

User Manual

for MPC5634M PORT Driver

Document Number: UM14PORTASR3.0R2.0.0

Rev. 1.1





Contents

Section Number	Title	Page
Chapter 1		
Revision History		
Chapter 2		
Introduction		
2.1	Supported Derivatives.....	9
2.2	Overview.....	9
2.3	About this Manual.....	10
2.4	Acronyms and Definitions.....	10
2.5	Reference List.....	11
Chapter 3		
Driver		
3.1	Requirements.....	13
3.2	Driver Design Summary.....	13
3.3	Deviation from Requirements.....	14
3.4	Runtime Errors.....	15
3.5	Software specification.....	15
3.5.1	Define Reference.....	15
3.5.1.1	Define PORT_E_DIRECTION_UNCHANGEABLE.....	15
3.5.1.2	Define PORT_E_INVALID_DIRECTION.....	15
3.5.1.3	Define PORT_E_INVALID_POINTER.....	16
3.5.1.4	Define PORT_E_MODE_UNCHANGEABLE.....	16
3.5.1.5	Define PORT_E_PARAM_CONFIG.....	16
3.5.1.6	Define PORT_E_PARAM_INVALID_MODE.....	16
3.5.1.7	Define PORT_E_PARAM_PIN.....	17
3.5.1.8	Define PORT_E_UNINIT.....	17
3.5.1.9	Define PORT_GETVERSIONINFO_ID.....	17
3.5.1.10	Define PORT_INIT_ID.....	18
3.5.1.11	Define PORT_REFRESHPIN_DIRECTION_ID.....	18

Section Number	Title	Page
3.5.1.12	Define PORT_SETPINDIRECTION_ID.....	18
3.5.1.13	Define PORT_SETPINMODE_ID.....	18
3.5.2	Enum Reference.....	19
3.5.3	Function Reference.....	19
3.5.3.1	Function FUNC.....	19
3.5.4	Structs Reference.....	20
3.5.5	Types Reference.....	20
3.5.6	Variables Reference.....	20
3.5.7	Port_SetPinMode function parameters.....	20
3.6	Symbolic Names DISCLAIMER.....	29

Chapter 4 Tresos Configuration Plug-in

4.1	Configuration elements of Port.....	31
4.2	Form IMPLEMENTATION_CONFIG_VARIANT.....	31
4.3	Form PortConfigSet.....	32
4.3.1	Form PortContainer.....	33
4.3.1.1	PortNumberOfPortPins (PortContainer).....	33
4.3.1.2	Form PortPin.....	34
4.3.1.2.1	PortPinWpe (PortPin).....	34
4.3.1.2.2	PortPinWps (PortPin).....	35
4.3.1.2.3	PortPinOde (PortPin).....	35
4.3.1.2.4	PortPinHysteresis (PortPin).....	35
4.3.1.2.5	PortPinDirectionChangeable (PortPin).....	36
4.3.1.2.6	PortPinId (PortPin).....	36
4.3.1.2.7	PortPinPcr (PortPin).....	37
4.3.1.2.8	PortPinMode (PortPin).....	37
4.3.1.2.9	PortPinInitialMode (PortPin).....	38
4.3.1.2.10	PortPinDirection (PortPin).....	38
4.3.1.2.11	PortPinLevelValue (PortPin).....	39

Section Number	Title	Page
4.3.1.2.12	PortPinSlewRate (PortPin).....	39
4.3.1.2.13	PortPinDriveStrength (PortPin).....	40
4.4	Form CommonPublishedInformation.....	40
4.4.1	ArMajorVersion (CommonPublishedInformation).....	41
4.4.2	ArMinorVersion (CommonPublishedInformation).....	41
4.4.3	ArPatchVersion (CommonPublishedInformation).....	42
4.4.4	ModuleId (CommonPublishedInformation).....	42
4.4.5	SwMajorVersion (CommonPublishedInformation).....	43
4.4.6	SwMinorVersion (CommonPublishedInformation).....	43
4.4.7	SwPatchVersion (CommonPublishedInformation).....	44
4.4.8	VendorApiInfix (CommonPublishedInformation).....	44
4.4.9	VendorId (CommonPublishedInformation).....	45
4.5	Form PortGeneral.....	45
4.5.1	PortDevErrorDetect (PortGeneral).....	46
4.5.2	PortSetPinDirectionApi (PortGeneral).....	46
4.5.3	PortSetPinModeApi (PortGeneral).....	47
4.5.4	PortVersionInfoApi (PortGeneral).....	47



Chapter 1

Revision History

Table 1-1. Revision History

Revision	Date	Author	Description
1.0	04-Feb-2011	Giovanni di Martino	Monaco 1.5 MByte automatic documentation
1.1	19-Dec-2011	Tilak Basavaraju	Updated for Monaco RTM 2.0.0



Chapter 2

Introduction

This User Manual describes Freescale Semiconductor AUTOSAR Port (Port) for MPC5634M .

AUTOSAR Port driver configuration parameters and deviations from the specification are described in Port Driver chapter of this document. AUTOSAR Port driver requirements and APIs are described in the AUTOSAR Port driver software specification document.

2.1 Supported Derivatives

The software described in this document is intended to be used with the following microcontroller devices of Freescale Semiconductor .

Table 2-1. MPC5634M Derivatives.

Freescale Semiconductor	mpc5634m_bga208, mpc5634m_qfp144, mpc5634m_qfp176
-------------------------	--

All of the above microcontroller devices are collectively named as MPC5634M.

2.2 Overview

AUTOSAR (AUTomotive Open System ARchitecture) is an industry partnership working to establish standards for software interfaces and software modules for automobile electronic control systems.

AUTOSAR

- paves the way for innovative electronic systems that further improve performance, safety and environmental friendliness.

- is a strong global partnership that creates one common standard: "Cooperate on standards, compete on implementation".
- is a key enabling technology to manage the growing electrics/electronics complexity. It aims to be prepared for the upcoming technologies and to improve cost-efficiency without making any compromise with respect to quality.
- facilitates the exchange and update of software and hardware over the service life of the vehicle.

2.3 About this Manual

This Technical Reference employs the following typographical conventions:

Boldface type: Bold is used for important terms, notes and warnings.

Italic font: Italic typeface is used for code snippets in the text. Note that C language modifiers such "const" or "volatile" are sometimes omitted to improve readability of the presented code.

Notes and warnings are shown as below:

Note

This is a note.

2.4 Acronyms and Definitions

Table 2-2. Acronyms and Definitions

Term	Definition
API	Application Programming Interface
AUTOSAR	Automotive Open System Architecture
ASM	Assembler
BSMI	Basic Software Make file Interface
CAN	Controller Area Network
DEM	Diagnostic Event Manager
DET	Development Error Tracer
C/CPP	C and C++ Source Code
VLE	Variable Length Encoding
N/A	Not Applicable
MCU	Micro Controller Unit

2.5 Reference List

Table 2-3. Reference List

#	Title	Version
1	AUTOSAR 3.0Port Driver Software Specification Document.	V2.2.0 R3.0 Rev 0001
2	MPC5634M Reference Manual	Rev. 6, 4 October 2011

Chapter 3

Driver

3.1 Requirements

Requirements for this driver are detailed in the AUTOSAR 3.0Port Driver Software Specification document (See Table [Reference List](#)).

3.2 Driver Design Summary

This module provides the service for initializing the whole PORT structure of the microcontroller. Many ports and port pins can be assigned to various functionalities, e.g.

- General purpose I/O
- ADC
- SPI
- SCI
- PWM
- CAN
- LIN
- etc

For this reason, there is an overall configuration and initialization of this port structure. The configuration and mode of these port pins is microcontroller and ECU dependent.

Port initialisation data are written to each port as efficiently as possible. This PORT driver module completes the overall configuration and initialisation of the port structure which is used in the DIO driver module. Therefore, the DIO driver works on pins and ports which are configured by the PORT driver.

The PORT driver is initialised prior to use of the DIO functions. Otherwise DIO functions will exhibit undefined behaviour.

3.3 Deviation from Requirements

The driver deviates from the AUTOSAR Port Driver software specification in some places. Table identifies the AUTOSAR requirements that are not fully implemented, implemented differently, or out of scope for the Port driver. Table [Table 3-1](#) provides Status column description.

Table 3-1. Deviations Status Column Description

Term	Definition
N/A	Not available
N/T	Not testable
N/S	Out of scope
N/I	Not implemented
N/F	Not fully implemented

Below table identifies the AUTOSAR requirements that are not fully implemented, implemented differently, or out of scope for the driver.

Table 3-2. Driver Deviations Table

Requirement	Status	Description	Notes
PORT005	N/F	This module shall configure all ports and port pins that are not used (neither as GPIO nor special purpose IO) to be set to a defined state by the PORT module configuration (GPIO set by default).	All unused pins are set to Input only
PORT037	N/I	Production errors shall be reported to Diagnostic Event Manager.	Production errors are not available in Port
PORT041	N/F	Port_Init - This initialization function shall be called to initialize ALL ports and port pins with the configuration set pointed to by ConfigPtr.	User application requirement: User has to configure all available pins
PORT071	N/S	Port_Init: This function shall also be called after a reset, in order to reconfigure the ports and port pins of the MCU.	Application Code Requirement
PORT078	N/S	This function should be called first in order to initialize the port for use. If not called first then no operation can occur on the MCU ports and port pins.	Application Code Requirement
PORT082	N/S	The PORT driver shall not provide for the configuration of Level inversion. The default value shall be set (not inverted). The IO Hardware Abstraction layer shall carry out level Inversion.	Other module Requirement

Table continues on the next page...

Table 3-2. Driver Deviations Table (continued)

Requirement	Status	Description	Notes
PORT115	N/A	Values for production code Event Ids are assigned externally by the configuration of the DEM. They are published in the file Dem_IntErrId.h and included via Dem.h.	Production errors are not available in Port

3.4 Runtime Errors

This driver doesn't generate any runtime error.

3.5 Software specification

The following sections contains driver software specifications.

3.5.1 Define Reference

Constants supported by the driver are as per AUTOSAR Port Driver software specification Version 3.0 .

3.5.1.1 Define PORT_E_DIRECTION_UNCHANGEABLE

Port Pin not configured as changeable.

Details: Errors and exceptions that will be detected by the PORT driver.

**Table 3-3. Define PORT_E_DIRECTION_UNCHANGEABLE
Description**

Name	PORT_E_DIRECTION_UNCHANGEABLE
Initializer	(uint8)0x0B

3.5.1.2 Define PORT_E_INVALID_DIRECTION

API service called with invalid direction.

Details: Errors and exceptions that will be detected by the PORT driver.

Table 3-4. Define PORT_E_INVALID_DIRECTION Description

Name	PORT_E_INVALID_DIRECTION
Initializer	(uint8)0x11

3.5.1.3 Define PORT_E_INVALID_POINTER

API service called with NULL pointer.

Details: Errors and exceptions that will be detected by the PORT driver.

Table 3-5. Define PORT_E_INVALID_POINTER Description

Name	PORT_E_INVALID_POINTER
Initializer	(uint8)0x10

3.5.1.4 Define PORT_E_MODE_UNCHANGEABLE

API Port_SetPinMode() service called when mode is unchangeable. .

Details: Errors and exceptions that will be detected by the PORT driver.

Table 3-6. Define PORT_E_MODE_UNCHANGEABLE Description

Name	PORT_E_MODE_UNCHANGEABLE
Initializer	(uint8)0x0E

3.5.1.5 Define PORT_E_PARAM_CONFIG

API Port_Init() service called with wrong parameter. .

Details: Errors and exceptions that will be detected by the PORT driver.

Table 3-7. Define PORT_E_PARAM_CONFIG Description

Name	PORT_E_PARAM_CONFIG
Initializer	(uint8)0x0C

3.5.1.6 Define PORT_E_PARAM_INVALID_MODE

API Port_SetPinMode() service called when mode is unchangeable. .

Details: Errors and exceptions that will be detected by the PORT driver.

Table 3-8. Define PORT_E_PARAM_INVALID_MODE Description

Name	PORT_E_PARAM_INVALID_MODE
Initializer	(uint8)0x0D

3.5.1.7 Define PORT_E_PARAM_PIN

Invalid Port Pin ID requested.

Details: Errors and exceptions that will be detected by the PORT driver.

Table 3-9. Define PORT_E_PARAM_PIN Description

Name	PORT_E_PARAM_PIN
Initializer	(uint8)0x0A

3.5.1.8 Define PORT_E_UNINIT

API service called without module initialization.

Details: Errors and exceptions that will be detected by the PORT driver.

Table 3-10. Define PORT_E_UNINIT Description

Name	PORT_E_UNINIT
Initializer	(uint8)0x0F

3.5.1.9 Define PORT_GETVERSIONINFO_ID

API service ID for PORT get version info function.

Details: Parameters used when raising an error/exception.

Table 3-11. Define PORT_GETVERSIONINFO_ID Description

Name	PORT_GETVERSIONINFO_ID
Initializer	(uint8)0x03

3.5.1.10 Define PORT_INIT_ID

API service ID for PORT Init function.

Details: Parameters used when raising an error/exception.

Table 3-12. Define PORT_INIT_ID Description

Name	PORT_INIT_ID
Initializer	(uint8)0x00

3.5.1.11 Define PORT_REFRESHPINDIRECTION_ID

API service ID for PORT refresh pin direction function.

Details: Parameters used when raising an error/exception.

Table 3-13. Define PORT_REFRESHPINDIRECTION_ID Description

Name	PORT_REFRESHPINDIRECTION_ID
Initializer	(uint8)0x02

3.5.1.12 Define PORT_SETPINDIRECTION_ID

API service ID for PORT set pin direction function.

Details: Parameters used when raising an error/exception.

Table 3-14. Define PORT_SETPINDIRECTION_ID Description

Name	PORT_SETPINDIRECTION_ID
Initializer	(uint8)0x01

3.5.1.13 Define PORT_SETPINMODE_ID

API service ID for PORT set pin mode.

Details: Parameters used when raising an error/exception.

Table 3-15. Define PORT_SETPINMODE_ID Description

Name	PORT_SETPINMODE_ID
Initializer	(uint8)0x04

3.5.2 Enum Reference

Enumeration of all constants supported by the driver are as per AUTOSAR Port Driver software specification Version 3.0 .

3.5.3 Function Reference

Functions of all functions supported by the driver are as per AUTOSAR Port Driver software specification Version 3.0 .

3.5.3.1 Function FUNC

Sets the port pin direction.

Details:

Returns the version information of this module.Sets the port pin mode.Refreshes port direction.The function `Port_SetPinDirection()` will set the port pin direction during runtime.

Pre: must have been called first. In order to change the pin direction the `PortPinDirectionChangeable` flag must have been set to . .

Satisfied Requirements: PORT059, PORT063, PORT086, PORT117, PORT141.

This function will refresh the direction of all configured ports to the configured direction. The PORT driver will exclude from refreshing those port pins that are configured as "pin direction changeable during runtime".

Pre: must have been called first.

Satisfied Requirements: PORT060, PORT061, PORT117, PORT142.

The function `Port_SetPinMode()` will set the port pin mode of the referenced pin during runtime.

Pre: must have been called first.

Satisfied Requirements: PORT117, PORT145.

The function `Port_GetVersionInfo()` will return the version information of this module. The version information includes:

- Module Id,
- Vendor Id,
- Vendor specific version numbers.

Satisfied Requirements: PORT103

Prototype: `FUNC(void, PORT_CODE);`

Table 3-16. FUNC Arguments

Type	Name	Direction	Description
	Pin	input	Pin ID number.
	Direction	input	Port Pin direction.
	Pin	input	Pin ID number.
	Mode	input	New Port Pin mode to be set on port pin.
	pVersionInfo	input, output	Pointer to where to store the version information of this module.

3.5.4 Structs Reference

Data structures supported by the driver are as per AUTOSAR Port Driver software specification Version 3.0 .

3.5.5 Types Reference

Types supported by the driver are as per AUTOSAR Port Driver software specification Version 3.0 .

3.5.6 Variables Reference

Variables supported by the driver are as per AUTOSAR Port Driver software specification Version 3.0 .

3.5.7 Port_SetPinMode function parameters

This section contains information about how to use the Port_SetPinMode function. In particular, the table in below shows all the possible values that the user can use by calling the Port_SetPinMode function.

Pin: This value is the PortPin Id defined in the configuration.

Mode: Is the Mode that the user wishes to change for the selected Pin.

The table is indexed by PortPin Pcr which is associated to a PortPin Id in the configuration.

The screenshot shows the 'PortPin' configuration window. At the top, the 'Name' field is set to 'Pin0'. Below this is the 'General' tab. The configuration is organized into two columns of settings, each with a document icon, a checkbox or dropdown, and an edit icon.

Setting	Value
PortPin Wpe	<input type="checkbox"/>
PortPin Wps	<input type="checkbox"/>
PortPin Ode	<input type="checkbox"/>
PortPin Safe Mode	<input type="checkbox"/>
PortPin Direction Changeable	<input checked="" type="checkbox"/>
PortPin Readback	<input type="checkbox"/>
PortPin Id	0
PortPin Pcr	3
PortPin Mode	GPIO
PortPin Initial Mode	PORT_GPIO_MODE
PortPin Direction	PortPinDirectionIn
PortPin Level Value	PortPinLevelLow
PortPin Output Slew Rate	Slow

The 'PortPin Id' and 'PortPin Pcr' fields are highlighted with a red box in the original image.

Figure 3-1. PortPin Id and PortPin Pcr

Table 3-17. Mode List

PortPin Pcr	PortPin Mode	Mode Field
83	GPIO	PORT_GPIO_MODE
	CAN_A_TX	PORT_PRIMARY_MODE
	eSCI_A_TX	PORT_ALT1_FUNC_MODE
84	GPIO	PORT_GPIO_MODE
	CAN_A_RX	PORT_PRIMARY_MODE
	eSCI_A_RX	PORT_ALT1_FUNC_MODE
87	GPIO	PORT_GPIO_MODE
	CAN_C_TX	PORT_PRIMARY_MODE
88	GPIO	PORT_GPIO_MODE
	CAN_C_RX	PORT_PRIMARY_MODE
89	GPIO	PORT_GPIO_MODE
	eSCI_A_TX	PORT_PRIMARY_MODE
	eMIOS_13	PORT_ALT1_FUNC_MODE
90	GPIO	PORT_GPIO_MODE
	eSCI_A_RX	PORT_PRIMARY_MODE
	eMIOS_15	PORT_ALT1_FUNC_MODE
91	GPIO	PORT_GPIO_MODE
	eSCI_B_TX	PORT_PRIMARY_MODE
92	GPIO	PORT_GPIO_MODE
	eSCI_B_RX	PORT_PRIMARY_MODE
98	GPIO	PORT_GPIO_MODE
99	GPIO	PORT_GPIO_MODE
102	GPIO	PORT_GPIO_MODE
	DSPI_B_SCK	PORT_PRIMARY_MODE
	DSPI_C_PCS_1	PORT_ALT1_FUNC_MODE
103	GPIO	PORT_GPIO_MODE
	DSPI_B_SIN	PORT_PRIMARY_MODE
	DSPI_C_PCS_2	PORT_ALT1_FUNC_MODE
104	GPIO	PORT_GPIO_MODE
	DSPI_B_SOUT	PORT_PRIMARY_MODE
	DSPI_C_PCS_5	PORT_ALT1_FUNC_MODE
105	GPIO	PORT_GPIO_MODE
	DSPI_B_PCS_0	PORT_PRIMARY_MODE
106	GPIO	PORT_GPIO_MODE
	DSPI_B_PCS_1	PORT_PRIMARY_MODE

Table continues on the next page...

Table 3-17. Mode List (continued)

PortPin Pcr	PortPin Mode	Mode Field
107	GPIO	PORT_GPIO_MODE
	DSPI_B_PCS_2	PORT_PRIMARY_MODE
	DSPI_C_SOUT	PORT_ALT1_FUNC_MODE
108	GPIO	PORT_GPIO_MODE
	DSPI_B_PCS_3	PORT_PRIMARY_MODE
	DSPI_C_SIN	PORT_ALT1_FUNC_MODE
109	GPIO	PORT_GPIO_MODE
	DSPI_B_PCS_4	PORT_PRIMARY_MODE
	DSPI_C_SCK	PORT_ALT1_FUNC_MODE
110	GPIO	PORT_GPIO_MODE
	DSPI_B_PCS_5	PORT_PRIMARY_MODE
	DSPI_C_PCS_0	PORT_ALT1_FUNC_MODE
114	GPIO	PORT_GPIO_MODE
	eTPU_A_0	PORT_PRIMARY_MODE
	eTPU_A_12	PORT_ALT1_FUNC_MODE
	eTPU_A_19	PORT_ALT2_FUNC_MODE
115	GPIO	PORT_GPIO_MODE
	eTPU_A_1	PORT_PRIMARY_MODE
	eTPU_A_13	PORT_ALT1_FUNC_MODE
116	GPIO	PORT_GPIO_MODE
	eTPU_A_2	PORT_PRIMARY_MODE
	eTPU_A_14	PORT_ALT1_FUNC_MODE
117	GPIO	PORT_GPIO_MODE
	eTPU_A_3	PORT_PRIMARY_MODE
	eTPU_A_15	PORT_ALT1_FUNC_MODE
118	GPIO	PORT_GPIO_MODE
	eTPU_A_4	PORT_PRIMARY_MODE
	eTPU_A_16	PORT_ALT1_FUNC_MODE
119	GPIO	PORT_GPIO_MODE
	eTPU_A_5	PORT_PRIMARY_MODE
	eTPU_A_17	PORT_ALT1_FUNC_MODE
	DSPI_B_SCK_LVDC_MINUS	PORT_ALT2_FUNC_MODE

Table continues on the next page...

Table 3-17. Mode List (continued)

PortPin Pcr	PortPin Mode	Mode Field
120	GPIO	PORT_GPIO_MODE
	eTPU_A_6	PORT_PRIMARY_MODE
	eTPU_A_18	PORT_ALT1_FUNC_MODE
	DSPI_B_SCK_LVDS_PLUS	PORT_ALT2_FUNC_MODE
121	GPIO	PORT_GPIO_MODE
	eTPU_A_7	PORT_PRIMARY_MODE
	eTPU_A_19	PORT_ALT1_FUNC_MODE
	DSPI_B_SOUT_LVDS_MINUS	PORT_ALT2_FUNC_MODE
	eTPU_A_6	PORT_ALT3_FUNC_MODE
122	GPIO	PORT_GPIO_MODE
	eTPU_A_8	PORT_PRIMARY_MODE
	eTPU_A_20	PORT_ALT1_FUNC_MODE
	DSPI_B_SOUT_LVDS_PLUS	PORT_ALT2_FUNC_MODE
123	GPIO	PORT_GPIO_MODE
	eTPU_A_9	PORT_PRIMARY_MODE
	eTPU_A_21	PORT_ALT1_FUNC_MODE
124	GPIO	PORT_GPIO_MODE
	eTPU_A_10	PORT_PRIMARY_MODE
	eTPU_A_22	PORT_ALT1_FUNC_MODE
125	GPIO	PORT_GPIO_MODE
	eTPU_A_11	PORT_PRIMARY_MODE
	eTPU_A_23	PORT_ALT1_FUNC_MODE
126	GPIO	PORT_GPIO_MODE
	eTPU_A_12	PORT_PRIMARY_MODE
	DSPI_B_PCS_1	PORT_ALT1_FUNC_MODE
127	GPIO	PORT_GPIO_MODE
	eTPU_A_13	PORT_PRIMARY_MODE
	DSPI_B_PCS_3	PORT_ALT1_FUNC_MODE
128	GPIO	PORT_GPIO_MODE
	eTPU_A_14	PORT_PRIMARY_MODE
	DSPI_B_PCS_4	PORT_ALT1_FUNC_MODE
	eTPU_A_9	PORT_ALT2_FUNC_MODE
129	GPIO	PORT_GPIO_MODE
	eTPU_A_15	PORT_PRIMARY_MODE
	DSPI_B_PCS_5	PORT_ALT1_FUNC_MODE

Table continues on the next page...

Table 3-17. Mode List (continued)

PortPin Pcr	PortPin Mode	Mode Field
130	GPIO	PORT_GPIO_MODE
	eTPU_A_16	PORT_PRIMARY_MODE
131	GPIO	PORT_GPIO_MODE
	eTPU_A_17	PORT_PRIMARY_MODE
132	GPIO	PORT_GPIO_MODE
	eTPU_A_18	PORT_PRIMARY_MODE
133	GPIO	PORT_GPIO_MODE
	eTPU_A_19	PORT_PRIMARY_MODE
134	GPIO	PORT_GPIO_MODE
	eTPU_A_20	PORT_PRIMARY_MODE
	IRQ_8	PORT_ALT1_FUNC_MODE
135	GPIO	PORT_GPIO_MODE
	eTPU_A_21	PORT_PRIMARY_MODE
	IRQ_9	PORT_ALT1_FUNC_MODE
136	GPIO	PORT_GPIO_MODE
	eTPU_A_22	PORT_PRIMARY_MODE
	IRQ_10	PORT_ALT1_FUNC_MODE
	eTPU_A_17	PORT_ALT2_FUNC_MODE
137	GPIO	PORT_GPIO_MODE
	eTPU_A_23	PORT_PRIMARY_MODE
	IRQ_11	PORT_ALT1_FUNC_MODE
	eTPU_A_21	PORT_ALT2_FUNC_MODE
138	GPIO	PORT_GPIO_MODE
	eTPU_A_24	PORT_PRIMARY_MODE
	IRQ_12	PORT_ALT1_FUNC_MODE
	DSPI_C_SCK_LVDS_MINUS	PORT_ALT2_FUNC_MODE
139	GPIO	PORT_GPIO_MODE
	eTPU_A_25	PORT_PRIMARY_MODE
	IRQ_13	PORT_ALT1_FUNC_MODE
	DSPI_C_SCK_LVDS_PLUS	PORT_ALT2_FUNC_MODE
140	GPIO	PORT_GPIO_MODE
	eTPU_A_26	PORT_PRIMARY_MODE
	IRQ_14	PORT_ALT1_FUNC_MODE
	DSPI_C_SOUT_LVDS_MINUS	PORT_ALT2_FUNC_MODE

Table continues on the next page...

Table 3-17. Mode List (continued)

PortPin Pcr	PortPin Mode	Mode Field
141	GPIO	PORT_GPIO_MODE
	eTPU_A_27	PORT_PRIMARY_MODE
	IRQ_15	PORT_ALT1_FUNC_MODE
	DSPI_C_SOUT_LVDS_PLUS	PORT_ALT2_FUNC_MODE
	DSPI_B_SOUT	PORT_ALT3_FUNC_MODE
142	GPIO	PORT_GPIO_MODE
	eTPU_A_28	PORT_PRIMARY_MODE
	DSPI_C_PCS_1	PORT_ALT1_FUNC_MODE
143	GPIO	PORT_GPIO_MODE
	eTPU_A_29	PORT_PRIMARY_MODE
	DSPI_C_PCS_2	PORT_ALT1_FUNC_MODE
144	GPIO	PORT_GPIO_MODE
	eTPU_A_30	PORT_PRIMARY_MODE
	DSPI_C_PCS_3	PORT_ALT1_FUNC_MODE
	eTPU_A_11	PORT_ALT2_FUNC_MODE
145	GPIO	PORT_GPIO_MODE
	eTPU_A_31	PORT_PRIMARY_MODE
	DSPI_C_PCS_4	PORT_ALT1_FUNC_MODE
	eTPU_A_13	PORT_ALT2_FUNC_MODE
179	GPIO	PORT_GPIO_MODE
	eMIOS_0	PORT_PRIMARY_MODE
	eTPU_A_0	PORT_ALT1_FUNC_MODE
	eTPU_A_25	PORT_ALT2_FUNC_MODE
180	GPIO	PORT_GPIO_MODE
	eMIOS_1	PORT_PRIMARY_MODE
	eTPU_A_1	PORT_ALT1_FUNC_MODE
181	GPIO	PORT_GPIO_MODE
	eMIOS_2	PORT_PRIMARY_MODE
	eTPU_A_2	PORT_ALT1_FUNC_MODE
183	GPIO	PORT_GPIO_MODE
	eMIOS_4	PORT_PRIMARY_MODE
	eTPU_A_4	PORT_ALT1_FUNC_MODE

Table continues on the next page...

Table 3-17. Mode List (continued)

PortPin Pcr	PortPin Mode	Mode Field
187	GPIO	PORT_PRIMARY_MODE
	eMIOS_8	PORT_PRIMARY_MODE
	eTPU_A_8	PORT_ALT1_FUNC_MODE
	eSCI_B_TX	PORT_ALT2_FUNC_MODE
188	GPIO	PORT_GPIO_MODE
	eMIOS_9	PORT_PRIMARY_MODE
	eTPU_A_9	PORT_ALT1_FUNC_MODE
	eSCI_B_RX	PORT_ALT2_FUNC_MODE
189	GPIO	PORT_GPIO_MODE
	eMIOS_10	PORT_PRIMARY_MODE
190	GPIO	PORT_GPIO_MODE
	eMIOS_11	PORT_PRIMARY_MODE
191	GPIO	PORT_GPIO_MODE
	eMIOS_12	PORT_PRIMARY_MODE
191	DSPI_C_SOUT	PORT_ALT1_FUNC_MODE
	eTPU_A_27	PORT_ALT2_FUNC_MODE
192	GPIO	PORT_GPIO_MODE
	eMIOS_13	PORT_PRIMARY_MODE
193	GPIO	PORT_GPIO_MODE
	eMIOS_14	PORT_PRIMARY_MODE
	IRQ_0	PORT_ALT1_FUNC_MODE
	eTPU_A_29	PORT_ALT2_FUNC_MODE
194	GPIO	PORT_GPIO_MODE
	eMIOS_15	PORT_PRIMARY_MODE
	IRQ_1	PORT_ALT1_FUNC_MODE
202	GPIO	PORT_GPIO_MODE
	eMIOS_23	PORT_PRIMARY_MODE
206	GPIO	PORT_GPIO_MODE
207	GPIO	PORT_GPIO_MODE
208	GPIO	PORT_GPIO_MODE
	PLLREF	PORT_PRIMARY_MODE
	IRQ_4	PORT_ALT1_FUNC_MODE
	ETRIG_2	PORT_ALT2_FUNC_MODE

Table continues on the next page...

Table 3-17. Mode List (continued)

PortPin Pcr	PortPin Mode	Mode Field
212	GPIO	PORT_GPIO_MODE
	BOOTCFG1	PORT_PRIMARY_MODE
	IRQ_3	PORT_ALT1_FUNC_MODE
	ETRIG_3	PORT_ALT2_FUNC_MODE
213	GPIO	PORT_GPIO_MODE
	WKPCFG	PORT_PRIMARY_MODE
	NMI	PORT_ALT1_FUNC_MODE
	DSPI_B_SOUT	PORT_ANALOG_INPUT_MODE
215	SDS	PORT_GPIO_MODE
	eTPU_A_19	PORT_PRIMARY_MODE
	MA_0	PORT_ALT1_FUNC_MODE
	AN_12	PORT_ANALOG_INPUT_MODE
216	SDO	PORT_GPIO_MODE
	eTPU_A_21	PORT_PRIMARY_MODE
	MA_1	PORT_ALT1_FUNC_MODE
	AN_13	PORT_ANALOG_INPUT_MODE
217	SDI	PORT_GPIO_MODE
	eTPU_A_27	PORT_PRIMARY_MODE
	MA_2	PORT_ALT1_FUNC_MODE
	AN_14	PORT_ANALOG_INPUT_MODE
218	eTPU_A_29	PORT_PRIMARY_MODE
	FCK	PORT_ALT1_FUNC_MODE
	AN_15	PORT_ANALOG_INPUT_MODE
219	GPIO	PORT_GPIO_MODE
220	GPIO	PORT_GPIO_MODE
	NEXUS_MDO	PORT_PRIMARY_MODE
	eTPU_A13	PORT_ALT1_FUNC_MODE
221	GPIO	PORT_GPIO_MODE
	eTPU_A_19	PORT_ALT1_FUNC_MODE
222	GPIO	PORT_GPIO_MODE
	eTPU_A_21	PORT_ALT1_FUNC_MODE
223	GPIO	PORT_GPIO_MODE
	eTPU_a_25	PORT_ALT1_FUNC_MODE
224	GPIO	PORT_GPIO_MODE
	eTPU_A_27	PORT_ALT1_FUNC_MODE

Table continues on the next page...

Table 3-17. Mode List (continued)

PortPin Pcr	PortPin Mode	Mode Field
225	GPIO	PORT_GPIO_MODE
	eTPU_A_29	PORT_ALT1_FUNC_MODE
227	GPIO	PORT_GPIO_MODE
	eTPU_A_4	PORT_ALT1_FUNC_MODE
228	GPIO	PORT_GPIO_MODE
	eMIOS_A_6	PORT_ALT1_FUNC_MODE
231	GPIO	PORT_GPIO_MODE
	NEXUS_EVTI	PORT_PRIMARY_MODE
	eTPU_A_2	PORT_ALT1_FUNC_MODE
232	GPIO	PORT_GPIO_MODE
	JTAG_TDI	PORT_PRIMARY_MODE
	eMIOS_5	PORT_ALT1_FUNC_MODE

3.6 Symbolic Names DISCLAIMER

All containers having the symbolic name tag set as true in the Autosar schema will generate defines like:

```
#define <Container_Short_Name> <Container_ID>
```

For this reason it is forbidden to duplicate the name of such containers across the MCAL configuration, or to use names that may trigger other compile issues (e.g. match existing #ifdefs arguments).

Chapter 4

Tresos Configuration Plug-in

This chapter describes the Tresos configuration plug-in for the Port Driver. The most of the parameters are described below.

4.1 Configuration elements of Port

Included forms :

- IMPLEMENTATION_CONFIG_VARIANT
- CommonPublishedInformation
- PortGeneral
- PortConfigSet

Table 4-1. Revision table

Revision	Date
1.9.0	2009-05-14T17:00:00

4.2 Form IMPLEMENTATION_CONFIG_VARIANT

VariantPreCompile: Only precompile time configuration parameters. Only one set of parameters. VariantPostBuild: Mix of precompile and postbuild time configuration parameters. More sets of parameters. If Config Variant = VariantPreCompile, the files Port_Cfg.h and Port_Cfg.c should be used. If Config Variant = VariantPostBuild, the files Port_Cfg.h and Port_PBcfg.c should be used.

Figure 4-1. Tresos Plugin snapshot for IMPLEMENTATION_CONFIG_VARIANT form.

Table 4-2. Attribute IMPLEMENTATION_CONFIG_VARIANT detailed description

Property	Value
Label	Config Variant
Default	VariantPreCompile
Range	VariantPreCompile VariantPostBuild

4.3 Form PortConfigSet

This container contains a configuration of the PORT driver / SIUL module. This container is a MultipleConfigurationContainer, i.e. this container and its sub-containers exit once per configuration set. NOTE:"User should configure only ONE instance of PortConfigSet container and not MULTIPLE."

Included forms :

- PortContainer

Figure 4-2. Tresa Plugin snapshot for PortConfigSet form.

4.3.1 Form PortContainer

Container collecting the PortPins.

Is included by form : PortConfigSet

Included forms :

- PortPin

Figure 4-3. Tresa Plugin snapshot for PortContainer form.

4.3.1.1 PortNumberOfPortPins (PortContainer)

The number of specified PortPins in this PortContainer.

Table 4-3. Attribute PortNumberOfPortPins (PortContainer) detailed description

Property	Value
Label	PortNumberOfPortPins
Type	INTEGER
Origin	AUTOSAR_ECUC

Table continues on the next page...

Table 4-3. Attribute PortNumberOfPortPins (PortContainer) detailed description (continued)

Property	Value
Symbolic Name	false
Lower Multiplicity	1
Upper Multiplicity	1
Invalid	Range <=500 >=1

4.3.1.2 Form PortPin

Configuration of the individual port pins.

Is included by form : PortContainer

The screenshot shows the 'PortPin' configuration window. At the top, the 'Name' field is set to 'Pin0'. Below this is the 'General' tab. The configuration options are arranged in two columns:

- Left Column:**
 - PortPin Wpe: ☐ (disabled)
 - PortPin Ode: ☐ (disabled)
 - PortPin Safe Mode: ☐ (disabled)
 - PortPin Readback: ☐ (disabled)
 - PortPin Id: (disabled)
 - PortPin Pcr: (disabled)
 - PortPin Mode: (disabled)
 - PortPin Initial Mode: (disabled)
 - PortPin Direction: (disabled)
 - PortPin Level Value: (disabled)
 - PortPin Output Slew Rate: (disabled)
- Right Column:**
 - PortPin Wps: ☐ (disabled)
 - PortPin Input Hysteresis: ☐ (disabled)
 - PortPin Direction Changeable: ☒ (disabled)

Figure 4-4. Tresos Plugin snapshot for PortPin form.

4.3.1.2.1 PortPinWpe (PortPin)

Enable Weak Pull Up/Down for the configured Pin. Checked box means any of the Weak Pull Up or Weak Pull Down configurations is enabled for the pin. This is an implementation specific parameter.

Table 4-4. Attribute PortPinWpe (PortPin) detailed description

Property	Value
Label	PortPin Wpe
Origin	Custom
Symbolic Name	false
Default	false
Lower Multiplicity	1
Upper Multiplicity	1

4.3.1.2.2 PortPinWps (PortPin)

Select Weak Pull Up/Down for the configured Pin. Checked box means the Weak Pull Up configuration is set. Unchecked box means the Weak Pull Down configuration is set. This is an implementation specific parameter.

Table 4-5. Attribute PortPinWps (PortPin) detailed description

Property	Value
Label	PortPin Wps
Origin	Custom
Symbolic Name	false
Default	false
Lower Multiplicity	1
Upper Multiplicity	1

4.3.1.2.3 PortPinOde (PortPin)

Enable Open Drain Output for the configured Pin. Checked box means the Open Drain configuration is set. This is an implementation specific parameter.

Table 4-6. Attribute PortPinOde (PortPin) detailed description

Property	Value
Label	PortPin Ode
Origin	Custom
Symbolic Name	false
Default	false
Lower Multiplicity	1
Upper Multiplicity	1

4.3.1.2.4 PortPinHysteresis (PortPin)

Enable/Disable Input Hysteresis for the configured Pin. This is an implementation specific parameter.

Table 4-7. Attribute PortPinHysteresis (PortPin) detailed description

Property	Value
Label	PortPin Input Hysteresis
Origin	Custom
Symbolic Name	false
Default	false
Lower Multiplicity	1
Upper Multiplicity	1

4.3.1.2.5 PortPinDirectionChangeable (PortPin)

Enable/Disable the changeability for the configured Pin. Checked box means the Direction Changeability is enabled. This is an implementation specific parameter.

Table 4-8. Attribute PortPinDirectionChangeable (PortPin) detailed description

Property	Value
Label	PortPin Direction Changeable
Type	BOOLEAN
Origin	AUTOSAR_ECUC
Symbolic Name	false
Default	true
Lower Multiplicity	1
Upper Multiplicity	1

4.3.1.2.6 PortPinId (PortPin)

Pin Id of the port pin. This value will be assigned to the symbolic name.

Table 4-9. Attribute PortPinId (PortPin) detailed description

Property	Value
Label	PortPin Id
Type	INTEGER
Origin	AUTOSAR_ECUC

Table continues on the next page...

Table 4-9. Attribute PortPinId (PortPin) detailed description (continued)

Property	Value
Symbolic Name	true
Lower Multiplicity	1
Upper Multiplicity	1
Invalid	Range >=0 <=0xFFFF

4.3.1.2.7 PortPinPcr (PortPin)

Pin Id of the port pin. This value will be assigned to the symbolic name derived from the port pin container short name.

Table 4-10. Attribute PortPinPcr (PortPin) detailed description

Property	Value
Label	PortPin Pcr
Origin	Custom
Symbolic Name	false
Lower Multiplicity	1
Upper Multiplicity	1
Invalid	Range >=0 <=0xFFFF

4.3.1.2.8 PortPinMode (PortPin)

Selects the PORT pin mode from the modes list. By default more than one mode are allowed. That way it is e.g. possible to combine DIO with another mode such as ICU. For the Alternative Function modes (not a GPIO mode) the OUT direction is hw selected for that pin. If the IN direction is needed too, it can be set at runtime. NOTE: To set the IN direction take care, please, that all the possible module inputs, possible as Alternative Functions for the pad mode, are hw connected together, if IN direction is enabled, to the pad.

Table 4-11. Attribute PortPinMode (PortPin) detailed description

Property	Value
Label	PortPin Mode
Type	ENUMERATION
Origin	AUTOSAR_ECUC

Table continues on the next page...

Table 4-11. Attribute PortPinMode (PortPin) detailed description (continued)

Property	Value
Symbolic Name	false
Default	GPIO
Lower Multiplicity	1
Upper Multiplicity	1

4.3.1.2.9 PortPinInitialMode (PortPin)

Port pin mode from mode list for use with Port_Init() function. NOTE: This parameter is not used in the current implementation and is retained as per std AUTOSAR_EcucParamDef.arxml file.

Table 4-12. Attribute PortPinInitialMode (PortPin) detailed description

Property	Value
Label	PortPin Initial Mode
Type	ENUMERATION
Origin	AUTOSAR_ECUC
Symbolic Name	false
Default	PORT_GPIO_MODE
Lower Multiplicity	1
Upper Multiplicity	1
Enable	false
Range	PORT_GPIO_MODE PORT_PRIMARY_FUNC_MODE PORT_ANALOG_INPUT_MODE PORT_ALT1_FUNC_MODE PORT_ALT2_FUNC_MODE PORT_ALT3_FUNC_MODE

4.3.1.2.10 PortPinDirection (PortPin)

Selects the initial direction of the pin (IN or OUT). If the direction is not changeable, the value configured here is fixed. The pin direction can be set only for the GPIO pins. For the Alternative Function modes the OUT pin direction is hw selected. If the IN direction is needed too, it can be set at runtime. NOTE: To set the IN direction take care, please, that all the possible module inputs, possible as Alternative Functions for the pad mode, are hw connected together, if IN direction is enabled, to the pad.

Table 4-13. Attribute PortPinDirection (PortPin) detailed description

Property	Value
Label	PortPin Direction
Type	ENUMERATION
Origin	AUTOSAR_ECUC
Symbolic Name	false
Default	PortPinDirectionIn
Lower Multiplicity	1
Upper Multiplicity	1

4.3.1.2.11 PortPinLevelValue (PortPin)

Port Pin Level value from Port pin list.

Table 4-14. Attribute PortPinLevelValue (PortPin) detailed description

Property	Value
Label	PortPin Level Value
Type	ENUMERATION
Origin	AUTOSAR_ECUC
Symbolic Name	false
Default	PortPinLevelLow
Lower Multiplicity	1
Upper Multiplicity	1
Range	PortPinLevelHigh PortPinLevelLow

4.3.1.2.12 PortPinSlewRate (PortPin)

Enable/Disable Slew Rate for the configured Pin. This is an implementation specific parameter.

Table 4-15. Attribute PortPinSlewRate (PortPin) detailed description

Property	Value
Label	PortPin Output Slew Rate
Origin	Custom
Symbolic Name	false
Default	MaximumValue
Lower Multiplicity	1

Table continues on the next page...

Table 4-15. Attribute PortPinSlewRate (PortPin) detailed description (continued)

Property	Value
Upper Multiplicity	1
Range	MaximumValue MediumValue MinimumValue

4.3.1.2.13 PortPinDriveStrength (PortPin)

Select the pad drive strength. Drive strength control pertains to pins with the fast I/O pad type.

Table 4-16. Attribute PortPinDriveStrength (PortPin) detailed description

Property	Value
Label	PortPin Drive Strength Control
Origin	Custom
Symbolic Name	false
Default	DriveStrength_10pF
Range	DriveStrength_10pF DriveStrength_20pF DriveStrength_30pF DriveStrength_50pF

4.4 Form CommonPublishedInformation

Common container, aggregated by all modules. It contains published information about vendor and versions.

The screenshot shows the 'Port' configuration window. At the top, there's a 'Name' field with the value 'Port'. Below it are three tabs: 'General', 'PortConfigSet', and 'Published Information'. The 'Published Information' tab is selected. Under this tab, there's a section titled 'CommonPublishedInformation' with a folder icon. Below this, there are several input fields, each with a lightbulb icon to its left, indicating they are required or have a default value. The fields and their values are: 'AUTOSAR Major Version' (3), 'AUTOSAR Minor Version' (0), 'AUTOSAR Patch Version' (2), 'Module Id.' (124), 'Software Major Version' (0), 'Software Minor Version' (9), 'Software Patch Version' (0), 'Vendor Api Infix' (0), and 'Vendor Id.' (27).

Figure 4-5. Tresos Plugin snapshot for CommonPublishedInformation form.

4.4.1 ArMajorVersion (CommonPublishedInformation)

Major version number of AUTOSAR specification on which the appropriate implementation is based on.

Table 4-17. Attribute ArMajorVersion (CommonPublishedInformation) detailed description

Property	Value
Label	AUTOSAR Major Version
Origin	Custom
Symbolic Name	false
Default	3
Lower Multiplicity	1
Upper Multiplicity	1
Invalid	Range <div>>=3</div> <div><=3</div>

4.4.2 ArMinorVersion (CommonPublishedInformation)

Minor version number of AUTOSAR specification on which the appropriate implementation is based on.

Table 4-18. Attribute ArMinorVersion (CommonPublishedInformation) detailed description

Property	Value
Label	AUTOSAR Minor Version
Origin	Custom
Symbolic Name	false
Default	0
Lower Multiplicity	1
Upper Multiplicity	1
Invalid	Range ≥ 0 ≤ 0

4.4.3 ArPatchVersion (CommonPublishedInformation)

Patch version number of AUTOSAR specification on which the appropriate implementation is based on.

Table 4-19. Attribute ArPatchVersion (CommonPublishedInformation) detailed description

Property	Value
Label	AUTOSAR Patch Version
Origin	Custom
Symbolic Name	false
Default	2
Lower Multiplicity	1
Upper Multiplicity	1
Invalid	Range ≥ 2 ≤ 2

4.4.4 ModuleId (CommonPublishedInformation)

Module ID of this module from Module List.

Table 4-20. Attribute ModuleId (CommonPublishedInformation) detailed description

Property	Value
Label	Module Id.
Origin	Custom
Symbolic Name	false
Default	124
Lower Multiplicity	1
Upper Multiplicity	1
Invalid	Range ≥ 124 ≤ 124

4.4.5 SwMajorVersion (CommonPublishedInformation)

Major version number of the vendor specific implementation of the module.

Table 4-21. Attribute SwMajorVersion (CommonPublishedInformation) detailed description

Property	Value
Label	Software Major Version
Origin	Custom
Symbolic Name	false
Default	1
Lower Multiplicity	1
Upper Multiplicity	1
Invalid	Range ≥ 1 ≤ 1

4.4.6 SwMinorVersion (CommonPublishedInformation)

Minor version number of the vendor specific implementation of the module.

Table 4-22. Attribute SwMinorVersion (CommonPublishedInformation) detailed description

Property	Value
Label	Software Minor Version

Table continues on the next page...

Table 4-22. Attribute SwMinorVersion (CommonPublishedInformation) detailed description (continued)

Property	Value
Origin	Custom
Symbolic Name	false
Default	9
Lower Multiplicity	1
Upper Multiplicity	1
Invalid	Range >=9 <=9

4.4.7 SwPatchVersion (CommonPublishedInformation)

Patch version number of the vendor specific implementation of the module.

Table 4-23. Attribute SwPatchVersion (CommonPublishedInformation) detailed description

Property	Value
Label	Software Patch Version
Origin	Custom
Symbolic Name	false
Default	0
Lower Multiplicity	1
Upper Multiplicity	1
Invalid	Range >=0 <=0

4.4.8 VendorApiInfix (CommonPublishedInformation)

In driver modules which can be instantiated several times on a single ECU, BSW00347 requires that the name of APIs is extended by the VendorId and a vendor specific name. This parameter is used to specify the vendor specific name. In total, the implementation specific name is generated as follows:

<ModuleName>_>VendorId>_<VendorApiInfix><Api name from SWS>. E.g. assuming that the VendorId of the implementor is 123 and the implementer chose a

VendorApiInfix of "v11r456" a api name Can_Write defined in the SWS will translate to Can_123_v11r456Write. This parameter is mandatory for all modules with upper multiplicity > 1. It shall not be used for modules with upper multiplicity =1.

Table 4-24. Attribute VendorApiInfix (CommonPublishedInformation) detailed description

Property	Value
Label	Vendor Api Infix
Origin	AUTOSAR_ECUC V1.0.0
Symbolic Name	false
Default	
Enable	false

4.4.9 VendorId (CommonPublishedInformation)

Vendor ID of the dedicated implementation of this module.

Table 4-25. Attribute VendorId (CommonPublishedInformation) detailed description

Property	Value
Label	Vendor Id.
Origin	Custom
Symbolic Name	false
Default	27
Lower Multiplicity	1
Upper Multiplicity	1
Invalid	Range ≥ 27 ≤ 27

4.5 Form PortGeneral

Module wide configuration parameters of the PORT driver.

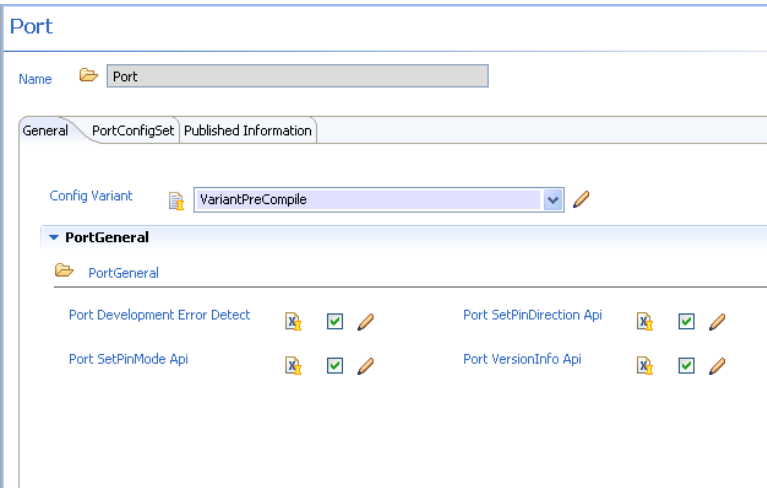


Figure 4-6. Tresos Plugin snapshot for PortGeneral form.

4.5.1 PortDevErrorDetect (PortGeneral)

Switches the Development Error Detection and Notification ON or OFF.

Table 4-26. Attribute PortDevErrorDetect (PortGeneral) detailed description

Property	Value
Label	Port Development Error Detect
Type	BOOLEAN
Origin	AUTOSAR_ECUC
Symbolic Name	false
Default	true
Lower Multiplicity	1
Upper Multiplicity	1

4.5.2 PortSetPinDirectionApi (PortGeneral)

Pre-processor switch to enable/disable the use of the function Port_SetPinDirection().

Table 4-27. Attribute PortSetPinDirectionApi (PortGeneral) detailed description

Property	Value
Label	Port SetPinDirection Api
Type	BOOLEAN
Origin	AUTOSAR_ECUC

Table continues on the next page...

Table 4-27. Attribute PortSetPinDirectionApi (PortGeneral) detailed description (continued)

Property	Value
Symbolic Name	false
Default	true
Lower Multiplicity	1
Upper Multiplicity	1

4.5.3 PortSetPinModeApi (PortGeneral)

Pre-processor switch to enable/disable the use of the function Port_SetPinMode().

Table 4-28. Attribute PortSetPinModeApi (PortGeneral) detailed description

Property	Value
Label	Port SetPinMode Api
Type	BOOLEAN
Origin	AUTOSAR_ECUC
Symbolic Name	false
Default	true
Lower Multiplicity	1
Upper Multiplicity	1

4.5.4 PortVersionInfoApi (PortGeneral)

Pre-processor switch to enable/disable the API to read out the modules version information.

Table 4-29. Attribute PortVersionInfoApi (PortGeneral) detailed description

Property	Value
Label	Port VersionInfo Api
Type	BOOLEAN
Origin	AUTOSAR_ECUC
Symbolic Name	false
Default	true
Lower Multiplicity	1
Upper Multiplicity	1

How to Reach Us:

Home Page:

www.freescale.com

Web Support:

<http://www.freescale.com/support>

USA/Europe or Locations Not Listed:

Freescale Semiconductor
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
+1-800-521-6274 or +1-480-768-2130
www.freescale.com/support

Europe, Middle East, and Africa:

Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
www.freescale.com/support

Japan:

Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:

Freescale Semiconductor China Ltd.
Exchange Building 23F
No. 118 Jianguo Road
Chaoyang District
Beijing 100022
China
+86 10 5879 8000
support.asia@freescale.com

For Literature Requests Only:

Freescale Semiconductor Literature Distribution Center
1-800-441-2447 or +1-303-675-2140
Fax: +1-303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductors products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals", must be validated for each customer application by customer's technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claims alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

RoHS-compliant and/or Pb-free versions of Freescale products have the functionality and electrical characteristics as their non-RoHS-complaint and/or non-Pb-free counterparts. For further information, see <http://www.freescale.com> or contact your Freescale sales representative.

For information on Freescale's Environmental Products program, go to <http://www.freescale.com/epp>.

Freescale™ and the Freescale logo are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners.

© 2011 Freescale Semiconductor, Inc.



AUTOSAR and AUTOSAR logo are registered trademarks of AUTOSAR GbR (www.autosar.org)