_MASK_ALL_PINS_PORT16 (0x0000U)
	Port 41 has pins available from 0-7	#define DIO_MASK_ALL_PINS_PORT41 (0x00FFU)

## 1.2 File: Dio\_Lcfg.c

The generated file contains all link time configuration parameters. Link time configuration mechanism allows configurable functionality of DIO driver that is delivered as object code. The file is generated in the 'src' folder.

### 1.2.1 Structure: Dio\_Config

**Table 25 Dio\_Config**

Name	Dio_Config	
Type	Dio_ConfigType	
Description	Root configuration structure of DIO driver which will be used during initialization.	
Verification method	The generated structure is present in Dio_Lcfg.c file.	
Example(s)	Action	Generated output
	Set Port Configuration at DioConfig/DioPort and Channel group configuration at DioConfig/DioPort/DioPort/DioChannelGroup	<pre>const Dio_ConfigType Dio_Config = { /*Dio Port and Channelconfiguration*/     &amp;Dio_kPortChannelConfig[0],  /* Dio Channelgroup configuration */     &amp;Dio_kChannelGroupConfig[0],  /* Configured number of Dio Channelgroups for configuration */     DIO_CHANNELGROUPCOUNT };</pre>
	Set Configuration only for Port at DioConfig/DioPort (No channel group is configured)	<pre>const Dio_ConfigType Dio_Config = { /*Dio Port and Channelconfiguration*/     &amp;Dio_kPortChannelConfig[0], /* Dio Channelgroup configuration */     Null PTR, /* Configured number of Dio Channelgroups for configuration */     DIO_CHANNELGROUPCOUNT };</pre>

### 1.2.1.1 Member: Dio\_kPortChannelConfig

**Table 26** Dio\_kPortChannelConfig[]

<b>Name</b>	Dio_kPortChannelConfig	
<b>Type</b>	Dio_PortChannelIdType *	
<b>Description</b>	Pointer to an array of a structure containing port and channel configuration.	
<b>Verification method</b>	The generated structure member is present in the Dio_Config structure and It is always generated as the base address of the Dio_kportChannelConfig.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure any available port at container DioConfig/DioPort	<pre>{ /*Dio Port and Channelconfiguration*/     &amp;Dio_kPortChannelConfig[0], }</pre>

### 1.2.1.2 Member: Dio\_kChannelGroupConfig

**Table 27** Dio\_kChannelGroupConfig

<b>Name</b>	Dio_kChannelGroupConfig	
<b>Type</b>	Dio_ChannelGroupType *	
<b>Description</b>	Pointer to the array of the structure containing channel group configuration.	
<b>Verification method</b>	The generated structure member is present in the Dio_Config structure and It is always generated as the base address of the Dio_kChannelGroupConfig.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure Channel Groups at container DioConfig/DioPort/DioPort_x/DioChannelGroup	<pre>{ /* Dio Channelgroup configuration */     &amp;Dio_kChannelGroupConfig[0] }</pre>
	No Channel Groups are configured at container DioConfig/DioPort/DioPort_x/DioChannelGroup then it is generated as null pointer	<pre>{ /* Dio Channelgroup configuration */     Null PTR, }</pre>

### 1.2.1.3 Member: Dio\_ChannelGroupConfigSize

**Table 28** Dio\_ChannelGroupConfigSize

<b>Name</b>	Dio_ChannelGroupConfigSize
<b>Type</b>	UInt32
<b>Description</b>	Contains the number of channel group configured
<b>Verification method</b>	The structure member is generated as a macro. Kindly refer 1.1.10

Example(s)	Action	Generated output
	Set of number of Channel groups are configured then a macro is generated	<pre>{ /* Configured the number of Dio Channel groups for configuration */ DIO_CHANNELGROUPCOUNT }</pre>

## 1.2.2 Structure: Dio\_kPortChannelConfig

**Table 29** Dio\_kPortChannelConfig

<b>Name</b>	Dio_kPortChannelConfig
<b>Type</b>	Dio_PortChannelIdType *
<b>Description</b>	An array of a structure containing port and channel configuration.
<b>Verification method</b>	The structure generated is an array of available ports having port information.
	<i>Note: The size of the array is equal to the number of available port.</i>

Example(s)	Action	Generated output
	Number of available port =0,6,13,20,41 3 ports are configured at DioConf/DioPort = 0,20,41 Configure all pins for ports 0,20 and 41 at DioConfig/DioPort/DioPort/DioChannel for respective ports	<pre>static const Dio_PortChannelIdType Dio_kPortChannelConfig[] = { { /* Port0 */ DIO_PORT_CONFIGURED, (0xffffU) }, { /* Port6 */ DIO_PORT_NOT_CONFIGURED, (0x0000U) }, { /* Port13 */ DIO_PORT_NOT_CONFIGURED, (0x0000U) }, { /* Port20 */ DIO_PORT_CONFIGURED, (0xffffU) }, { /* Port41 */ DIO_PORT_CONFIGURED,</pre>



	(0xffffU)
	}
	};

### 1.2.2.1 Member: Dio\_PortIdConfig

**Table 30** Dio\_PortIdConfig

<b>Name</b>	Dio_PortIdConfig	
<b>Type</b>	UInt8	
<b>Description</b>	Indicates whether a port is configured or not.	
<b>Verification method</b>	The structure member is generated with DIO_PORT_CONFIGURED when a port is configured, else as DIO_PORT_NOT_CONFIGURED	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Available ports = 0,10,13,20,41 3 ports are configured at DioConf/DioPort = 10,13,41	<pre> {   { /* Port0 */     DIO_PORT_NOT_CONFIGURED,   },   { /* Port10 */     DIO_PORT_CONFIGURED,   },   { /* Port13 */     DIO_PORT_CONFIGURED,   },   { /* Port20 */     DIO_PORT__NOT_CONFIGURED,   },   { /* Port41 */     DIO_PORT_CONFIGURED,   } } </pre>

### 1.2.2.2 Member: Dio\_ChannelIdConfig

**Table 31** Dio\_ChannelIdConfig

<b>Name</b>	Dio_ChannelIdConfig
<b>Type</b>	UInt16
<b>Description</b>	Indicates the number of channels configured within a port.

## DIO driver

<b>Verification method</b>	The structure member is generated as a bit-wise numeric representation of pins configured within a port. Bit 0 corresponds to Channel 0 up to Bit15 which corresponds to Channel 15. 1 indicates the channel is configured else 0.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure all channels of Port 0 i.e., 0-15	<pre>/* Port0 */ (0xffffU)</pre>
	No channels are configured for port 11	<pre>/* Port11 */ (0x0000U)</pre>
	Channels 1,3,7,8,10,12,13,15, are configured for port 36	<pre>/* Port36 */ (0xa58aU)</pre>

## 1.2.3 Structure: Dio\_kChannelGroupConfig

Table 32 Dio\_kChannelGroupConfig

<b>Name</b>	Dio_kChannelgroupConfig	
<b>Type</b>	Dio_ChannelGroupType *	
<b>Description</b>	An array of a structure containing a channel group configuration.	
<b>Verification method</b>	The structure generated is an array of channel group configured.	
	<i>Note:</i> The size of the array is equal to the total number of Channel groups configured.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure 5 Channel Groups for Port 0 at DioConf\DioPort\DioPort_0	<pre>static const Dio_ChannelGroupType Dio_kChannelGroupConfig[DIO_CHANNELG ROUPCOUNT]= {     {         /* DioPort_0, DioChGrpId_P0_0 */         (Dio_PortLevelType)0xf, /* Mask */         (uint8)0,                /* Offset */         (Dio_PortType)0          /* Port Id */     },     {         /* DioPort_1, DioChGrpId_P11_0 */</pre>

```

        (Dio_PortLevelType)0xf0, /* Mask
*/
        (uint8)4,                /* Offset
*/
        (Dio_PortType)11         /*
Port Id */
    },
    {
        /* DioPort_2, DioChGrpId_P20_0
*/
        (Dio_PortLevelType)0xf00, /*
Mask */
        (uint8)8,                /* Offset
*/
        (Dio_PortType)20         /*
Port Id */
    },
    {
        /* DioPort_3, DioChGrpId_P33_0
*/
        (Dio_PortLevelType)0xf80, /*
Mask */
        (uint8)7,                /* Offset
*/
        (Dio_PortType)33         /*
Port Id */
    },
    {
        /* DioPort_4, DioChGrpId_P41_0
*/
        (Dio_PortLevelType)0xe000, /*
Mask */
        (uint8)13,               /*
Offset */
        (Dio_PortType)11         /*
Port Id */
    }
};

```

### 1.2.3.1 Member: Mask

**Table 33** Mask

## DIO driver

<b>Name</b>	Mask	
<b>Type</b>	Dio_PortLevelType	
<b>Description</b>	Indicates the number of channels configured for the group.	
<b>Verification method</b>	The structure member is generated as a bit-wise numeric representation of channels present in a group.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure Channel Group at DioConf\DioPort\DioPort_0\ DioChannelGroup with mask 15	<pre>/* DioPort_0, DioChGrpId_P0_0 */  (Dio_PortLevelType)0xf, /* Mask */</pre>
	Configure Channel Group at DioConf\DioPort\DioPort_21\ DioChannelGroup with mask 240	<pre>/* DioPort_0, DioChGrpId_P21_0 */  (Dio_PortLevelType)0xf0, /* Mask */</pre>
	Configure Channel Group at DioConf\DioPort\DioPort_33\ DioChannelGroup with mask 3968	<pre>/* DioPort_0, DioChGrpId_P33_0 */  (Dio_PortLevelType)0xf10, /* Mask */</pre>

## 1.2.3.2 Member: Offset

Table 34 Offset

<b>Name</b>	Offset	
<b>Type</b>	uint8	
<b>Description</b>	The position of the Channel Group on the port counted from the LSB	
<b>Verification method</b>	The structure member is generated as a numeric value which indicates the lowest channel configured for the port.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure Channel Group at DioConf\DioPort\DioPort_0\ DioChannelGroup with mask 15	<pre>/* DioPort_0, DioChGrpId_P0_0 */  (uint8)0, /* Offset */</pre>
	Configure Channel Group at DioConf\DioPort\DioPort_0\ DioChannelGroup with mask 240	<pre>/* DioPort_0, DioChGrpId_P0_0 */  (uint8)4, /* Offset */</pre>

## DIO driver

Configure Channel Group at DioConf\DioPort\DioPort_0\ DioChannelGroup with mask 3968	/* DioPort_0, DioChGrpId_P0_0 */ (uint8)7, /* Offset */
---	--

## 1.2.3.3 Member: Port

Table 35 Port

<b>Name</b>	Port	
<b>Type</b>	PortType	
<b>Description</b>	The id of the port where channel group is configured.	
<b>Verification method</b>	The structure member is generated as the numeric value of the port number on which channel group is defined.	
<b>Example(s)</b>	<b>Action</b>	<b>Generated output</b>
	Configure Channel Group at DioConf\DioPort\DioPort_0\ DioChannelGroup	(Dio_PortType) 0 /* Port Id */
	Configure Channel Group at DioConf\DioPort\DioPort_11\ DioChannelGroup	(Dio_PortType) 11 /* Port Id */
	Configure Channel Group at DioConf\DioPort\DioPort_33\ DioChannelGroup	(Dio_PortType) 33 /* Port Id */
	Configure Channel Group at DioConf\DioPort\DioPort_42\ DioChannelGroup	(Dio_PortType) 42 /* Port Id */

## Revision history

### Major changes since the last revision

Date	Version	Description
2020-08-10	4.0	Document Released
2020-08-06	3.1	<ul style="list-style-type: none"><li>Added new macro DIO_MASKED_WRITE_PORT_API (Section 1.1.9)</li><li>DIO driver chapter moved from MC-ISAR_TC3xx_Config_Verification_Manual_BASIC.pdf to this document</li></ul>
2019-07-19	3.0	Version and revision history update. No other changes identified from previous releases.
2019-02-27	1.10.0_2.0	Added Common published information
2019-02-22	1.10.0_1.0	Initial Release

#### Trademarks

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

**Edition 2019-07-19**

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

**Document reference**

**Doc\_Number**

#### IMPORTANT NOTICE

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

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

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

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

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

#### WARNINGS

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

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