User Manual

for S32K1 SPI Driver

Document Number: UM2SPIASR4.4 Rev0000R1.0.1 Rev. 1.0

Chapter 1

Revision History

Revision	Date	Author	Description
1.0	24.02.2022	NXP RTD Team	Prepared for release RTD S32K1 Version 1.0.1

Chapter 2

Introduction

This User Manual describes NXP Semiconductor AUTOSAR Serial Peripheral Interface (SPI) for S32K1XX .

AUTOSAR SPI driver configuration parameters and deviations from the specification are described in SPI Driver chapter of this document. AUTOSAR SPI driver requirements and APIs are described in the AUTOSAR SPI 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 NXP Semiconductors:

- s32k116_qfn32
- s32k116_lqfp48
- s32k118_lqfp48
- s32k118_lqfp64
- s32k142_lqfp48
- s32k142_lqfp64
- s32k142_lqfp100
- s32k142w_lqfp48
- s32k142w_lqfp64

- s32k144_lqfp48
- s32k144_lqfp64
- s32k144_lqfp100
- s32k144_mapbga100
- s32k144w_lqfp48
- s32k144w_lqfp64
- s32k146_lqfp64
- s32k146_lqfp100
- s32k146_mapbga100
- s32k146_lqfp144
- s32k148_lqfp100
- s32k148_mapbga100
- s32k148_lqfp144
- s32k148_lqfp176

All of the above microcontroller devices are collectively named as S32K1.

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** style: Used for important terms, notes and warnings.
- *Italic* style: 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.

Warning

This is a warning

2.4 Acronyms and Definitions

Term	Definition
API	Application Programming Interface
AUTOSAR	Automotive Open System Architecture
BSMI	Basic Software Make file Interface
CS	Chip Select
DEM	Diagnostic Event Manager
DET	Development Error Tracer
DMA	Direct Memory Access
ECU	Electronic Control Unit
FIFO	First In First Out
LSB	Least Signifigant Bit
MCU	Micro Controller Unit
MIDE	Multi Integrated Development Environment
MSB	Most Significant Bit
N/A	Not Applicable
RAM	Random Access Memory
SIU	Systems Integration Unit
SWS	Software Specification
SPI	Serial Peripheral Interface
XML	Extensible Markup Language
BSW	Basic Software
ISR	Interrupt Service Routine
OS	Operating System
GUI	Graphical User Interface
PB Variant	Post Build Variant
PC Variant	Pre Compile Variant
LT Variant	Link Time Variant

2.5 Reference List

#	Title	Version
1	Specification of SPI Driver	AUTOSAR Release 4.↩ 4.0
2	S32K1XX Series Reference Manual	Rev. 14, 09/2021
3	Errata S32K116_0N96V	Rev. 22/OCT/2021
4	Errata S32K118_0N97V	Rev. 22/OCT/2021
5	Errata S32K142_0N33V	Rev. 22/OCT/2021
6	Errata S32K144_0N57U	Rev. 22/OCT/2021
7	Errata S32K144W_0P64A	Rev. 22/OCT/2021
8	Errata S32K146_0N73V	Rev. 22/OCT/2021
9	Errata S32K148_0N20V	Rev. 22/OCT/2021
10	S32K1xx Data Sheet	Rev. 14, 08/2021

Chapter 3

Driver

3.1 Requirements

Requirements for this driver are detailed in the AUTOSAR 4.4 Rev0000 SPI Driver Software Specification document (See Table Reference List).

AUTOSAR deviations from requirements are described in Deviations from Requirements chapter of this document..

3.2 Driver Design Summary

The SPI Handler and Driver provide services for reading from and writing to devices connected via SPI busses. It provides access to SPI communication to several users (e.g., EEPROM, Watchdog, I/O ASICs). It also provides the required mechanism to configure the on-chip SPI peripheral.

This specification describes the API, Mapping to SWS requirements for a monolithic SPI Handler and Driver. This software module includes handling and driving functionalities. Main objectives of this monolithic SPI Driver are to take the best of each microcontroller features and to allow implementation optimization depending on static configuration to fit as much as possible to ECU needs.

The general behavior of the SPI Handler and Driver could be asynchronous or synchronous according to the level of functionality selected.

The specification covers the Handler and Driver functionalities combined in one single module. The SPI handler controls multiple accesses to busses that could be located in the ECU Abstraction layer. The other part is the SPI driver that accesses the microcontroller hardware directly that could be located in the Microcontroller Abstraction layer.

SPI Dual Clock Mode

The SPI Driver allows to be used in Dual Clock Mode. This mode permits to change the clock reference(referred by the field SpiPhyUnitAlternateClockRef) and to keep the basic characteristics of the transmission(like baudrate). This is useful when it wants to be crossed to a low frequency(low power) or higher frequency.

Notification usage

To be able to use the SPI driver with DMA functionality, the following function need to be set as notification for the DMA channels used: Lpspi_Ip_LPSPI_X_IrqTxDmaHandler, Flexio_Spi_Ip_FLEXIO_SPI_X_IrqTxDma↩ Handler and Lpspi_Ip_LPSPI_X_IrqRxDmaHandler, Flexio_Spi_Ip_FLEXIO_SPI_0_IrqRxDmaHandler where X is the SPI unit used (eg: Lpspi_Ip_LPSPI_0_IrqTxDmaHandler if LPSPI0 is used).

Interrupt request usage

Every interrupt is guarded by #ifdef definitions that specify if the corresponded SPI is used. If not, the interrupt function is removed. A template of the #ifdef guard is:

#if (<SPI_X_ENABLED> == STD_ON)

<ISR_function_name()>

#endif

Description of the symbolic names

When the plugin is generated, symbolic names of the sequences, jobs and channels are created by define macros. The templates of the defines are:

- for sequences:

  #define SpiConf_SpiSequence_<SpiSequenceName> ((Spi_SequenceType)<SpiSequenceID>)

- for jobs:

  #define SpiConf_SpiJob_<SpiJobName> ((Spi_JobType)<SpiJobID>)

- for channels:

  #define SpiConf_SpiChannel_<SpiChannelName> ((Spi_ChannelType)<SpiChannelID>)

  Name is the name of the container and the ID is configurable by the user

3.3 Hardware Resources

The hardware configured by the Spi driver is the same between derivatives.

SPI Physical Units: Has a different on number instance between each derivatives of the chip S32K1XX

- Only S32K116 has LPSPI_0, FLEXIO_SPI_0 and FLEXIO_SPI_1 hardware instances.
- Both S32K118 and S32K142 have LPSPI_0, LPSPI1, FLEXIO_SPI_0 and FLEXIO_SPI_1 hardware instances.
- And the rest S32k142w, S32K144, S32K144w, S32K146 and S32K148 have LPSPI_0, LPSPI_1, LPSPI_2, FLEXIO_SPI_0, FLEXIO_SPI_1 hardware instances.

Note

In EB tresos, SPI Physical Unit has selected by SpiPhyUnitMapping.

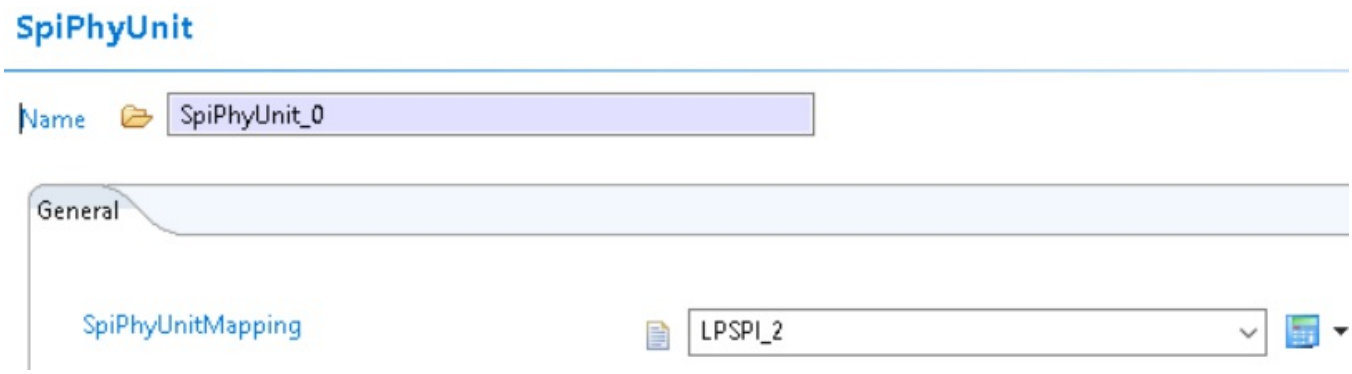

Figure 3.1 SPI Physical Unit has selected by SpiPhyUnitMapping in EB Tresos

For example with the chip S32K1XX:

The LPSPI_0, LPSPI_1 and LPSPI_2 use the pins correspondingly with naming is LPSPI0, LPSPI1, LPSPI2. The FLEXIO_SPI_0 and FLEXIO_SPI_1 use the pins correspondingly with naming is FXIO. The pins can find in the file IO Signal Description for each derivatives (e.g. "S32K148_IO_Signal_Description_Input_Multiplexing.xlsx" from attached file of S32K1XX Reference Manual Reference List.

LPSPI1 can be found in the xlsx file with naming is LPSPI1. And the Pin-Muxing is:

LPSPI1	LPSPI1_PCS0	1	0000_0011	IO_PAD	PTD3	D10	101	122	70
		2	0000_0101	IO_PAD	PTE1	A4	137	165	93
		3	0000_0100	IO_PAD	PTA21			6	
		4	0000_0011	IO_PAD	PTA26		19	26	
		5		-	disable low				
LPSPI1	LPSPI1_PCS1	1	0000_0011	IO_PAD	PTA6	G9	85	104	58
		2	0000_0100	IO_PAD	PTB18		36	43	
		3	0000_0100	IO_PAD	PTA22			9	
		4		-	disable low				
LPSPI1	LPSPI1_SCK	1	0000_0011	IO_PAD	PTD0	C2	4	8	4
		2	0000_0011	IO_PAD	PTB14	F10	96	116	66
		3	0000_0100	IO_PAD	PTA19			2	
		4	0000_0011	IO_PAD	PTA28		24	31	
		5		-	disable low				
LPSPI1	LPSPI1_SIN	1	0000_0011	IO_PAD	PTB15	F9	95	115	65
		2	0000_0011	IO_PAD	PTD1	B1	3	7	3
		3	0000_0100	IO_PAD	PTA20			3	
		4	0000_0101	IO_PAD	PTA29		26	33	
		5		-	disable low				
LPSPI1	LPSPI1_SOUT	1	0000_0011	IO_PAD	PTD2	D9	102	123	71
		2	0000_0011	IO_PAD	PTB16	F8	94	114	64
		3	0000_0101	IO_PAD	PTE0	B4	138	166	94
		4	0000_0100	IO_PAD	PTA18			1	
		5	0000_0011	IO_PAD	PTA27		22	29	
		6		-	disable low				

Figure 3.2 IO Signal Description for LPSPI_1

3.4 Deviations from Requirements

The driver deviates from the AUTOSAR Spi Driver software specification in some places. The table identifies the AUTOSAR requirements that are not fully implemented, implemented differently, not available, not testable or out of scope for the Spi Driver.

Term	Definition
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, not available, not testable or out of scope for the driver.

Table 3.2 Driver Deviations Table

Requirement	Status	Description	Notes
SWS_Spi_00040	N/S	The SPI Handler/Driver handles only the Master mode.	Both Master and Slave are supported by SPI driver.
SWS_Spi_00342	N/S	Depending on microcontrollers, the SPI peripheral could share registers with other peripherals. In this typical case, the SPI Handler/Driver has a relationship with MCU module [REF] for initialising and de-initialising those registers.	There is no register that is shared with other peripherals.

Requirement	Status	Description	Notes
SWS_Spi_00270	N/S	In case call end notification function and rescheduling are fully done by software, the order between these shall be first scheduling and then the call of end notification function executed.	Job and sequences notifications are performed before the scheduling of the next job (contrary to the recommendation given by SPI270) . In this way, calls like Spi_SetupIB() or Spi_WriteIB() can be targeted on the next schedulable jobs, before the starting of the job transfer.
SWS_Spi_00195	N/S	SPI Handler/driver shall be able to detect the error SPI_E↩ _HARDWARE_ERROR when an hardware error occur during asynchronous or synchronous transmit. Please see also SW↩ S_Spi_00267 and SWS_Spi_↩ 00384.	SPI_E_HARDWARE_↩ ERROR is not supported for Spi_Async_Transmit() Function. It is supported for Spi_Sync_Transmit() function. To implement this requirement a timer should be set to that estimated value and if the timer expires, then it can be assumed that a hardware error occurred. This would add a dependency of Gpt.

Requirement	Status	Description	Notes
SWS_Spi_00383	N/S	Error Name: SPI_E_HA↩ RDWARE_ERROR Short Description: An hardware error occurred during asynchronous or synchronous SPI transmit. Long Description: This Extended Production Error shall be issued when any error bit inside the S↩ PI hardware transmit status register is raised Detection Criteria: Fail The SPI transmit status register information shall be reported to DEM as Dem_ReportErrorStatus (SP↩ I_E_HARDWARE_ERROR, DEM_EVENT_STATUS_↩ FAILED) when any error bit inside the SPI transmit status register is set. (SWS_Spi_↩ 00385) Pass The SPI transmit status register information shall be reported to DEM as Dem_ReportErrorStatus (SP↩ I_E_HARDWARE_ERROR, DEM_EVENT_STATUS_↩ PASSED) when no error bit inside the SPI transmit status register is set. (SWS_Spi_↩ 00386)	SPI_E_HARDWARE_↩ ERROR is not supported for Spi_Async_Transmit() Function. It is supported for Spi_Sync_Transmit() function. To implement this requirement a timer should be set to that estimated value and if the timer expires, then it can be assumed that a hardware error occurred. This would add a dependency of Gpt.
SWS_Spi_00385	N/S	When any error bit inside the SPI transmit status register is set, the SPI transmit status register information shall be reported to DEM as Dem_ReportErrorStatus (SP↩ I_E_HARDWARE_ERROR, DEM_EVENT_STATUS_F↩ AILED)	SPI_E_HARDWARE_↩ ERROR is not supported for Spi_Async_Transmit() Function. It is supported for Spi_Sync_Transmit() function. To implement this requirement a timer should be set to that estimated value and if the timer expires, then it can be assumed that a hardware error occurred. This would add a dependency of Gpt.

Requirement	Status	Description	Notes
SWS_Spi_00386	N/S	When no error bit inside the SPI transmit status register is set, the SPI transmit status register information shall be reported to DEM as Dem_ReportErrorStatus (SPI_E_HARDWARE_ERROR, DEM_EVENT_STATUS_PASSED)	SPI_E_HARDWARE_ERROR is not supported for Spi_Async_Transmit() Function. It is supported for Spi_Sync_Transmit() function. To implement this requirement a timer should be set to that estimated value and if the timer expires, then it can be assumed that a hardware error occurred. This would add a dependency of Gpt.
SWS_Spi_00293	N/S	When the function Spi_AsyncTransmit is called, the SPI Handler/Driver shall handle the Job results. Result shall be SPI_JOB_FAILED when the transmission of Jobs is failed.	The Spi_AsyncTransmit can only schedule Jobs to be sent. So the function itself cannot detect if a job is failed.
SWS_Spi_00999	N/S	These requirements are not applicable to this specification.	This is not a requirement.
ECUC_Spi_00239	N/S	When the Chip select handling is enabled (see SpiEnableCs), then this parameter specifies if the chip select is handled automatically by Peripheral HW engine or via general purpose IO by Spi driver.	If user selects paraeter CS_VIA_GPIO, the user has to use SpiJobStartNotification' & 'SpiJobEndNotification' to toggle the CS (chip select pin) using DIO drivers .
SWS_Spi_CONSTR_00001	N/S	DRATF: The ECUC partitions referenced by SpiKernelEcucPartitionRef shall be a subset of the ECUC partitions referenced by SpiEcucPartitionRef.	Type IV Autosar multicore not implemented for current module. Multicore Type II is implemented. AAI-445.

3.5 Driver Limitations

3.5.1 Some LPSPI's features are not supported:

- Host request can be used to control the start of a SPI bus transfer
- Receive data match logic supporting wakeup on data match

3.5.2 Limitation of LPSPI Slave mode:

- In Slave mode and use interrupt mode without DMA, the application needs to make sure Slave's interrupt service is not delayed to avoid errors underflow and overflow occur.

3.5.3 Some limitations of SPI over FLEXIO:

- Do not support CS Non-continous mode.
- Do not support the polarity of CS is HIGH (only LOW is supported).
- "SpiCsIdentifier" (in SpiExternalDevice) is not used to select the no. of CS. It will be configured by "Spi↩ FlexioRxAndCsChannelsConfig" in "SpiPhyUnit". Please refer "How to configure SPI over FLEXIO" section for more details
- SPI FLEXIO does not support to configure the timing delay.
- In DMA mode, Spi over FLEXIO does not support MSB mode in both transmission and reception modes except data width differents to 8, 16 or 32 bits.

3.6 Driver usage and configuration tips

3.6.1 How to use dual clock mode

This mode permits to change the clock reference(referred by the field SpiPhyUnitAlternateClockRef) and to keep the basic characteristics of the transmission(like baudrate, delay time). The user can configuration two McuClock↩ SettingConfig are McuClockSettingConfig_Normal and McuClockSettingConfig_Alternate:

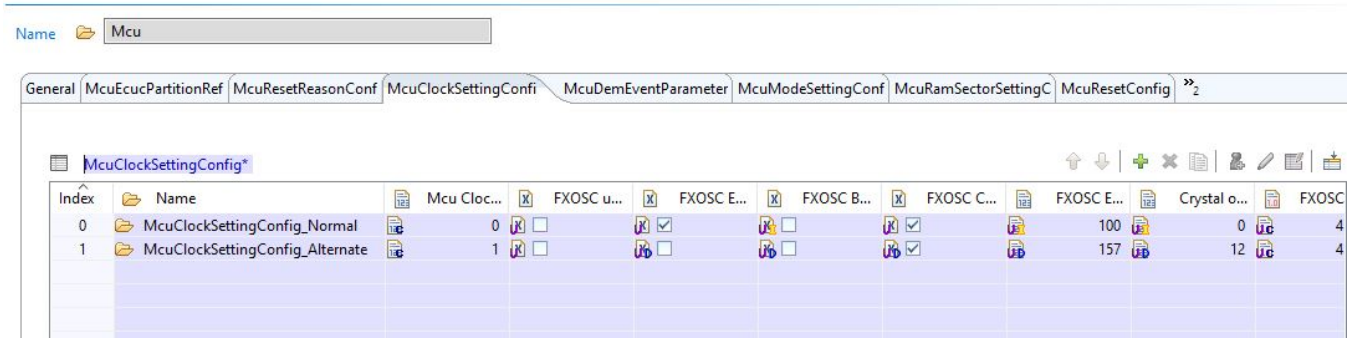

Figure 3.3 Configuration two McuClockSettingConfig

The field SpiPhyUnitClockRef will be referred to McuClockSettingConfig_Normal.

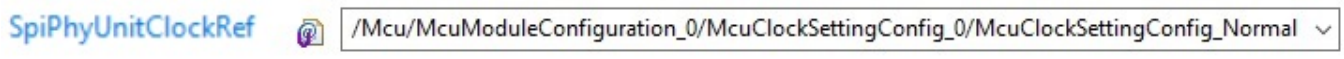

Figure 3.4 SpiPhyUnitClockRef will be referred to McuClockSettingConfig_Normal

The field SpiPhyUnitAlternateClockRef will be referred to McuClockSettingConfig_Alternate.

Figure 3.5 SpiPhyUnitAlternateClockRef will be referred to McuClockSettingConfig_Alternate

The default, Spi driver will use Normal Clock with the clock reference by the field SpiPhyUnitClockRef. If the user changes the clock setting to McuClockSettingConfig_Alternate by the function Mcu_InitClock(McuClockSetting↩ Config_Alternate), the user also changes clock mode for Spi driver by the function Spi_SetClockMode(SPI_ALT↩ ERNATE).

3.6.2 How to configure handling Chip Select via general purpose IO (SpiCsSelection: CS_VIA_GPIO)

The chip select is handled automatically by Pe-ripheral HW engine or via general purpose IO by Spi driver. In the case of the hardware does not support to keep chip select asserted between frame transfers. The user can use the CS_VIA_GPIO feature (selected by node SpiCsSelection) and the driver will call functions notification(defined by nodes SpiJobStartNotification and SpiJobEndNotification) to control CS pin via GPIO for each Job. By this way, SPI driver can communication with external device requires Continuous CS.

The configuration steps use the CS_VIA_GPIO feature as below:

- Select the CS_VIA_GPIO feature by node SpiCsSelection

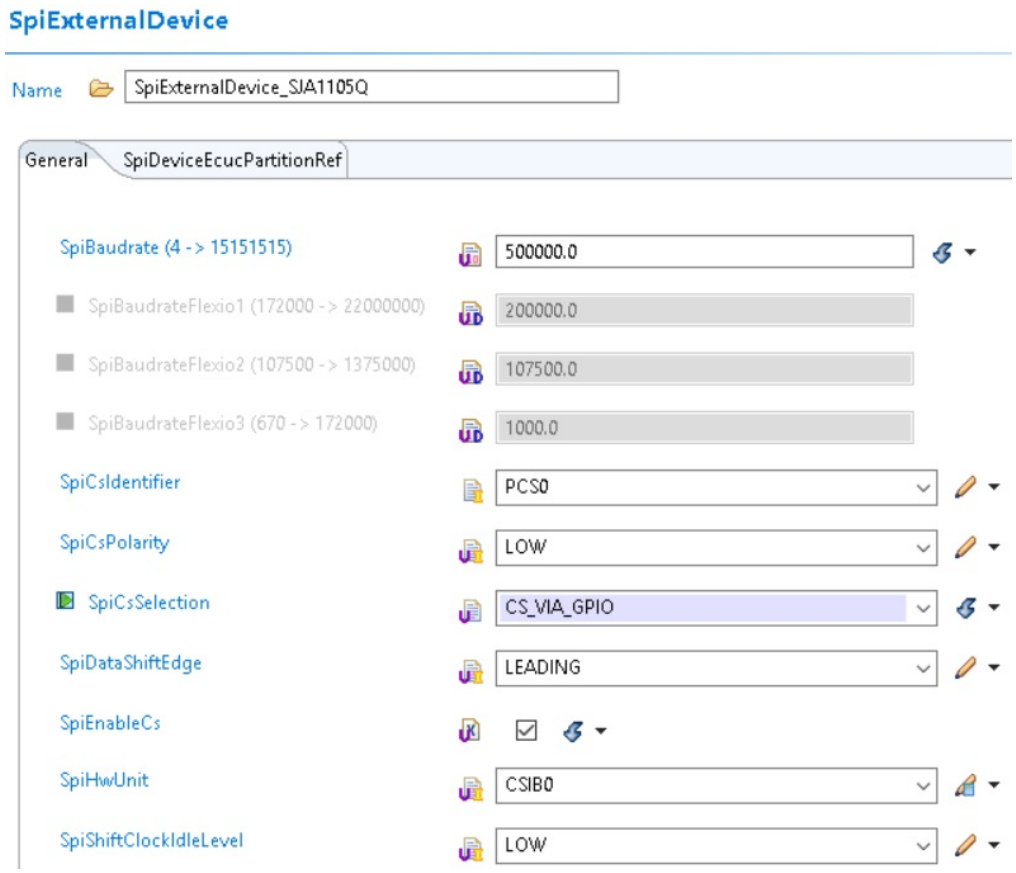

Figure 3.6 Select the CS_VIA_GPIO feature

- Enable Job Start Notification by node SpiJobStartNotificationEnable

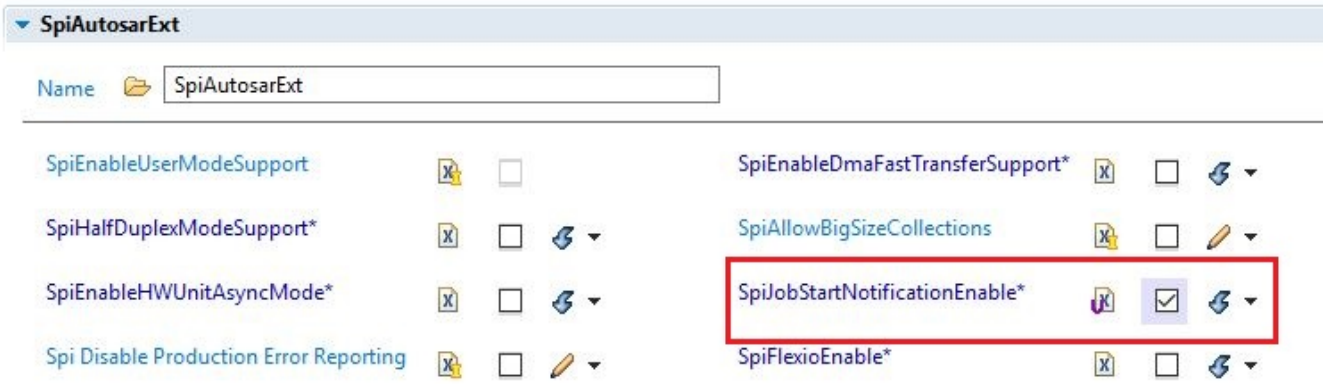

Figure 3.7 Enable Job Start Notification on SpiAutosarExt form

- Define two functions Job Start Notification and Job End Notification. Ex: Start_Job_Notification_Function, End_Job_Notification_Function. Start_Job_Notification_Function function will must assert CS via GPIO. End_Job_Notification_Function will must de-assert CS via GPIO. After that, enter name of two functions on nodes SpiJobStartNotification and SpiJobEndNotification.

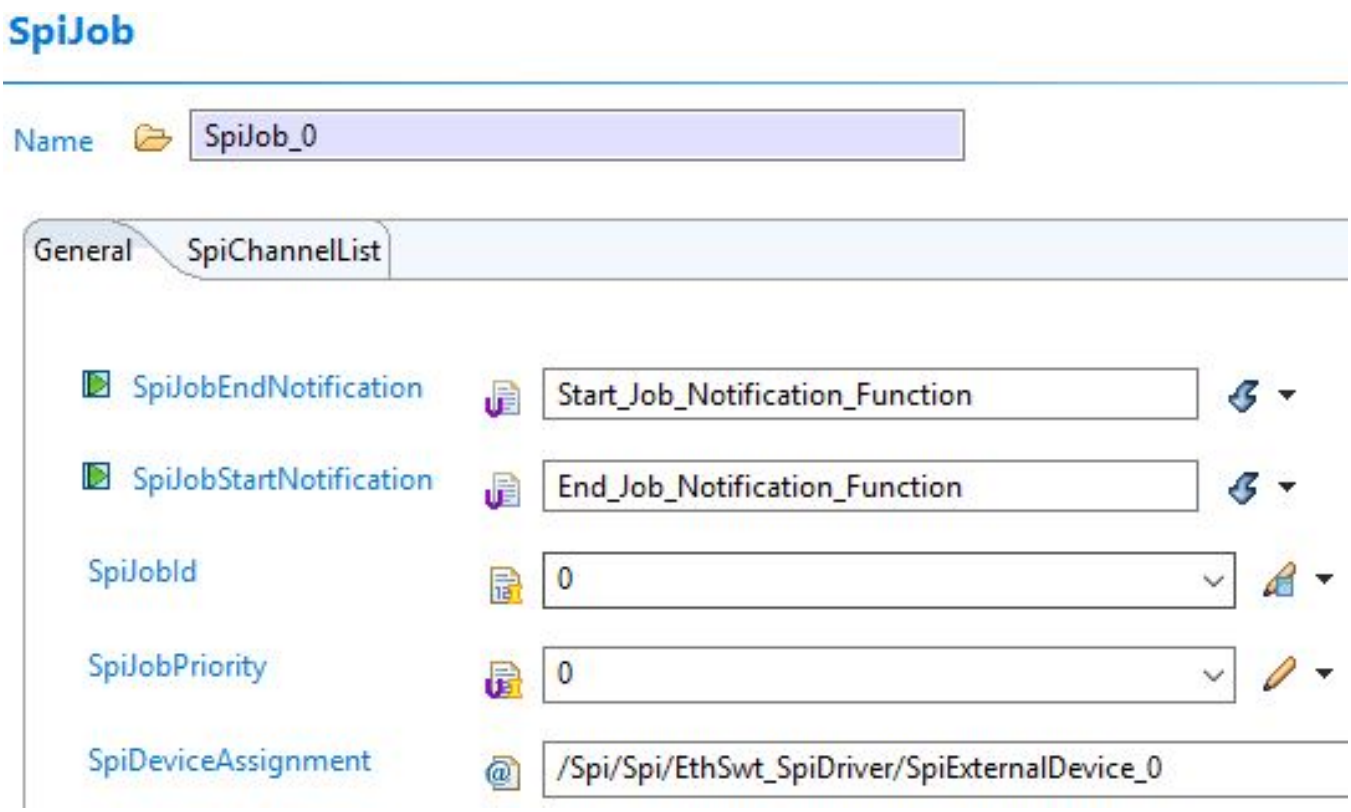

Figure 3.8 Enter name of two functions SpiJobStartNotification and SpiJobEndNotification on SpiJob form

3.6.3 How to configure in multiple post build variants

To use multiple post build variants, the configurations need to have the same symbolic name. So, all Names and IDs of Sequences, Jobs, Channels must the same between all post build variants. Names and Index of ExternalDevices must the same between all post build variants.

Scenario:

Let's assume there are 2 post build variant configurations, VS_0 and VS_1.

- VS_0 defines 2 Sequences, 2 Jobs, 2 Channels and 2 ExternalDevices:
- Name of sequence is SEQ_DSPI0_1J_C0_1 and SpiSequenceId is 0.
- Name of job is JOB_DSPI0_C0_1 and SpiJobId is 0.
- Name of ExternalDevice is DEV_EXP_100K_LEAD_0 and Index is 0.
- Name of channel is CH_EB_10K and SpiChannelId is 0.
- Name of sequence is SEQ_DSPI1_1J_C0_2 and SpiSequenceId is 1.
- Name of job is JOB_DSPI1_C0_2 and SpiJobId is 1.
- Name of ExternalDevice is DEV_EXP_100K_LEAD_1 and Index is 1.

- Name of channel is CH_EB_1K and SpiChannelId is 1.

So, VS_1 must has configuration the same Names, IDs and Indexs with VS_0 for Sequences, Jobs, Channels and ExternalDevices.

- VS_1 defines 2 Sequences, 2 Jobs, 2 Channels and 2 ExternalDevices:
- Name of sequence is SEQ_DSPI0_1J_C0_1 and SpiSequenceId is 0.
- Name of job is JOB_DSPI0_C0_1 and SpiJobId is 0.
- Name of ExternalDevice is DEV_EXP_100K_LEAD_0 and Index is 0.
- Name of channel is CH_EB_10K and SpiChannelId is 0.
- Name of sequence is SEQ_DSPI1_1J_C0_2 and SpiSequenceId is 1.
- Name of job is JOB_DSPI1_C0_2 and SpiJobId is 1.
- Name of ExternalDevice is DEV_EXP_100K_LEAD_1 and Index is 1.
- Name of channel is CH_EB_1K and SpiChannelId is 1.

The generated symbolic names for VS_0 and VS_1 will be:

```
#define SpiConf_SpiSequence_SEQ_DSPI0_1J_C0_1 0

#define SpiConf_SpiSequence_SEQ_DSPI1_1J_C0_2 1

#define SpiConf_SpiJob_JOB_DSPI0_C0_1 0

#define SpiConf_SpiJob_JOB_DSPI1_C0_2 1

#define SpiConf_SpiChannel_CH_EB_10K 0

#define SpiConf_SpiChannel_CH_EB_1K 1
```

It allows the upper layer to use same channel symbolic name(s) across multiple configurations.

3.6.4 How to configure Dma Fast transfer for a Sequence

The SPI driver can be able to transfer a Sequence with multiple Channels, Jobs without any CPU intervention between the channels and jobs. The CPU is used only to start the transmission (Spi_AsyncTransmit) and to process end of sequence notification. The driver will use DMA Scatter/Gather feature for all 2 DMA channels Spi↩ PhyTxDmaChannel and SpiPhyRxDmaChannel. SpiPhyTxDmaChannel will fill data to TCR and TDR registers. SpiPhyRxDmaChannel will read data from RDR register.

This feature requires:

- The parameters SpiBaudrate, SpiHwUnit, SpiTimeClk2Cs, SpiTimeCs2Clk, SpiTimeCs2Cs in External Device linked to each Job in this Sequence must be the same.
- The parameters SpiDataWidth and SpiTransferStart in Channel assigned to each Job in this Sequence must be the same.

- In each Channel, the number of data buffers is NOT higher than 32767 if SpiDataWidth < 9. So, SpiIbNBuffers and SpiEbMaxLength must be assigned to suitable values.
- Only Master mode is supported(SpiPhyUnitMode = SPI_MASTER).
- Make sure that SpiMaxDmaFastTransfer value in SpiPhyUnit allocated to this Sequence must NOT lower than total of Channels in this Sequence.
- Make sure that number of ScatterGathers configuration in each SpiPhyUnit/SpiPhyTxDmaChannel must NOT lower than (total of Channels ∗ 2) plus total of Jobs in this Sequence.
- Make sure that number of ScatterGathers configuration in each SpiPhyUnit/SpiPhyRxDmaChannel must NOT lower than total of Channels in this Sequence.
- Only SpiJobStartNotification and SpiJobEndNotification can be supported for first Job in a Sequence. But SpiJobEndNotification will be called at the end of Sequence as SpiSeqEndNotification.

Configuration example for 4 Channels and 2 Jobs in Sequence:

- Enable Dma Fast transfer support.

Figure 3.9 Enable Dma Fast transfer support

- Enable Dma Fast transfer for Sequence.

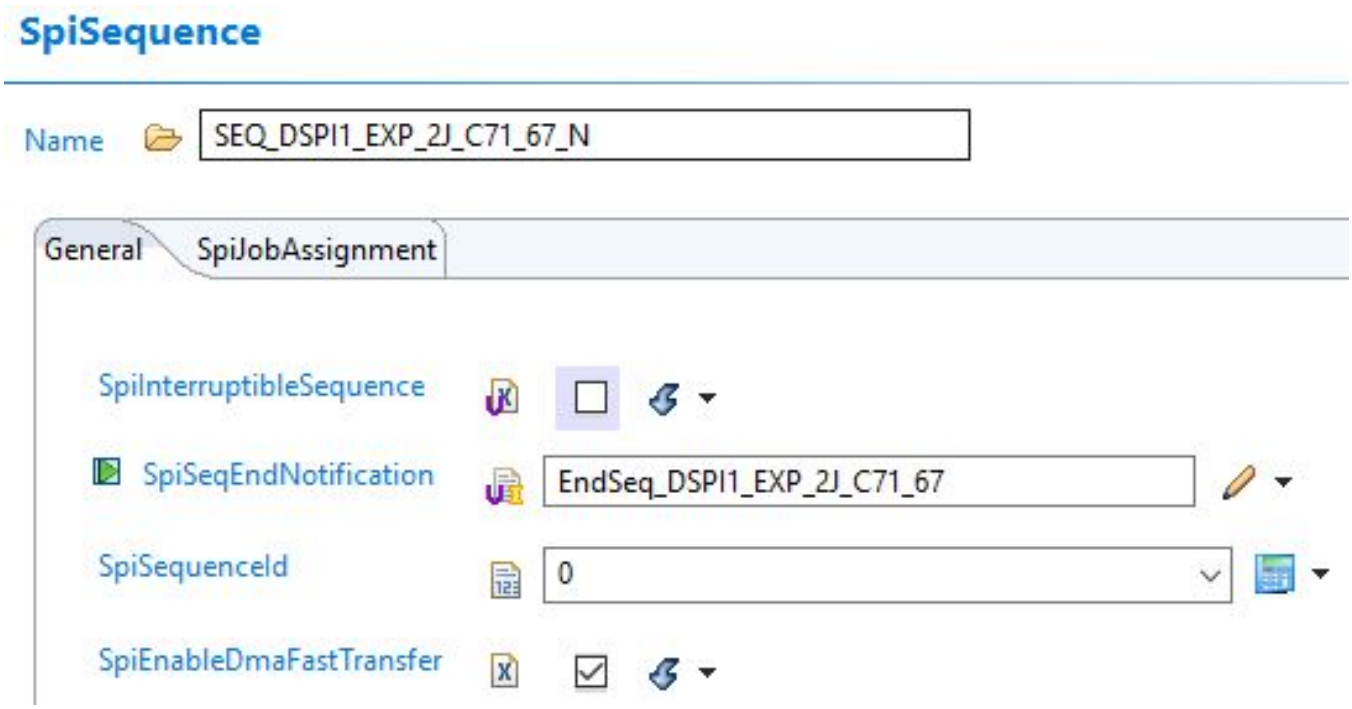

Figure 3.10 Enable Dma Fast transfer for Sequence

- Enable Dma mode and set maximum number of Dma Fast transfers session. Make sure that the value of SpiMaxDmaFastTransfer is higher or equal total number of Channels in a Sequence.

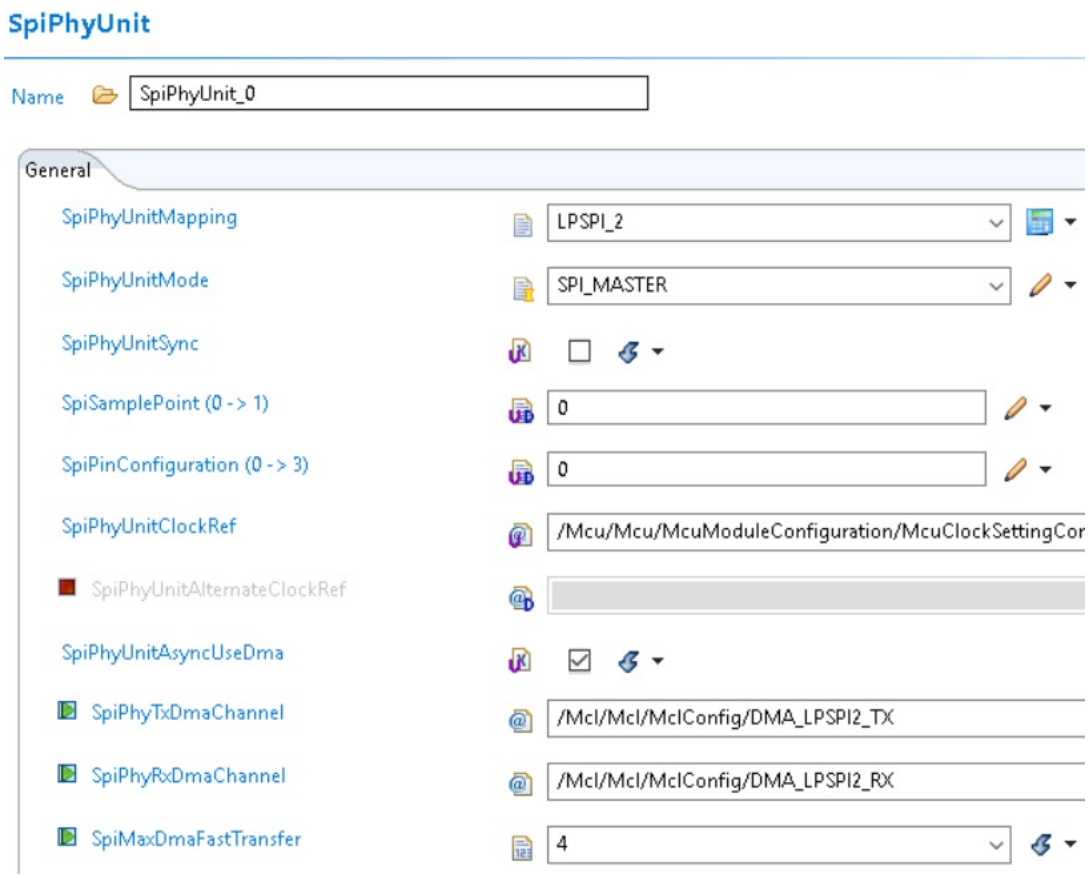

Figure 3.11 Enable Dma mode and set maximum number of Dma Fast transfers session

- Configure for SpiPhyTxDmaChannel in MCL.

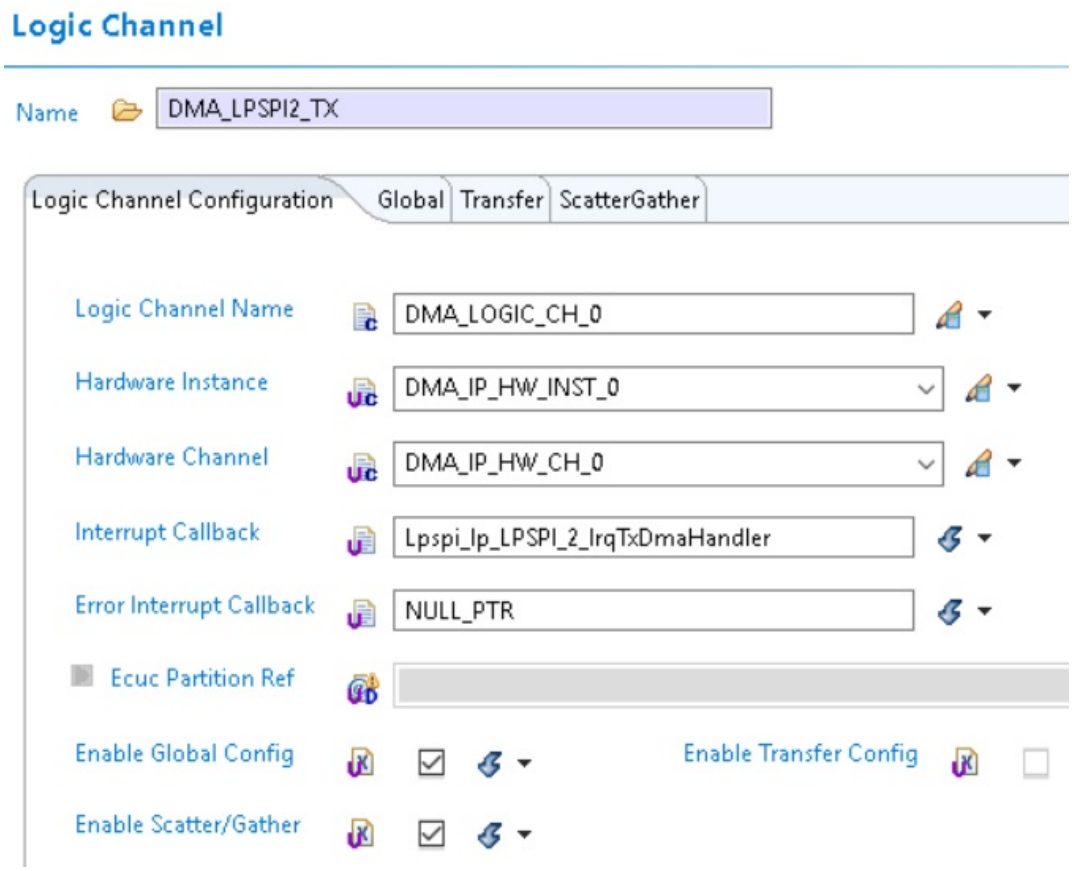

Figure 3.12 Configure for SpiPhyTxDmaChannel in MCL

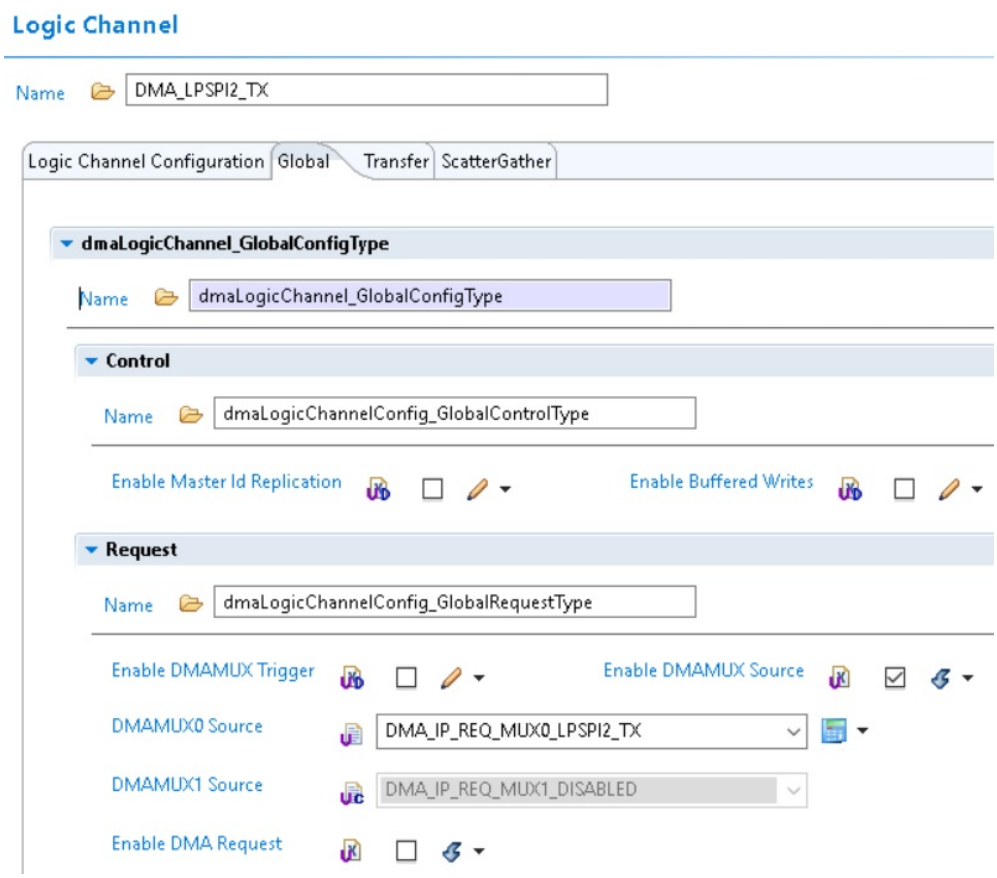

Figure 3.13 Global configuration for SpiPhyTxDmaChannel in MCL

Number of ScatterGather elements will equal to SpiMaxDmaFastTransfer ∗ 2 + total number of Jobs in Sequence.

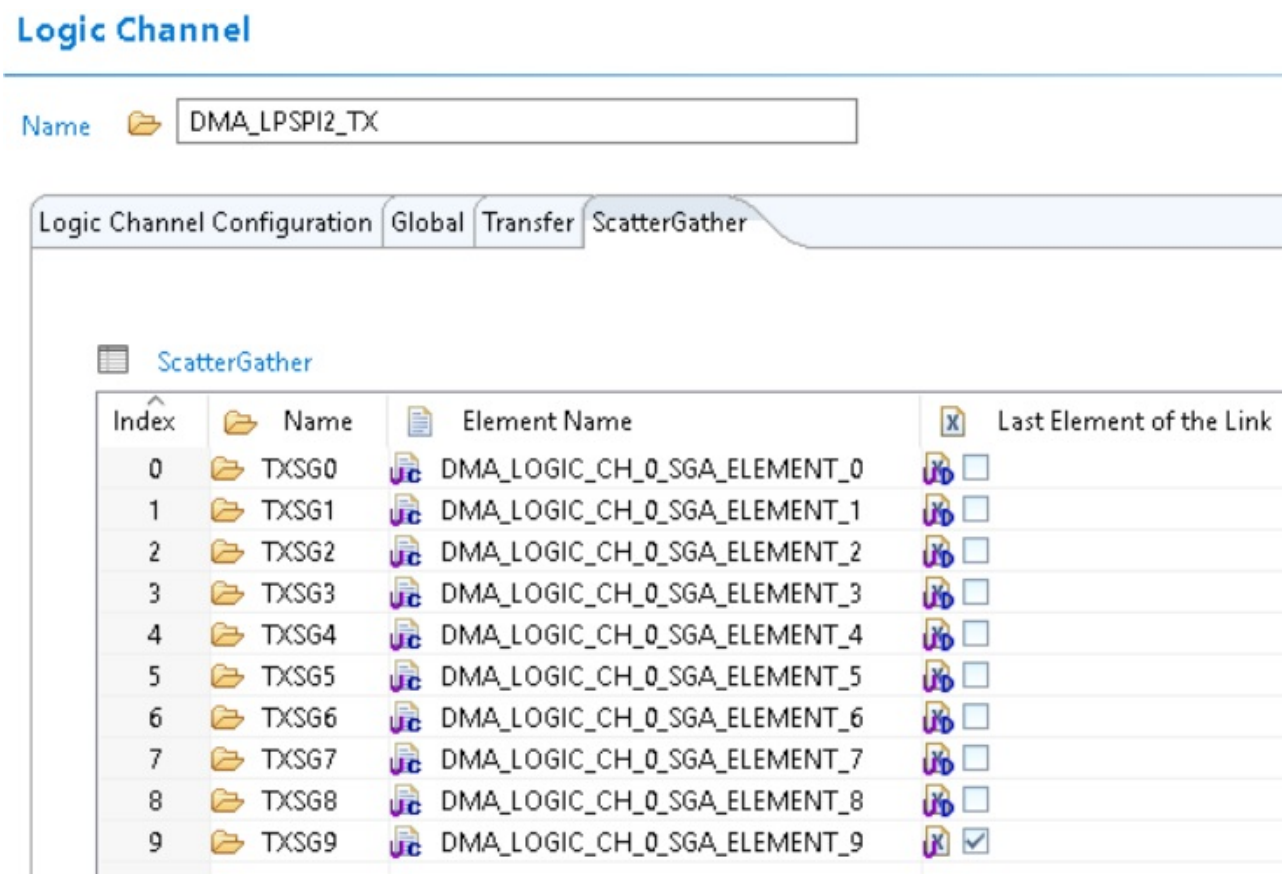

Figure 3.14 ScatterGather configuration for SpiPhyTxDmaChannel in MCL

"Element Link" node must be configured for each ScatterGather element as a continuation sequence: TXSG0->TXSG1->TXSG2->TXSG3->TXSG4->TXSG5->TXSG6->TXSG7->TXSG8->TXSG9.

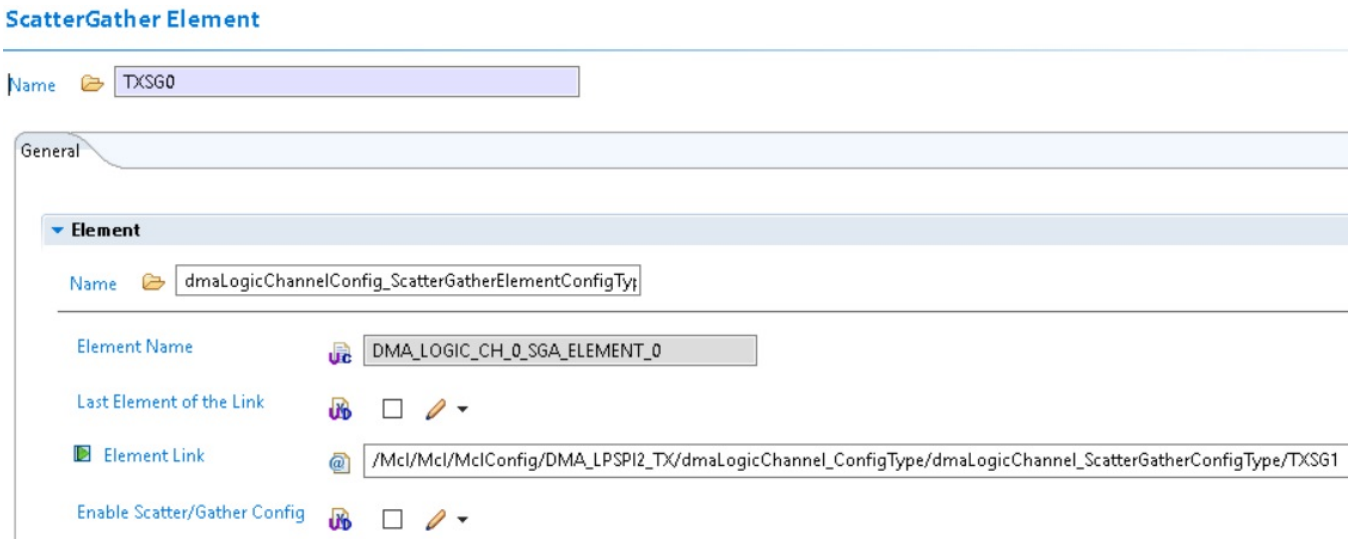

Figure 3.15 Link each ScatterGather element as a continuation sequence in MCL

- Configure for SpiPhyRxDmaChannel in MCL is similar to SpiPhyTxDmaChannel. But number of Scatter↩ Gather elements will equal to SpiMaxDmaFastTransfer.

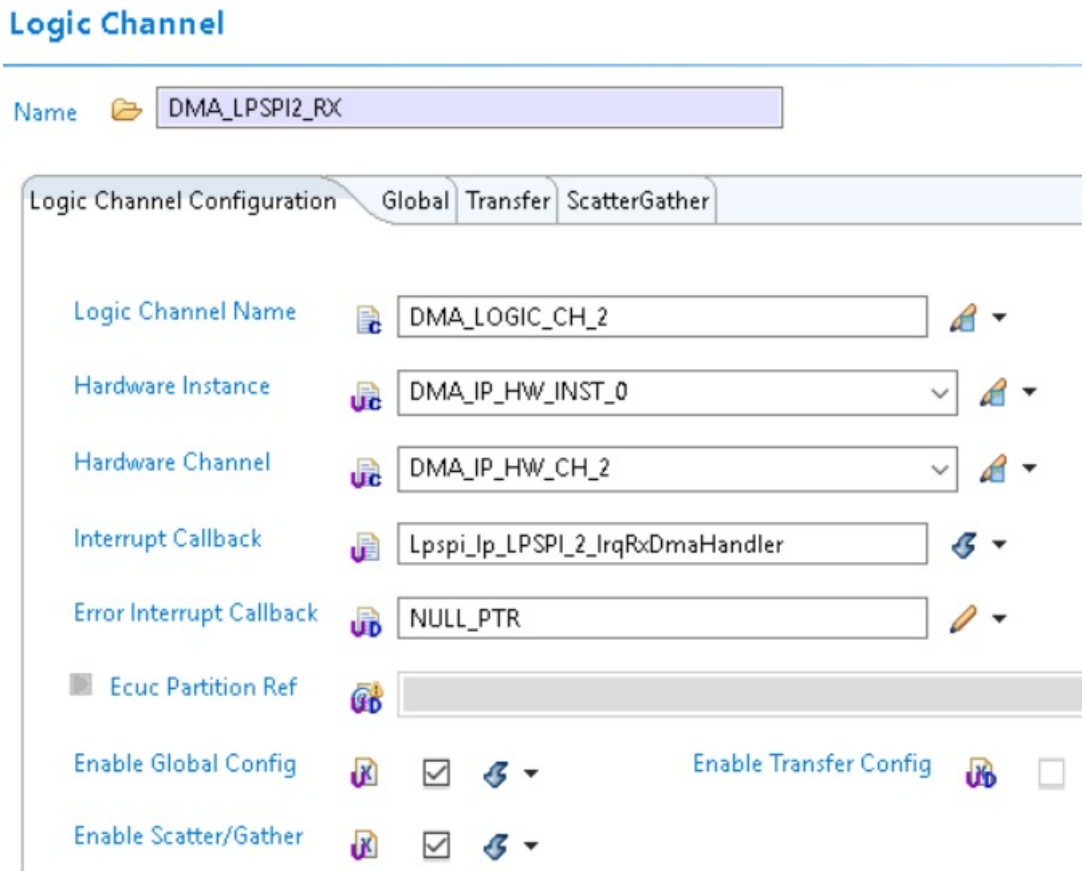

Figure 3.16 Configure for SpiPhyRxDmaChannel in MCL

- Notice that "Level Priority" value of SpiPhyRxDmaChannel is higher than SpiPhyTxDmaChannel.

3.6.5 How to configure SPI over FLEXIO

First of all, SPI over FLEXIO must use MCL module to enable/disable Flexio module and configure Flexio Channels (Select PIN, Shifter, Timer registers) and ISR handler. SPI can not work by itself. Bellowing is guiding to configure SPI over FLEXIO for both EB tresos and CT configuration tool

Note

Mcl_Init() functions must be called before using SPI over FLEXIO.

3.6.5.1 EB Tresos

- Enable flexio mode first by set SpiFlexioEnable: (bellow picture)

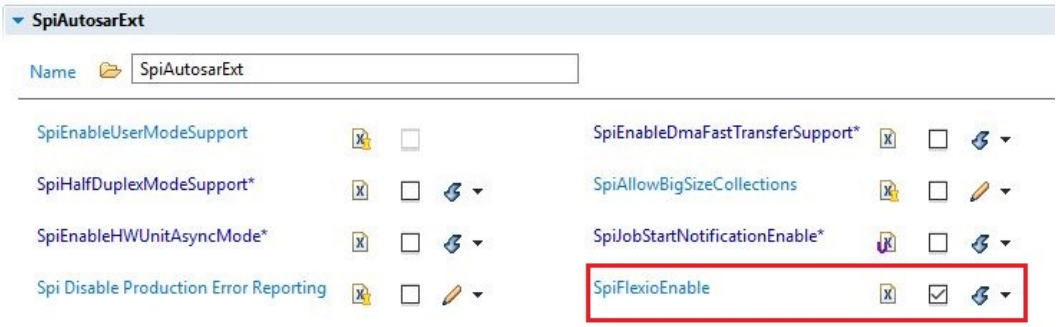

Figure 3.17 Enable SPI over FLEXIO

- Enable Flexio support on MCL site: (bellow picture)

Figure 3.18 Enable Mcl FLEXIO support

- In MCL, setup a set for SPI FLEXO Channels like: (bellow picture)

- Flexio Channel

  is no. of Shifter and Timer registers will be selected for each FLEXIO channel. For example: CHANNEL_0 selected means Shifter_0 and Timer_0 will be selected and locked, other channels cannot select those registers.

- Flexio Pin

  is no. of PIN will be selected for each FLEXIO channel. For example: PIN_0 selected means FXIO_D0 will be selected and locked, other channels cannot select this PIN.

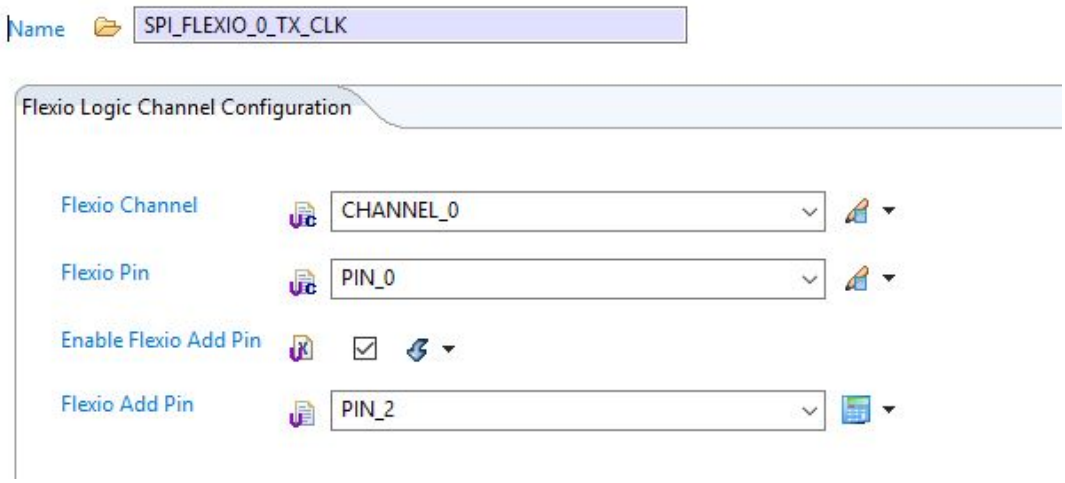

Figure 3.19 A set of FLEXIO SPI channels from MCL

- From SPI, each FLEXIO_SPI_x is selected in SpiPhyUnit then those set must be enabled and selected correctly value (refer from MCL which was configured above) (bellow picture)
  - SpiFlexioTxAndClkChannelsConfig: FLEXIO Logical Channel for TX(MOSI) and CLK(Clock). This selects 1 PIN to setup as TX(MOSI) channel and 1 PIN to set up as CLK channel.If you select CHAN↩ NEL_X, PIN_Y(FlexioMclPinId) and PIN_Z(FlexioMclAddPinId) that means:
    - ShifterX is selected for TX(MOSI) channel
    - TimerX is selected for CLK channel.
    - PIN_Y is selected for TX(MOSI) channel
    - PIN_Z is selected for CLK channel
  - SpiFlexioRxAndCsChannelsConfig: FLEXIO Logical Channel for RX(MISO) and CS(Chip select). This selects 1 PIN to setup as MISO channel and 1 PIN to set up as CS channel. If you select CHANNEL_X, PIN_Y(FlexioMclPinId) and PIN_Z(FlexioMclAddPinId) that mean:
    - ShifterX is selected for RX(MISO) channel
    - TimerX is selected for CS channel.
    - PIN_Y is selected for RX(MISO) channel
    - PIN_Z is selected for CS channel

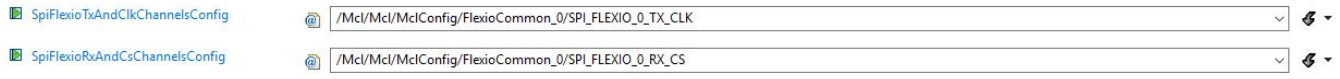

Figure 3.20 A set of FLEXIO SPI channels from SPI

3.6.5.2 Design Studio Tool:

It is the same with EB tresos

- Enable flexio mode first by set SpiFlexioEnable: (bellow picture)

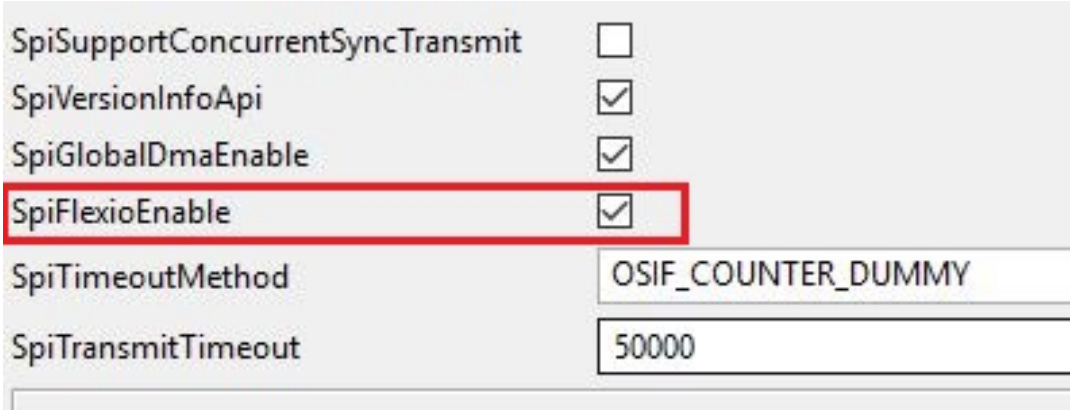

Figure 3.21 Enable SPI over FLEXIO

- Enable Flexio support on MCL site: (bellow picture)

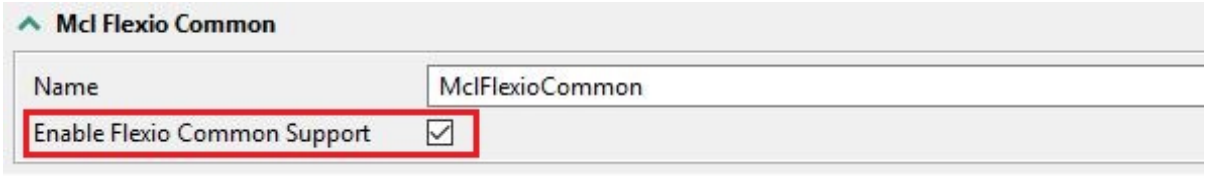

Figure 3.22 Enable Mcl FLEXIO support

- In MCL, setup a set for SPI FLEXO Channels like: (bellow picture)

- Flexio Channel

  is no. of Shifter and Timer registers will be selected for each FLEXIO channel. For example: CHANNEL_0 selected means Shifter_0 and Timer_0 will be selected and locked, other channels cannot select those registers.

- Flexio Pin

  is no. of PIN will be selected for each FLEXIO channel. For example: PIN_0 selected means FXIO_D0 will be selected and locked, other channels cannot select this PIN.

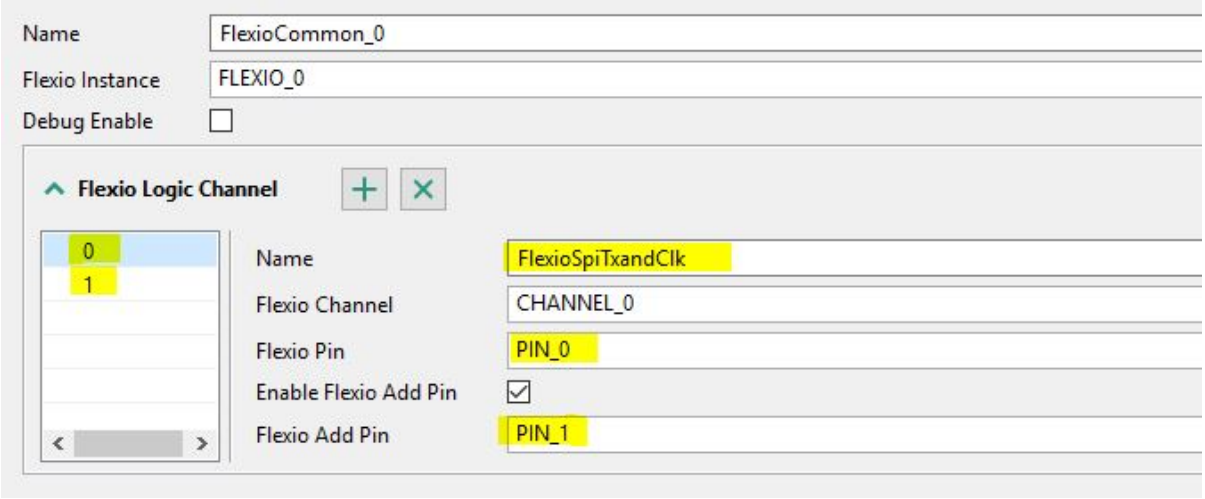

Figure 3.23 A set of FLEXIO SPI channels from MCL

- From SPI, each FLEXIO_SPI_x is selected in SpiPhyUnit then those set must be enabled and selected correctly value (refer from MCL which was configured above) (bellow picture)
  - SpiFlexioTxAndClkChannelsConfig: FLEXIO Logical Channel for TX(MOSI) and CLK(Clock). This selects a PIN to setup as TX(MOSI) channel and another for CLK channel. If you select CHANNEL_X, PIN_Y(FlexioMclPinId) and PIN_Z(FlexioMclAddPinId) that means:
    - ShifterX is selected for TX(MOSI) channel
    - TimerX is selected for CLK channel.
    - PIN_Y is selected for TX(MOSI) channel
    - PIN_Z is selected for CLK channel
  - SpiFlexioRxAndCsChannelsConfig: FLEXIO Logical Channel for RX(MISO) and CS(Chip select). This selects 1 PIN to setup as MISO channel and 1 PIN to set up as CS channel. If you select CHANNEL_X, PIN_Y(FlexioMclPinId) and PIN_Z(FlexioMclAddPinId) that mean:
    - ShifterX is selected for RX(MISO) channel
    - TimerX is selected for CS channel.
    - PIN_Y is selected for RX(MISO) channel
    - PIN_Z is selected for CS channel

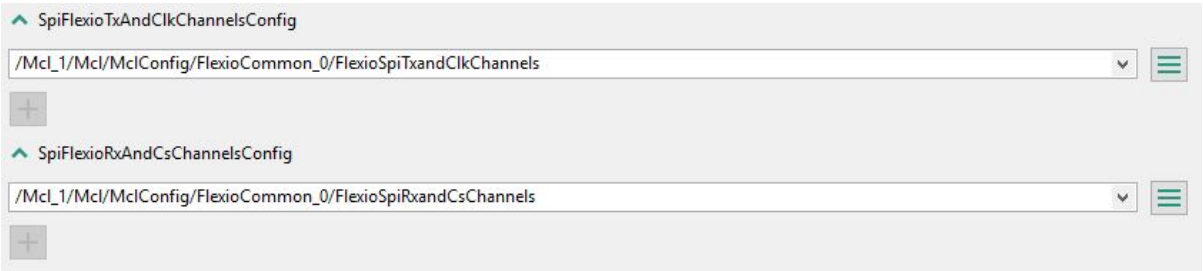

Figure 3.24 A set of FLEXIO SPI channels from SPI

3.6.5.3 Register ISR for INTERRUPT mode

- ISR name: **MCL_FLEXIO_ISR**

- Register FLEXIO ISR like document. For example:

  sys_disableIsrSource(69);

  sys_enableIsrSource(69, 7);

  sys_registerIsrHandler(69, MCL_FLEXIO_ISR);

  where:
  139: NVIC Interrupt ID of FLEXIO module
  7: The priority

3.6.6 How to configure to use half duplex mode

To configure Spi in half duplex mode that has some different to normal configuration of full duplex mode. The different will proposed below:

- First of all, the SpiHalfDuplexModeSupport checkbox in SpiAutosarExt must be enabled to half-duplex mode can be configured in configurational tools:

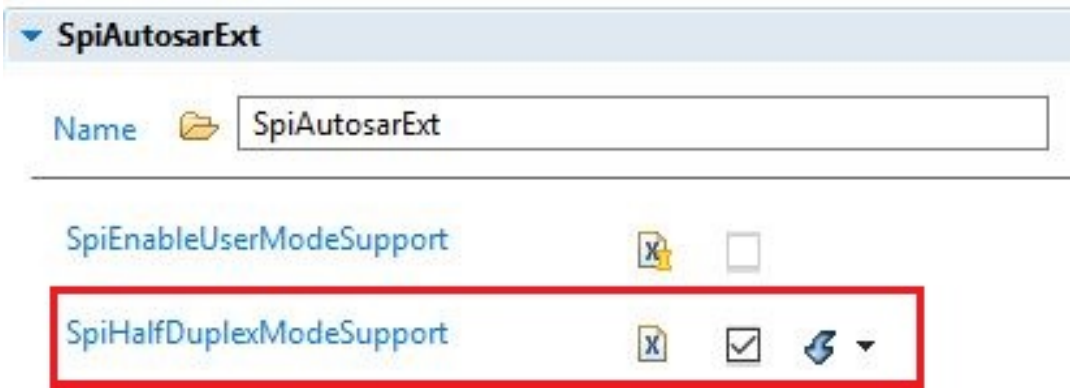

Figure 3.25 Enable Half-duplex mode support on EBTresos

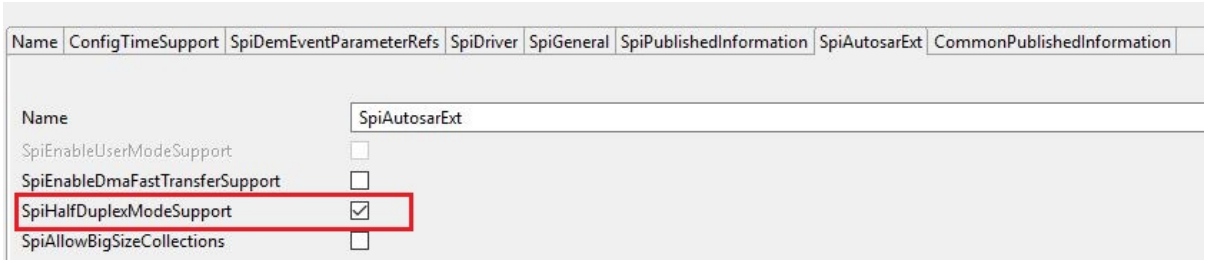

Figure 3.26 Enable Half-duplex mode support on S32 Design Studio

- Second, an external device has to be created with half-duplex transfer support by enabling SpiDeviceHalf↩ DuplexSupport checkbox. After that, selecting number of bit transmited and received on each clock edge. There are 3 types of transfer are supported: 1-bit, 2-bits or 4 bits.

Note

- Half-duplex mode only supports in external devices which are using Lpspi.
- With 1-bit transfer, the CIN or COUT pin could be chosen by SpiHalfDuplexPinSelect on External↩ Device configuration.
- With 2-bits transfer, the CIN and COUT pins will be used.
- And with 4-bits transfer, the CIN, COUT, PCS2 and PCS3 will be used.

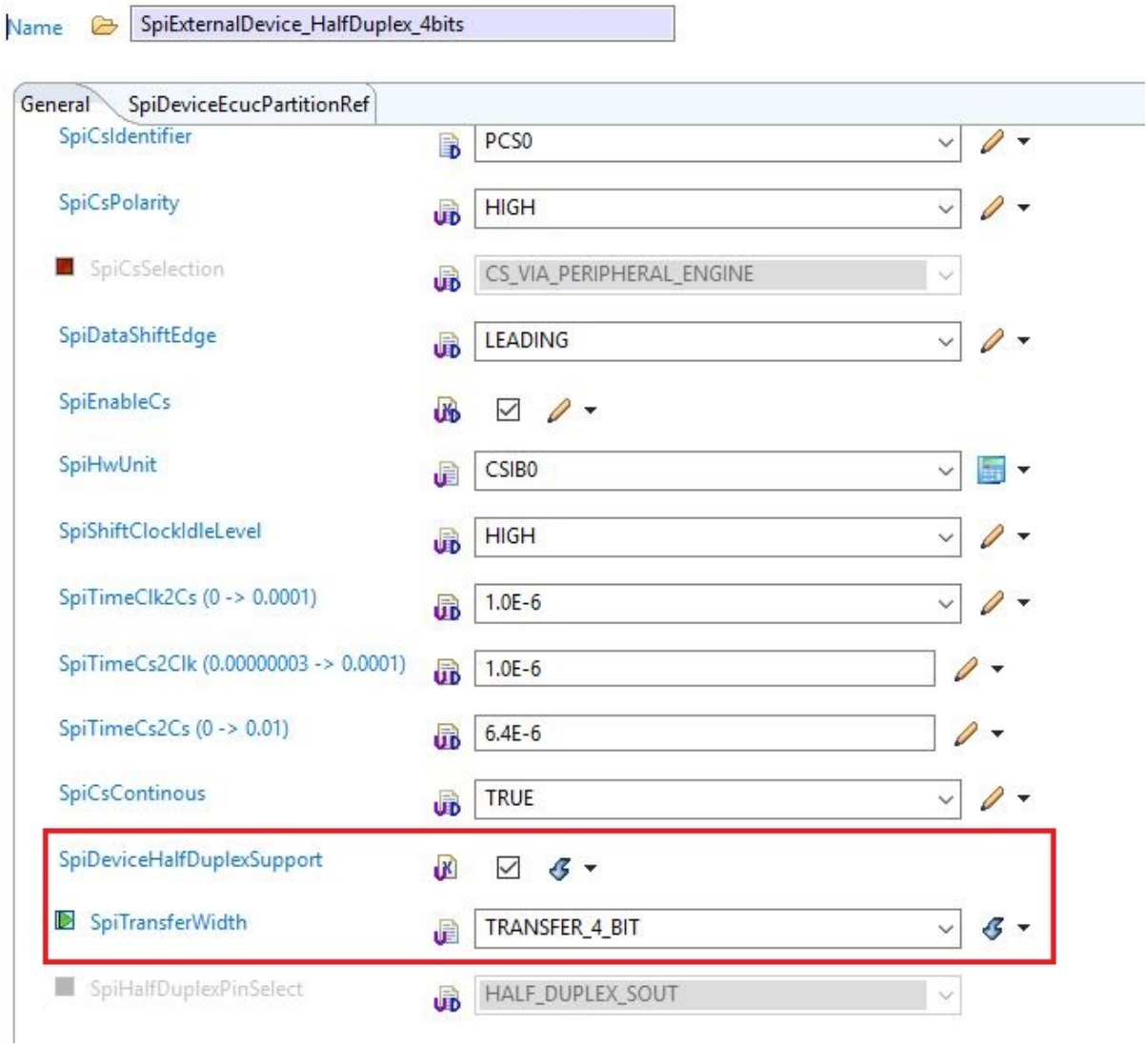

Figure 3.27 Configure Half-duplex mode for external device on EBTresos

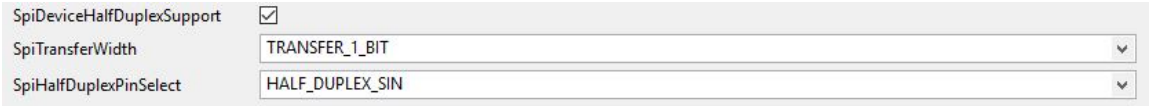

Figure 3.28 Configure Half-duplex mode for external device on S32 Design Studio

- Finish, a channel should be selected as transmission or reception in half-duplex mode by configuring the Spi↩ ChannelHalfDuplexDirection.

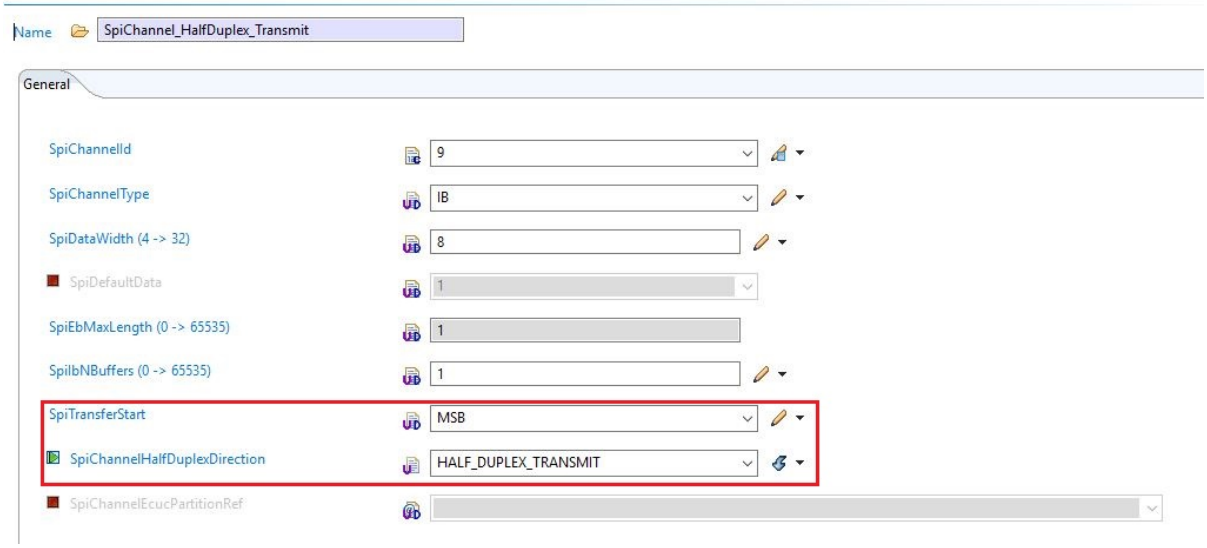

Figure 3.29 Configure Half-duplex mode for external device on EBTresos

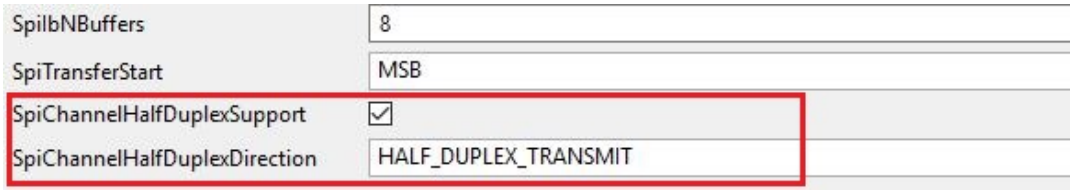

Figure 3.30 Configure Half-duplex mode for external device on S32 Design Studio

3.6.7 How to use Half Duplex mode

Below is presenting on EB tresos, this is same on CT.

- Enable Spi half duplex mode support:

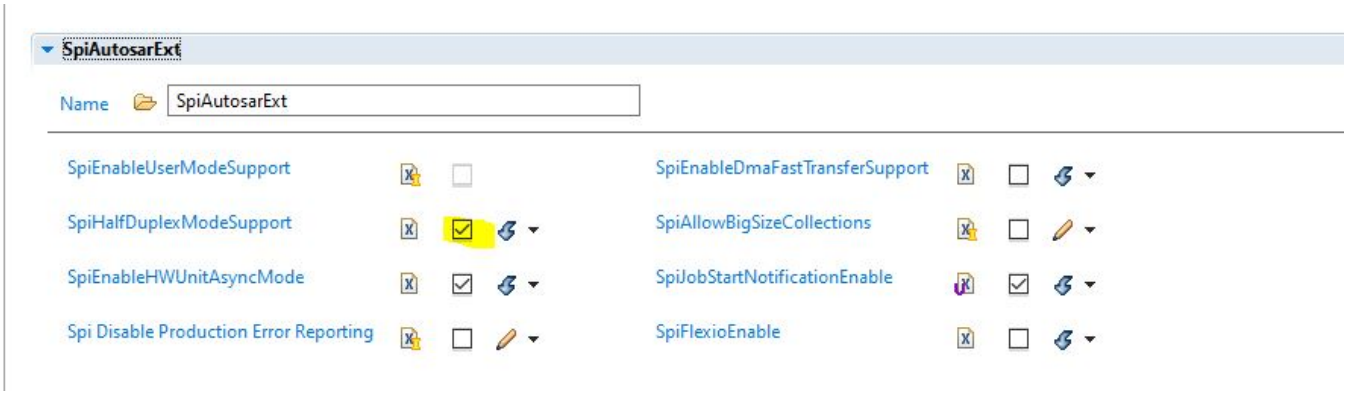

Figure 3.31 Enable Spi half duplex mode support

- Select direction for each channel:
  - HALF_DUPLEX_TRANSMIT: Transmit only mode
  - HALF_DUPLEX_RECEIVE: Receive only mode

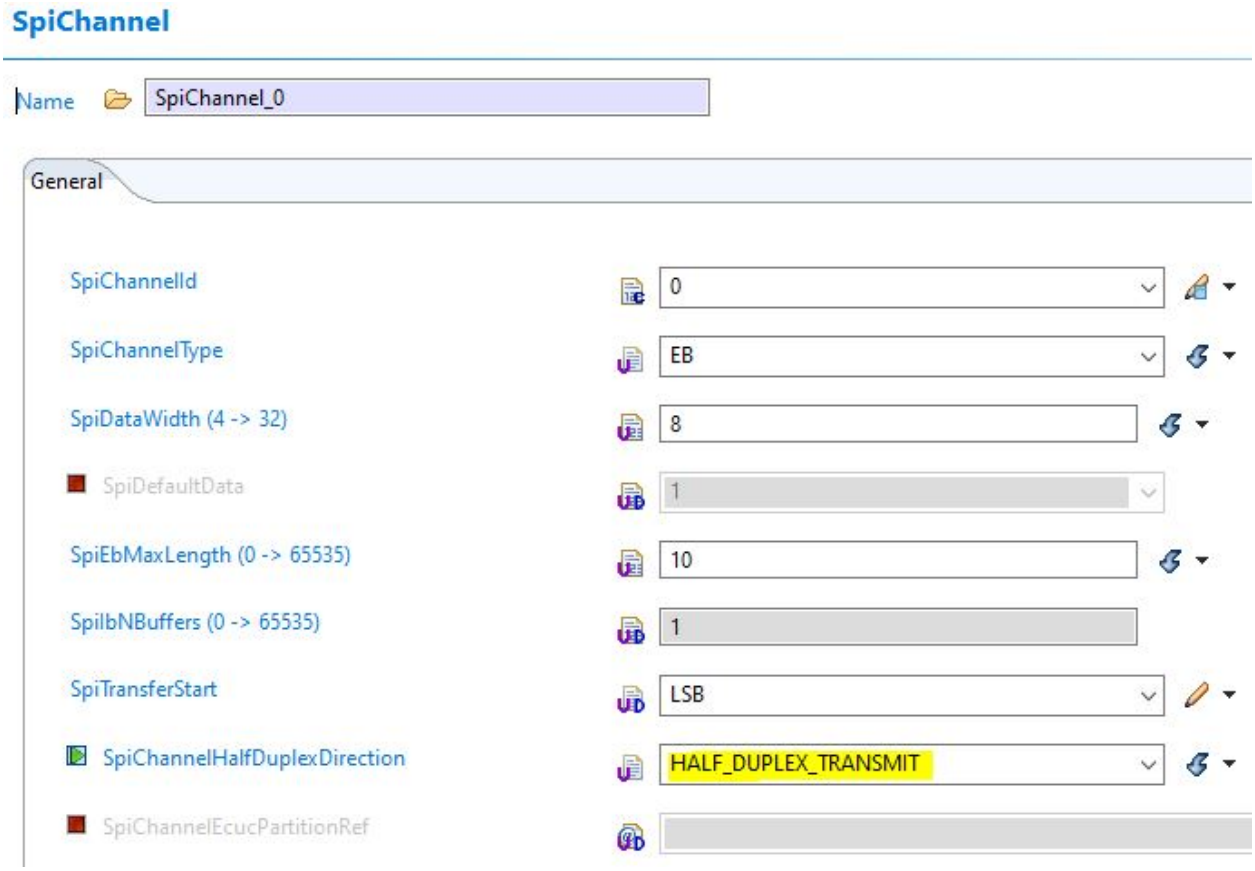

Figure 3.32 Direction of channels

- Configure Externaldevice:
  - SpiTransferWidth: Number of bits will be transfered on per cycle clock. It supports 1/2/4 bits on Master mode, Slave mode only supports 1 bit mode.
  - SpiHalfDuplexPinSelect: Pin will be selected for the transfer (it is HALF_DUPLEX_SOUT means MOSI will be selected).

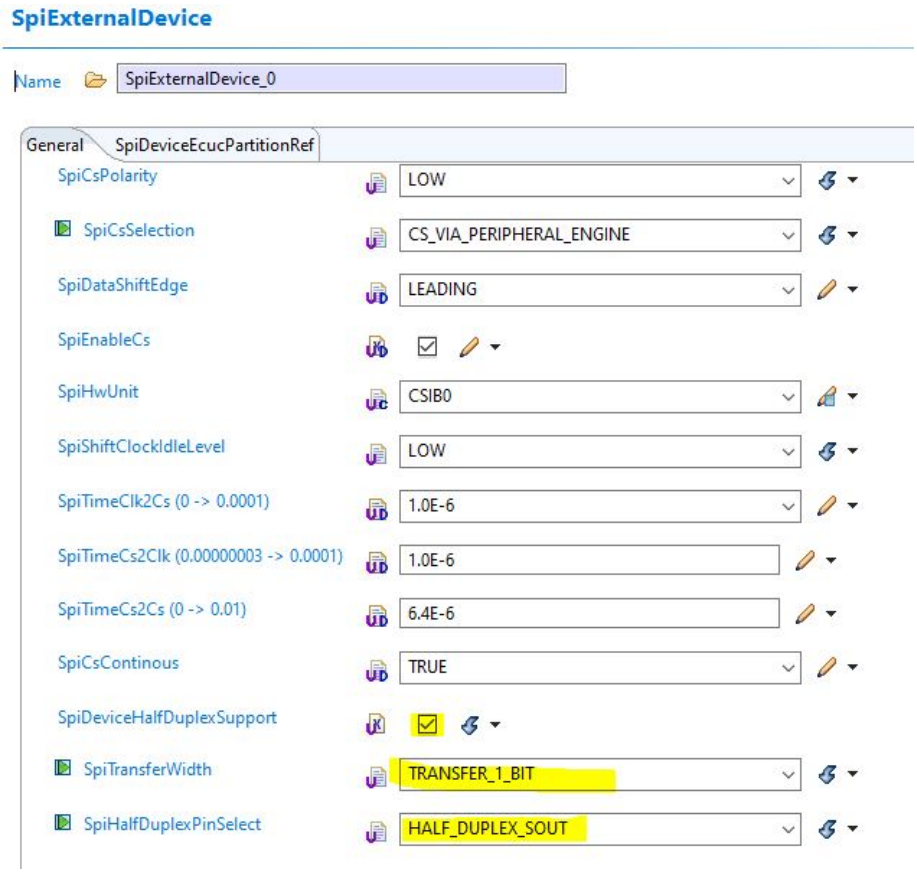

Figure 3.33 Direction of channels

3.7 Runtime errors

The driver generates the following DEM errors at runtime.

Function	Error Code	Condition triggering the error
Spi_SyncTransmit	Spi_E_Hardware_ErrorCfg	The SPI driver cannot transmit complete or receive complete one frame in the allocated time defined by "SPI_TIMEOUT_COUNTER" parameter in configuration. Baud rate of HW might be low speed or timeout value to short. Timeout occurred.
Spi_SyncTransmit	SPI_E_SEQ_IN_PROCESS	Synchronous transmission service called at wrong time.
Spi_AsyncTransmit	SPI_E_SEQ_PENDING	Services called in a wrong sequence.

3.8 Symbolic Names Disclaimer

All containers having symbolicNameValue set to TRUE in the AUTOSAR schema will generate defines like:

#define <Mip>Conf_<Container_ShortName>_<Container_ID>

For this reason it is forbidden to duplicate the names of such containers across the RTD configurations 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 driver. All the parameters are described below.

- Module Spi
  - Container SpiDemEventParameterRefs
    - Reference SPI_E_HARDWARE_ERROR
  - Container SpiDriver
    - Parameter SpiMaxChannel
    - Parameter SpiMaxJob
    - Parameter SpiMaxSequence
    - Container SpiChannel
      - Parameter SpiChannelId
      - Parameter SpiChannelType
      - Parameter SpiDataWidth
      - Parameter SpiDefaultData
      - Parameter SpiEbMaxLength
      - Parameter SpiIbNBuffers
      - Parameter SpiTransferStart
      - Parameter SpiChannelHalfDuplexDirection
      - Reference SpiChannelEcucPartitionRef
    - Container SpiExternalDevice
      - Parameter SpiBaudrate
      - Parameter SpiCsIdentifier
      - Parameter SpiCsPolarity
      - Parameter SpiCsSelection
      - Parameter SpiDataShiftEdge
      - Parameter SpiEnableCs
      - Parameter SpiHwUnit
      - Parameter SpiShiftClockIdleLevel
      - Parameter SpiTimeClk2Cs
      - Parameter SpiTimeCs2Clk
      - Parameter SpiTimeCs2Cs
      - Parameter SpiCsContinous

- Parameter SpiDeviceHalfDuplexSupport
- Parameter SpiTransferWidth
- Parameter SpiHalfDuplexPinSelect
- Reference SpiDeviceEcucPartitionRef

* Container SpiJob
  - Parameter SpiJobEndNotification
  - Parameter SpiJobStartNotification
  - Parameter SpiJobId
  - Parameter SpiJobPriority
  - Reference SpiDeviceAssignment
  - Container SpiChannelList
  - Parameter SpiChannelIndex
  - Reference SpiChannelAssignment
* Container SpiSequence
  - Parameter SpiInterruptibleSequence
  - Parameter SpiSeqEndNotification
  - Parameter SpiSequenceId
  - Parameter SpiEnableDmaFastTransfer
  - Reference SpiJobAssignment

– Container SpiGeneral

* Parameter SpiMulticoreSupport
* Parameter SpiCancelApi
* Parameter SpiChannelBuffersAllowed
* Parameter SpiDevErrorDetect
* Parameter SpiHwStatusApi
* Parameter SpiInterruptibleSeqAllowed
* Parameter SpiLevelDelivered
* Parameter SpiMainFunctionPeriod
* Parameter SpiSupportConcurrentSyncTransmit
* Parameter SpiVersionInfoApi
* Parameter SpiGlobalDmaEnable
* Parameter SpiTimeoutMethod
* Parameter SpiTransmitTimeout
* Reference SpiEcucPartitionRef
* Reference SpiKernelEcucPartitionRef
* Container SpiPhyUnit
  - Parameter SpiPhyUnitMapping
  - Parameter SpiPhyUnitMode
  - Parameter SpiPhyUnitSync
  - Parameter SpiSamplePoint
  - Parameter SpiPinConfiguration
  - Parameter SpiPhyUnitAsyncUseDma
  - Parameter SpiMaxDmaFastTransfer
  - Reference SpiPhyUnitClockRef

        - Reference SpiPhyUnitAlternateClockRef
        - Reference SpiPhyTxDmaChannel
        - Reference SpiPhyRxDmaChannel
        - Reference SpiFlexioTxAndClkChannelsConfig
        - Reference SpiFlexioRxAndCsChannelsConfig
    - Container SpiPublishedInformation
        - Parameter SpiMaxHwUnit
    - Container CommonPublishedInformation
        - Parameter ArReleaseMajorVersion
        - Parameter ArReleaseMinorVersion
        - Parameter ArReleaseRevisionVersion
        - Parameter ModuleId
        - Parameter SwMajorVersion
        - Parameter SwMinorVersion
        - Parameter SwPatchVersion
        - Parameter VendorApiInfix
        - Parameter VendorId
    - Container SpiAutosarExt
        - Parameter SpiEnableUserModeSupport
        - Parameter SpiEnableDmaFastTransferSupport
        - Parameter SpiHalfDuplexModeSupport
        - Parameter SpiAllowBigSizeCollections
        - Parameter SpiEnableHWUnitAsyncMode
        - Parameter SpiJobStartNotificationEnable
        - Parameter SpiDisableDemReportErrorStatus
        - Parameter SpiFlexioEnable

4.1 Module Spi

Configuration of the Spi (Serial Peripheral Interface) module.

Included containers:

- SpiDemEventParameterRefs
- SpiDriver
- SpiGeneral
- SpiPublishedInformation
- CommonPublishedInformation
- SpiAutosarExt

Property	Value
type	ECUC-MODULE-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantSupport	true
supportedConfigVariants	VARIANT-PRE-COMPILE, VARIANT-LINK-TIME, VARIANT-POST-BUILD

4.2 Container SpiDemEventParameterRefs

Container for the references to DemEventParameter elements which shall be invoked using the Dem_SetEventStatus API in case the corresponding error occurs.

The EventId is taken from the referenced DemEventParameter's DemEventId value. The standardized errors are provided in the container and can be extended by vendor specific error references.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.3 Reference SPI_E_HARDWARE_ERROR

Reference to configured DEM event to report "Hardware failure". If the reference is not configured the error shall not be reported.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE

Property	Value
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Dem/DemConfigSet/DemEventParameter

4.4 Container SpiDriver

This container contains the configuration parameters and sub containers of the AUTOSAR Spi module.

Included subcontainers:

- SpiChannel
- SpiExternalDevice
- SpiJob
- SpiSequence

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.5 Parameter SpiMaxChannel

This parameter contains the number of Channels configured. It will be gathered by tools during the configuration stage.

NoteThis parameter is not used, instead max channel value is derived from number of channels configured.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	255
min	0

4.6 Parameter SpiMaxJob

This parameter contains the number of Jobs configured. It will be gathered by tools during the configuration stage.

NoteThis parameter is not used, instead max jobs value is derived from number of jobs configured.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	65535
min	0

4.7 Parameter SpiMaxSequence

This parameter contains the number of Sequences configured. It will be gathered by tools during the configuration stage.

NoteThis parameter is not used, instead max Sequences value is derived from number of sequences configured.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	255
min	0

4.8 Container SpiChannel

All data needed to configure one SPI-channel.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.9 Parameter SpiChannelId

SPI Channel ID, used as parameter in SPI API functions.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	true
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	0
max	255
min	0

4.10 Parameter SpiChannelType

Buffer usage with EB/IB channel.

NoteThis parameter is dependant on SpiChannelBuffersAllowed parameter.

When SpiChannelBuffersAllowed = 0; SpiChannelType should be IB

When SpiChannelBuffersAllowed = 1; SpiChannelType should be EB

When SpiChannelBuffersAllowed = 2; SpiChannelType can be IB or EB

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	IB
literals	['EB', 'IB']

4.11 Parameter SpiDataWidth

This parameter is the width of a transmitted data unit.

NoteThe hardware supports data width from 4 to 32 bit.The unit is in bits.

When SpiChannelBuffersAllowed = 0; SpiChannelType should be IB

When SpiChannelBuffersAllowed = 1; SpiChannelType should be EB

When SpiChannelBuffersAllowed = 2; SpiChannelType can be IB or EB

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	8
max	32
min	4

4.12 Parameter SpiDefaultData

The default data to be transmitted when (for internal buffer or external buffer)

the pointer passed to Spi_WriteIB (for internal buffer) or to Spi_SetupEB (for external buffer) is NULL.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1
max	4294967295
min	0

4.13 Parameter SpiEbMaxLength

This parameter contains the maximum size (number of data elements) of data buffers in case of EB Channels and only.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1
max	65535
min	0

4.14 Parameter SpiIbNBuffers

This parameter contains the maximum number of data buffers in case of IB Channels and only.

In case of channel's Spi_DataWidth ranges from 9 to 16, this parameter reffers to

the number of bytes allocated to the buffers and MUST be even. Or divisible by 4 if the range from 17 to 32.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false

Property	Value
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1
max	65535
min	0

4.15 Parameter SpiTransferStart

This parameter defines the first starting bit for transmission.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	LSB
literals	['LSB', 'MSB']

4.16 Parameter SpiChannelHalfDuplexDirection

This parameter select direction of transfer in half duplex mode

HALF_DUPLEX_TRANSMIT: Transmit only.

HALF_DUPLEX_RECEIVE: Receive only.

NoteThis parameter only is used in Half Duplex mode.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	HALF_DUPLEX_TRANSMIT
literals	['HALF_DUPLEX_TRANSMIT', 'HALF_DUPLEX_RECEIVE']

4.17 Reference SpiChannelEcucPartitionRef

Maps an SPI Channel to zero or ECUC partition to limit the access to this Channel. The ECUC partition referenced is a

subset of the ECUC partitions where the SPI driver is mapped to.

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/EcuC/EcucPartitionCollection/EcucPartition

4.18 Container SpiExternalDevice

The communication settings of an external device. Closely linked to SpiJob.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.19 Parameter SpiBaudrate

This parameter is the communication baudrate - This parameter allows using a range of values, from the point of view of configuration tools, from Hz up to MHz.

This field is used only in MASTER mode.

This field is used only for LPSPI

The baudrate must be in allowed range

LPSPI in normal Master mode:

- RUN MODE: [1 to 10Mhz]

- HSRUN MODE: [1 to 14Mhz] only support for K14x

- VLPR MODE: [1 to 2Mhz] for K1xx and [1 to 500Khz] for K14xW

FLEXIO in Master:

Note: Due to synchronization delays, the setup time for the serial input data is 1.5 FlexIO clock cycles, so the maximum baud rate is divide by 4 of the FlexIO clock frequency.

Theoretical baud rate =

FLEXIO_CLK (BUS_CLK) ? (16 ? 2 ? (TIMCMPn[CMP] + 1))

FLEXIO_CLK (BUS_CLK) ? (2 ? (TIMCMPn[CMP] + 1))

- RUN MODE: [1 to (BUS_CLK/4) ? (2 ? (TIMCMPn[CMP] + 1) = 6Mhz]

- HSRUN MODE: [1 to (BUS_CLK/4) ? (2 ? (TIMCMPn[CMP] + 1) = 7Mhz] only support for K14x

- VLPR MODE: [1 to (BUS_CLK/4) ? (2 ? (TIMCMPn[CMP] + 1) = 0.5Mhz] for K11x, K14x and [1 to (BUS_CLK/4) ? (2 ? (TIMCMPn[CMP] + 1) = 62.5KhzMhz] for K14xW

NoteThe precision of this value depends SPI clock source configuration. If the driver cannot generate correct of the value, approximate value will be used.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	100000.0
max	1.4E7
min	1.0

4.20 Parameter SpiCsIdentifier

This parameter is the symbolic name to identify the Chip Select (CS) allocated to this Job.

The chip selects are specific per HwUnit. Please check in Reference Manual for information on available chip selects.

If FLEXIO channel used, Chip Select will be configured by SpiPhyUnit\SpiFlexioCsPinSelect and this parameter will be not used.

If SpiEnableCs is not set, value of this node will not be used by driver code. It should set to default value (PCS0)

Property	Value
type	ECUC-STRING-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	true
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	PCS0

4.21 Parameter SpiCsPolarity

This parameter defines the active polarity of Chip Select.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	HIGH
literals	['HIGH', 'LOW']

4.22 Parameter SpiCsSelection

When the Chip select handling is enabled (see SpiEnableCs), then this

parameter specifies if the chip select is handled automatically by

Pe-ripheral HW engine or via general purpose IO by Spi driver.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	CS_VIA_PERIPHERAL_ENGINE
literals	['CS_VIA_PERIPHERAL_ENGINE', 'CS_VIA_GPIO']

4.23 Parameter SpiDataShiftEdge

This parameter defines the SPI data shift edge.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	LEADING
literals	['LEADING', 'TRAILING']

4.24 Parameter SpiEnableCs

This parameter enables or not the Chip Select handling functions.

This parameter is closely linked to Job.If This parameter is True,then chip select is asserted and

if False No chip select is asserted.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.25 Parameter SpiHwUnit

This parameter is the symbolic name to identify the HW SPI Hardware microcontroller peripheral allocated to this Job.

CSIBn references the n-th logical unit configured in SpiPhyUnit container. For example: CSIB0 references the first logical unit

(not the first SPI_0 HW unit).

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	CSIB0
literals	['CSIB0', 'CSIB1', 'CSIB2', 'CSIB3', 'CSIB4']

4.26 Parameter SpiShiftClockIdleLevel

This parameter defines the SPI shift clock idle level.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	HIGH
literals	['HIGH', 'LOW']

4.27 Parameter SpiTimeClk2Cs

Timing between clock and chip select in seconds (tASC) - This parameter allows to use a range of values

from 0 up to 0.0001 Sec. The real configuration-value used in software BSW-SPI is calculated out of this by the generator-tools.

If use continuous transfer(PCS signals remain asserted between transfers), tASC and tCSC will insert between transfers.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1.0E-6
max	1.0E-4
min	0.0

4.28 Parameter SpiTimeCs2Clk

Timing between chip select and clock in seconds (tCSC) - This parameter allows to use a range of values from 0.00000003 up to 0.0001 Sec.

If use continuous transfer(PCS signals remain asserted between transfers), tASC and tCSC will insert between transfers.

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1.0E-6
max	1.0E-4
min	3.0E-8

4.29 Parameter SpiTimeCs2Cs

Timing between chip select assertions in seconds (tDT) - This parameter allows to use a range of values from 0 up to 0.01 Sec.

If use continuous transfer(PCS signals remain asserted between transfers), tDT is not inserted between the transfers.

NoteThis is an implementation parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	6.4E-6
max	0.01
min	0.0

4.30 Parameter SpiCsContinous

This field determines to keep chip select asserted between frame transfers.

NoteThis is an implementation parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	TRUE
literals	['TRUE', 'FALSE']

4.31 Parameter SpiDeviceHalfDuplexSupport

This parameter enables or not half duplex mode for this external device.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.32 Parameter SpiTransferWidth

In half duplex mode, this will select the number of bits are transfered on per cycle clock

TRANSFER_1_BIT: 1 bit will be transfered on per cycle clock

TRANSFER_2_BIT: 2 bits will be transfered on per cycle clock

TRANSFER_4_BIT: 4 bits will be transfered on per cycle clock

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	TRANSFER_1_BIT
literals	['TRANSFER_1_BIT', 'TRANSFER_2_BIT', 'TRANSFER_4_BIT']

4.33 Parameter SpiHalfDuplexPinSelect

This node will support to select which Pin (SIN or SOUT) will be used for half duplex 1 bit mode.

HALF_DUPLEX_SIN: SIN will be used to transfer

HALF_DUPLEX_SOUT: SOUT will be used to transfer

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	HALF_DUPLEX_SOUT
literals	['HALF_DUPLEX_SIN', 'HALF_DUPLEX_SOUT']

4.34 Reference SpiDeviceEcucPartitionRef

ECUC_Spi_00246. Maps an SPI external device to zero or multiple ECUC partitions to limit

the access to this external device. The ECUC partitions referenced are a

subset of the ECUC partitions where the SPI driver is mapped to.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	Infinite
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/EcuC/EcucPartitionCollection/EcucPartition

4.35 Container SpiJob

All data needed to configure one SPI-Job, amongst others the connection between the internal SPI unit and the special settings for an external device is done.

Included subcontainers:

- SpiChannelList

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.36 Parameter SpiJobEndNotification

This parameter is a reference to a notification function.

Property	Value
type	ECUC-FUNCTION-NAME-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	NULL_PTR

4.37 Parameter SpiJobStartNotification

This parameter is a reference to a notification function.

Property	Value
type	ECUC-FUNCTION-NAME-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	NULL_PTR

4.38 Parameter SpiJobId

SPI Job ID, used as parameter in SPI API functions.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	true
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	0
max	65535
min	0

4.39 Parameter SpiJobPriority

Priority set accordingly to SPI093: 0, lowest, 3, highest priority

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	3
min	0

4.40 Reference SpiDeviceAssignment

Reference to the external device used by this job.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Spi/SpiDriver/SpiExternalDevice

4.41 Container SpiChannelList

References to SPI channels and their order within the Job.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.42 Parameter SpiChannelIndex

This parameter specifies the order of Channels within the Job.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	255
min	0

4.43 Reference SpiChannelAssignment

A job references several channels.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Spi/SpiDriver/SpiChannel

4.44 Container SpiSequence

All data needed to configure one SPI-sequence.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.45 Parameter SpiInterruptibleSequence

This parameter allows or not this Sequence to be suspended by another one.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.46 Parameter SpiSeqEndNotification

This parameter is a reference to a notification function.

Property	Value
type	ECUC-FUNCTION-NAME-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	NULL_PTR

4.47 Parameter SpiSequenceId

Sequence ID of configured SPI Sequence.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	true
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	0
max	255
min	0

4.48 Parameter SpiEnableDmaFastTransfer

When this parameter is enabled, this Sequence will be transferred using DMA ScatterGather and CPU used only for processing end of Sequence.

SpiAutosarExt/SpiEnableDmaFastTransferSupport must be checked to support this feature.

Note: This feature requires:

1. The parameters SpiBaudrate, SpiHwUnit, SpiTimeClk2Cs, SpiTimeCs2Clk, SpiTimeCs2Cs in External Device linked to each Job in this Sequence must be the same.

2. The parameters SpiDataWidth and SpiTransferStart in Channel assigned to each Job in this Sequence must be the same.

3. In each Channel, the number of data buffers is NOT higher than 32767 if SpiDataWidth < 9. So, SpiIbNBuffers and SpiEbMaxLength must be assigned to suitable values.

4. Only Master mode is supported.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.49 Reference SpiJobAssignment

A sequence references several jobs, which are executed during a communication sequence.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Spi/SpiDriver/SpiJob

4.50 Container SpiGeneral

General configuration settings for SPI-Handler.

Included subcontainers:

- SpiPhyUnit

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.51 Parameter SpiMulticoreSupport

SpiMulticoreEnable

When this parameter is enabled, multi-core feature will be used in SPI driver.

That means mapping the SPI driver to multiple ECUC partitions to make the module API available in this partition.

The SPI driver will operate as an independent instance in each of the partitions.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.52 Parameter SpiCancelApi

Switches the Spi_Cancel function ON or OFF.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1

Property	Value
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.53 Parameter SpiChannelBuffersAllowed

Selects the SPI Handler/Driver Channel Buffers usage allowed and delivered.

Note

0 - Only Internal Buffers (IB) are allowed

1 - Only External buffers (EB) are allowed

2 - Both Internal (IB) and External (EB) buffers are allowed

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	0
max	2
min	0

4.54 Parameter SpiDevErrorDetect

Switches the Development Error Detection and Notification ON or OFF.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.55 Parameter SpiHwStatusApi

Switches the Spi_GetHWUnitStatus function ON or OFF.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.56 Parameter SpiInterruptibleSeqAllowed

Switches the Interruptible Sequences handling functionality ON or OFF.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false

Property	Value
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.57 Parameter SpiLevelDelivered

Selects the SPI Handler/Driver level of scalable functionality that is available and delivered.

Note

Level 0 Only Simple Synchronous Behavior

Level 1 Basic Asynchronous Behaviour

Level 2 Enhanced Behaviour

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	2
max	2
min	0

4.58 Parameter SpiMainFunctionPeriod

This parameter defines the cycle time of the function Spi_MainFunction_Handling in seconds.

The parameter is not used by the driver it self, but it is used by upper layer.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	0.01
max	1.0
min	1.0E-7

4.59 Parameter SpiSupportConcurrentSyncTransmit

Specifies whether concurrent Spi_SyncTransmit() calls for different se-quences shall be configurable.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.60 Parameter SpiVersionInfoApi

Switches the Spi_GetVersionInfo function ON or OFF.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.61 Parameter SpiGlobalDmaEnable

If checked, it allows using the DMA module during the transfer.

For each SPI unit a transfering method can be configured: FIFO or DMA.

If not checked, all SPI units will use FIFO transfering mode.

NoteThis is an implementation parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.62 Parameter SpiTimeoutMethod

SpiTimeoutMethod

Configures the timeout method.

Based on this selection a certain timeout method from OsIf will be used in the driver.

Note: If SystemTimer or CustomTimer are selected make sure the corresponding timer is enabled in OsIf General configuration.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	OSIF_COUNTER_DUMMY
literals	['OSIF_COUNTER_DUMMY', 'OSIF_COUNTER_SYSTEM', 'OSIF_COU↩ NTER_CUSTOM']

4.63 Parameter SpiTransmitTimeout

Timeout value (microseconds) used to wait for TX/RX transmission to complete one frame in both full duplex and half duplex mode NoteThis is an implementation parameter. The transmission will be unsuccessful if the Chip cannot completly transfer one frame during this timeout.

The precision of this value is quite low, it must be greater than the time needed to completly transmit one frame.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

Property	Value
defaultValue	50000
max	900000
min	1

4.64 Reference SpiEcucPartitionRef

ECUC_Spi_00244.Maps the SPI driver to zero or multiple ECUC partitions to make the driver

API available in the according partition.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	Infinite
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/EcuC/EcucPartitionCollection/EcucPartition

4.65 Reference SpiKernelEcucPartitionRef

ECUC_Spi_00245.Maps the SPI kernel to zero or one ECUC partitions to assign the driver

kernel to a certain core. The ECUC partition referenced is a subset of the

ECUC partitions where the SPI driver is mapped to. SPI driver is implemented according to multicore type II, so this node is not used.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true

Property	Value
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/EcuC/EcucPartitionCollection/EcucPartition

4.66 Container SpiPhyUnit

Logical to Physical SPI Bus mapping.

NoteThis is an implementation specific container.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.67 Parameter SpiPhyUnitMapping

Logical SpiHWunit to physical LPSPI_[0|1|2|3|4] or LPSPI_[0|1|2|3|4] assignment. It depends on the number of units present in the chip version.

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false

Property	Value
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	LPSPI_1
literals	['LPSPI_0', 'LPSPI_1', 'LPSPI_2', 'FLEXIO_SPI_0', 'FLEXIO_SPI_1']

4.68 Parameter SpiPhyUnitMode

Select between SPI_MASTER and SPI_SLAVE modes.

SPI Slave mode support only if SpiGeneral/SpiLevelDelivered is 1 or 2

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	SPI_MASTER
literals	['SPI_MASTER', 'SPI_SLAVE']

4.69 Parameter SpiPhyUnitSync

Specific if this HwUnit can only do sync transfers.

If true then this hardware unit is dedicated for Synchronous transfers only.

If false then this hardware unit is dedicated for Asynchronous transfers only.

False is applicable only if SpiGeneral/SpiLevelDelivered is either 1 or 2

and true is applicable only if SpiGeneral/SpiLevelDelivered is 0 or 2.

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.70 Parameter SpiSamplePoint

When set, the LPSPI master will sample the input data on a delayed LPSPI_SCK edge, which improves the setup time when sampling data.

? The input data setup time in master mode with delayed LPSPI_SCK edge is equal to the input data setup time in slave mode

? In slave mode, the SAMPLE bit is ignored

0b - Input data is sampled on SCK edge

1b - Input data is sampled on delayed SCK edge

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1

Property	Value
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	1
min	0

4.71 Parameter SpiPinConfiguration

Configures which pins are used for input and output data during serial transfers. When performing parallel transfers, the Pin Configuration field is ignored.

00b - SIN is used for input data and SOUT is used for output data

01b - SIN is used for both input and output data, only half-duplex serial transfers are supported

10b - SOUT is used for both input and output data, only half-duplex serial transfers are supported

11b - SOUT is used for input data and SIN is used for output data

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	3
min	0

4.72 Parameter SpiPhyUnitAsyncUseDma

Select Asynchronous mechanism with DMA or not.

NoteThis is an implementation parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.73 Parameter SpiMaxDmaFastTransfer

Number of transfer section allowed in Dma Fast transfer.

Note: This feature will be supported if SpiEnableDmaFastTransferSupport and SpiPhyUnitAsyncUseDma checked.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	1
max	10
min	1

4.74 Reference SpiPhyUnitClockRef

Reference to the SPI clock source configuration, which is set into the MCU driver configuration.

This clock source is used for configure SPI baudrate.

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Mcu/McuModuleConfiguration/McuClockSetting↩ Config/McuClockReferencePoint

4.75 Reference SpiPhyUnitAlternateClockRef

Reference to the alternate clock configuration, retrieved from the MCU plugin.

Use to enable Spi_SetClockMode() function, which allows dual MCU clock configuration settings.

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Mcu/McuModuleConfiguration/McuClockSetting↩ Config/McuClockReferencePoint

4.76 Reference SpiPhyTxDmaChannel

SPI Master Transmit DMA Logical Channel as configured by MCL plug-in, used to prepare the SPI transmission dataframes

starting from the TX buffer content.

This parameter is required only if SpiPhyUnitAsyncUseDma is checked.

NoteThis is an implementation specific parameter. The current SPI TX source

needs be configured for enabling this DMA channel.

Property	Value
type	ECUC-CHOICE-REFERENCE-DEF
origin	NXP
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destinations	['/TS_T40D2M10I1R0/Mcl/MclConfig/dmaLogicChannel_Type']

4.77 Reference SpiPhyRxDmaChannel

SPI Receive DMA Logical Channel as configured by MCL plug-in, used to read the deserialized dataframes into the RX buffers.

This parameter is required only if SpiPhyUnitAsyncUseDma is checked.

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-CHOICE-REFERENCE-DEF
origin	NXP
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
	VARIANT-LINK-TIME: PRE-COMPILE

multiplicityConfigClasses

Property	Value
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destinations	['/TS_T40D2M10I1R0/Mcl/MclConfig/dmaLogicChannel_Type']

4.78 Reference SpiFlexioTxAndClkChannelsConfig

FLEXIO Logical Channel for TX(MOSI) and CLK(Clock). This selects 1 PIN to setup as TX(MOSI) channel and 1 PIN to set up as CLK channel

If you select CHANNEL_X, PIN_Y(FlexioMclPinId) and PIN_Z(FlexioMclAddPinId) that mean:

- ShifterX is selected for TX(MOSI) channel

- TimerX is selected for CLK channel.

- PIN_Y is selected for TX(MOSI) channel

- PIN_Z is selected for CLK channel

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-CHOICE-REFERENCE-DEF
origin	NXP
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destinations	['/TS_T40D2M10I1R0/Mcl/MclConfig/FlexioCommon/FlexioMclLogicChannels']

4.79 Reference SpiFlexioRxAndCsChannelsConfig

FLEXIO Logical Channel for RX(MISO) and CS(Chip select). This selects 1 PIN to setup as MISO channel and 1 PIN to set up as CS channel

If you select CHANNEL_X, PIN_Y(FlexioMclPinId) and PIN_Z(FlexioMclAddPinId) that mean:

- ShifterX is selected for RX(MISO) channel

- TimerX is selected for CS channel.

- PIN_Y is selected for RX(MISO) channel

- PIN_Z is selected for CS channel

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-CHOICE-REFERENCE-DEF
origin	NXP
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destinations	['/TS_T40D2M10I1R0/Mcl/MclConfig/FlexioCommon/FlexioMclLogic↩ Channels', '/TS_T40D2M10I1R0/Mcl/MclConfig/FlexioCommon/Flexio↩ MclLogicChannels']

4.80 Container SpiPublishedInformation

Container holding all SPI specific published information parameters.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF

Property	Value
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.81 Parameter SpiMaxHwUnit

Number of different SPI hardware microcontroller peripherals (units/busses) available and handled by this SPI Handler/Driver module.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	0
max	65535
min	0

4.82 Container CommonPublishedInformation

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

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.83 Parameter ArReleaseMajorVersion

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

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	4
max	4
min	4

4.84 Parameter ArReleaseMinorVersion

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

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	4
max	4
min	4

4.85 Parameter ArReleaseRevisionVersion

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

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	0
min	0

4.86 Parameter ModuleId

Module ID of this module.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	83
max	83
min	83

4.87 Parameter SwMajorVersion

Major version number of the vendor specific implementation of the module. The numbering is vendor specific.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	1
max	1
min	1

4.88 Parameter SwMinorVersion

Minor version number of the vendor specific implementation of the module. The numbering is vendor specific.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	0
min	0

4.89 Parameter SwPatchVersion

Patch level version number of the vendor specific implementation of the module. The numbering is vendor specific.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	1
max	1
min	1

4.90 Parameter VendorApiInfix

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>.

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.

Property	Value
type	ECUC-STRING-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD

Property	Value
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	

4.91 Parameter VendorId

Vendor ID of the dedicated implementation of this module according to the AUTOSAR vendor list.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	43
max	43
min	43

4.92 Container SpiAutosarExt

Enabling the settings of this section will configure the driver in a mode not compliant with AUTOSAR requirements.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.93 Parameter SpiEnableUserModeSupport

When this parameter is enabled, the Spi module will adapt to run from User Mode.

Note Spi module does not include registers protection. So, It is accessible to all registers in any public mode.

SPI is not affected by this field.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.94 Parameter SpiEnableDmaFastTransferSupport

When this parameter is enabled, the SPI module can support to transfer a Sequence with multiple Channels, Jobs using DMA ScatterGather and CPU used only for processing end of sequence transfer.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.95 Parameter SpiHalfDuplexModeSupport

When this parameter is enabled, the SPI module can support to transfer in half duplex mode.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.96 Parameter SpiAllowBigSizeCollections

A feature to allow more than 256 sequences, jobs, and channels.

Note Enabling this option will violate the following requirements: SPI166, SPI167, SPI168.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.97 Parameter SpiEnableHWUnitAsyncMode

Enable Spi_SetHWUnitAsyncMode() function, which allows defining distinct operation mode (POLLING or INTERRUPT) for each HWUnit.

Note This feature is not required by Autosar, which defines asynchronous mode configuration at driver level only.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.98 Parameter SpiJobStartNotificationEnable

settings.

Note This feature is a SpiAutosarExt feature to enable the start job notification.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.99 Parameter SpiDisableDemReportErrorStatus

SpiDisableDemReportErrorStatus

Switches the Diagnostic Error Reporting and Notification OFF.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.100 Parameter SpiFlexioEnable

SpiFlexioEnable

If it is true, FLEXIO feature is enabled

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

This chapter describes the Tresos configuration plug-in for the driver. All the parameters are described below.

- Module Spi
  - Container SpiDemEventParameterRefs
    - Reference SPI_E_HARDWARE_ERROR
  - Container SpiDriver

* Parameter SpiMaxChannel
* Parameter SpiMaxJob
* Parameter SpiMaxSequence
* Container SpiChannel
  - Parameter SpiChannelId
  - Parameter SpiChannelType
  - Parameter SpiDataWidth
  - Parameter SpiDefaultData
  - Parameter SpiEbMaxLength
  - Parameter SpiIbNBuffers
  - Parameter SpiTransferStart
  - Reference SpiChannelEcucPartitionRef
* Container SpiExternalDevice
  - Parameter SpiBaudrate
  - Parameter SpiBaudrateFlexio1
  - Parameter SpiBaudrateFlexio2
  - Parameter SpiBaudrateFlexio3
  - Parameter SpiCsIdentifier
  - Parameter SpiCsPolarity
  - Parameter SpiCsSelection
  - Parameter SpiDataShiftEdge
  - Parameter SpiEnableCs
  - Parameter SpiHwUnit
  - Parameter SpiShiftClockIdleLevel
  - Parameter SpiTimeClk2Cs
  - Parameter SpiTimeCs2Clk
  - Parameter SpiTimeCs2Cs
  - Parameter SpiCsContinous
  - Reference SpiDeviceEcucPartitionRef
* Container SpiJob
  - Parameter SpiJobEndNotification
  - Parameter SpiJobStartNotification
  - Parameter SpiJobId
  - Parameter SpiJobPriority
  - Reference SpiDeviceAssignment
  - Container SpiChannelList
  - Parameter SpiChannelIndex
  - Reference SpiChannelAssignment
* Container SpiSequence
  - Parameter SpiInterruptibleSequence
  - Parameter SpiSeqEndNotification
  - Parameter SpiSequenceId
  - Parameter SpiEnableDmaFastTransfer
  - Reference SpiJobAssignment

- Container SpiGeneral
  - Parameter SpiMulticoreSupport
  - Parameter SpiCancelApi
  - Parameter SpiChannelBuffersAllowed
  - Parameter SpiDevErrorDetect
  - Parameter SpiHwStatusApi
  - Parameter SpiInterruptibleSeqAllowed
  - Parameter SpiLevelDelivered
  - Parameter SpiMainFunctionPeriod
  - Parameter SpiSupportConcurrentSyncTransmit
  - Parameter SpiVersionInfoApi
  - Parameter SpiGlobalDmaEnable
  - Parameter SpiTimeoutMethod
  - Parameter SpiTransmitTimeout
  - Reference SpiEcucPartitionRef
  - Reference SpiKernelEcucPartitionRef
  - Container SpiPhyUnit
    - Parameter SpiPhyUnitMapping
    - Parameter SpiPhyUnitMode
    - Parameter SpiPhyUnitSync
    - Parameter SpiSamplePoint
    - Parameter SpiPinConfiguration
    - Parameter SpiPhyUnitAsyncUseDma
    - Parameter SpiMaxDmaFastTransfer
    - Reference SpiPhyUnitClockRef
    - Reference SpiPhyUnitAlternateClockRef
    - Reference SpiPhyTxDmaChannel
    - Reference SpiPhyRxDmaChannel
    - Reference SpiFlexioTxChannelConfig
    - Reference SpiFlexioRxChannelConfig
    - Reference SpiFlexioClkChannelConfig
    - Reference SpiFlexioCsChannelConfig
- Container SpiPublishedInformation
  - Parameter SpiMaxHwUnit
- Container CommonPublishedInformation

        * Parameter ArReleaseMajorVersion
        * Parameter ArReleaseMinorVersion
        * Parameter ArReleaseRevisionVersion
        * Parameter ModuleId
        * Parameter SwMajorVersion
        * Parameter SwMinorVersion
        * Parameter SwPatchVersion
        * Parameter VendorApiInfix
        * Parameter VendorId
    - Container SpiAutosarExt
        * Parameter SpiEnableUserModeSupport
        * Parameter SpiEnableDmaFastTransferSupport
        * Parameter SpiAllowBigSizeCollections
        * Parameter SpiEnableHWUnitAsyncMode
        * Parameter SpiJobStartNotificationEnable
        * Parameter SpiDisableDemReportErrorStatus
        * Parameter SpiFlexioEnable

4.101 Module Spi

Configuration of the Spi (Serial Peripheral Interface) module.

Included containers:

- SpiDemEventParameterRefs
- SpiDriver
- SpiGeneral
- SpiPublishedInformation
- CommonPublishedInformation
- SpiAutosarExt

Property	Value
type	ECUC-MODULE-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantSupport	true
supportedConfigVariants	VARIANT-PRE-COMPILE, VARIANT-LINK-TIME, VARIANT-POST-BUILD

4.102 Container SpiDemEventParameterRefs

Container for the references to DemEventParameter elements which shall be invoked using the Dem_SetEventStatus API in case the corresponding error occurs.

The EventId is taken from the referenced DemEventParameter's DemEventId value. The standardized errors are provided in the container and can be extended by vendor specific error references.

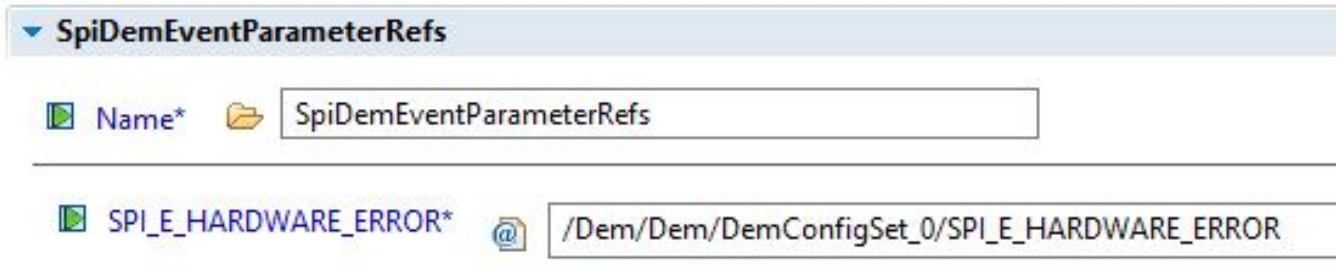

Figure 4.1 Tresos Plugin snapshot for SpiDemEventParameterRefs form.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.103 Reference SPI_E_HARDWARE_ERROR

Reference to configured DEM event to report "Hardware failure". If the reference is not configured the error shall not be reported.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	1

Property	Value
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Dem/DemConfigSet/DemEventParameter

4.104 Container SpiDriver

This container contains the configuration parameters and sub containers of the AUTOSAR Spi module.

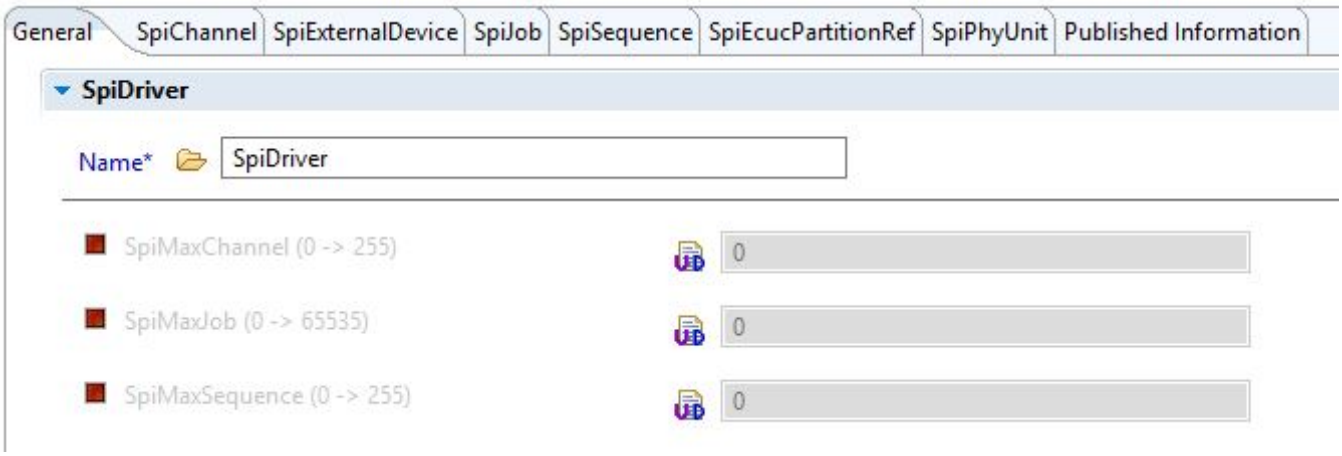

Figure 4.2 Tresos Plugin snapshot for SpiDriver form.

Included subcontainers:

- SpiChannel
- SpiExternalDevice
- SpiJob
- SpiSequence

Property	Value

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.105 Parameter SpiMaxChannel

This parameter contains the number of Channels configured. It will be gathered by tools during the configuration stage.

NoteThis parameter is not used, instead max channel value is derived from number of channels configured.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	255
min	0

4.106 Parameter SpiMaxJob

This parameter contains the number of Jobs configured. It will be gathered by tools during the configuration stage.

NoteThis parameter is not used, instead max jobs value is derived from number of jobs configured.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	65535
min	0

4.107 Parameter SpiMaxSequence

This parameter contains the number of Sequences configured. It will be gathered by tools during the configuration stage.

NoteThis parameter is not used, instead max Sequences value is derived from number of sequences configured.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	255
min	0

4.108 Container SpiChannel

All data needed to configure one SPI-channel.

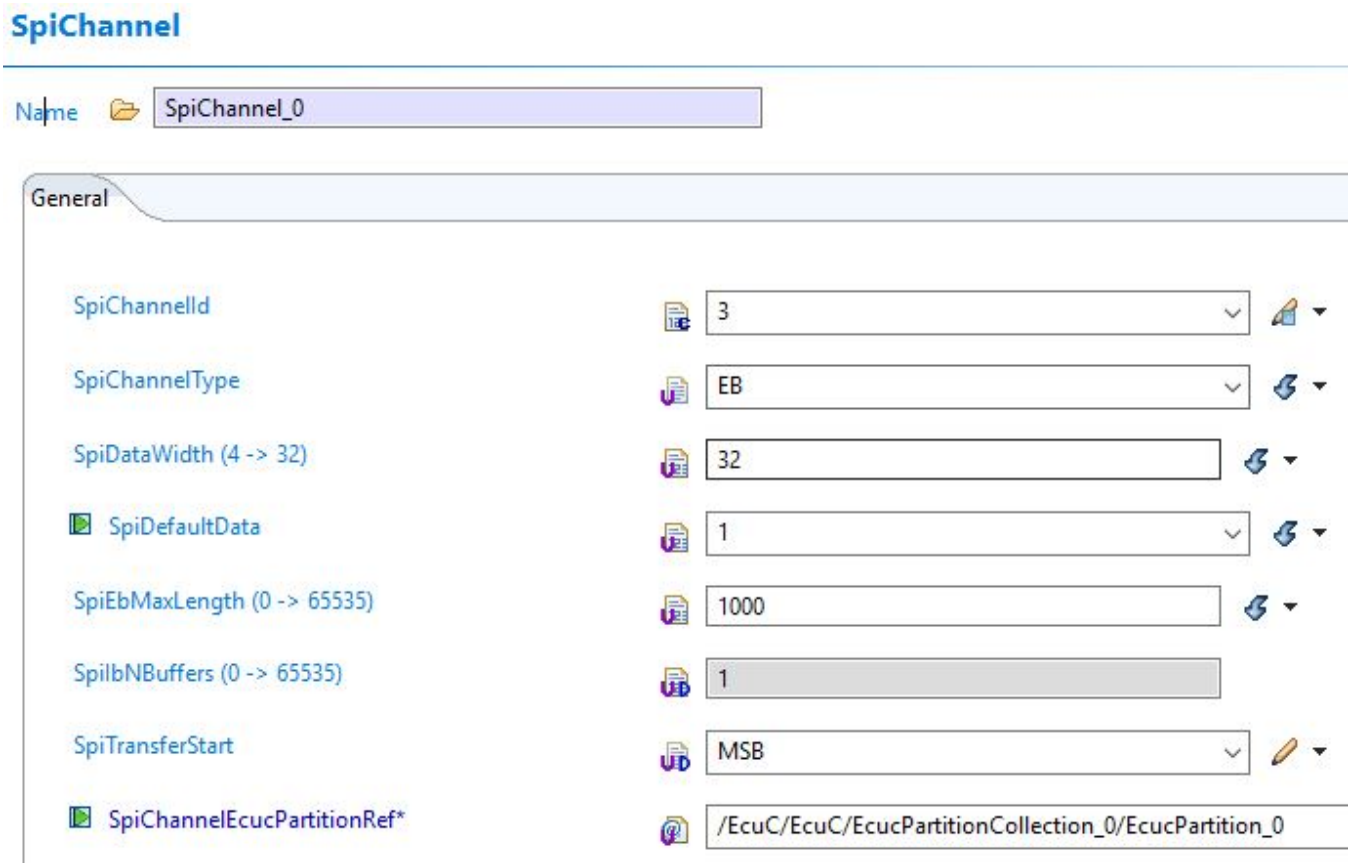

Figure 4.3 Tresos Plugin snapshot for SpiChannel form.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.109 Parameter SpiChannelId

SPI Channel ID, used as parameter in SPI API functions.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	true
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	0
max	255
min	0

4.110 Parameter SpiChannelType

Buffer usage with EB/IB channel.

NoteThis parameter is dependant on SpiChannelBuffersAllowed parameter.

When SpiChannelBuffersAllowed = 0; SpiChannelType should be IB

When SpiChannelBuffersAllowed = 1; SpiChannelType should be EB

When SpiChannelBuffersAllowed = 2; SpiChannelType can be IB or EB

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	IB
literals	['EB', 'IB']

4.111 Parameter SpiDataWidth

This parameter is the width of a transmitted data unit.

NoteThe hardware supports data width from 4 to 32 bit.The unit is in bits.

When SpiChannelBuffersAllowed = 0; SpiChannelType should be IB

When SpiChannelBuffersAllowed = 1; SpiChannelType should be EB

When SpiChannelBuffersAllowed = 2; SpiChannelType can be IB or EB

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	8
max	32
min	4

4.112 Parameter SpiDefaultData

The default data to be transmitted when (for internal buffer or external buffer)

the pointer passed to Spi_WriteIB (for internal buffer) or to Spi_SetupEB (for external buffer) is NULL.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1
max	4294967295
min	0

4.113 Parameter SpiEbMaxLength

This parameter contains the maximum size (number of data elements) of data buffers in case of EB Channels and only.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1
max	65535
min	0

4.114 Parameter SpiIbNBuffers

This parameter contains the maximum number of data buffers in case of IB Channels and only.

In case of channel's Spi_DataWidth ranges from 9 to 16, this parameter reffers to

the number of bytes allocated to the buffers and MUST be even. Or divisible by 4 if the range from 17 to 32.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false

Property	Value
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1
max	65535
min	0

4.115 Parameter SpiTransferStart

This parameter defines the first starting bit for transmission.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	MSB
literals	['LSB', 'MSB']

4.116 Reference SpiChannelEcucPartitionRef

Maps an SPI Channel to zero or ECUC partition to limit the access to this Channel. The ECUC partition referenced is a

subset of the ECUC partitions where the SPI driver is mapped to.

Property	Value
type	ECUC-REFERENCE-DEF

Property	Value
origin	NXP
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/EcuC/EcucPartitionCollection/EcucPartition

4.117 Container SpiExternalDevice

The communication settings of an external device. Closely linked to SpiJob.

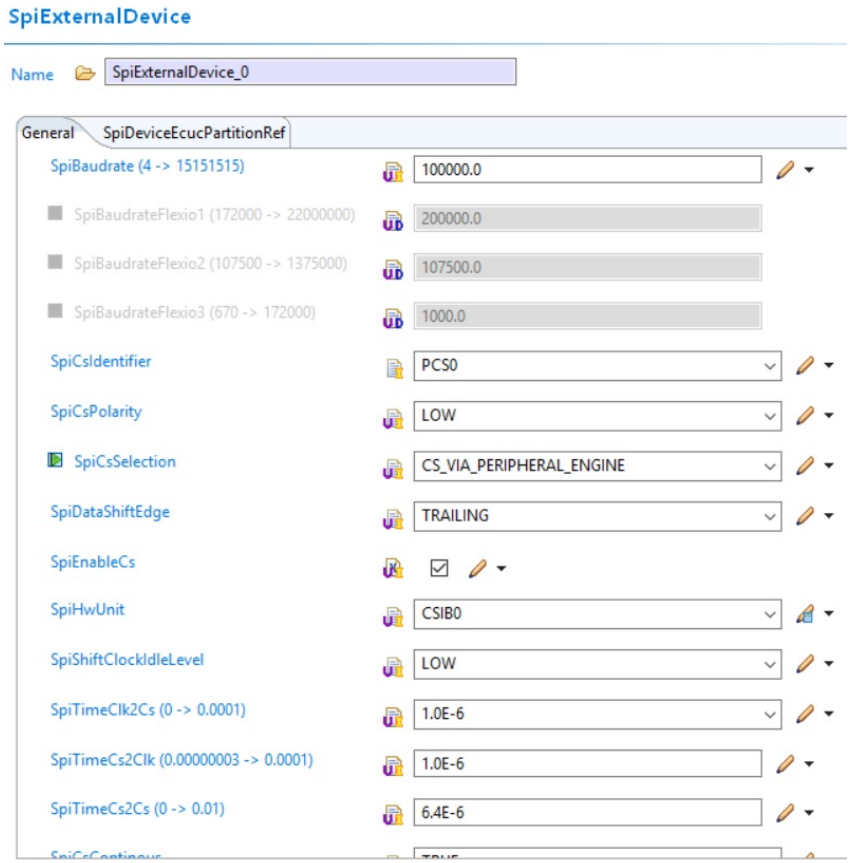

Figure 4.4 Tresos Plugin snapshot for SpiExternalDevice form.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.118 Parameter SpiBaudrate

This parameter is the communication baudrate - This parameter allows using a range of values, from the point of view of configuration tools, from Hz up to MHz.

This field is used only in MASTER mode.

This field is used only for LPSPI

NoteThe precision of this value depends SPI clock source configuration. If the driver cannot generate correct of the value, approximate value will be used.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	100000.0
max	1.5151515E7
min	4.0

4.119 Parameter SpiBaudrateFlexio1

This parameter is the communication baudrate - This parameter allows using a range of values, from the point of view of configuration tools, from Hz up to MHz.

This field is used only in MASTER mode.

This field is used only in FLEXIO SPI with TIMDEC = 0, range [FLEXIO_CLK/(256*2) to FLEXIO_CLK/4] = [172K - 22M]

NoteThe precision of this value depends SPI clock source configuration. If the driver cannot generate correct of the value, approximate value will be used.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	200000.0
max	2.2E7
min	172000.0

4.120 Parameter SpiBaudrateFlexio2

This parameter is the communication baudrate - This parameter allows using a range of values, from the point of view of configuration tools, from Hz up to MHz.

This field is used only in MASTER mode.

This field is used only in FLEXIO SPI with TIMDEC = 4, range [FLEXIO_CLK/(256*2*16) to FLEXIO_CLK/32] = [10.75K - 1.375M]

NoteThe precision of this value depends SPI clock source configuration. If the driver cannot generate correct of the value, approximate value will be used.

Property	Value
type	ECUC-FLOAT-PARAM-DEF

Property	Value
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	107500.0
max	1375000.0
min	107500.0

4.121 Parameter SpiBaudrateFlexio3

This parameter is the communication baudrate - This parameter allows using a range of values, from the point of view of configuration tools, from Hz up to MHz.

This field is used only in MASTER mode.

This field is used only in FLEXIO SPI with TIMDEC = 4, range [FLEXIO_CLK/(256*2*256) to FLEXIO_CLK/512] = [670 - 172K]

NoteThe precision of this value depends SPI clock source configuration. If the driver cannot generate correct of the value, approximate value will be used.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1000.0
max	172000.0
min	670.0

4.122 Parameter SpiCsIdentifier

This parameter is the symbolic name to identify the Chip Select (CS) allocated to this Job.

The chip selects are specific per HwUnit. Please check in Reference Manual for information on available chip selects.

If FLEXIO channel used, Chip Select will be configured by SpiPhyUnit\SpiFlexioCsPinSelect and this parameter will be not used.

If SpiEnableCs is not set, value of this node will not be used by driver code. It should set to default value (PCS0)

Property	Value
type	ECUC-STRING-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	true
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	PCS0

4.123 Parameter SpiCsPolarity

This parameter defines the active polarity of Chip Select.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	HIGH
literals	['HIGH', 'LOW']

4.124 Parameter SpiCsSelection

When the Chip select handling is enabled (see SpiEnableCs), then this

parameter specifies if the chip select is handled automatically by

Pe-ripheral HW engine or via general purpose IO by Spi driver.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	CS_VIA_PERIPHERAL_ENGINE
literals	['CS_VIA_PERIPHERAL_ENGINE', 'CS_VIA_GPIO']

4.125 Parameter SpiDataShiftEdge

This parameter defines the SPI data shift edge.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	LEADING
literals	['LEADING', 'TRAILING']

4.126 Parameter SpiEnableCs

This parameter enables or not the Chip Select handling functions.

This parameter is closely linked to Job.If This parameter is True,then chip select is asserted and

if False No chip select is asserted.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.127 Parameter SpiHwUnit

This parameter is the symbolic name to identify the HW SPI Hardware microcontroller peripheral allocated to this Job.

CSIBn references the n-th logical unit configured in SpiPhyUnit container. For example: CSIB0 references the first logical unit

(not the first SPI_0 HW unit).

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	CSIB0
literals	['CSIB0', 'CSIB1', 'CSIB2', 'CSIB3', 'CSIB4', 'CSIB5', 'CSIB6', 'CSIB7', 'CSIB8', 'CSIB9']

4.128 Parameter SpiShiftClockIdleLevel

This parameter defines the SPI shift clock idle level.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	HIGH
literals	['HIGH', 'LOW']

4.129 Parameter SpiTimeClk2Cs

Timing between clock and chip select in seconds (tASC) - This parameter allows to use a range of values

from 0 up to 0.0001 Sec. The real configuration-value used in software BSW-SPI is calculated out of this by the generator-tools.

If use continuous transfer(PCS signals remain asserted between transfers), tASC and tCSC will insert between transfers.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1.0E-6
max	1.0E-4
min	0.0

4.130 Parameter SpiTimeCs2Clk

Timing between chip select and clock in seconds (tCSC) - This parameter allows to use a range of values from 0.00000003 up to 0.0001 Sec.

If use continuous transfer(PCS signals remain asserted between transfers), tASC and tCSC will insert between transfers.

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1.0E-6
max	1.0E-4
min	3.0E-8

4.131 Parameter SpiTimeCs2Cs

Timing between chip select assertions in seconds (tDT) - This parameter allows to use a range of values from 0 up to 0.01 Sec.

If use continuous transfer(PCS signals remain asserted between transfers), tDT is not inserted between the transfers.

NoteThis is an implementation parameter.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	6.4E-6
max	0.01
min	0.0

4.132 Parameter SpiCsContinous

This field determines to keep chip select asserted between frame transfers.

NoteThis is an implementation parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	TRUE
literals	['TRUE', 'FALSE']

4.133 Reference SpiDeviceEcucPartitionRef

ECUC_Spi_00246. Maps an SPI external device to zero or multiple ECUC partitions to limit the access to this external device. The ECUC partitions referenced are a subset of the ECUC partitions where the SPI driver is mapped to.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC

Property	Value
lowerMultiplicity	0
upperMultiplicity	Infinite
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/EcuC/EcucPartitionCollection/EcucPartition

4.134 Container SpiJob

All data needed to configure one SPI-Job, amongst others the connection between the internal SPI unit and the special settings for an external device is done.

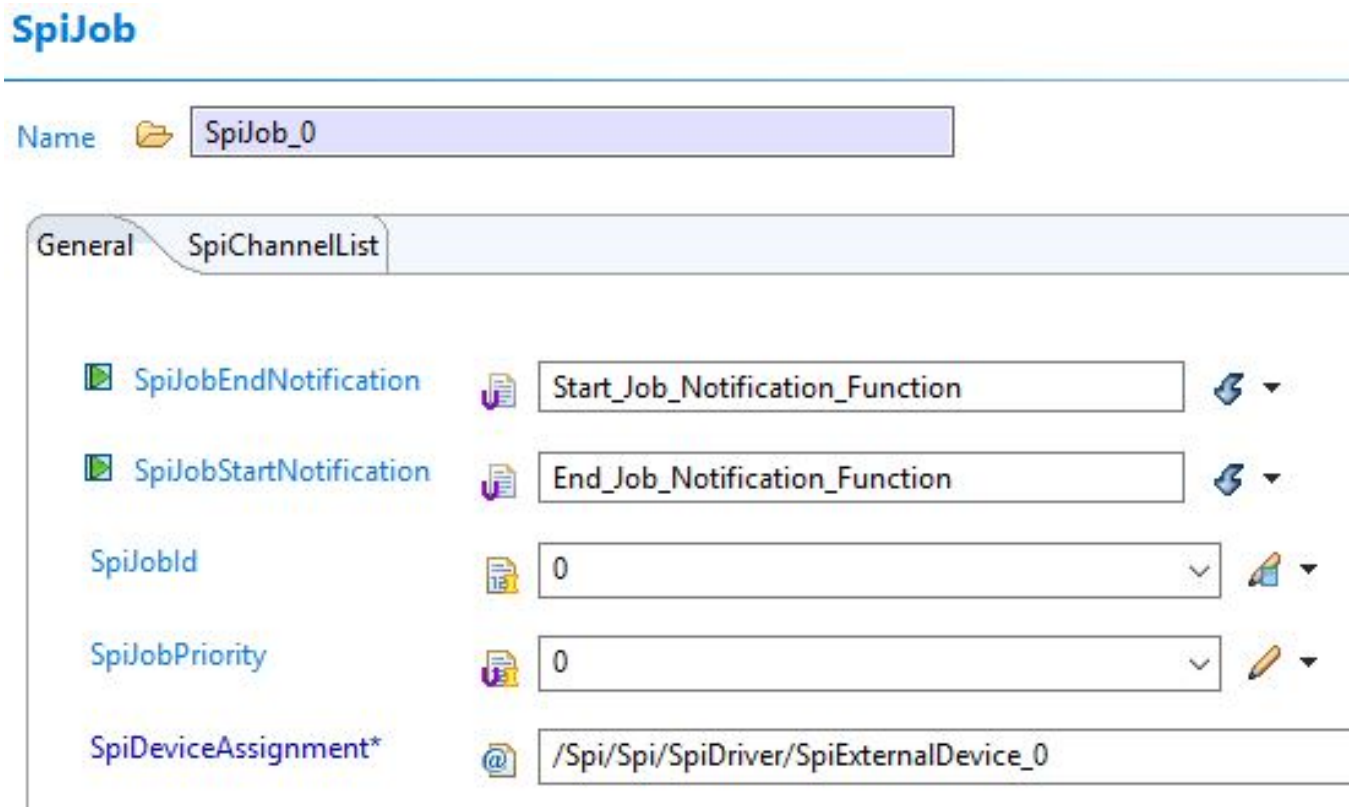

Figure 4.5 Tresos Plugin snapshot for SpiJob form.

Included subcontainers:

- SpiChannelList

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.135 Parameter SpiJobEndNotification

This parameter is a reference to a notification function.

Property	Value
type	ECUC-FUNCTION-NAME-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	NULL_PTR

4.136 Parameter SpiJobStartNotification

This parameter is a reference to a notification function.

Property	Value
type	ECUC-FUNCTION-NAME-DEF
origin	NXP

Property	Value
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	NULL_PTR

4.137 Parameter SpiJobId

SPI Job ID, used as parameter in SPI API functions.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	true
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	0
max	65535
min	0

4.138 Parameter SpiJobPriority

Priority set accordingly to SPI093: 0, lowest, 3, highest priority

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC

Property	Value
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	3
min	0

4.139 Reference SpiDeviceAssignment

Reference to the external device used by this job.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Spi/SpiDriver/SpiExternalDevice

4.140 Container SpiChannelList

References to SPI channels and their order within the Job.

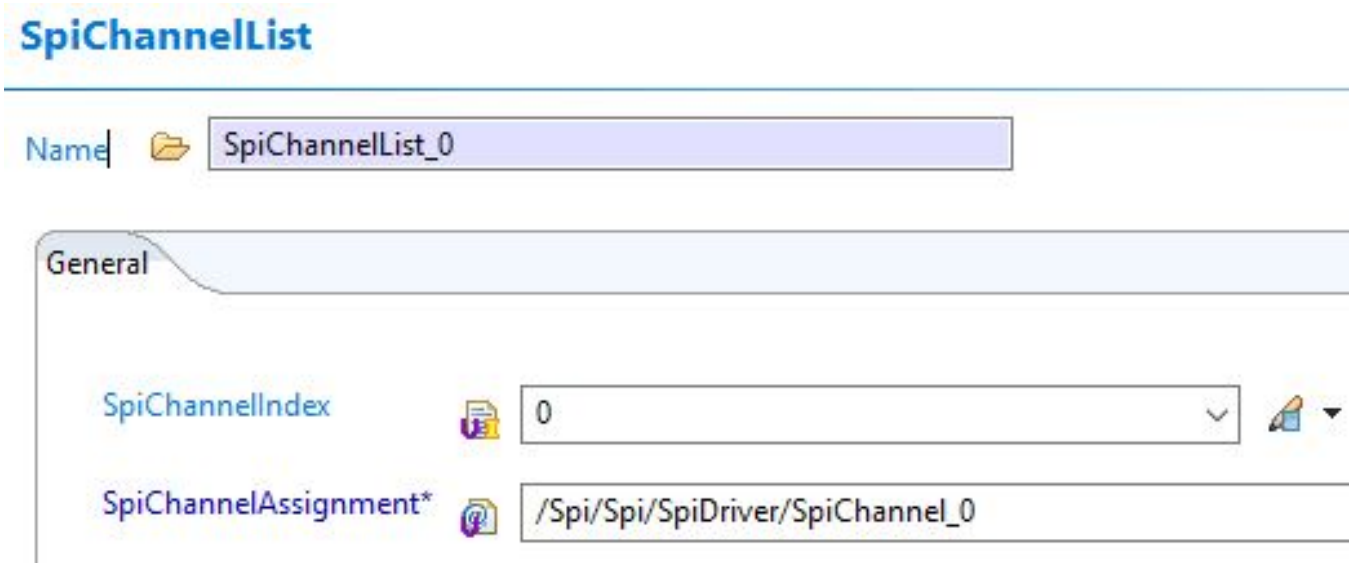

Figure 4.6 Tresos Plugin snapshot for SpiChannelList form.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.141 Parameter SpiChannelIndex

This parameter specifies the order of Channels within the Job.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1

Property	Value
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	255
min	0

4.142 Reference SpiChannelAssignment

A job references several channels.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Spi/SpiDriver/SpiChannel

4.143 Container SpiSequence

All data needed to configure one SPI-sequence.

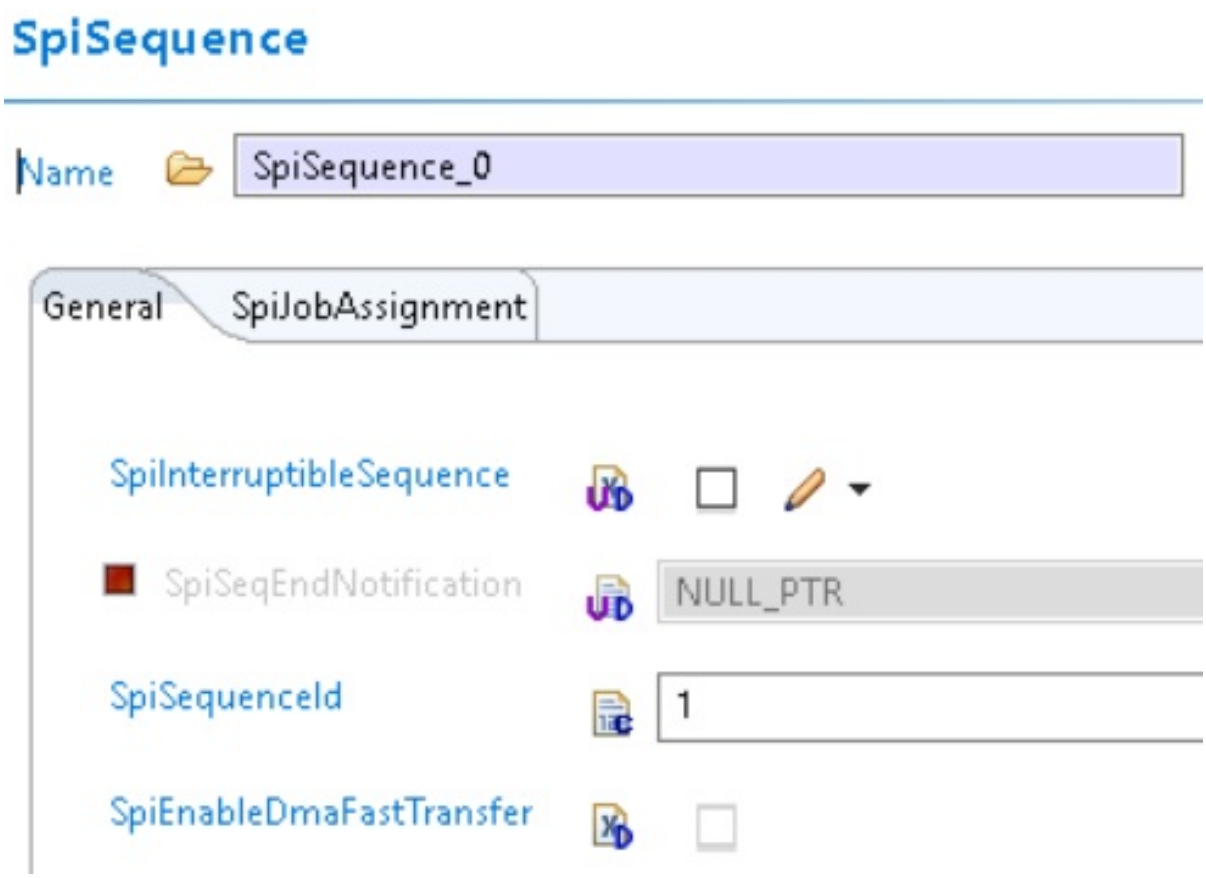

Figure 4.7 Tresos Plugin snapshot for SpiSequence form.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.144 Parameter SpiInterruptibleSequence

This parameter allows or not this Sequence to be suspended by another one.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.145 Parameter SpiSeqEndNotification

This parameter is a reference to a notification function.

Property	Value
type	ECUC-FUNCTION-NAME-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	NULL_PTR

4.146 Parameter SpiSequenceId

Sequence ID of configured SPI Sequence.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC

Property	Value
symbolicNameValue	true
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	0
max	255
min	0

4.147 Parameter SpiEnableDmaFastTransfer

When this parameter is enabled, this Sequence will be transferred using DMA ScatterGather and CPU used only for processing end of Sequence.

SpiAutosarExt/SpiEnableDmaFastTransferSupport must be checked to support this feature.

Note: This feature requires:

1. The parameters SpiBaudrate, SpiHwUnit, SpiTimeClk2Cs, SpiTimeCs2Clk, SpiTimeCs2Cs in External Device linked to each Job in this Sequence must be the same.

2. The parameters SpiDataWidth and SpiTransferStart in Channel assigned to each Job in this Sequence must be the same.

3. In each Channel, the number of data buffers is NOT higher than 32767 if SpiDataWidth < 9. So, SpiIbNBuffers and SpiEbMaxLength must be assigned to suitable values.

4. Only Master mode is supported.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.148 Reference SpiJobAssignment

A sequence references several jobs, which are executed during a communication sequence.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Spi/SpiDriver/SpiJob

4.149 Container SpiGeneral

General configuration settings for SPI-Handler.

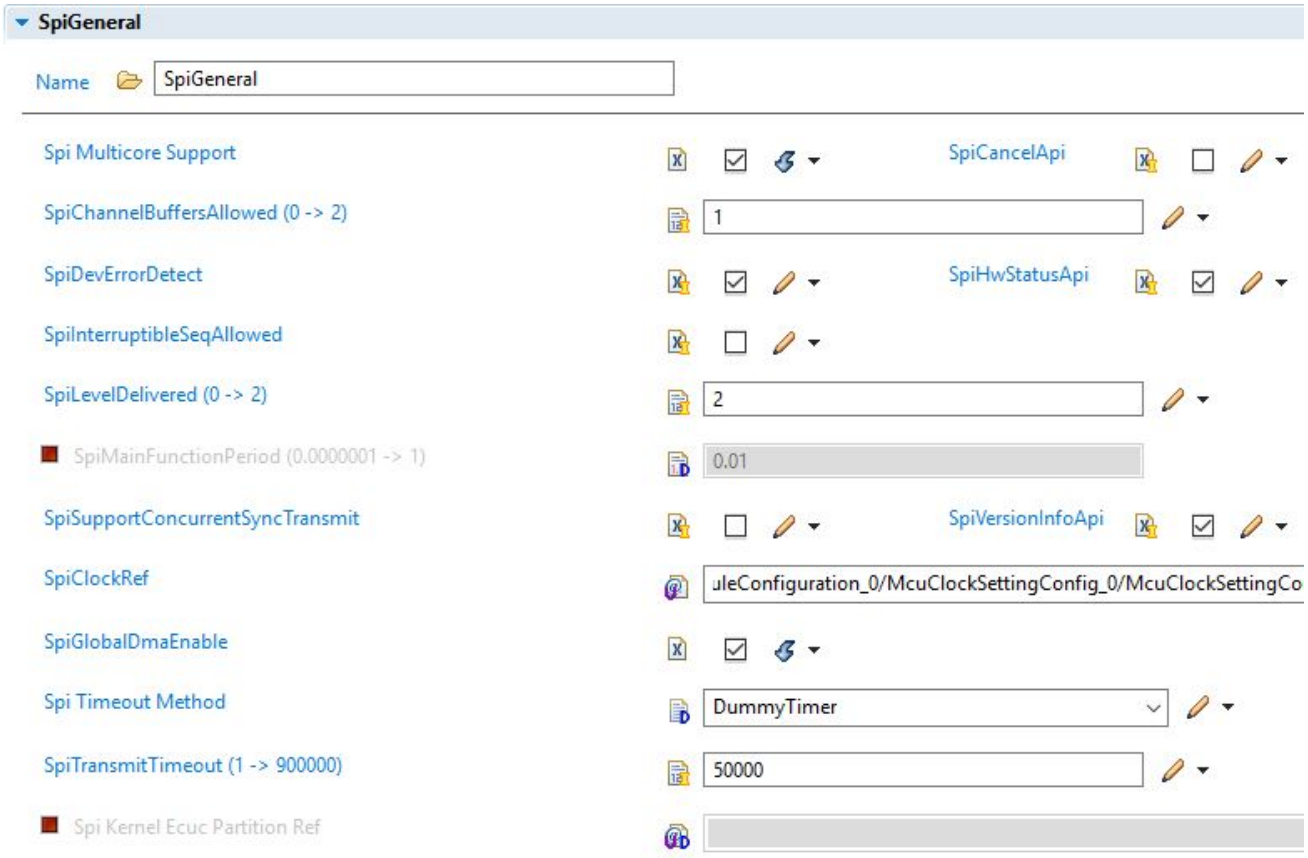

Figure 4.8 Tresos Plugin snapshot for SpiGeneral form.

Included subcontainers:

- SpiPhyUnit

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.150 Parameter SpiMulticoreSupport

SpiMulticoreEnable

When this parameter is enabled, multi-core feature will be used in SPI driver.

That means mapping the SPI driver to multiple ECUC partitions to make the module API available in this partition.

The SPI driver will operate as an independent instance in each of the partitions.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.151 Parameter SpiCancelApi

Switches the Spi_Cancel function ON or OFF.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.152 Parameter SpiChannelBuffersAllowed

Selects the SPI Handler/Driver Channel Buffers usage allowed and delivered.

Note

0 - Only Internal Buffers (IB) are allowed

1 - Only External buffers (EB) are allowed

2 - Both Internal (IB) and External (EB) buffers are allowed

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	0
max	2
min	0

4.153 Parameter SpiDevErrorDetect

Switches the Development Error Detection and Notification ON or OFF.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.154 Parameter SpiHwStatusApi

Switches the Spi_GetHWUnitStatus function ON or OFF.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.155 Parameter SpiInterruptibleSeqAllowed

Switches the Interruptible Sequences handling functionality ON or OFF.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.156 Parameter SpiLevelDelivered

Selects the SPI Handler/Driver level of scalable functionality that is available and delivered.

Note

Level 0 Only Simple Synchronous Behavior

Level 1 Basic Asynchronous Behaviour

Level 2 Enhanced Behaviour

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	2
max	2
min	0

4.157 Parameter SpiMainFunctionPeriod

This parameter defines the cycle time of the function Spi_MainFunction_Handling in seconds.

The parameter is not used by the driver it self, but it is used by upper layer.

Property	Value
type	ECUC-FLOAT-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	0.01
max	1.0
min	1.0E-7

4.158 Parameter SpiSupportConcurrentSyncTransmit

Specifies whether concurrent Spi_SyncTransmit() calls for different se-quences shall be configurable.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.159 Parameter SpiVersionInfoApi

Switches the Spi_GetVersionInfo function ON or OFF.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.160 Parameter SpiGlobalDmaEnable

If checked, it allows using the DMA module during the transfer.

For each SPI unit a transfering method can be configured: FIFO or DMA.

If not checked, all SPI units will use FIFO transfering mode.

NoteThis is an implementation parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.161 Parameter SpiTimeoutMethod

SpiTimeoutMethod

Configures the timeout method.

Based on this selection a certain timeout method from OsIf will be used in the driver.

Note: If SystemTimer or CustomTimer are selected make sure the corresponding timer is enabled in OsIf General configuration.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	DummyTimer
literals	['DummyTimer', 'SystemTimer', 'CustomTimer']

4.162 Parameter SpiTransmitTimeout

Timeout value (microseconds) used to wait for TX/RX transmission to complete one frame.

NoteThis is an implementation parameter. The transmission will be unsuccessful if the Chip cannot completly transfer one frame during this timeout.

The precision of this value is quite low, it must be greater than the time needed to completly transmit one frame.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	50000
max	900000
min	1

4.163 Reference SpiEcucPartitionRef

ECUC_Spi_00244.Maps the SPI driver to zero or multiple ECUC partitions to make the driver

API available in the according partition.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	Infinite
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/EcuC/EcucPartitionCollection/EcucPartition

4.164 Reference SpiKernelEcucPartitionRef

ECUC_Spi_00245.Maps the SPI kernel to zero or one ECUC partitions to assign the driver

kernel to a certain core. The ECUC partition referenced is a subset of the

ECUC partitions where the SPI driver is mapped to. SPI driver is implemented according to multicore type II, so this node is not used.

Property	Value
type	ECUC-REFERENCE-DEF
origin	AUTOSAR_ECUC
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	true
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/EcuC/EcucPartitionCollection/EcucPartition

4.165 Container SpiPhyUnit

Logical to Physical SPI Bus mapping.

NoteThis is an implementation specific container.

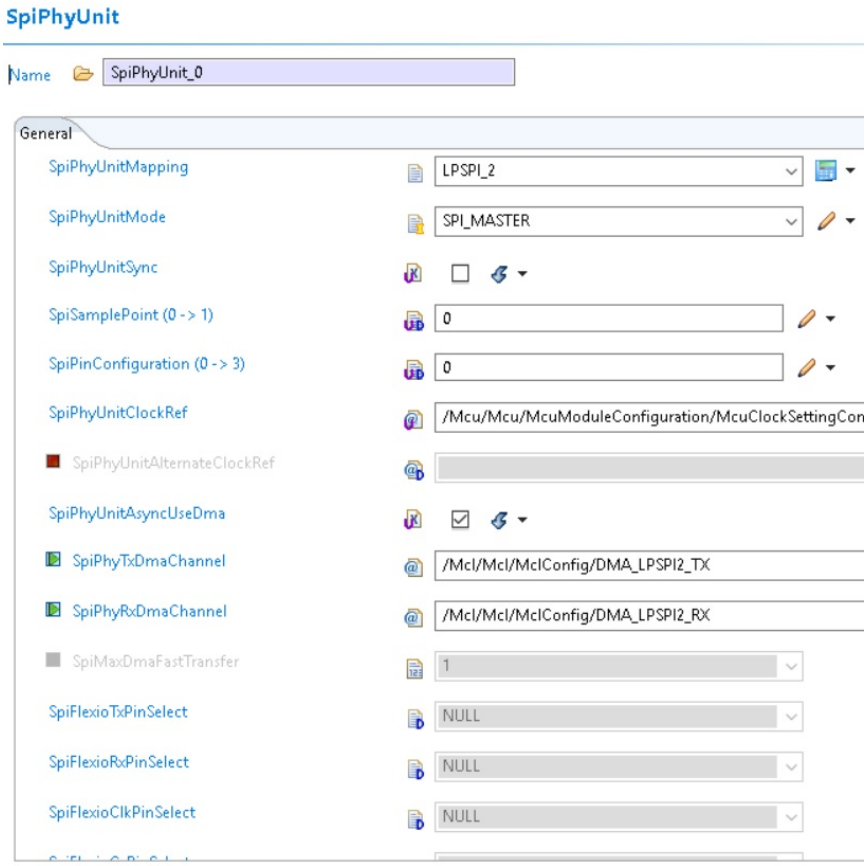

Figure 4.9 Tresos Plugin snapshot for SpiPhyUnit form.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	Infinite
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE

4.166 Parameter SpiPhyUnitMapping

Logical SpiHWunit to physical LPSPI_[0|1|2|3|4] or LPSPI_[0|1|2|3|4] assignment. It depends on the number of units present in the chip version.

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	LPSPI_1
literals	['LPSPI_0', 'LPSPI_1', 'LPSPI_2', 'LPSPI_3', 'LPSPI_4', 'LPSPI_5', 'FLE↩ XIO_SPI_0', 'FLEXIO_SPI_1']

4.167 Parameter SpiPhyUnitMode

Select between SPI_MASTER and SPI_SLAVE modes.

SPI Slave mode support only if SpiGeneral/SpiLevelDelivered is 1 or 2

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	SPI_MASTER
literals	['SPI_MASTER', 'SPI_SLAVE']

4.168 Parameter SpiPhyUnitSync

Specific if this HwUnit can only do sync transfers.

If true then this hardware unit is dedicated for Synchronous transfers only.

If false then this hardware unit is dedicated for Asynchronous transfers only.

False is applicable only if SpiGeneral/SpiLevelDelivered is either 1 or 2

and true is applicable only if SpiGeneral/SpiLevelDelivered is 0 or 2.

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.169 Parameter SpiSamplePoint

When set, the LPSPI master will sample the input data on a delayed LPSPI_SCK edge, which improves

the setup time when sampling data.

? The input data setup time in master mode with delayed LPSPI_SCK edge is equal to the input data

setup time in slave mode

? In slave mode, the SAMPLE bit is ignored

0b - Input data is sampled on SCK edge

1b - Input data is sampled on delayed SCK edge

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	1
min	0

4.170 Parameter SpiPinConfiguration

Configures which pins are used for input and output data during serial transfers. When performing parallel transfers, the Pin Configuration field is ignored.

00b - SIN is used for input data and SOUT is used for output data

01b - SIN is used for both input and output data, only half-duplex serial transfers are supported

10b - SOUT is used for both input and output data, only half-duplex serial transfers are supported

11b - SOUT is used for input data and SIN is used for output data

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	3
min	0

4.171 Parameter SpiPhyUnitAsyncUseDma

Select Asynchronous mechanism with DMA or not.

NoteThis is an implementation parameter.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.172 Parameter SpiMaxDmaFastTransfer

Number of transfer section allowed in Dma Fast transfer.

Note: This feature will be supported if SpiEnableDmaFastTransferSupport and SpiPhyUnitAsyncUseDma checked.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	1
max	10
min	1

4.173 Reference SpiPhyUnitClockRef

Reference to the SPI clock source configuration, which is set into the MCU driver configuration.

This clock source is used for configure SPI baudrate.

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Mcu/McuModuleConfiguration/McuClockSetting↩ Config/McuClockReferencePoint

4.174 Reference SpiPhyUnitAlternateClockRef

Reference to the alternate clock configuration, retrieved from the MCU plugin.

Use to enable Spi_SetClockMode() function, which allows dual MCU clock configuration settings.

Property	Value
type	ECUC-REFERENCE-DEF
origin	NXP
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destination	/AUTOSAR/EcucDefs/Mcu/McuModuleConfiguration/McuClockSetting↩ Config/McuClockReferencePoint

4.175 Reference SpiPhyTxDmaChannel

SPI Master Transmit DMA Logical Channel as configured by MCL plug-in, used to prepare the SPI transmission dataframes

starting from the TX buffer content.

This parameter is required only if SpiPhyUnitAsyncUseDma is checked.

NoteThis is an implementation specific parameter. The current SPI TX source

needs be configured for enabling this DMA channel.

Property	Value
type	ECUC-CHOICE-REFERENCE-DEF
origin	NXP
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destinations	['/TS_T40D2M10I0R0/Mcl/MclConfig/dmaLogicChannel_Type']

4.176 Reference SpiPhyRxDmaChannel

SPI Receive DMA Logical Channel as configured by MCL plug-in, used to read the deserialized dataframes into the RX buffers.

This parameter is required only if SpiPhyUnitAsyncUseDma is checked.

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-CHOICE-REFERENCE-DEF
origin	NXP
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
	VARIANT-LINK-TIME: PRE-COMPILE

multiplicityConfigClasses

Property	Value
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destinations	['/TS_T40D2M10I0R0/Mcl/MclConfig/dmaLogicChannel_Type']

4.177 Reference SpiFlexioTxChannelConfig

FLEXIO Logical Channel PIN. This selects 1 PIN to setup as Serial Data Output channel

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-CHOICE-REFERENCE-DEF
origin	NXP
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destinations	['/TS_T40D2M10I0R0/Mcl/MclConfig/FlexioCommon/FlexioMclLogicChannels']

4.178 Reference SpiFlexioRxChannelConfig

FLEXIO Logical Channel PIN. This selects 1 PIN to setup as Serial Data Output channel

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-CHOICE-REFERENCE-DEF
origin	NXP

Property	Value
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destinations	['/TS_T40D2M10I0R0/Mcl/MclConfig/FlexioCommon/FlexioMclLogicChannels']

4.179 Reference SpiFlexioClkChannelConfig

FLEXIO Logical Channel PIN. This selects 1 PIN to setup as Serial Data Output channel

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-CHOICE-REFERENCE-DEF
origin	NXP
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destinations	['/TS_T40D2M10I0R0/Mcl/MclConfig/FlexioCommon/FlexioMclLogicChannels']

4.180 Reference SpiFlexioCsChannelConfig

FLEXIO Logical Channel PIN. This selects 1 PIN to setup as Serial Data Output channel

NoteThis is an implementation specific parameter.

Property	Value
type	ECUC-CHOICE-REFERENCE-DEF
origin	NXP
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
requiresSymbolicNameValue	False
destinations	['/TS_T40D2M10I0R0/Mcl/MclConfig/FlexioCommon/FlexioMclLogicChannels']

4.181 Container SpiPublishedInformation

Container holding all SPI specific published information parameters.

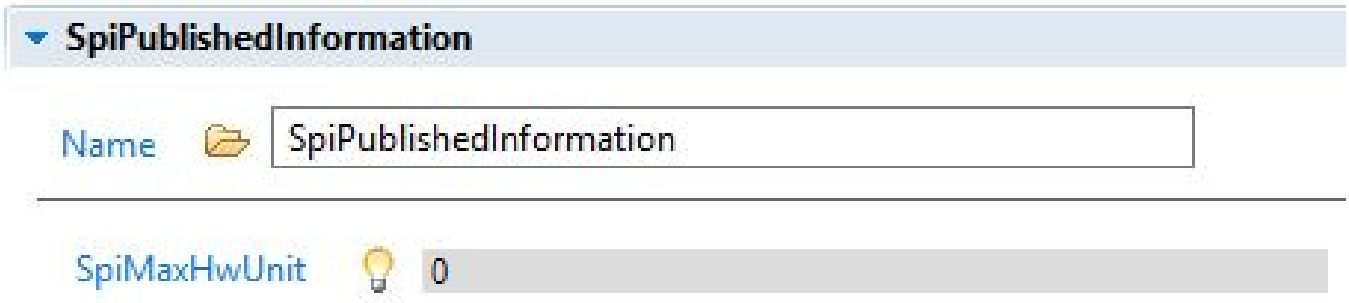

Figure 4.10 Tresos Plugin snapshot for SpiPublishedInformation form.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.182 Parameter SpiMaxHwUnit

Number of different SPI hardware microcontroller peripherals (units/busses) available and handled by this SPI Handler/Driver module.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
defaultValue	0
max	65535
min	0

4.183 Container CommonPublishedInformation

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

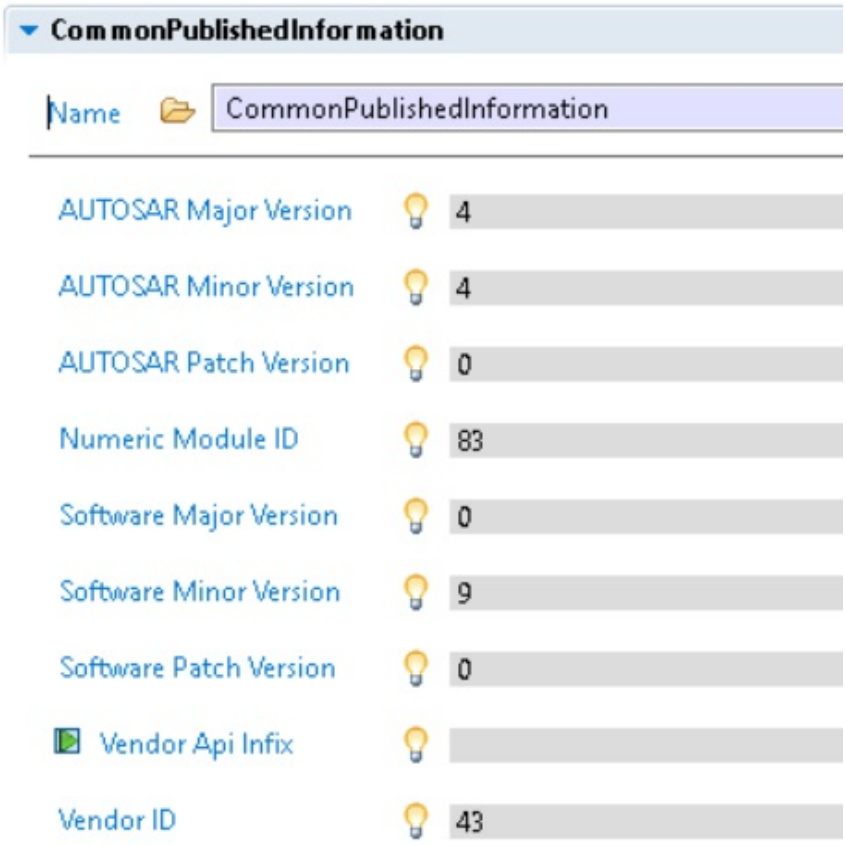

Figure 4.11 Tresos Plugin snapshot for CommonPublishedInformation form.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.184 Parameter ArReleaseMajorVersion

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

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	4
max	4
min	4

4.185 Parameter ArReleaseMinorVersion

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

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	4
max	4
min	4

4.186 Parameter ArReleaseRevisionVersion

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

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	0
min	0

4.187 Parameter ModuleId

Module ID of this module.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	83
max	83
min	83

4.188 Parameter SwMajorVersion

Major version number of the vendor specific implementation of the module. The numbering is vendor specific.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	0
min	0

4.189 Parameter SwMinorVersion

Minor version number of the vendor specific implementation of the module. The numbering is vendor specific.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	9
max	9
min	9

4.190 Parameter SwPatchVersion

Patch level version number of the vendor specific implementation of the module. The numbering is vendor specific.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	0
max	0
min	0

4.191 Parameter VendorApiInfix

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>.

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.

Property	Value
type	ECUC-STRING-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	

4.192 Parameter VendorId

Vendor ID of the dedicated implementation of this module according to the AUTOSAR vendor list.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-POST-BUILD: POST-BUILD
	VARIANT-PRE-COMPILE: PRE-COMPILE
defaultValue	43
max	43
min	43

4.193 Container SpiAutosarExt

Enabling the settings of this section will configure the driver in a mode not compliant with AUTOSAR requirements.

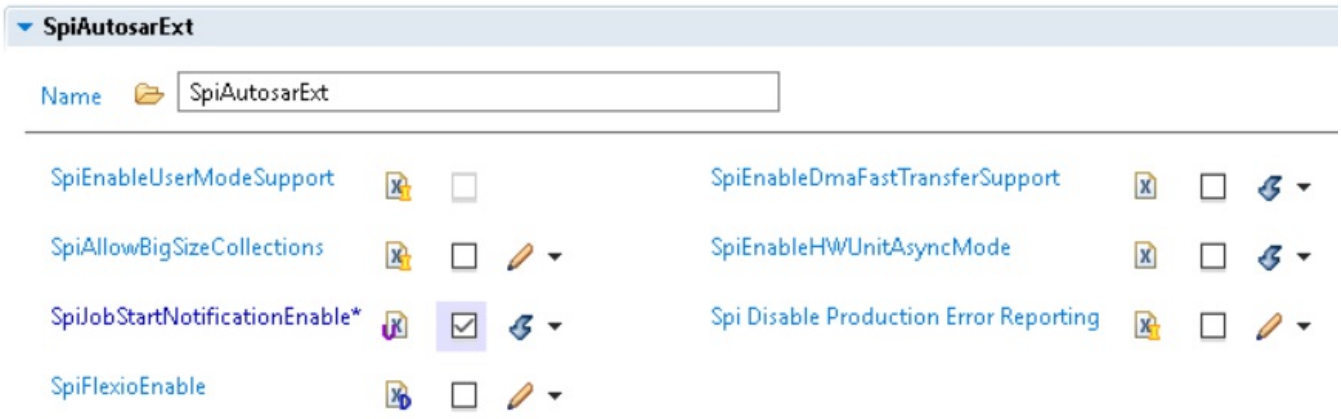

Figure 4.12 Tresos Plugin snapshot for SpiAutosarExt form.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.194 Parameter SpiEnableUserModeSupport

When this parameter is enabled, the Spi module will adapt to run from User Mode.

Note Spi module does not include registers protection. So, It is accessible to all registers in any public mode.

SPI is not affected by this field.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.195 Parameter SpiEnableDmaFastTransferSupport

When this parameter is enabled, the SPI module can support to transfer a Sequence with multiple Channels, Jobs using DMA ScatterGather and CPU used only for processing end of sequence transfer.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1

Property	Value
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.196 Parameter SpiAllowBigSizeCollections

A feature to allow more than 256 sequences, jobs, and channels.

Note Enabling this option will violate the following requirements: SPI166, SPI167, SPI168.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.197 Parameter SpiEnableHWUnitAsyncMode

Enable Spi_SetHWUnitAsyncMode() function, which allows defining distinct operation mode (POLLING or INTERRUPT) for each HWUnit.

Note This feature is not required by Autosar, which defines asynchronous mode configuration at driver level only.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.198 Parameter SpiJobStartNotificationEnable

settings.

Note This feature is a SpiAutosarExt feature to enable the start job notification.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-LINK-TIME: LINK
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.199 Parameter SpiDisableDemReportErrorStatus

SpiDisableDemReportErrorStatus

Switches the Diagnostic Error Reporting and Notification OFF.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1

Property	Value
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

4.200 Parameter SpiFlexioEnable

SpiFlexioEnable

If it is true, FLEXIO feature is enabled

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-LINK-TIME: PRE-COMPILE
	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	false

Chapter 5

Module Index

5.1 Software Specification

Here is a list of all modules:

Chapter 6

Module Documentation

6.1 Flexio_Spi Driver

6.1.1 Detailed Description

6.2 Lpspi Driver

6.2.1 Detailed Description

Data Structures

- struct Lpspi_Ip_DeviceParamsType

  Structure defining some parameters often change of the spi bus. *More...*
- struct Lpspi_Ip_ExternalDeviceType

  Structure defining the parameters of the spi bus. *More...*
- struct Lpspi_Ip_ConfigType

  Structure defining information needed for SPI driver initialization. *More...*
- struct Lpspi_Ip_StateStructureType

  Structure defining information needed for internal state of the driver. *More...*

Macros

- #define SPI_STOP_SEC_CONFIG_DATA_UNSPECIFIED

  Export Post-Build configurations.

Types Reference

- typedef void(∗ Lpspi_Ip_CallbackType) (uint8 Instance, Lpspi_Ip_EventType Event)

  Callback for all peripherals which supports SPI features.

Enum Reference

- enum Lpspi_Ip_EventType

  Enum defining the possible events which triggers end of transfer callback.
- enum Lpspi_Ip_ModeType

  Enum defining the possible transfer modes.
- enum Lpspi_Ip_HwStatusType

  Enum defining the possible states of SPI/DSPI hardware unit.
- enum Lpspi_Ip_StatusType

  Enum defining the possible return types.

Function Reference

- Lpspi_Ip_StatusType Lpspi_Ip_Init (const Lpspi_Ip_ConfigType *PhyUnitConfigPtr)

  LPSPI peripheral initialization.
- Lpspi_Ip_StatusType Lpspi_Ip_DeInit (uint8 Instance)

  LPSPI peripheral deinitialization.
- Lpspi_Ip_StatusType Lpspi_Ip_SyncTransmit (const Lpspi_Ip_ExternalDeviceType *ExternalDevice, uint8 *TxBuffer, uint8 *RxBuffer, uint16 Length, uint32 TimeOut)

  LPSPI synchronous transmission.
- Lpspi_Ip_StatusType Lpspi_Ip_AsyncTransmit (const Lpspi_Ip_ExternalDeviceType *ExternalDevice, uint8 *TxBuffer, uint8 *RxBuffer, uint16 Length, Lpspi_Ip_CallbackType EndCallback)

  LPSPI asynchronous transmission.
- Lpspi_Ip_HwStatusType Lpspi_Ip_GetStatus (uint8 Instance)

  Get status of HW unit.
- void Lpspi_Ip_ManageBuffers (uint8 Instance)

  Process transfer in POLLING mode.
- Lpspi_Ip_StatusType Lpspi_Ip_UpdateFrameSize (const Lpspi_Ip_ExternalDeviceType *ExternalDevice, uint8 FrameSize)

  LPSPI change frame size.
- Lpspi_Ip_StatusType Lpspi_Ip_UpdateLsb (const Lpspi_Ip_ExternalDeviceType *ExternalDevice, boolean Lsb)

  LPSPI change bit order.
- Lpspi_Ip_StatusType Lpspi_Ip_UpdateDefaultTransmitData (const Lpspi_Ip_ExternalDeviceType *ExternalDevice, uint32 DefaultData)

  LPSPI change default transmit data.
- Lpspi_Ip_StatusType Lpspi_Ip_UpdateTransferMode (uint8 Instance, Lpspi_Ip_ModeType Mode)

  LPSPI change transfer mode.
- void Lpspi_Ip_Cancel (uint8 Instance)

  LPSPI cancel current asynchronous transmission.

6.2.2 Data Structure Documentation

6.2.2.1 struct Lpspi_Ip_DeviceParamsType

Structure defining some parameters often change of the spi bus.

Definition at line 162 of file Lpspi_Ip_Types.h.

Data Fields

Type	Name	Description
uint8	FrameSize	Frame size configured
boolean	Lsb	Transfer LSB first or MSB first
uint32	DefaultData	Default data to send when TxBuffer is NULL_PTR

6.2.2.2 struct Lpspi_Ip_ExternalDeviceType

Structure defining the parameters of the spi bus.

Definition at line 173 of file Lpspi_Ip_Types.h.

Data Fields

Type	Name	Description
uint8	Instance	Instance of the hardware unit.
uint32	Ccr	CCR register which contains clocking and frame size configuration.
uint32	Tcr	TCR register which contains clock polarities, frame size, which PCS and continuous mode.
Lpspi_Ip_DeviceParamsType *	DeviceParams	Contain configuration for bit order, frame size, default transmit data.

6.2.2.3 struct Lpspi_Ip_ConfigType

Structure defining information needed for SPI driver initialization.

Definition at line 226 of file Lpspi_Ip_Types.h.

Data Fields

Type	Name	Description
uint8	Instance	Instance of the hardware unit.
uint32	Cr	It contains only debug enable.
uint32	Cfgr1	It contains PCS polarities.
Lpspi_Ip_ModeType	TransferMode	Transfer mode for HWunit
uint8	StateIndex	State of current transfer

6.2.2.4 struct Lpspi_Ip_StateStructureType

Structure defining information needed for internal state of the driver.

Definition at line 252 of file Lpspi_Ip_Types.h.

Data Fields

Type	Name	Description
Lpspi_Ip_ModeType	TransferMode	Store current transfer mode for HWunit
Lpspi_Ip_HwStatusType	Status	0 = available, 1 = busy, 2 = fail due to overflow or underflow

Data Fields

Type	Name	Description
uint8 *	RxBuffer	Store pointer for Rx buffer
uint8 *	TxBuffer	Store pointer for Tx buffer
Lpspi_Ip_CallbackType	Callback	Store pointer for call back function
uint16	RxIndex	Store current Rx index to receive data in Rx buffer
uint16	TxIndex	Store current Tx index to transmit data in Tx buffer
uint16	ExpectedFifoReads	Store number of frames needs to be receive for current transfer
uint16	ExpectedFifoWrites	Store number of frames needs to be transmit for current transfer
boolean	KeepCs	Keep CS signal after tranfers completed.
boolean	FirstCmd	
const Lpspi_Ip_ConfigType *	PhyUnitConfig	
const Lpspi_Ip_ExternalDeviceType *	ExternalDevice	
uint8	TxFrameSize	Store current frame size for HWunit
boolean	TxLsb	Transfer LSB first or MSB first
uint8 *	TxBufferNext	Store pointer for Tx buffer
uint8	FrameSizeNext	Store current frame size for HWunit
boolean	LsbNext	Transfer LSB first or MSB first
uint32	DefaultDataNext	Default data to send when TxBuffer is NULL_PTR
uint16	LengthNext	Store number of frames needs to be transmit for current transfer
boolean	NextTransferConfigAvailable	Flag to check next transfer configuration is available
boolean	NextTransferDone	Flag to check next transfer done
uint8	CurrentTxFifoSlot	Number of TX FIFO slots are current available.
boolean	TxDoneFlag	Flag to check TX is done

6.2.3 Macro Definition Documentation

6.2.3.1 SPI_STOP_SEC_CONFIG_DATA_UNSPECIFIED

```
#define SPI_STOP_SEC_CONFIG_DATA_UNSPECIFIED
```

Export Post-Build configurations.

Definition at line 112 of file Lpspi_Ip.h.

6.2.4 Types Reference

6.2.4.1 Lpspi_Ip_CallbackType

```
typedef void(* Lpspi_Ip_CallbackType) (uint8 Instance, Lpspi_Ip_EventType Event)
```

Callback for all peripherals which supports SPI features.

Definition at line 116 of file Lpspi_Ip_Types.h.

6.2.5 Enum Reference

6.2.5.1 Lpspi_Ip_EventType

```
enum Lpspi_Ip_EventType
```

Enum defining the possible events which triggers end of transfer callback.

Enumerator

LPSPI_IP_EVENT_END_TRANSFER	The transfer is successfully done.
LPSPI_IP_EVENT_FAULT	The transfer failed due to overflow/underflow.

Definition at line 109 of file Lpspi_Ip_Types.h.

6.2.5.2 Lpspi_Ip_ModeType

```
enum Lpspi_Ip_ModeType
```

Enum defining the possible transfer modes.

Enumerator

LPSPI_IP_POLLING	For polling mode the application must call periodically Spi_Ip_ManageBuffers after asynchronous transfers.
LPSPI_IP_INTERRUPT	For interrupt mode the application doesn't need to perform any additional operations after asynchronous transfers. The application must enable the interrupt requests and install the right callbacks.

Definition at line 121 of file Lpspi_Ip_Types.h.

6.2.5.3 Lpspi_Ip_HwStatusType

```
enum Lpspi_Ip_HwStatusType
```

Enum defining the possible states of SPI/DSPI hardware unit.

Enumerator

LPSPI_IP_UNINIT	Module is not initialized.
LPSPI_IP_IDLE	Module is not used.
LPSPI_IP_BUSY	A transfer is in progress.
LPSPI_IP_FAULT	During last transfer a fault occurred.

Definition at line 131 of file Lpspi_Ip_Types.h.

6.2.5.4 Lpspi_Ip_StatusType

```
enum Lpspi_Ip_StatusType
```

Enum defining the possible return types.

Enumerator

LPSPI_IP_STATUS_SUCCESS	Successful operation.
LPSPI_IP_STATUS_FAIL	Failed operation.
LPSPI_IP_FIFO_ERROR	Overflow or underflow error.
LPSPI_IP_TIMEOUT	Timeout error.

Definition at line 142 of file Lpspi_Ip_Types.h.

6.2.6 Function Reference

6.2.6.1 Lpspi_Ip_Init()

```
Lpspi_Ip_StatusType Lpspi_Ip_Init (
            const Lpspi_Ip_ConfigType * PhyUnitConfigPtr )
```

LPSPI peripheral initialization.

The function initialize the SPI Unit specified in the configuration.

Parameters

in	*PhyUnitConfigPtr*	- pointer to the specified SPI Unit configuration.

Returns

LPSPI_IP_STATUS_SUCCESS: Initialization command has been accepted. LPSPI_IP_STATUS_FAIL: Initialization command has not been accepted.

6.2.6.2 Lpspi_Ip_DeInit()

```
Lpspi_Ip_StatusType Lpspi_Ip_DeInit (
            uint8 Instance )
```

LPSPI peripheral deinitialization.

The function de-initialize the SPI peripheral instance specified. All registers of SPI peripheral will be reset.

Parameters

in	*Instance*	- SPI peripheral instance number.

Returns

LPSPI_IP_STATUS_SUCCESS: De-initialization command has been accepted. LPSPI_IP_STATUS_F↩ AIL: De-initialization command has not been accepted.

6.2.6.3 Lpspi_Ip_SyncTransmit()

```
Lpspi_Ip_StatusType Lpspi_Ip_SyncTransmit (
            const Lpspi_Ip_ExternalDeviceType * ExternalDevice,
            uint8 * TxBuffer,
            uint8 * RxBuffer,
            uint16 Length,
            uint32 TimeOut )
```

LPSPI synchronous transmission.

This function initializes a synchronous transfer using the bus parameters provided by external device.

Parameters

in	*ExternalDevice*	- pointer to the external device where data is transmitted.
in	*TxBuffer*	- pointer to transmit buffer.
	[in-out]	RxBuffer - pointer to receive buffer.
in	*Length*	- number of bytes to be sent.
in	*TimeOut*	- duration for sending one frame.

Returns

LPSPI_IP_STATUS_SUCCESS: Transmission command has been accepted. LPSPI_IP_FIFO_ERROR↩ : Overflow or underflow error occurred. LPSPI_IP_STATUS_FAIL: Transmission command has not been accepted. LPSPI_IP_TIMEOUT: Timeout error occurred.

6.2.6.4 Lpspi_Ip_AsyncTransmit()

```
Lpspi_Ip_StatusType Lpspi_Ip_AsyncTransmit (
            const Lpspi_Ip_ExternalDeviceType * ExternalDevice,
            uint8 * TxBuffer,
            uint8 * RxBuffer,
            uint16 Length,
            Lpspi_Ip_CallbackType EndCallback )
```

LPSPI asynchronous transmission.

This function initializes an asynchronous transfer using the bus parameters provided by external device.

Parameters

in	*ExternalDevice*	- pointer to the external device where data is transmitted
in	*TxBuffer*	- pointer to transmit buffer.
	[in-out]	RxBuffer - pointer to receive buffer.
in	*Length*	- number of bytes to be sent.
in	*EndCallback*	- callback function is called at the end of transfer.

Returns

LPSPI_IP_STATUS_SUCCESS: Transmission command has been accepted. LPSPI_IP_STATUS_FAIL: Transmission command has not been accepted.

6.2.6.5 Lpspi_Ip_GetStatus()

```
Lpspi_Ip_HwStatusType Lpspi_Ip_GetStatus (
            uint8 Instance )
```

Get status of HW unit.

This function returns the status of the specified SPI Hardware microcontroller peripheral.

Parameters

in	*Instance*	- SPI peripheral instance number.

Returns

Lpspi_Ip_HwStatusType

6.2.6.6 Lpspi_Ip_ManageBuffers()

```
void Lpspi_Ip_ManageBuffers (
            uint8 Instance )
```

Process transfer in POLLING mode.

This function shall polls the SPI interrupts linked to SPI peripheral instance allocated to the transmission of data to enable the evolution of transmission state machine.

Parameters

in	*Instance*	- SPI peripheral instance number.

Returns

void

6.2.6.7 Lpspi_Ip_UpdateFrameSize()

```
Lpspi_Ip_StatusType Lpspi_Ip_UpdateFrameSize (
            const Lpspi_Ip_ExternalDeviceType * ExternalDevice,
            uint8 FrameSize )
```

LPSPI change frame size.

This function updates frame size of specific external device configuration for next transfers.

Parameters

in	*ExternalDevice*	- pointer to the external device configuration.
in	*FrameSize*	- Frame size.

Returns

LPSPI_IP_STATUS_SUCCESS: Setting command has been accepted. LPSPI_IP_STATUS_FAIL: Setting command has not been accepted.

6.2.6.8 Lpspi_Ip_UpdateLsb()

```
Lpspi_Ip_StatusType Lpspi_Ip_UpdateLsb (
            const Lpspi_Ip_ExternalDeviceType * ExternalDevice,
            boolean Lsb )
```

LPSPI change bit order.

This function updates bits order LSB or MSB of specific external device configuration for next transfer.

Parameters

in	*ExternalDevice*	- pointer to the external device configuration.
in	*Lsb*	- Data is transferred LSB first or not.

Returns

LPSPI_IP_STATUS_SUCCESS: Setting command has been accepted. LPSPI_IP_STATUS_FAIL: Setting command has not been accepted.

6.2.6.9 Lpspi_Ip_UpdateDefaultTransmitData()

```
Lpspi_Ip_StatusType Lpspi_Ip_UpdateDefaultTransmitData (
            const Lpspi_Ip_ExternalDeviceType * ExternalDevice,
            uint32 DefaultData )
```

LPSPI change default transmit data.

This function updates default transmit data of specific external device configuration for next transfer.

Parameters

in	*ExternalDevice*	- pointer to the external device configuration.
in	*DefaultData*	- New default transmit data.

Returns

LPSPI_IP_STATUS_SUCCESS: Setting command has been accepted. LPSPI_IP_STATUS_FAIL: Setting command has not been accepted.

6.2.6.10 Lpspi_Ip_UpdateTransferMode()

```
Lpspi_Ip_StatusType Lpspi_Ip_UpdateTransferMode (
            uint8 Instance,
            Lpspi_Ip_ModeType Mode )
```

LPSPI change transfer mode.

This function updates the asynchronous mechanism mode for the specified SPI Hardware microcontroller peripheral.

Parameters

in	*Instance*	- SPI peripheral instance number.
in	*Mode*	- new mode (interrupt or polling).

Returns

LPSPI_IP_STATUS_SUCCESS: Setting command has been accepted. LPSPI_IP_STATUS_FAIL: Setting command has not been accepted.

6.2.6.11 Lpspi_Ip_Cancel()

```
void Lpspi_Ip_Cancel (
            uint8 Instance )
```

LPSPI cancel current asynchronous transmission.

This function cancels an asynchronous transmission in progress for the specified SPI Hardware microcontroller peripheral.

Parameters

in	*Instance*	- SPI peripheral instance number.

Returns

void

6.3 Spi Driver

6.3.1 Detailed Description

Data Structures

- struct Spi_SequenceConfigType

  This structure contains all the needed data to configure one SPI Sequence. *More...*
- struct Spi_SeqsConfigType

  This structure contains Sequence configuration. *More...*
- struct Spi_SequenceStateType

  Internal structure used to manage the sequence state. *More...*
- struct Spi_JobStateType

  Internal structure used to manage the job state. *More...*
- struct Spi_JobConfigType

  This is the structure containing all the parameters needed to completely define a Job. *More...*
- struct Spi_JobsCfgType

  This is the structure containing Job configuration. *More...*
- struct Spi_BufferDescriptorType

  The structure contains the pointers to the Tx/Rx memory locations for the given buffer (IB or EB). *More...*
- struct Spi_ChannelStateType

  Internal structure used to manage the channel state. *More...*
- struct Spi_HWUnitQueue

  This structure holds the HWUnit scheduling queue. *More...*
- struct Spi_ChannelConfigType

  The structure contains the channel configuration parameters. *More...*
- struct Spi_ChannelsCfgType

  The structure contains the channel configuration. *More...*
- struct Spi_ConfigType

  This is the top level structure containing all the needed parameters for the SPI Handler Driver. *More...*
- struct Spi_Ipw_IpConfigType
- struct Spi_Ipw_ExternalDeviceType
- struct Spi_HWUnitConfigType

  This structure holds the HWUnit configuration parameters. *More...*
- struct Spi_PhyUnitsConfigType

  This structure holds the PhyUnit configuration. *More...*
- struct Spi_ExternalDeviceConfigType

  This structure holds the configuration parameters for each ExternalDevice. *More...*
- struct Spi_ExDevicesConfigType

  This structure holds the ExternalDevice configuration. *More...*

Macros

- #define SPI_PHYUNIT_SYNC_U32

  Define state of hardware unit for synchronous transmission.
- #define SPI_E_PARAM_CHANNEL

  API service called with wrong parameter of Channel.
- #define SPI_E_PARAM_JOB

  API service called with wrong parameter of Job.
- #define SPI_E_PARAM_SEQ

  API service called with wrong parameter of Sequence.
- #define SPI_E_PARAM_LENGTH

  API service called with wrong parameter of external buffer length.
- #define SPI_E_PARAM_UNIT

  API service called with wrong parameter of HWUnit.
- #define SPI_E_PARAM_CONFIG

  API service called with wrong resource assigned.
- #define SPI_E_UNINIT

  API service used without module initialization.
- #define SPI_E_SEQ_PENDING

  Services called in a wrong sequence.
- #define SPI_E_SEQ_IN_PROCESS

  Synchronous transmission service called at wrong time.
- #define SPI_E_ALREADY_INITIALIZED

  API SPI_Init service called while the SPI driver has already been initialized.
- #define SPI_E_CONFIG_OUT_OF_RANGE

  The number of sequences, jobs or channels exceeds precompile time sizes.
- #define SPI_E_INIT_FAILED

  API Spi_Init was called with wrong configuration pointer.
- #define SPI_E_PARAM_EB_UNIT

  When a sequence contains uninitialized external buffers.
- #define SPI_E_SEQ_EMPTY

  No job in sequence.
- #define SPI_E_JOB_EMPTY

  No channel in job.
- #define SPI_E_PARAM_POINTER

  If the parameter versioninfo or Spi configuration is NULL_PTR.
- #define SPI_INIT_ID

  API service ID for SPI Init function.
- #define SPI_DEINIT_ID

  API service ID for SPI DeInit function.
- #define SPI_WRITEIB_ID

  API service ID for SPI write IB function.
- #define SPI_ASYNCTRANSMIT_ID

  API service ID for SPI async transmit function.
- #define SPI_READIB_ID

  API service ID for SPI read IB function.
- #define SPI_SETUPEB_ID

API service ID for SPI setup EB function.

- #define SPI_GETSTATUS_ID

  API service ID for SPI get status function.
- #define SPI_GETJOBRESULT_ID

  API service ID for SPI get job result function.
- #define SPI_GETSEQUENCERESULT_ID

  API service ID for SPI get sequence result function.
- #define SPI_GETVERSIONINFO_ID

  API service ID for SPI get version info function.
- #define SPI_SYNCTRANSMIT_ID

  API service ID for SPI sync transmit function.
- #define SPI_GETHWUNITSTATUS_ID

  API service ID for SPI get hwunit status function.
- #define SPI_CANCEL_ID

  API service ID for SPI cancel function.
- #define SPI_SETASYNCMODE_ID

  API service ID for SPI set async mode function.
- #define SPI_MAINFUNCTION_HANDLING_ID

  API service ID for SPI main function.
- #define SPI_SETHWUNITASYNCMODE_ID

  API service ID for SPI set HW Unit async mode.
- #define SPI_SETCLOCKMODE_ID

  API service ID for SPI Set Clock Mode.
- #define SPI_JOB_PRIORITY_LEVELS_COUNT

  The number of allowed job priority levels (0..3).
- #define SPI_STOP_SEC_CONFIG_DATA_UNSPECIFIED

  Export Post-Build configurations.

Types Reference

- typedef uint8 Spi_DataBufferType

  Type of application data buffer elements.
- typedef uint16 Spi_NumberOfDataType

  Type for defining the number of data elements of the type Spi_DataBufferType.
- typedef uint8 Spi_ChannelType

  Specifies the identification (ID) for a Channel.
- typedef uint16 Spi_JobType

  Specifies the identification (ID) for a Job.
- typedef uint8 Spi_SequenceType

  Specifies the identification (ID) for a sequence of jobs.
- typedef uint8 Spi_HWUnitType

  Specifies the ID for a SPI Hardware microcontroller peripheral unit.
- typedef uint8 Spi_ExternalDeviceType

  Contains the ID of an external device.

Enum Reference

- enum Spi_StatusType
  This type defines a range of specific status for SPI Driver.
- enum Spi_JobResultType
  This type defines a range of specific Jobs status for SPI Driver.
- enum Spi_SeqResultType
  This type defines a range of specific Sequences status for SPI Driver.
- enum Spi_BufferType
  The enumeration containing the designated values for buffer types (internal or external).
- enum Spi_AsyncModeType
  Specifies the asynchronous mechanism mode for SPI buses handled asynchronously in Level 2.
- enum Spi_Ipw_SupportedIpsType
  This enum contains all IPs which can integrate SPI functionalityes.

Function Reference

- void Spi_JobTransferFinished (const Spi_JobConfigType *JobConfig, Spi_JobResultType JobResult)
  This function is called after a Job has been executed.
- void Spi_Init (const Spi_ConfigType *ConfigPtr)
  This function initializes the SPI driver.
- Std_ReturnType Spi_DeInit (void)
  This function de-initializes the SPI driver.
- Std_ReturnType Spi_WriteIB (Spi_ChannelType Channel, const Spi_DataBufferType *DataBufferPtr)
  This function writes the given data into the buffer of a specific channel.
- Std_ReturnType Spi_ReadIB (Spi_ChannelType Channel, Spi_DataBufferType *DataBufferPointer)
  This function reads the data from the buffer of a channel and puts at the memory location.
- Std_ReturnType Spi_AsyncTransmit (Spi_SequenceType Sequence)
  This function triggers the asynchronous transmission for the given sequence.
- Std_ReturnType Spi_SetupEB (Spi_ChannelType Channel, Spi_DataBufferType *SrcDataBufferPtr, Spi_DataBufferType *DesDataBufferPtr, Spi_NumberOfDataType Length)
  This function setup an external buffer to be used by a specific channel.
- Spi_StatusType Spi_GetStatus (void)
  This function returns the status of the SPI driver.
- Spi_JobResultType Spi_GetJobResult (Spi_JobType Job)
  This function is used to request the status of a specific job.
- Spi_SeqResultType Spi_GetSequenceResult (Spi_SequenceType Sequence)
  This function is used to request the status of a specific sequence.
- Std_ReturnType Spi_SyncTransmit (Spi_SequenceType Sequence)
  This function is used for synchronous transmission of a given sequence.
- Std_ReturnType Spi_SetAsyncMode (Spi_AsyncModeType Mode)
  This function specifies the asynchronous mode for the SPI busses handled asynchronously.

Variables

- Spi_JobStateType Spi_axSpiJobState [SPI_MAX_JOB]
  Extern arrays contain the state of Sequences, Jobs and Channels.

6.3.2 Data Structure Documentation

6.3.2.1 struct Spi_SequenceConfigType

This structure contains all the needed data to configure one SPI Sequence.

Definition at line 567 of file Spi.h.

Data Fields

- Spi_JobType NumJobs
  Number of jobs in the sequence.
- uint32 SpiCoreUse
  CoreID used.
- const Spi_JobType * JobIndexList
  Job index list.
- Spi_NotifyType * EndNotification
  Job notification handler.
- uint8 Interruptible
  Boolean indicating if the Sequence is interruptible or not.

6.3.2.1.1 Field Documentation

6.3.2.1.1.1 NumJobs `Spi_JobType NumJobs`

Number of jobs in the sequence.

Definition at line 570 of file Spi.h.

6.3.2.1.1.2 SpiCoreUse `uint32 SpiCoreUse`

CoreID used.

Definition at line 572 of file Spi.h.

6.3.2.1.1.3 JobIndexList `const Spi_JobType* JobIndexList`

Job index list.

Definition at line 574 of file Spi.h.

6.3.2.1.1.4 EndNotification `Spi_NotifyType* EndNotification`

Job notification handler.

Definition at line 576 of file Spi.h.

6.3.2.1.1.5 Interruptible `uint8 Interruptible`

Boolean indicating if the Sequence is interruptible or not.

Definition at line 578 of file Spi.h.

6.3.2.2 struct Spi_SeqsConfigType

This structure contains Sequence configuration.

Definition at line 588 of file Spi.h.

Data Fields

Type	Name	Description
const Spi_SequenceConfigType *	SeqConfig	Point to Sequence configuration.

6.3.2.3 struct Spi_SequenceStateType

Internal structure used to manage the sequence state.

Definition at line 599 of file Spi.h.

Data Fields

- Spi_SeqResultType Result
  Sequence Result.
- const Spi_SequenceConfigType * Sequence
  Pointer to the configuration.
- const Spi_JobType * CurrentJobIndexPointer
  Position in JobIndexList to the job in transmission of an async sequence.
- Spi_JobType RemainingJobs
  Number of jobs in a pending async sequence, not yet transmitted.

6.3.2.3.1 Field Documentation

6.3.2.3.1.1 Result Spi_SeqResultType Result

Sequence Result.

Definition at line 602 of file Spi.h.

6.3.2.3.1.2 Sequence const Spi_SequenceConfigType* Sequence

Pointer to the configuration.

Definition at line 605 of file Spi.h.

6.3.2.3.1.3 CurrentJobIndexPointer const Spi_JobType* CurrentJobIndexPointer

Position in JobIndexList to the job in transmission of an async sequence.

Definition at line 607 of file Spi.h.

6.3.2.3.1.4 RemainingJobs Spi_JobType RemainingJobs

Number of jobs in a pending async sequence, not yet transmitted.

Definition at line 609 of file Spi.h.

6.3.2.4 struct Spi_JobStateType

Internal structure used to manage the job state.

Definition at line 617 of file Spi.h.

Data Fields

Type	Name	Description
Spi_JobResultType	Result	Job Result.
Spi_SequenceStateType *	AsyncCrtSequenceState	Pointer to the state information of the async sequence.
Spi_JobType	AsyncNextJob	Pointer to the next async job planned for transmission.

6.3.2.5 struct Spi_JobConfigType

This is the structure containing all the parameters needed to completely define a Job.

Definition at line 635 of file Spi.h.

Data Fields

- Spi_ChannelType NumChannels
  Number of channels in the job.
- const Spi_ChannelType * ChannelIndexList
  Channel index list.
- Spi_NotifyType * EndNotification
  Job end notification.
- Spi_NotifyType * StartNotification
  Job start notification.
- sint8 Priority
  Priority.
- uint32 SpiCoreUse
  CoreID used.
- Spi_JobStateType * JobState
  Implementation specific field referencing the channel internal state.
- Spi_HWUnitType HWUnit
  HWUnit.
- Spi_ExternalDeviceType ExternalDevice
  ExternalDevice.
- const Spi_ExDevicesConfigType * ExternalDeviceConfig
  Implementation specific field: cached LLD device attributes.

6.3.2.5.1 Field Documentation

6.3.2.5.1.1 NumChannels `Spi_ChannelType NumChannels`

Number of channels in the job.

Definition at line 638 of file Spi.h.

6.3.2.5.1.2 ChannelIndexList `const Spi_ChannelType* ChannelIndexList`

Channel index list.

Definition at line 640 of file Spi.h.

6.3.2.5.1.3 EndNotification `Spi_NotifyType* EndNotification`

Job end notification.

Definition at line 642 of file Spi.h.

6.3.2.5.1.4 StartNotification `Spi_NotifyType* StartNotification`

Job start notification.

Definition at line 644 of file Spi.h.

6.3.2.5.1.5 Priority `sint8 Priority`

Priority.

Definition at line 646 of file Spi.h.

6.3.2.5.1.6 SpiCoreUse `uint32 SpiCoreUse`

CoreID used.

Definition at line 648 of file Spi.h.

6.3.2.5.1.7 JobState `Spi_JobStateType* JobState`

Implementation specific field referencing the channel internal state.

Definition at line 650 of file Spi.h.

6.3.2.5.1.8 HWUnit `Spi_HWUnitType HWUnit`

HWUnit.

Definition at line 652 of file Spi.h.

6.3.2.5.1.9 ExternalDevice `Spi_ExternalDeviceType ExternalDevice`

ExternalDevice.

Definition at line 654 of file Spi.h.

6.3.2.5.1.10 ExternalDeviceConfig `const Spi_ExDevicesConfigType* ExternalDeviceConfig`

Implementation specific field: cached LLD device attributes.

Definition at line 656 of file Spi.h.

6.3.2.6 struct Spi_JobsCfgType

This is the structure containing Job configuration.

Definition at line 662 of file Spi.h.

Data Fields

Type	Name	Description
const Spi_JobConfigType *	JobCfg	Point to Job configuration.

6.3.2.7 struct Spi_BufferDescriptorType

The structure contains the pointers to the Tx/Rx memory locations for the given buffer (IB or EB).

Definition at line 672 of file Spi.h.

Data Fields

Type	Name	Description
Spi_DataBufferType *	BufferTX	Transmit buffer pointer.
Spi_DataBufferType *	BufferRX	Receive buffer pointer.

6.3.2.8 struct Spi_ChannelStateType

Internal structure used to manage the channel state.

Definition at line 684 of file Spi.h.

Data Fields

Type	Name	Description
uint8	Flags	Default Transmit Enabled.
Spi_NumberOfDataType	Length	Actual Transfer size for EB.

6.3.2.9 struct Spi_HWUnitQueue

This structure holds the HWUnit scheduling queue.

For async transmissions, this structure holds the HWUnit scheduling queue . For sync transmissions, only HWUnit Status is managed.

Definition at line 696 of file Spi.h.

Data Fields

Type	Name	Description
Spi_JobType	ScheduledJobsListHead[(4)]	Array of the IDs of jobs to be scheduled, for each priority level.
Spi_JobType	ScheduledJobsListTail[(4)]	Array of the IDs of last jobs in queues, for each priority level.
sint8	MaxScheduledPriority	Array of the IDs of last jobs in queues, for each priority level.
Spi_StatusType	Status	DSPI state.
Spi_ChannelType	Channel	Current channel index in Job
Spi_JobType	Job	Current job index

6.3.2.10 struct Spi_ChannelConfigType

The structure contains the channel configuration parameters.

Definition at line 716 of file Spi.h.

Data Fields

- Spi_BufferType BufferType
  Buffer Type IB/EB.
- uint8 FrameSize
  Data frame size.
- boolean Lsb
  Bite order (MSB/LSB).
- uint32 DefaultTransmitValue
  Default Transmit Value.
- Spi_NumberOfDataType Length
  Data length.

- Spi_BufferDescriptorType * BufferDescriptor
  Buffer Descriptor.
- uint32 SpiCoreUse
  CoreID assigned.
- Spi_ChannelStateType * ChannelState
  Implementation specific field referencing the channel internal state.

6.3.2.10.1 Field Documentation

6.3.2.10.1.1 BufferType `Spi_BufferType BufferType`

Buffer Type IB/EB.

Definition at line 719 of file Spi.h.

6.3.2.10.1.2 FrameSize `uint8 FrameSize`

Data frame size.

Definition at line 721 of file Spi.h.

6.3.2.10.1.3 Lsb `boolean Lsb`

Bite order (MSB/LSB).

Definition at line 723 of file Spi.h.

6.3.2.10.1.4 DefaultTransmitValue `uint32 DefaultTransmitValue`

Default Transmit Value.

Definition at line 731 of file Spi.h.

6.3.2.10.1.5 Length `Spi_NumberOfDataType Length`

Data length.

Definition at line 733 of file Spi.h.

6.3.2.10.1.6 BufferDescriptor `Spi_BufferDescriptorType* BufferDescriptor`

Buffer Descriptor.

Definition at line 735 of file Spi.h.

6.3.2.10.1.7 SpiCoreUse `uint32 SpiCoreUse`

CoreID assigned.

Definition at line 737 of file Spi.h.

6.3.2.10.1.8 ChannelState `Spi_ChannelStateType* ChannelState`

Implementation specific field referencing the channel internal state.

Definition at line 739 of file Spi.h.

6.3.2.11 struct Spi_ChannelsCfgType

The structure contains the channel configuration.

Definition at line 745 of file Spi.h.

Data Fields

Type	Name	Description
const Spi_ChannelConfigType *	ChannelCfg	Point to Channel configuration.

6.3.2.12 struct Spi_ConfigType

This is the top level structure containing all the needed parameters for the SPI Handler Driver.

Definition at line 765 of file Spi.h.

Data Fields

- uint16 MaxExternalDevice
  Number of external devices defined in the configuration.
- Spi_ChannelType SpiMaxChannel

Number of channels defined in the configuration.

- Spi_JobType SpiMaxJob

  Number of jobs defined in the configuration.
- Spi_SequenceType SpiMaxSequence

  Number of sequences defined in the configuration.
- uint32 SpiCoreUse

  CoreID used.
- const Spi_ChannelsCfgType * ChannelConfig

  Pointer to Array of channels defined in the configuration.
- const Spi_JobsCfgType * JobConfig

  Pointer to Array of jobs defined in the configuration.
- const Spi_SeqsConfigType * SequenceConfig

  Pointer to Array of sequences defined in the configuration.
- const Spi_ExDevicesConfigType * ExternalDeviceConfig

  External device unit attributes.
- const Spi_PhyUnitsConfigType * HWUnitConfig

  Pointer to Array of LLD DSPI device instances.
- const Mcal_DemErrorType SpiErrorHardwareCfg

  SPI Driver DEM Error: SPI_E_HARDWARE_ERROR.

6.3.2.12.1 Field Documentation

6.3.2.12.1.1 MaxExternalDevice `uint16 MaxExternalDevice`

Number of external devices defined in the configuration.

Definition at line 768 of file Spi.h.

6.3.2.12.1.2 SpiMaxChannel `Spi_ChannelType SpiMaxChannel`

Number of channels defined in the configuration.

Definition at line 770 of file Spi.h.

6.3.2.12.1.3 SpiMaxJob `Spi_JobType SpiMaxJob`

Number of jobs defined in the configuration.

Definition at line 772 of file Spi.h.

6.3.2.12.1.4 SpiMaxSequence `Spi_SequenceType SpiMaxSequence`

Number of sequences defined in the configuration.

Definition at line 774 of file Spi.h.

6.3.2.12.1.5 SpiCoreUse `uint32 SpiCoreUse`

CoreID used.

Definition at line 776 of file Spi.h.

6.3.2.12.1.6 ChannelConfig `const Spi_ChannelsCfgType* ChannelConfig`

Pointer to Array of channels defined in the configuration.

Definition at line 778 of file Spi.h.

6.3.2.12.1.7 JobConfig `const Spi_JobsCfgType* JobConfig`

Pointer to Array of jobs defined in the configuration.

Definition at line 780 of file Spi.h.

6.3.2.12.1.8 SequenceConfig `const Spi_SeqsConfigType* SequenceConfig`

Pointer to Array of sequences defined in the configuration.

Definition at line 782 of file Spi.h.

6.3.2.12.1.9 ExternalDeviceConfig `const Spi_ExDevicesConfigType* ExternalDeviceConfig`

External device unit attributes.

Definition at line 784 of file Spi.h.

6.3.2.12.1.10 HWUnitConfig `const Spi_PhyUnitsConfigType* HWUnitConfig`

Pointer to Array of LLD DSPI device instances.

Definition at line 786 of file Spi.h.

6.3.2.12.1.11 SpiErrorHardwareCfg `const Mcal_DemErrorType SpiErrorHardwareCfg`

SPI Driver DEM Error: SPI_E_HARDWARE_ERROR.

Definition at line 789 of file Spi.h.

6.3.2.13 struct Spi_Ipw_IpConfigType

@bried This union contains config structure for all IPs available.

Definition at line 147 of file Spi_Ipw_Types.h.

6.3.2.14 struct Spi_Ipw_ExternalDeviceType

@bried This union contains config structure for all external device available.

Definition at line 163 of file Spi_Ipw_Types.h.

6.3.2.15 struct Spi_HWUnitConfigType

This structure holds the HWUnit configuration parameters.

Definition at line 176 of file Spi_Ipw_Types.h.

6.3.2.16 struct Spi_PhyUnitsConfigType

This structure holds the PhyUnit configuration.

Definition at line 188 of file Spi_Ipw_Types.h.

Data Fields

Type	Name	Description
const Spi_HWUnitConfigType *	PhyUnitConfig	Point to PhyUnit configuration.

6.3.2.17 struct Spi_ExternalDeviceConfigType

This structure holds the configuration parameters for each ExternalDevice.

Definition at line 197 of file Spi_Ipw_Types.h.

6.3.2.18 struct Spi_ExDevicesConfigType

This structure holds the ExternalDevice configuration.

Definition at line 208 of file Spi_Ipw_Types.h.

Data Fields

Type	Name	Description
const Spi_ExternalDeviceConfigType *	ExDeviceConfig	Point to ExternalDevice configuration.

6.3.3 Macro Definition Documentation

6.3.3.1 SPI_PHYUNIT_SYNC_U32

```
#define SPI_PHYUNIT_SYNC_U32
```

Define state of hardware unit for synchronous transmission.

Definition at line 129 of file Spi.h.

6.3.3.2 SPI_E_PARAM_CHANNEL

```
#define SPI_E_PARAM_CHANNEL
```

API service called with wrong parameter of Channel.

Definition at line 137 of file Spi.h.

6.3.3.3 SPI_E_PARAM_JOB

```
#define SPI_E_PARAM_JOB
```

API service called with wrong parameter of Job.

Definition at line 143 of file Spi.h.

6.3.3.4 SPI_E_PARAM_SEQ

```
#define SPI_E_PARAM_SEQ
```

API service called with wrong parameter of Sequence.

Definition at line 149 of file Spi.h.

6.3.3.5 SPI_E_PARAM_LENGTH

```
#define SPI_E_PARAM_LENGTH
```

API service called with wrong parameter of external buffer length.

Definition at line 155 of file Spi.h.

6.3.3.6 SPI_E_PARAM_UNIT

```
#define SPI_E_PARAM_UNIT
```

API service called with wrong parameter of HWUnit.

Definition at line 161 of file Spi.h.

6.3.3.7 SPI_E_PARAM_CONFIG

```
#define SPI_E_PARAM_CONFIG
```

API service called with wrong resource assigned.

Definition at line 167 of file Spi.h.

6.3.3.8 SPI_E_UNINIT

```
#define SPI_E_UNINIT
```

API service used without module initialization.

Definition at line 173 of file Spi.h.

6.3.3.9 SPI_E_SEQ_PENDING

```
#define SPI_E_SEQ_PENDING
```

Services called in a wrong sequence.

Definition at line 179 of file Spi.h.

6.3.3.10 SPI_E_SEQ_IN_PROCESS

```
#define SPI_E_SEQ_IN_PROCESS
```

Synchronous transmission service called at wrong time.

Definition at line 185 of file Spi.h.

6.3.3.11 SPI_E_ALREADY_INITIALIZED

```
#define SPI_E_ALREADY_INITIALIZED
```

API SPI_Init service called while the SPI driver has already been initialized.

Definition at line 191 of file Spi.h.

6.3.3.12 SPI_E_CONFIG_OUT_OF_RANGE

```
#define SPI_E_CONFIG_OUT_OF_RANGE
```

The number of sequences, jobs or channels exceeds precompile time sizes.

The number of sequences, jobs or channels in the configuration exceeds precompile time related sizes: SPI_MAX↩ _SEQUENCE, SPI_MAX_JOB or SPI_MAX_CHANNEL.

Definition at line 201 of file Spi.h.

6.3.3.13 SPI_E_INIT_FAILED

```
#define SPI_E_INIT_FAILED
```

API Spi_Init was called with wrong configuration pointer.

Definition at line 208 of file Spi.h.

6.3.3.14 SPI_E_PARAM_EB_UNIT

```
#define SPI_E_PARAM_EB_UNIT
```

When a sequence contains uninitialized external buffers.

Definition at line 215 of file Spi.h.

6.3.3.15 SPI_E_SEQ_EMPTY

```
#define SPI_E_SEQ_EMPTY
```

No job in sequence.

Definition at line 222 of file Spi.h.

6.3.3.16 SPI_E_JOB_EMPTY

```
#define SPI_E_JOB_EMPTY
```

No channel in job.

Definition at line 229 of file Spi.h.

6.3.3.17 SPI_E_PARAM_POINTER

```
#define SPI_E_PARAM_POINTER
```

If the parameter versioninfo or Spi configuration is NULL_PTR.

Definition at line 235 of file Spi.h.

6.3.3.18 SPI_INIT_ID

```
#define SPI_INIT_ID
```

API service ID for SPI Init function.

Parameters used when raising an error or exception.

Definition at line 252 of file Spi.h.

6.3.3.19 SPI_DEINIT_ID

```
#define SPI_DEINIT_ID
```

API service ID for SPI DeInit function.

Parameters used when raising an error or exception.

Definition at line 259 of file Spi.h.

6.3.3.20 SPI_WRITEIB_ID

```
#define SPI_WRITEIB_ID
```

API service ID for SPI write IB function.

Parameters used when raising an error or exception.

Definition at line 266 of file Spi.h.

6.3.3.21 SPI_ASYNCTRANSMIT_ID

```
#define SPI_ASYNCTRANSMIT_ID
```

API service ID for SPI async transmit function.

Parameters used when raising an error or exception.

Definition at line 273 of file Spi.h.

6.3.3.22 SPI_READIB_ID

```
#define SPI_READIB_ID
```

API service ID for SPI read IB function.

Parameters used when raising an error or exception.

Definition at line 280 of file Spi.h.

6.3.3.23 SPI_SETUPEB_ID

```
#define SPI_SETUPEB_ID
```

API service ID for SPI setup EB function.

Parameters used when raising an error or exception.

Definition at line 287 of file Spi.h.

6.3.3.24 SPI_GETSTATUS_ID

```
#define SPI_GETSTATUS_ID
```

API service ID for SPI get status function.

Parameters used when raising an error or exception.

Definition at line 294 of file Spi.h.

6.3.3.25 SPI_GETJOBRESULT_ID

```
#define SPI_GETJOBRESULT_ID
```

API service ID for SPI get job result function.

Parameters used when raising an error or exception.

Definition at line 301 of file Spi.h.

6.3.3.26 SPI_GETSEQUENCERESULT_ID

```
#define SPI_GETSEQUENCERESULT_ID
```

API service ID for SPI get sequence result function.

Parameters used when raising an error or exception.

Definition at line 308 of file Spi.h.

6.3.3.27 SPI_GETVERSIONINFO_ID

```
#define SPI_GETVERSIONINFO_ID
```

API service ID for SPI get version info function.

Parameters used when raising an error or exception.

Definition at line 315 of file Spi.h.

6.3.3.28 SPI_SYNCTRANSMIT_ID

```
#define SPI_SYNCTRANSMIT_ID
```

API service ID for SPI sync transmit function.

Parameters used when raising an error or exception.

Definition at line 322 of file Spi.h.

6.3.3.29 SPI_GETHWUNITSTATUS_ID

```
#define SPI_GETHWUNITSTATUS_ID
```

API service ID for SPI get hwunit status function.

Parameters used when raising an error or exception.

Definition at line 329 of file Spi.h.

6.3.3.30 SPI_CANCEL_ID

```
#define SPI_CANCEL_ID
```

API service ID for SPI cancel function.

Parameters used when raising an error or exception.

Definition at line 336 of file Spi.h.

6.3.3.31 SPI_SETASYNCMODE_ID

```
#define SPI_SETASYNCMODE_ID
```

API service ID for SPI set async mode function.

Parameters used when raising an error or exception.

Definition at line 343 of file Spi.h.

6.3.3.32 SPI_MAINFUNCTION_HANDLING_ID

```
#define SPI_MAINFUNCTION_HANDLING_ID
```

API service ID for SPI main function.

Parameters used when raising an error or exception

Definition at line 350 of file Spi.h.

6.3.3.33 SPI_SETHWUNITASYNCMODE_ID

```
#define SPI_SETHWUNITASYNCMODE_ID
```

API service ID for SPI set HW Unit async mode.

Parameters used when raising an error or exception.

Definition at line 357 of file Spi.h.

6.3.3.34 SPI_SETCLOCKMODE_ID

```
#define SPI_SETCLOCKMODE_ID
```

API service ID for SPI Set Clock Mode.

Parameters used when raising an error or exception.

Definition at line 364 of file Spi.h.

6.3.3.35 SPI_JOB_PRIORITY_LEVELS_COUNT

```
#define SPI_JOB_PRIORITY_LEVELS_COUNT
```

The number of allowed job priority levels (0..3).

The Priority has to be sint8.

Definition at line 372 of file Spi.h.

6.3.3.36 SPI_STOP_SEC_CONFIG_DATA_UNSPECIFIED

```
#define SPI_STOP_SEC_CONFIG_DATA_UNSPECIFIED
```

Export Post-Build configurations.

Definition at line 804 of file Spi.h.

6.3.4 Types Reference

6.3.4.1 Spi_DataBufferType

```
typedef uint8 Spi_DataBufferType
```

Type of application data buffer elements.

Definition at line 486 of file Spi.h.

6.3.4.2 Spi_NumberOfDataType

```
typedef uint16 Spi_NumberOfDataType
```

Type for defining the number of data elements of the type Spi_DataBufferType.

Type for defining the number of data elements of the type Spi_DataBufferType to send or receive by Channel.

Definition at line 495 of file Spi.h.

6.3.4.3 Spi_ChannelType

```
typedef uint8 Spi_ChannelType
```

Specifies the identification (ID) for a Channel.

Definition at line 526 of file Spi.h.

6.3.4.4 Spi_JobType

```
typedef uint16 Spi_JobType
```

Specifies the identification (ID) for a Job.

Definition at line 533 of file Spi.h.

6.3.4.5 Spi_SequenceType

```
typedef uint8 Spi_SequenceType
```

Specifies the identification (ID) for a sequence of jobs.

Definition at line 540 of file Spi.h.

6.3.4.6 Spi_HWUnitType

```
typedef uint8 Spi_HWUnitType
```

Specifies the ID for a SPI Hardware microcontroller peripheral unit.

This type is used for specifying the identification (ID) for a SPI Hardware microcontroller peripheral unit.

Definition at line 550 of file Spi.h.

6.3.4.7 Spi_ExternalDeviceType

```
typedef uint8 Spi_ExternalDeviceType
```

Contains the ID of an external device.

This contains the identification (ID) of the external device for which there's a collection of particular settings

Definition at line 558 of file Spi.h.

6.3.5 Enum Reference

6.3.5.1 Spi_StatusType

```
enum Spi_StatusType
```

This type defines a range of specific status for SPI Driver.

Enumerator

SPI_UNINIT	Not initialized or not usable.
SPI_IDLE	Not currently transmitting any jobs.
SPI_BUSY	Is performing a SPI Job(transmit).

Definition at line 383 of file Spi.h.

6.3.5.2 Spi_JobResultType

enum Spi_JobResultType

This type defines a range of specific Jobs status for SPI Driver.

Enumerator

SPI_JOB_OK	The last transmission of the Job has been finished successfully.
SPI_JOB_PENDING	The SPI handler/Driver is performing a SPI Job.
SPI_JOB_FAILED	The last transmission of the Job has failed.
SPI_JOB_QUEUED	An asynchronous transmit Job has been accepted, while actual transmission for this Job has not started yet.

Definition at line 395 of file Spi.h.

6.3.5.3 Spi_SeqResultType

enum Spi_SeqResultType

This type defines a range of specific Sequences status for SPI Driver.

Enumerator

SPI_SEQ_OK	The last transmission of the Sequence has been finished successfully.
SPI_SEQ_PENDING	The SPI handler/Driver is performing a SPI Sequence.
SPI_SEQ_FAILED	The last transmission of the Sequence has failed.
SPI_SEQ_CANCELLED	The last transmission of the Sequence has been cancelled by the user.

Definition at line 409 of file Spi.h.

6.3.5.4 Spi_BufferType

enum Spi_BufferType

The enumeration containing the designated values for buffer types (internal or external).

Enumerator

IB	The Channel is configured using Internal Buffer.
EB	The Channel is configured using External Buffer.

Definition at line 421 of file Spi.h.

6.3.5.5 Spi_AsyncModeType

enum Spi_AsyncModeType

Specifies the asynchronous mechanism mode for SPI buses handled asynchronously in Level 2.

#if (SPI_LEVEL2 == SPI_LEVEL_DELIVERED) Specifies the asynchronous mechanism mode for SPI buses handled asynchronously in LEVEL 2. SPI150: This type is available or not according to the pre compile time parameter: SPI_LEVEL_DELIVERED. This is only relevant for LEVEL 2.

Enumerator

SPI_POLLING_MODE	The asynchronous mechanism is ensured by polling, so interrupts related to SPI buses handled asynchronously are disabled.
SPI_INTERRUPT_MODE	The asynchronous mechanism is ensured by interrupt, so interrupts related to SPI buses handled asynchronously are enabled.

Definition at line 437 of file Spi.h.

6.3.5.6 Spi_Ipw_SupportedIpsType

enum Spi_Ipw_SupportedIpsType

This enum contains all IPs which can integrate SPI functionalityes.

Definition at line 136 of file Spi_Ipw_Types.h.

6.3.6 Function Reference

6.3.6.1 Spi_JobTransferFinished()

```
void Spi_JobTransferFinished (
            const Spi_JobConfigType * JobConfig,
            Spi_JobResultType JobResult )
```

This function is called after a Job has been executed.

The function calls Job and Sequence end notifications and schedules the next job of the sequence or on the liberated HW Unit.

Parameters

in	*JobConfig*	The just transmited job pointer.

Returns

void

Precondition

Pre-compile parameter SPI_LEVEL_DELIVERED shall be SPI_LEVEL1 or SPI_LEVEL2.

6.3.6.2 Spi_Init()

```
void Spi_Init (
            const Spi_ConfigType * ConfigPtr )
```

This function initializes the SPI driver.

This function initializes the SPI driver using the pre-established configurations

- Service ID: 0x00
- Sync or Async: Synchronous
- Reentrancy: Non-Reentrant

Parameters

in	*ConfigPtr*	Specifies the pointer to the configuration set

Returns

void

6.3.6.3 Spi_DeInit()

```
Std_ReturnType Spi_DeInit (
            void  )
```

This function de-initializes the SPI driver.

This function de-initializes the SPI driver using the pre-established configurations

- Service ID: 0x01
- Sync or Async: Synchronous
- Reentrancy: Non-Reentrant

Returns

Std_ReturnType

Return values

E_OK	de-initialisation command has been accepted
E_NOT_OK	de-initialisation command has not been accepted

Precondition

The driver needs to be initialized before calling Spi_DeInit() otherwise, the function Spi_DeInit() shall raise the development error if SPI_DEV_ERROR_DETECT is STD_ON.

6.3.6.4 Spi_WriteIB()

```
Std_ReturnType Spi_WriteIB (
            Spi_ChannelType Channel,
            const Spi_DataBufferType * DataBufferPtr )
```

This function writes the given data into the buffer of a specific channel.

This function writes the given data into the buffer of a specific channel.

- Service ID: 0x02
- Sync or Async: Synchronous
- Reentrancy: Reentrant

Parameters

in	*Channel*	Channel ID
in	*DataBufferPtr*	Pointer to source data buffer

Returns

Std_ReturnType

Return values

E_OK	Command has been accepted
E_NOT_OK	Command has not been accepted

Precondition

The driver needs to be initialized before calling Spi_WriteIB() otherwise, the function Spi_WriteIB() shall raise the development error if SPI_DEV_ERROR_DETECT is STD_ON.

Pre-compile parameter SPI_CHANNEL_BUFFERS_ALLOWED shall be SPI_USAGE0 or SPI_USAGE2.

6.3.6.5 Spi_ReadIB()

```
Std_ReturnType Spi_ReadIB (
            Spi_ChannelType Channel,
            Spi_DataBufferType * DataBufferPointer )
```

This function reads the data from the buffer of a channel and puts at the memory location.

This function reads the data from the buffer of a specific channel and puts at the specified memory location.

- Service ID: 0x04
- Sync or Async: Synchronous
- Reentrancy: Reentrant

Parameters

in	*Channel*	Channel ID
in,out	*DataBufferPointer*	Pointer to the memory location that will be written with the data in the internal buffer

Returns

Std_ReturnType

Return values

E_OK	read command has been accepted
E_NOT_OK	read command has not been accepted

Precondition

The driver needs to be initialized before calling Spi_ReadIB() otherwise, the function Spi_ReadIB() shall raise the development error if SPI_DEV_ERROR_DETECT is STD_ON.

Pre-compile parameter SPI_CHANNEL_BUFFERS_ALLOWED shall be SPI_USAGE0 or SPI_USAGE2.

6.3.6.6 Spi_AsyncTransmit()

```
Std_ReturnType Spi_AsyncTransmit (
            Spi_SequenceType Sequence )
```

This function triggers the asynchronous transmission for the given sequence.

This function triggers the asynchronous transmission for the given sequence.

- Service ID: 0x03
- Sync or Async: Asynchronous
- Reentrancy: Reentrant

Parameters

in	*Sequence*	Sequence ID

Returns

Std_ReturnType

Return values

E_OK	Transmission command has been accepted
E_NOT_OK	Transmission command has not been accepted

Precondition

The driver needs to be initialized before calling Spi_AsyncTransmit() otherwise, the function Spi_AsyncTransmit() shall raise the development error if SPI_DEV_ERROR_DETECT is STD_ON.

Pre-compile parameter SPI_LEVEL_DELIVERED shall be SPI_LEVEL1 or SPI_LEVEL2.

6.3.6.7 Spi_SetupEB()

```
Std_ReturnType Spi_SetupEB (
            Spi_ChannelType Channel,
            Spi_DataBufferType * SrcDataBufferPtr,
            Spi_DataBufferType * DesDataBufferPtr,
            Spi_NumberOfDataType Length )
```

This function setup an external buffer to be used by a specific channel.

This function setup an external buffer to be used by a specific channel.

- Service ID: 0x05
- Sync or Async: Synchronous
- Reentrancy: Reentrant

Parameters

in	*Channel*	Channel ID
in	*SrcDataBufferPtr*	Pointer to the memory location that will hold the transmitted data
in	*Length*	Length of the data in the external buffer
out	*DesDataBufferPtr*	Pointer to the memory location that will hold the received data

Returns

Std_ReturnType

Return values

E_OK	Setup command has been accepted
E_NOT_OK	Setup command has not been accepted

Precondition

The driver needs to be initialized before calling Spi_SetupEB() otherwise, the function Spi_SetupEB() shall raise the development error if SPI_DEV_ERROR_DETECT is STD_ON.

Pre-compile parameter SPI_CHANNEL_BUFFERS_ALLOWED shall be SPI_USAGE1 or SPI_USAGE2.

6.3.6.8 Spi_GetStatus()

```
Spi_StatusType Spi_GetStatus (
            void  )
```

This function returns the status of the SPI driver.

This function returns the status of the SPI driver.

- Service ID: 0x06
- Sync or Async: Synchronous
- Reentrancy: Reentrant

Returns

Spi_StatusType

Return values

SPI_UNINIT	The driver is un-initialized
SPI_IDLE	The driver has no pending transfers
SPI_BUSY	The driver is busy

Precondition

The driver needs to be initialized before calling Spi_GetStatus() otherwise, the function Spi_GetStatus() shall raise the development error if SPI_DEV_ERROR_DETECT is STD_ON.

6.3.6.9 Spi_GetJobResult()

```
Spi_JobResultType Spi_GetJobResult (
            Spi_JobType Job )
```

This function is used to request the status of a specific job.

This function is used to request the status of a specific job.

- Service ID: 0x07
- Sync or Async: Synchronous
- Reentrancy: Reentrant

Parameters

in	*Job*	Job ID

Returns

Spi_JobResultType

Return values

SPI_JOB_OK	The job ended successfully
SPI_JOB_PENDING	The job is pending
SPI_JOB_FAILED	The job has failed

Precondition

The driver needs to be initialized before calling Spi_GetJobResult() otherwise, the function Spi_GetJobResult() shall raise the development error if SPI_DEV_ERROR_DETECT is STD_ON.

6.3.6.10 Spi_GetSequenceResult()

```
Spi_SeqResultType Spi_GetSequenceResult (
            Spi_SequenceType Sequence )
```

This function is used to request the status of a specific sequence.

This function is used to request the status of a specific sequence.

- Service ID: 0x08
- Sync or Async: Synchronous
- Reentrancy: Reentrant

Parameters

in	*Sequence*	Sequence ID

Returns

Spi_SeqResultType

Return values

SPI_SEQ_OK	The sequence ended successfully
SPI_SEQ_PENDING	The sequence is pending
SPI_SEQ_FAILED	The sequence has failed

Precondition

The driver needs to be initialized before calling Spi_GetSequenceResult() otherwise, the function Spi_GetSequenceResult() shall raise the development error if SPI_DEV_ERROR_DETECT is STD_↩ ON.

6.3.6.11 Spi_SyncTransmit()

```
Std_ReturnType Spi_SyncTransmit (
            Spi_SequenceType Sequence )
```

This function is used for synchronous transmission of a given sequence.

This function is used for synchronous transmission of a given sequence.

- Service ID: 0x0a
- Sync or Async: Synchronous
- Reentrancy: Reentrant

Parameters

in	*Sequence*	Sequence ID

Returns

Std_ReturnType

Return values

E_OK	Transmission command has been completed successfully
E_NOT_OK	Transmission command has not been accepted

Precondition

The driver needs to be initialized before calling Spi_SyncTransmit(). otherwise, the function Spi_SyncTransmit() shall raise the development error if SPI_DEV_ERROR_DETECT is STD_ON.

Pre-compile parameter SPI_LEVEL_DELIVERED shall be SPI_LEVEL0 or SPI_LEVEL2

6.3.6.12 Spi_SetAsyncMode()

```
Std_ReturnType Spi_SetAsyncMode (
            Spi_AsyncModeType Mode )
```

This function specifies the asynchronous mode for the SPI busses handled asynchronously.

This function specifies the asynchronous mode for the SPI busses handled asynchronously.

- Service ID: 0x0d
- Sync or Async: Synchronous
- Reentrancy: Non-Reentrant

Parameters

in	*Mode*	This parameter specifies the asynchronous operating mode (SPI_POLLING_MODE or SPI_INTERRUPT_MODE)

Returns

Std_ReturnType

Return values

E_OK	The command ended successfully
E_NOT_OK	The command has failed

Precondition

The driver needs to be initialized before calling Spi_SetAsyncMode() otherwise, the function Spi_SetAsyncMode() shall raise the development error if SPI_DEV_ERROR_DETECT is STD_ON.

Pre-compile parameter SPI_LEVEL_DELIVERED shall be SPI_LEVEL2

6.3.7 Variable Documentation

6.3.7.1 Spi_axSpiJobState

```
Spi_JobStateType Spi_axSpiJobState[SPI_MAX_JOB]  [extern]
```

Extern arrays contain the state of Sequences, Jobs and Channels.

How to Reach Us:

Home Page:
nxp.com

Web Support:
nxp.com/support

Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in NXP 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. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.