S32K1_S32M24x Real-Time Drivers AUTOSAR R21-11

Version 3.0.0

21 March 2025 **Release notes**

1 Getting Started

1.1 Package content

This release contains the NXP S32K1_S32M24x Real-Time Drivers Version 3.0.0:

- "eclipse/plugins/<mod>_TS_T40D2M30I0R0" directories - Tresos Plugins, 1 per module.
- "SW32K1_S32M24x_RTD_R21-11_3.0.0_D2503.exe"
- "SW32K1_S32M24x_RTD_R21-11_3.0.0_D2503_ReleaseNotes.pdf" - This file.
- "SW32K1_S32M24x_RTD_R21-11_3.0.0_D2503_DesignStudio_updatesite.zip"
- "SW32K1_S32M24x_RTD_R21-11_3.0.0_D2503_SafetyPackage.zip" - contains FMEA reports and Safety Manual.
- "SW32K1_S32M24x_RTD_R21-11_3.0.0_D2503_QualityPackage.zip"
- "SW32K1_S32M24x_RTD_R21-11_3.0.0_D2503_SCR.txt"
- Various other files: GettingStarted.htm start page and associated images, the license.txt EULA file and the Uninstall.exe utility for removing the RTD installation.

1.2 Installation Design Studio

1.2.1 Bundled in S32 Design Studio

S32 RTD is delivered bundled in the S32 Design Studio. In this case, it's already configured and ready to use. New S32DS project wizard and New S32DS project from Example can be used to create S32 RTD based projects.

1.2.2 Delivered as an extension for S32 Design Studio

S32 RTD is delivered as an Update Site for S32 Design Studio "*S32DS 3.6.0*". In this case, it must be installed by opening Help -> S32 Design Studio Extensions and Updates -> Add Update Sites and selecting the archive file containing the S32 RTD software and then check the S32 RTD software package to be installed and continue the installation process. After it is installed, New S32DS project wizard and New S32DS project from Example can be used to create S32 RTD based projects.

S32DS v3.6 installation flow:

1. Install S32Design Studio 3.6: S32DS_3.6.0_win32.x86_64.exe

2. Install updatesite for this release: SW32K1_S32M24x_RTD_R21-11_3.0.0_D2503_DesignStudio_updatesite.zip

1.3 Installation EB Tresos

Follow the installer steps. By default the installer will create a link between the installation target directory and a selected EB Tresos installation. If you choose not to create a link, you can later create one manually or you can copy all "<mod>_TS_T40D2M30I0R0" directories and .JAR files to the "<Tresos Install Path>\plugins" directory.

2 Release Specifics

The S32K1_S32M24x Real-Time Drivers Version 3.0.0 is AUTOSAR R21-11 compliant. The AUTOSAR Configuration ARXML specification takes precedence over AUTOSAR SWS PDF Specifications if there are discrepancies.

The S32K1_S32M24x Real-Time Drivers Version 3.0.0 can be used also in non-AUTOSAR environment, as a collection of peripheral drivers designed to simplify and accelerate application development on NXP microcontrollers.

2.1 Release Details

This is the S32K1_S32M24X Real-Time Drivers release Version 3.0.0, supporting AUTOSAR R21-11 and non-AUTOSAR, with functionality and testing on S32K1 and S32M24x family of processor.

This release contains a deviation from AUTOSAR recommended version check inside source files.

This release has RFP quality in terms of testing and quality documentation. RFP release contains all features and is fully tested on Customer Engineering Samples or Qual Intend Samples.

RFP release is software release that can be used in cars production. RFP release is delivered with complete software documentation, quality package and safety package.

In all source files, Software Version values are checked (major, minor, patch). AUTOSAR release or SWS versions are not checked during preprocessing/template generation.

The correct SWS versions are exported by each module.

The functions contained in the Dem, Det, EcuC, EcuM, Rte and interface plugins are sample stub functions.

These functions should be replaced by the user developed code during integration.

The Resource module is needed to select the MCU derivative.

The derivatives supported can be found in the Resource module definition file, parameter 'ResourceSubderivative'.

This release was developed and tested using:

- Silicon Chip S32K116 (0N96V), 48 LQFP
- Silicon Chip S32K118 cut 1.0 (0N97V), 64 LQFP
- Silicon Chip S32K142 cut 1.0 (0N33V), 100 LQFP
- Silicon Chip S32K144 cut 2.1 (0N57U), 100 LQFP
- Silicon Chip S32K144W SWLH (0P64A) CTXP1950B
- Silicon Chip S32K146 cut 1.0 (0N73V), 144 LQFP
- Silicon Chip S32K148 cut 1.0 (0N20V), 144 LQFP
- Silicon Chip P32M244CC WKHS 0P69K 20231008
- Silicon Chip P32M244LC WKHS 0P69K 20231008
- Silicon Chip P32M242CC MKHS 0P69K 20231201
- Silicon Chip P32M242LC WKHS 0P69K 20231201
- S32K-MB PCB 28767 RevA SCH RevB
- S32K1XXCVD-Q048 PCB 30838 RevX1 SCH RevA
- S32K14XCVD-Q064 PCB 29454 RevX2 SCH RevA1
- S32K144-100LQFP PCB 28768 RevA SCH RevX1
- XS32K14XCVD-Q144 PCB 29559 RevX2 SCH RevB
- XS32M2XXCVB-Q064 PCB 53099 RevX1 SCH RevA

The following limitations are present in this release:

- Known issue with GCC compiler, all RTD drivers : Warning regarding enum size is thrown by the linker due to usage of "-fno-short-enums" option: "use of enum values across objects may fail". The drivers do not use any library enum types - no functional impact.
- Known issue with IAR compiler, all RTD drivers : Warning regarding stack usage is thrown for reference implementations of core exceptions in startup files when drivers are compiled with IAR. These functions are provided as reference code and can be replace/modified by the application.
- Usage of IAR compiler option "-enable_stack_usage " will issue warnings regarding uncalled functions (eg : interrupt handlers). This should be disregarded.
- Due to RAM size on devices S32K11x it is not recommend to build the RTD drivers project targeting RAM.
- Known issue with S32DS: Values in ecpd output file are duplicated when using S32DS version 3.6. A workaround is adding the following argument in s32ds.ini "Dcom.nxp.swtools.periph.components.loading.new=false”.
- During this SW product development, the achieved code coverage is between 82.45% min and 100% max, as it can be seen in the individual Code Coverage Summary reports. Nevertheless, the whole code went through a diligently inspection-based review.

2.2 Used Documentation

This release was developed and tested with the following documents:

Table 1. Reference Manuals

Document Title	Version and Date
S32K1xx Series Reference Manual	Rev.14.1, 01/2024
S32M24x Reference Manual	Rev. 5, 12/2024
S32K1xx Data Sheet	Rev. 14, 08/2021
S32M2xx Data Sheet	Rev. 7, 12/2024

Table 2. Implemented Errata

Document Title	Maskset	Date
S32K116 Mask Set Errata for Mask	0N96V	Rev. 17/JAN/2024, 1/2024
S32K118 Mask Set Errata for Mask	0N97V	Rev. 15/JAN/2024, 1/2024
S32K142 Mask Set Errata for Mask	0N33V	Rev. 15/JAN/2024, 1/2024
S32K144W Mask Set Errata for Mask	0P64A	Rev. 03/JUL/2024, 7/2024
S32K144 Mask Set Errata for Mask	0N57U	Rev. 15/JAN/2024, 1/2024
S32K146 Mask Set Errata for Mask	0N73V	Rev. 15/JAN/2024, 1/2024
S32K148 Mask Set Errata for Mask	0N20V	Rev. 15/JAN/2024, 1/2024
S32M244 Mask Set Errata for Mask	P64A+P69K	Rev.1, 8/2024
S32M242 Mask Set Errata for Mask	N33V+P69K	Rev.1, 8/2024

2.3 Supported Derivatives

The software described in this document is intended to be used with the following microcontroller devices of NXP :

- 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
- s32m244_lqfp64
- s32m243_lqfp64
- s32m242_lqfp64
- s32m241_lqfp64

The mapping between MWCT1xxxS parts and S32K1XX is showed in the table below:

MWCT1xxxS derivative	S32K1xx derivative
MWCT1014S_LQFP64	S32K144_LQFP64
MWCT1014S_LQFP100	S32K144_LQFP100
MWCT1015S_LQFP100	S32K146_LQFP100
MWCT1015S_MAPBGA100	S32K146_MAPBGA100
MWCT1016S_MAPBGA100	S32K148_MAPBGA100

Note: MWCT part numbers contain NXP confidential IP for Qi Wireless Power, and they are tested with:

- Silicon Chip S32K144 cut 2.1 (maskset 0N57U), 100 LQFP (FS32K144UAVLL)

2.4 Modules Configuration

2.4.1 EB Tresos

Modules configurations were developed and tested using the Tresos Configuration Tool version "*EB tresos Studio 29.0.0 b220329-0119*"

Configuration definition files were developed according to AUTOSAR 4.7.0, AUTOSAR_EcucParamDef.arxml

A folder named "<mod>_TS_TtDdMmIiRr" exists for each delivered module (<mod>). It is called a Tresos plugin for the module. A plugin contains the AUTOSAR module definition file (epd), the Tresos Xpath Data Model module definition file (xdm), the module user and integration manuals, the module configuration generation template source files, and the module driver static source files. Additional necessary Tresos specific tooling files are also included.

Plugin Encoding: <mod>_TS_TtDdMmIiRr

Important change related to the plugin notation:
- "m" = coding major and minor version number, can contain 1 or more digits
- "i" = patch number.
The major version number will be left out, if it is "0", in this case "m" contains 1 digit only,
otherwise it contains 2 digits

For this release:

- t=40, CortexM Architecture
- d=2, S32K1XX (derivative)
- m=30, Release major and minor version
- i=0, Release patch version
- r=0, Reserved

2.4.2 S32 Design Studio

Configuration components were developped using "*S32DS 3.6.0*".

The components are split in three tools inside S32 Design Studio: Pins Tool, Clocks Tool, Peripherals tool which enable the generation of configuration structures to be used by both Autosar and low-level drivers.

2.5 Support and Driver Plugins Delivered

Table 3. Support and Driver Plugins Delivered

Plugin	Low level interface	SW Version	Description
ADC	Adc_Ip, Pdb_Ip	sw version 3.0.0	Driver, Analog to Digital Conversion
AE	Hvm_Ip, Ae_Ip	sw version 3.0.0	Driver, Application Extension
BASENXP	N/A	sw version 3.0.0	Base Module, General AUTOSAR and Hardware Specific register files
CAN_43_FLEXCAN	FlexCAN_Ip	sw version 3.0.0	Driver, Controller Area Network
CANTRCV_43_AE	N/A	sw version 3.0.0	Driver, Can Transceiver
CRC	Crc_Ip	sw version 3.0.0	Driver, Cyclic Redundancy Check
DIO	Gpio_Ip	sw version 3.0.0	Driver, Digital Input Output
DPGA	Dpga_Ip	sw version 3.0.0	Driver, Differential Programmable Gain Amplifier
ETH_43_ENET	ENET	sw version 3.0.0	Driver, Ethernet
FEE	N/A	sw version 3.0.0	Driver, Flash EEPROM Emulation
GPT	Ftm_Gpt_Ip, Lpit_Gpt_Ip, Lptmr_Gpt_Ip, Srtc_Ip	sw version 3.0.0	Driver, General Purpose Timer
GDU	Gdu_Ip	sw version 3.0.0	Driver, Gate Driver Uinit
I2C	FlexIo_I2C, LPI2C_Ip	sw version 3.0.0	Driver, Inter-Integrated Circuit
ICU	Cmp_Ip, Ftm_Icu_Ip, Lptmr_Icu_Ip, Lpit_Icu_Ip ,Port_Ci_Icu_Ip	sw version 3.0.0	Driver, Input Capture Unit

Table 3. Support and Driver Plugins Delivered*...continued*

Plugin	Low level interface	SW Version	Description
LIN_43_LPUART_FLEXIO	Lpuart_Lin_Ip, Flexio_Lin	sw version 3.0.0	Driver, Local Interconnect Network
LINTRCV_43_AE	N/A	sw version 3.0.0	Driver, Lin Transceiver
MCL	Dma_Ip, Cache_Ip, Trgmux_Ip	sw version 3.0.0	Driver, Microcontroller Library
MCU	Clock_Ip, Power_Ip, Ram_Ip	sw version 3.0.0	Driver, Microcontroller Unit
MEMACC	NA	sw version 3.0.0	Driver, Memory Access
MEM_43_INFLS	Ftfe_Ip	sw version 3.0.0	Driver, Mem Internal Flash
MEM_43_EXFLS	Qspi_Ip	sw version 3.0.0	Driver, Mem External Flash
MEM_43_EEP	Ftfe_Ip	sw version 3.0.0	Driver, Mem EEPROM
OCOTP	AeMemOtp_Ip	sw version 3.0.0	Driver, On-Chip One Time Programmable Controller
OCU	Ftm_Ocu_Ip	sw version 3.0.0	Driver, Output Control Unit
PLATFORM	startup, System_Ip, IntCtrl_Ip, Mpu_Ip	sw version 3.0.0	Driver, Platform
PORT	Gpio_Ip, Port_Ip	sw version 3.0.0	Driver, Port
PWM	FlexIO_Pwm_Ip, Ftm_Pwm_Ip	sw version 3.0.0	Driver, Pulse Width Modulation
QDEC	Ftm_Qdec_Ip	sw version 3.0.0	Driver, Quadrature Decoder
RESOURCE	N/A	sw version 3.0.0	Resource Module, Required by all other modules to select MCU derivative
RM	Dma_Mux_Ip	sw version 3.0.0	Driver, Resource Manager
I2S	Sai_Ip, FlexIO_Ip	sw version 3.0.0	Driver, Inter-IC Sound
SPI	FlexIO_Spi, Lpspi	sw version 3.0.0	Driver, Serial Peripheral Interface
UART	Flexlo_UART,Lpuart_Uart_Ip	sw version 3.0.0	Driver, UART Driver
WDG	Ewm_Ip (External WDG monitor),Wdog_Ip,AeWdog_Ip	sw version 3.0.0	Driver, Watchdog
CANIF	N/A	sw version 3.0.0	Support Stub, Controller Area Network Interface
DEM	N/A	sw version 3.0.0	Support Stub, Diagnostic Event Manager
DET	N/A	sw version 3.0.0	Support Stub, Development Error Tracer
ECUC	N/A	sw version 3.0.0	Support Stub, ECU Configuration
ECUM	N/A	sw version 3.0.0	Support Stub, ECU State Manager
ETHIF	N/A	sw version 3.0.0	Support Stub, Ethernet Interface
ETHTRCV	N/A	sw version 3.0.0	Support Stub, Ethernet Transceiver
ETHSWT	N/A	sw version 3.0.0	Support Stub, Ethernet Switch
LINIF	N/A	sw version 3.0.0	Support Stub, Local Interconnect Network Interface
MEMIF	N/A	sw version 3.0.0	Support Stub, Memory Interface
OS	N/A	sw version 3.0.0	Support Stub, Operating System

Table 3. Support and Driver Plugins Delivered*...continued*

Plugin	Low level interface	SW Version	Description
RTE	N/A	sw version 3.0.0	Support Stub, only for Schedule Manager
WDGIF	N/A	sw version 3.0.0	Support Stub, Watchdog Interface
MKA	N/A	sw version 3.0.0	Support Stub, Macsec Key Assignment

2.6 Module Plugin Folder Structure

Table 4. Module Plugin Folder Structure

Folder or file	Description
<mod>_TS_TtDdMmIiRr\anchors.xml	Tresos Configuration tooling documentation data file
<mod>_TS_TtDdMmIiRr\plugin.xml	Tresos Configuration tooling data file
<mod>_TS_TtDdMmIiRr\ant_generator.xml	Tresos Configuration tooling data file
<mod>_TS_TtDdMmIiRr\autosar\<mod>.epd	Module Parameter Definition in AUTOSAR format
<mod>_TS_TtDdMmIiRr\config\<mod>.xdm	Module Parameter Definition in Tresos XDM format
<mod>_TS_TtDdMmIiRr\config_ext\<mod>PreConfiguration.xdm	Module Parameter Default Configuration in Tresos XDM format[1]
<mod>_TS_TtDdMmIiRr\doc\AUTOSAR_RTD_<mod>_IM.pdf	Module Integration Manual
<mod>_TS_TtDdMmIiRr\doc\AUTOSAR_RTD_<mod>_UM.pdf	Module User's Manual
<mod>_TS_TtDdMmIiRr\generate_PB	Post-build source files (only if applicable)
<mod>_TS_TtDdMmIiRr\generate_PB\src	Post-build source file templates (only if applicable)
<mod>_TS_TtDdMmIiRr\generate_PB\include	Post-build source file header templates (only if applicable)
<mod>_TS_TtDdMmIiRr\generate_LT	Link-time source files (only if applicable)
<mod>_TS_TtDdMmIiRr\generate_LT\src	Link-time source file templates (only if applicable)
<mod>_TS_TtDdMmIiRr\generate_LT\include	Link-time source file templates (only if applicable)
<mod>_TS_TtDdMmIiRr\generate_PC\	Pre-compile source files
<mod>_TS_TtDdMmIiRr\generate_PC\src	Pre-compile source files templates
<mod>_TS_TtDdMmIiRr\generate_PC\include	Pre-compile source files templates
<mod>_TS_TtDdMmIiRr\generate_swcd	Module BSWMD file
<mod>_TS_TtDdMmIiRr\include\	Module driver header files
<mod>_TS_TtDdMmIiRr\META-INF	Tresos Configuration tooling data and signature files
<mod>_TS_TtDdMmIiRr\src\	Module driver source files[2]

Notes:

[1] Not available for all plugins.

[2] The Support Stub Resource contains the "resource" folder instead of the "src" folder.

2.7 Compiler Options

This release was developed and tested with:

- NXP GCC 10.2.0 20200723 (Build 1728 Revision g5963bc8)
- Green Hills Multi 7.1.6d / Compiler 2020.1.4
- IAR ANSI C/C++ Compiler V8.40.3.228/W32 for ARM Functional Safety

2.8 Examples and Demos

The drivers provide a set of examples. For details, please refer to Examples\...\readme.txt file from each driver folder.

3 Known Issues for S32K1 RTD 3.0.0

3.1 Known Issues for 3.0.0

ID	Headline
ARTD-172639	[S32K1_M24X 3.0.0 FSRA Improvement] GPT Traceability
ARTD-169518	[SPI] Cannot config more SpiExternalDevice on S32DS with S32G-RTD 5.0.0
ARTD-164644	[ETH][ENET][K148] Enet_Ip_SetSpeed only effective for RMII mode
ARTD-163592	[ETH] The configuration for the SIM subsystem registers should be added to the IPL driver in ETH module
ARTD-92326	[MCU] S32DS import from arxml file does not change the settings in Clock Tool
ARTD-54102	[SPI][S32K][FLEXIO] Master CS is only continuous between datas in channel but not between channels in job
ARTD-173086	[MemAcc] Exclusive Areas are missing for guarding global variables in MainFunction
ARTD-173074	[MEM_EXFLS][S32K1][ITG] QSPI_IP_031_001 is not completely tested
ARTD-173068	[MEM_EXFLS][S32K1] QSPI_IP_031_001 is not completely implemented
ARTD-172836	[S32K1][MEM_EEP] Remove and update redundant DET lines in driver
ARTD-172849	[MCU] Fixing MCU Traceability finding
ARTD-172842	[AE] Fixing misra rule 8.13
ARTD-168328	[BLN_RTD_4.7_S32K1XX_S32M24x_3.0.0][S32K1xx] Other derivatives than S32K148(with ENET) are present in DS
ARTD-163584	[MCU] K1_M2 S32CT configuration between Clocks tool and Mcu component not match
ARTD-161118	[S32K1][gpt] Generated headers cannot be built in case sensitive file systems
ARTD-144240	[I2S]: EBT/DS Comparison: Separate Rx/Tx enabling and Separate Rx/Tx Callbacks not available in EBT
ARTD-143126	[S32K3XX_S32M27x_5.0.0][I2S]: Flexio DMA transmits incorrect data when BytesWidth is 8
ARTD-133588	[Platform] Core power down after loading the image of the low power mode test
ARTD-168018	[S32K1_M24X 3.0.0][PORT] Port_Init: setting the pin direction is not the final step of the pad initialization
ARTD-167941	[S32K1_M24X 3.0.0][ADC]: Starting a conversion during async calibration fails the calibration

ID	Headline
ARTD-167028	[S32K1_M24X 3.0.0][SPI]: Need to limit configuration on Tresos and CT
ARTD-166496	[I2S] Compiler Directives #define 's are generating in single line in EB Tresos 29.7
ARTD-167042	[PWM] After calling Pwm_DeInit, the signal pin does not return to the idle state
ARTD-165932	[PLATFORM][S32K1_M24X 3.0.0] Platform_init funtion occur hard fault when user init out of range Partition ID
ARTD-167083	[PWM] Difference code gen EB and CT
ARTD-166562	[ADC] AdcPdbInterruptDelayPeriod not conditioned by the adc group delays
ARTD-166051	[Spi] Description for spi source clocks are not correct
ARTD-165936	[GPT][S32K1-M24X] Different code generate between EB and CT
ARTD-164268	[MCU] CPR_RTD_00110.mcu is not implemented for all modes
ARTD-166062	[S32K1_M24X 3.0.0] [CRC] Lack of constrain with node Algorithm type at IPL and HLD layer
ARTD-165219	[PWM][S32K1XX_M24X_3.0.0] Incorrect output signal when using channel fault
ARTD-164854	[SPI] Set SpiDevErrorDetect node default value to false to follow Autosar
ARTD-162870	Incorrect define value for the LOL_INT bit
ARTD-154948	[Mem_ExFls] Bswmd file should not have MemMainFunctionPeriod configurable
ARTD-152987	[MCU] Incorrect implementation of the req CPR_RTD_00110.mcu
ARTD-151686	[Spi] ecvd missing some node relative of SpiBaudrateConfig
ARTD-140322	[MCU] GetClockFrequency returns invalid value when clock is off
ARTD-148179	[MCU]: The description for some configuration item is inappropriate.
ARTD-134229	MCU: Type of function "Clock_Ip_GetClockFrequency" not follow Requirement
ARTD-145593	[memacc] Locking mechanism needs rework part 2 - Follow up
ARTD-7913	[MCU] Fix VSMD errors related to XPath expression for MIN attributes - resource file not found
ARTD-144278	[Mcu] when import McuPreconfiguration.epc to project, node POST_BUILD_VARIANT_USED is always read-only
ARTD-112976	[OCOTP][S32K3XX] Fix finding from Safety Assessment for UML design
ARTD-107833	[i2s] Driver makefile needs to be updated with the correct version of AUTOSAR SWS specification
ARTD-102221	[i2s] Validation of partition reference should be OS version independent
ARTD-126874	[Mem_ExFls] "Return value" field in ReqExport for Mem_Suspend and Mem_Resume is not correct - follow up
ARTD-141362	[DIO] S32CT doesn't raise an error when config a UnAvailablePortPins
ARTD-99266	[MemAcc] Interface of MemAcc_RequestLock is inconsistent between ReqImport.txt and MemAcc.h files (Follow-up)
ARTD-101475	[Mem_INFLS]: The performance for read is too slow.
ARTD-109062	[S32DS 3.5.6][BASE]Paths of RTD and other settings should be removed after detaching
ARTD-96711	[S32K3xx_S32M27x_4.0.0] LIN: Fix findings in Code Review Checklist
ARTD-100060	[FEE] Implementation of Fee_JobEndNotification is not correct
ARTD-100206	[LINTRCV] Wakeup flag should be checked when init driver
ARTD-99025	[S32K3xx_S32M27x_4.0.0] I2S: Different the syntax name of callback function

ID	Headline
ARTD-98831	[I2S] Build fail when using multicore and disable dev error detect node
ARTD-86699	[MCU]: The description for some configuration item is inappropriate.
ARTD-34888	[MCU] Requirement CPR_RTD_00011 is implemented incorrectly
ARTD-76065	[S32K1XX_M24X RTM 2.0.0] Dio: The example project build fail on K144
ARTD-66280	[LINTRCV] Bswmd file contains dummy implementation
ARTD-66277	[CANTRCV] Bswmd file contains dummy implementation
ARTD-10501	[MCL] Mcl_GetDmaInstanceStatus function can not get active bit on CR register
ARTD-172268	[LPI2C] Unable to set Clock Hold in slave mode.
ARTD-140779	[LINTRCV] difference between driver files that generated by S32CT and EB tresos
ARTD-140384	[S32K3_M27x 5.0.0] AE: function prototypes in Ae.h are different from requirement
ARTD-117881	[AE]Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT
ARTD-101563	[AE] Fix warnings in VSMD report.
ARTD-97136	[GPT] Build code fail on S32DS

4 Changes List for S32K1_M24x RTD 3.0.0

4.1 Change List for 3.0.0

ID	Subtype	Headline and Description
ARTD-8089	Bug	[I2S] For I2S over SAI the DMA transferred data is incorrect when FIFO packing enabled SAI set DMA transfer and FIFO packing: MUX mode: MUX_DISABLED: not working MUX_LINE: not working MUX_MEM: not working
ARTD-13848	New Feature	[MCL] FTM global time base feature NewWorkDescription: Implement into MCL component the support for FTM global time base feature. MCL should provide an API for enabling/disabling SIM > FTM_GTB_SPLIT_EN Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: N/A
ARTD-19536	Bug	[S32XX][PORT] Port driver have to add some Det_reportError function following requirement Detailed description (how to reproduce it): If Det is enabled, some function shall report specify error and return without any other action. Detail: CPR_RTD_00423.port: The function Port_ResetPinMode shall report PORT_E_MODE_UNCHANGEABLE error and return without any other action, if the parameter PortPinModeChangeable is set to FALSE. CPR_RTD_00426.port*: The function Port_SetAsUnusedPin shall report PORT_E_MODE_UNCHANGEABLE error and return without any other action, if the parameter PortPinModeChangeable is set to FALSE. CPR_RTD_00428.port*: The function Port_SetAsUsedPin shall report PORT_E_MODE_UNCHANGEABLE error and return without any other action, if the parameter PortPinModeChangeable is set to FALSE. SWS_Port_00223: The function Port_SetPinMode shall reportPORT_E_MODE_UNCHANGEABLE error and return without any other action, if the parameter PortPinModeChangeable is set to FALSE. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...]

ID	Subtype	Headline and Description
		Observed behavior: [...] Expected behavior: Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Port driver should add some Det_reportError function with specify error following the requirement in Detailed Description.
ARTD-22762	New Feature	[MCU] Implement interrupts in IP component NewWorkDescription: Implement interrupts in IP component. Interrupts shouldn't be implemented in MCU HLD component. MCU HLD implements wrappers over IP components. ISR is shared between clock and power IPs, a solution must be found to implement it in one of the drivers and send notifications from one driver to the other. Requirement source: [...] NA Proposed solution optional: Implement ISR in one of the drivers and send notifications from one driver to the other.
ARTD-25932	New Feature	[BASE][S32DS] Remove the reference clock from BaseNXP in IPL to Mcu/EcuC in HLD Detailed description (how to reproduce it): S32DS: Get rid of the reference clock from BaseNXP in IPL to Mcu/EcuC in HLD. !image-2022-04-15-16-45-54-723.png! {_}*Preconditions:*{_}{*}{*} Test Case ID (internal TC that caught the defect) optional: Observed behavior: The component from IPL depends on the other in HLD Expected behavior: The component from IPL can work independently, without any reference from HLD.{*}{*} Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-29209	Bug	[DIO] Investigate the violations from CWEReport Detailed description (how to reproduce it): There are 2 violations were found in the RTD_DIO_CWEReport. Please come to see the attachment for more details Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: There are 2 violations were found in the RTD_DIO_CWEReport. Please come to see the attachment for more details Expected behavior: All the violations need to be fixed Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-38632	Bug	[Dio] S32DS doesn't raise an error when naming two Channel Group Identification that are similar. Detailed description (how to reproduce it): at port 0: !screenshot-1.png|thumbnail! at port 1: !screenshot-2.png|thumbnail! Dio Channel Group Indentification are similar names. But S32DS doesn't raise an error. Preconditions: Make a new S32Z project, add Dio, and configure as described above. Test Case ID (internal TC that caught the defect) optional: Dio_TC_COT_201 Observed behavior: Like above

ID	Subtype	Headline and Description
		Expected behavior:*. An error must be raised Proposed solution optional:
ARTD-52191	New Feature	[WDGIF] Generate a new macro to represent all instances used NewWorkDescription: When we develop a new platform that has more WDG instances in the past, the Wdgif needs to be updated as the condition bellow : !image-2022-12-23-13-33-39-040.png|width=892,height=191! To avoid wasting effort on this, the code generation can generate a new macro to represent all instances which are being used. Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Generate a new macro if there is at least one WDG instance is being used: *{color:#00875a}USE_WDG_INSTANCE_ANY Use this new macro instead of repeating each macro for each instance
ARTD-53051	Bug	[SPI] Inconsistent SCK in continuous transfer Detailed description (how to reproduce it): When SpiExternalDevice is configured with SpiDataShiftEdge = TRAILING, SpiShiftClockIdleLevel = LOW (CPOL = CPHA = 0), and SpiExternalDevice/SpiCsContinuous = TRUE, generated SCK low level duration may be incorrect between subsequent series of SCK pulses, defined by SpiChannel/SpiDataWidth. See attachment (SpiBaudRate = 1.0E-7, SpiChannel/SpiDataWidth = 8, LPSPI2 input clock 40MHz, SCK yellow, SOUT purple, CS green). This can be worked-around by configuring SpiExternalDevice/SpiTimeCs2Cs with desired SCK low level duration, but ideally, it should be handled automatically by the driver code. As stated in SpiTimeCs2Cs description, this parameter shouldn't affect continuous transfers anyway. Preconditions: SpiExternalDevice/SpiBaudRate configured with resulting LPSPI_CCR1[SCKSET] > 0. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Incorrect SCK low pulse (inter-frame delay) in continuous transfer, Expected behavior: SCK low pulse duration consistent in continuous transfer. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: LPSPI_CCR1[SCKSCK] should be set equal to LPSPI_CCR1[SCKSET].
ARTD-54761	New Feature	[WDG] Update documentation with note about fix for Multiple instances of Wdg cannot generate in parallel NewWorkDescription: An issue will be occurred with multiple Wdg instances generation in parallel. The reason for this issue is, only a single plugin folder for Wdg is used to supports multiple instances, and the folders like "generate", "generate_PC", "generate_PB", "generate_swcd" are common. They are all accessed from plugin xml for each instance from the TemplateBasedCodeGenerator. We should warn users about Multiple instances of Wdg cannot generate in parallel. Requirement source: Customer Request Proposed solution optional: # Add the following setting in TemplateBasedCodeGenerator of WDG instances: !image-2022-12-01-18-51-26-654.png! 2. Add a note in UM/IM about this limitation.
ARTD-59652	New Feature	[I2S] Merge Slave and Master LLD APIs NewWorkDescription: Develop Slave Flexio I2s Requirement source: FLEXIO_I2S_IP_016_001 FLEXIO_I2S_IP_015_001 FLEXIO_I2S_IP_013_001

ID	Subtype	Headline and Description
		FLEXIO_I2S_IP_011_001 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: NA

ARTD-62490	New Feature	[S32K1][WDG] Add validation code for Wdg_Service()
		
		NewWorkDescription:
		Add validations for Wdg_Service() calls, in Wdg_Channel.c. This should satisfy requirements SWS_Wdg_00035, SWS_Wdg_00052.
		
		These should be similar to other Wdg functions.
		
		Requirement source:
		SWS_Wdg_00035: When development error detection is enabled for the Wdg Driver module: the watchdog servicing routine shall check whether the Wdg module's state is WDG_IDLE (meaning the watchdog driver and hardware are initialized and the watchdog is currently not being triggered or switched). If this is not the case, the function shall not trigger the watchdog hardware but raise the development error WDG_E_DRIVER_STATE.
		SWS_Wdg_00052: When development error detection is enabled for the Wdg Driver module: the watchdog servicing routine shall set the Wdg module's state to WDG_BUSY during its execution (indicating, that the module is busy) and shall reset the module's state to WDG_IDLE (indicating, that the module is initialized and not busy) as last operation before it returns.
		
		Proposed solution optional:
		For example, as the verifications in Wdg_SetTriggerCondition in Wdg_Channel.c
		
		#if ((WDG_VALIDATE_GLOBAL_CALL == STD_ON) (WDG_MULTICORE_ENABLED == STD_ON))
		Std_ReturnType Valid = (Std_ReturnType)E_NOT_OK;
		#endif
		
		#if (WDG_MULTICORE_ENABLED == STD_ON)
		volatile uint32 CoreID;
		
		CoreID = (uint32) Wdg_GetCoreID();
		Valid = Wdg_ChannelValidateCoreUsed(CoreID, Instance,WDG_SETTRIGGERCONDITION_ID,WDG_E_PARAM_CONFIG);
		if ((Std_ReturnType)E_OK == Valid)
		{
		#endif
		#if (WDG_VALIDATE_GLOBAL_CALL == STD_ON)
		Valid = Wdg_ChannelValidateTrigerCondition(Instance);
		if ((Std_ReturnType)E_OK == Valid)
		{
		#endif / (WDG_VALIDATE_GLOBAL_CALL == STD_ON /

ARTD-63527	Bug	[MemAcc] MemAcc_JobRetrying return not correct value when blankcheck operation determined a mismatch Detailed description (how to reproduce it): [SAF85_S32R41 1.0.0][MemAcc] MemAcc_JobRetrying return not correct value when the blankcheck operation determined a mismatch !image-2023-04-19-06-28-45-505.png! Preconditions: Test Case ID (internal TC that caught the defect) optional: Observed behavior: JobResult return MEMACC_MEM_FAILED Expected behavior: JobResult return MEMACC_MEM_INCONSISTENT Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-66938	New Feature	[os] Update copyright template NewWorkDescription: Replace : “(c) Copyright \{year range} NXP Semiconductors All Rights Reserved.” with : Copyright \{year range} NXP !screenshot-2.png! to !screenshot-3.png! Requirement source: as described in “Updating Copyright Years” topic on this guide [https://confluence.sw.nxp.com/display/OSS/File+Headers+-+Copyright+and+License+Information]). (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)

ID	Subtype	Headline and Description
		Proposed solution optional: as above
ARTD-66959	New Feature	[rte] Update copyright template NewWorkDescription: Replace : “(c) Copyright \{year range} NXP Semiconductors All Rights Reserved.” with : Copyright \{year range} NXP !screenshot-2.png! to !screenshot-3.png! Requirement source: as described in “Updating Copyright Years” topic on this guide [https://confluence.sw.nxp.com/display/OSS/File+Headers+-+Copyright+and+License+Information]). (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: as above
ARTD-67371	Bug	[WDG] Add M4 to enable the ClearResetRequest functions according to CPR_RTD_00434.wdg Detailed description (how to reproduce it): Traceability matrix report of release SJA11xx ASR 4.4 RTM 1.0.0 has a warning about Undefined Requirement refer to 'Wdg_ChannelClearResetRequest_Activity'. This is because SJA11XX doesn't support Clear Reset Request feature, but source code of this feature is exposed for all platform. Preconditions: N/A. Test Case ID (internal TC that caught the defect) optional: N/A. Observed behavior: Warning is generated in Traceability matrix report of release SJA11xx ASR 4.4 RTM 1.0.0 Expected behavior: There is no warning in Traceability Matrix Report. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Add M4 to guard all code related to ClearResetRequest functions.
ARTD-70300	Bug	[mcu] Cannot wake up S32M244 if the debugger is not attached I have the following configuration XS32M2XXCVB-Q064 (SCH-53099 REV A) with an M244C. I have one button that puts the MCU in VLPS mode and another that wakes it up. Everything works fine if the debugger is attached. When I put the MCU in VLPS mode without the debugger attached, I can't wake it up. If I attach the debugger after putting the MCU in VLPS, it is "core power down", whereas with the debugger attached in VLPS it was "running (sleeping)". I also tried STOP1 and STOP2 modes, and when I have the debugger attached everything works fine. When not attached, it's the same, I can't wake it up. The difference is that when I have the debugger attached, after I put the MCU in STOP1 or STOP2, it is not in "core power down" as in the VLPS. The state is 'running (sleeping)', it is stuck at WFI and now I can wake it up using the button, which didn't work before I attached the debugger.
ARTD-72698	Bug	[BASE] StandardType are missing suffix U in definition Detailed description (how to reproduce it) There are many macro define in StandardTypes.h are missing U as suffix for definition. Where it actually need to define as unsigned int according to Autosar Spec !image.png|width=464,height=223! eg: brief Logical state active. implements SymbolDefinitions_enum*/#define STD_ACTIVE 0x01 / brief Logical state idle. implements SymbolDefinitions_enum*/#define STD_IDLE 0x00 / brief ON State. implements SymbolDefinitions_enum*/#define STD_ON 0x01 / brief OFF state. implements SymbolDefinitions_enum*/#define STD_OFF 0x00 / brief Return code for failure/error. implements SymbolDefinitions_enum*/ #define E_NOT_OK 0x01 Preconditions: Using Base module Test Case ID (internal TC that caught the defect) optional: NONE Observed behavior:

ID	Subtype	Headline and Description
		Miss match in type Expected behavior: No miss match in type Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Define as Autosar Specification
ARTD-75301	Bug	Update requirement and design UML with changing for Clock_Ip_PowerModeChangeNotification Detailed description (how to reproduce it): Function Clock_Ip_PowerModeChangeNotification}} is changed to Clock_Ip_PowerNotifications}} in ticket ARTD-22762 The function name must be applied also for Requirement on DOORs and UML design Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: N/A Expected behavior: N/A Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-75549	Bug	[MCU] Driver code should disallow the configuration of clocks while the chip is in other modes than RUN_MODE Detailed description (how to reproduce it): RM mentions: !image-2023-07-08-10-00-23-257.png! The current implementation of the driver code is only restricting configuring clocks while the chip is in other modes than RUN_MODE if Default Error Detect is Enabled !image-2023-07-08-10-03-11-374.png! In the purpose of increasing the safety of MCU module. The line of code: CLOCK_IP_DEV_ASSERT(CLOCK_IP_RUN_POWER_MODE_STATUS == ((IP_SMC->PMSTAT & SMC_PMSTAT_PMSTAT_MASK) >> SMC_PMSTAT_PMSTAT_SHIFT)) should be replaced by checking IP_SMC->PMSTAT in Mcu_InitClock() and return ClockStatus = E_NOT_OK if the chip is in other modes than RUN_MODE Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: The current implementation of the driver code is only restricting configuring clocks while the chip is in other modes than RUN_MODE if Default Error Detect is Enabled Expected behavior: The line of code: CLOCK_IP_DEV_ASSERT(CLOCK_IP_RUN_POWER_MODE_STATUS == ((IP_SMC->PMSTAT & SMC_PMSTAT_PMSTAT_MASK) >> SMC_PMSTAT_PMSTAT_SHIFT)) should be replaced by checking IP_SMC->PMSTAT in Mcu_InitClock() and return ClockStatus = E_NOT_OK if the chip is in other modes than RUN_MODE Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: The line of code: CLOCK_IP_DEV_ASSERT(CLOCK_IP_RUN_POWER_MODE_STATUS == ((IP_SMC->PMSTAT & SMC_PMSTAT_PMSTAT_MASK) >> SMC_PMSTAT_PMSTAT_SHIFT)) should be replaced by checking IP_SMC->PMSTAT in Mcu_InitClock() and return ClockStatus = E_NOT_OK if the chip is in other modes than RUN_MODE
ARTD-76172	Bug	[MCU] Missing code generation for GPIO_CLK gate in s32ct s32k118 Detailed description (how to reproduce it): Step 1: Create s32ds project for s32k118 Step 2: Disable GPIO gate in SIM module and update code (picture 1)

ID	Subtype	Headline and Description
		Issue : GPIO gate can not be generate on output file. On eb is ok (picture 2) Preconditions: s32ct, s32k118 Test Case ID (internal TC that caught the defect) optional: Mcu_TC_FCT_0514 Observed behavior: Missing GPIO gate configuration in output generation file Expected behavior: Possible to generate GPIO gate Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-79139	Bug	[AE] EVENTS_STATUS[FRAMEWIDTH _FL] bit is sometimes set after calling AE_Init() Detailed description (how to reproduce it): After calling AE_Init(), EVENTS_STATUS[FRAMEWIDTH FL] bit is sometimes set (Indicates that the duration of an SPI access was not as expected). This causes the test to be unstable (randomly passed or failed). Then I added a delay before calling Ae_Init() > the test passed. After investigating, I see that the driver code is not implemented to check if AE is booted successfully before accessing to it. The issue I mention above shall come from this. !image-2023-07-21-17-24-19-451.png! Preconditions: EVENTS_ENABLE[FRAMEWIDTH_EN] bit must be set Test Case ID (internal TC that caught the defect) optional: Ae_TS_014 Observed behavior: Test randomly failed at Eu_assert() Expected behavior: Test passed Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-79415	Bug	[GPT] SRTC interrupt can not be raised more 2 times Detailed description (how to reproduce it): [Tester modifies a test use SRTC with 5 interrupt times, but notification function has been called only 2 times. !image-2023-07-25-09-31-50-973.png! The detail issue is below: At the time start timer, the truly value was loaded to compare register is "Target_time 1" and first time interrupt with Target_time 1. !image-2023-07-25-09-09-09-426.png! So, interrupt process of SRTC resets value of TAR register (TAR value = 0) !image-2023-07-25-09-20-07-747.png! and loads this Target_time by Global variable and second time interrupt with Target_time by function Srtc_Ip_ProcessAlarmInterrupt( ). !image-2023-07-25-09-20-45-512.png! !image-2023-07-25-09-21-16-953.png! When compute newAlarmTime, it will be equal 0 Target_time = Target_time. Since then, counter register counts from Target_time value but TAR value was fixed at Target_time and cannot interrupt any more. Preconditions: [Config SRTC IP with continuous mode, enable notification and interrupt more 2 times.] Test Case ID (internal TC that caught the defect) optional: [Ip_SRTC_TS_001, GPT_TS_001] Observed behavior: [Test and review driver code.] Expected behavior: [Although spurious interrupt protection by write to TAR to clear TAF but it is not necessary is 0 value.]

ID	Subtype	Headline and Description
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [Driver shall maintain old value of TAR to compute newAlarmTime.]
ARTD-79575	Bug	[WDG] WdgSettingsOff uses EDITABLE = FALSE attribute instead of READONLY Detailed description (how to reproduce it): In Wdm.xml file, the Containter WdgSettingsOff is using attribute "<a:a name="EDITABLE" value="false"/>" The container WdgSettingsOff is not used, but it still is implemented to follow the ASR REQ ECU_Wdg_00122. !image-2023-07-26-14-11-57-499.png|width=556,height=115! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: Should be use "<a:a name="READONLY" value="true"/>" Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Replace <a:a name="EDITABLE" value="false"/> by <a:a name="READONLY" value="true"/>
ARTD-80581	New Feature	[I2C] Add guard LPI2C_IP_COMMON_IRQ_MASTER_AND_SLAVE for function Lpi2c_Ip_ModuleIRQHandler NewWorkDescription: The function Lpi2c_Ip_ModuleIRQHandler is only used if LPI2C_IP_COMMON_IRQ_MASTER_AND_SLAVE is defined. Please investigate and guard all the codes related to this macro to reduce code size Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: N/A
ARTD-81124	Bug	[MCL] Update resource files with additional fields related to cache types available Detailed description (how to reproduce it): During S32N development some restriction were added related to CacheEnable field in Tresos. Because not all cores support L1 caches the CacheEnable field shall be editable only for cores that support L1 cache. For this reason the following fields were added in resource files. " Mcl.Cache.L1RCore:1 Mcl.Cache.L1MCore:0 Mcl.Cache.Lmem:0 " Because the rest of the platforms are missing the above information in resource files the CacheEnable field in Tresos is not editable thus the user is not able to enable Cache support. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: CacheEnable field in Tresos is not editable for platforms that support L1 cache. Expected behavior: CacheEnable field shall be editable for cores that support L1 cache and for those that do not support L1Cache the field shall be greydout. Proposed Solution: Changes made in ARTD-79915 shall be reverted and the resource files shall be updated (with the fileds mentioned above) for all platfomrs with L1 cache. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-81555	Bug	[ocu] Ocu_SetPinAction(...) has no effect when the channel is not running Detailed description (how to reproduce it): Call to the {{Ocu_SetPinAction(...)}} has no effect if the channel is not running.

ID	Subtype	Headline and Description
		Ocu_StopChannel(MY_CH); ... Ocu_SetPinAction(MY_CH, OCU_TOGGLE); / no effect / Ocu_StartChannel(MY_CH); Preconditions: S32Gx platform OCU channel is configured to have an associated pin Ocu_Init(...) was called Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Pin action is not changed if the API is called when the channel is not running. Expected behavior: If the module is initialized, and both channel and action values are valid. then the pin action should be configured regardless the running state of the channel. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: !image-2023-08-07-14-15-09-818.png!
ARTD-81655	Bug	[I2s]FXIO I2S force to enable both Tx&Rx even when only Tx used Detailed description (how to reproduce it): [After adding an FlexIO I2S component in .mex file, trying to enable only Tx or Rx] Preconditions: [Configured FXIO_mcl for I2S pin.] Test Case ID (internal TC that caught the defect) optional: [N/A] Observed behavior: [Found configure error reported by .mex tool. Besides, if both Tx&Rx enabled, the callback function will be invoked by Rx even we only use Tx sendData but not invoke Rx receive API.] Expected behavior: [The FXIO I2S Tx / Rx can be independently enabled and use, the callback function will not be invoked twice if we didn't initiate both Tx&Rx process.] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [N/A]
ARTD-83862	New Feature	[Mem_Infls] update Mem_43_JobResultType Mem_43_IPW_GetJobResult for other platform NewWorkDescription: update Mem_43_JobResultType Mem_43_IPW_GetJobResult follow this change as in PR ARTD-81703. Mem_43_JobType JobType > Mem_43_JobRuntimeInfoType JobInfo Test Case ID (internal TC that caught the defect) optional: Mem_InFls_TC_0001 Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: update second input : Mem_43_JobType JobType > Mem_43_JobRuntimeInfoType JobInfo
ARTD-84809	Bug	[ENET][SJA11XX] ENET driver not according to UM/spec Detailed description (how to reproduce it): The ENET driver is not according to the UM. When setting the READY flag of a buffer descriptor, no more changes to the fields are allowed. See description of the ENET for SJA1110: !image-2023-08-31-16-09-20-218.png! But the driver changes some fields after setting the READY flag in line 1061

ID	Subtype	Headline and Description
		!image-2023-08-31-16-08-45-821.png! This can be an issue when trying to send/enqueue multiple frames fast. As the ENET is shared, this might affect also other products. However, many SW example hide this issue by sending frames "slowly". But this is unwanted in certain cases. Expected behavior: Driver is implemented according to UM. Proposed solution optional: Without verifying, but seems like setting READY (ENET_BUFFDESCR_TX_READY_MASK) should happen directly before ENET_ActivateTransmit, such that no changes to the BD are made after setting READY.
ARTD-85309	New Feature	[Base]Definition of E_MEM_SERVICE_NOT_AVAIL shouldn't be in Std_ReturnType Detailed description (how to reproduce it): Latest MCAL version SW32K3_RTD_R21-11_3.0.0_P07 has the define of E_MEM_SERVICE_NOT_AVAIL in StandardTypes.h. In our opinion, module specific result codes should not be added to this file. It should be added to a driver specific file like MemAcc_GeneralTypes.h. In ASR R21-11, we have: !image-2023-07-24-13-45-51-189.png|thumbnail! An example, This is also done for other extensions to Std_ReturnType, see for example the Crypto module. !image-2023-07-24-13-46-21-546.png|thumbnail! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: E_MEM_SERVICE_NOT_AVAIL added to generic type Std_ReturnType Expected behavior: E_MEM_SERVICE_NOT_AVAIL wasn't added to generic type Std_ReturnType, it should be added to specific file such as MemAcc_GeneralTypes.h. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Make Mem_Acc_General_Types.h into Base driver, remove E_MEM_SERVICE_NOT_AVAIL in StandardTypes.h
ARTD-85428	Bug	[Mem_ExFls] Jenkins build failure running on Linux OS due to inconsistent casing in the driver Observed Jenkins build failure while using SW32SAF85xx_RTD_R21-11_2.0.0_CD01 with Mem_43_EX_FLS module. The reason is that filenames are case sensitive. (Logs attached file name mem_43_flslog.txt) Environment* *used:* Jenkins node running on Linux OS RTD plugin* *issue:* Module: Mem_43_EX_FLS Source code includes a header file as Mem_43_EXFLS_{*}IPW{*}.h but the file saved in inc folder in lower case as Mem_43_EXFLS_{*}Ipw.h,{*} Source file location c file that is giving call to header file with Upper* *case:* : C:\NXP \SW32SAF85xx_RTD_R21-11_2.0.0_CD01\eclipse\plugins\Mem_43_EXFLS_TS_T40D47M20I0R0\src !image-2023-08-17-11-57-59-998.png! Header file location with lower case header file : C:\NXP\SW32SAF85xx_RTD_R21-11_2.0.0_CD01\eclipse\plugins \Mem_43_EXFLS_TS_T40D47M20I0R0\include !image-2023-08-17-12-04-07-703.png!
ARTD-86092	Bug	[GPT]: The API:Gpt_GetTimeElapsed will return wrong value for the counting up type timer Detailed description (how to reproduce it): The API:Gpt_GetTimeElapsed will return wrong value for the counting up type timer. !image-2023-09-14-18-29-46-370.png! Preconditions: None

ID	Subtype	Headline and Description
		Test Case ID (internal TC that caught the defect) optional: None Observed behavior: Return a big wrong value Expected behavior: return correct value Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: add exclusive proect
ARTD-86398	Bug	[S32ZE 1.0.0] CAN: Update Generate Code Template to generate CAN_CONFIG_EXT only in PB mode Detailed description (how to reproduce it): The collection of all configuration structure declarations now generates all of both PC and PB Modes. #define CAN_CONFIG_EXT With PC mode it is not created, it is only created in PB Mode. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: The macro still be genrated for PreCompile and Link Time Expected behavior: The macro is only generated for Post Build Proposed solution optional: [....]
ARTD-112685	Bug	[memacc] The driver should not be hardware-dependent Detailed description (how to reproduce it): # The driver should not be hardware-dependent. It is the job of the Mem drivers to abstract the hardware details. Currently the MemAcc configuration contains the device name, which is taken from resources: !image-2024-05-23-11-25-46-655.png! # Device name is imported from the Mem driver and used to ensure that two jobs for the same device are not done in parallel. But this is a non-standard configuration item, so it should not be imported, because the MemAcc should work with any Mem implementation. Instead, the Mem driver instance should be used. One Mem driver instance corresponds to a device and can only perform one job at a time. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: MemAcc configuration contains device name (e.g. XSPI) Expected behavior: The driver should not be hardware-dependent Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: the Mem driver instance should be used
ARTD-112683	Bug	[memacc] Fail status for pending job Detailed description (how to reproduce it): There is an inconsistency of the job status and job result (MemAcc_GetJobStatus, MemAcc_GetJobResult) when a Mem job fails and is retried. The result is updated in the Job info structure and from now on MemAcc_GetJobResult will return MEMACC_MEM_FAILED, even though MemAcc_GetJobStatus still returns MEMACC_JOB_PENDING. This can be confusing for the application, and not quite in line with the standard: Note: If a MemAcc job is still pending, the API returns the result of the last MemAcc job. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior:

ID	Subtype	Headline and Description
		MemAcc_GetJobResult will return MEMACC_MEM_FAILED, even though MemAcc_GetJobStatus still returns MEMACC_JOB_PENDING Expected behavior: MemAcc_GetJobResult should updated status only when job is complete Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:
ARTD-95839	New Feature	[memacc] [IMPLEMENTATION] Multicore Type 1 for MEMACC Driver NewWorkDescription: Implement the multicore "Type 1" for the MEMACC Driver Requirement source: (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Reference the way to implement the multicore "Type 1" on the FLS Driver (AUTOSAR 4.4) Document: NXP_RealTimeDrivers_Multicore.pptx; Autosar Memstack Multicore.pptx
ARTD-95842	New Feature	[MemAcc] [IMPLEMENTATION] Multicore Type 3 for MEMACC Driver NewWorkDescription: Implement the multi partition "Type 3" for the MEMACC Driver Each Address area will have a list of partition which is configurable. When a partition request a job on MemAcc, the driver will check and only accept if the address area is configured for current partition. Requirement source: (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Reference the way to implement the multicore "Type 3" on the FLS Driver (AUTOSAR 4.4) Document: NXP_RealTimeDrivers_Multicore.pptx; Autosar Memstack Multicore.pptx
ARTD-96050	Bug	[I2c] Configuration imported with error from Tresos to Design Studio All the errors have been fixed, the code generated, and build. When the Application was exported from Tresos to Design Studio, in DS we have the error from the image attached. The I2c Baud Rate in Tresos was set to 0 and in DS it have an outrange value. Also if you want set another value/default value it will be overwrite with the same outrange value. please take a look at the following picture, I saw that there is a setValue function called. So doesn't matter what value we set, all the time will have that outrange value. !image-2023-08-01-16-51-05-886.png!
ARTD-96056	New Feature	[I2C][k1] Investigate the interrupts will be used by LPI2C in the interrupt handlers NewWorkDescription: Currently the LPI2C driver is using all the interrupts from hardware register to validate the spurious interrupt in Lpi2c_Ip_MasterIRQHandler and {*}Lpi2c_Ip_SlaveIRQHandler{*}: !image-2023-07-13-11-16-49-766.png! But some of them are not used / not enabled by the driver and should be excluded from that list. Please investigate and keep only the necessary ones. Also investigate and find a better solution to replace the guard for S32K3XX: !image-2023-07-13-11-19-50-566.png! Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: N/A
ARTD-96358	New Feature	[Dpga] S32M24x implement the feature Functional self-test for the amplifier and the voltage monitoring NewWorkDescription: According S32M24x Reference Manual, Rev. 2 Draft A, 05/2023 implement the feature "Functional self-test for the amplifier and the voltage monitoring" !image-2023-06-23-18-00-50-737.png! Requirement source: NA

ID	Subtype	Headline and Description
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: NA
ARTD-96452	Bug	[platform] Remove adhoc CCOPT values and use generated values based on resource Detailed description (how to reproduce it): Remove custom CCOPT and use generated values based on resource used see [PR comment|https://bitbucket.sw.nxp.com/projects/ARTD/repos/platform/pull-requests/751/overview?commentId=2165839] Remove the mentions from UM/IM as well Preconditions: None Test Case ID (internal TC that caught the defect) optional: None Observed behavior: Custom CCOPT are requested to be used by customer to build the code for a specific NPI these values should be generated based on configuration and not forced in CCOPT in makefiles Expected behavior: Use generated values Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-96947	Bug	[S32K3xx_S32M27x_4.0.0] [DIO] Different sort order of Channel ID when importing epc file from EBT into S32DS Detailed description (how to reproduce it): Different sort order of channel ID when importing epc file from EBT into S32DS Preconditions: The order of channel ID arrangement when porting from EB to S32DS must be the same Test Case ID (internal TC that caught the defect) optional Dio_TS_010 Observed behavior: Config EB: !image-2023-10-18-09-51-11-444.png|width=751,height=447! Import to CT The order of channel id is changed leading to run test fail !image-2023-10-18-10-11-18-204.png|width=675,height=382! The difference of the generate file when porting from EB to CT leads to run test fail !image-2023-10-18-09-58-09-457.png|width=822,height=349! Expected behavior Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional
ARTD-97735	New Feature	[memexfls] [IMPLEMENTATION] Update code to support multicore Type 3 for MEMACC Driver NewWorkDescription: Move all global variable to Share no cache region. Add check core ID for each job. Only the core that requests the job can perform the job in the mainfunction Requirement source: (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Reference the way to implement the multicore "Type 3" on the FLS Driver (AUTOSAR 4.4) Document: NXP_RealTimeDrivers_Multicore.pptx; Autosar Memstack Multicore.pptx
ARTD-97810	Bug	[ETH] Schema file is not using VendorApiInfix In Eth_43_GMAC.xdm VendorApiInfix is configured as OPTIONAL true : <a:a name="OPTIONAL" value="true" <a:a name="READONLY" value="false" Since Eth driver is using VendorApiInfix the lines should be replaced with : <a:a name="OPTIONAL" value="false"

ID	Subtype	Headline and Description
		<a:a name="READONLY" value="true"
ARTD-98067	New Feature	[memInfls] [IMPLEMENTATION] Update code to support multicore Type 3 for MEMACC Driver NewWorkDescription: Move all global variable to Share no cache region. Add check core ID for each job. Only the core that requests the job can perform the job in the mainfunction Requirement source: (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Reference the way to implement the multicore "Type 3" on the FLS Driver (AUTOSAR 4.4) Document: NXP_RealTimeDrivers_Multicore.pptx; Autosar Memstack Multicore.pptx
ARTD-98070	New Feature	[mem_eep] Update code to support multicore Type 3 for MEMACC Driver NewWorkDescription: Move all global variable to Share no cache region. Add check core ID for each job. Only the core that requests the job can perform the job in the mainfunction Requirement source: (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Reference the way to implement the multicore "Type 3" on the FLS Driver (AUTOSAR 4.4) Document: NXP_RealTimeDrivers_Multicore.pptx; Autosar Memstack Multicore.pptx
ARTD-98313	Bug	[S32K1] OCOTP: No error for unsupported derivative Detailed description (how to reproduce it): This is a clone ticket from S32K3 bug ticket (Same issue). Install the release, create a new project, add Ocotp, set the resources s32k144_mapbga100, generate code for the current project. Preconditions: EBT 29.2.0 Test Case ID (internal TC that caught the defect) optional: Observed behavior: Failed to run generator for Ocotp: !image-2023-05-30-14-46-23-405.png! Expected behavior: Should receive an warning message which tells us that for S32K1 derivative, Ocotp is unsupported. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-98759	Bug	[GPT] Function Gpt_GetTimeElapsed() returns wrong value with FTM Detailed description (how to reproduce it): MCAL-22136, MCAL-24446, MCAL-24449 Kostal use S32K14X for thier BMW project. They meet the WDG and GPT issue which is similar with MCAL-22136, MCAL-24446, MCAL-24449. Gtp uses FTM, feed dog cycle is 500ms, WdgIf_SetTriggerCondition() call cycle is 10ms, and occasionally Gpt_GetTimeElapsed() returns a negative value during operation, resulting in the flowing problem: condition "if (((uint16)(uElapsedTime/Wdg_apConfigPtr[Wdg_Instance]->Wdg_u32TriggerSourceClock) > Wdg_au16Timeout[Wdg_Instance]) ((uint16)0 == u16Timeout))" in function Wdg_ChannelSetTriggerCondition() satisfies ; generate a reset caused by a watchdog. Debugging found that when the return value of Gpt_GetTimeElapsed() is negative, the values of u32CompareValue and u32CounterValue in the function Gpt_Ftm_GetTimeElapsed() are equal. At this time, an interrupt should be generated, but due to exclusive area protection, the interrupt is delayed and the acquired Elapedtime is also incorrect. Preconditions: [None] Test Case ID (internal TC that caught the defect) optional: [..None.] Observed behavior: [...None] Expected behavior: Gpt_GetTimeElapsed should be called out side exclusive are.

ID	Subtype	Headline and Description
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-98780	Bug	[AE] Fix build fail in CT config Detailed description (how to reproduce it): The raw bit of node AE does not exist so it causes this error. !image-2023-11-02-14-31-28-239.png! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Build fail on CT Expected behavior: Build pass on CT Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-98828	New Feature	CLONE - [SPI] Add supporting Byte Swap configuration NewWorkDescription: Add supporting Byte Swap configuration !image-2023-08-11-16-12-56-569.png|thumbnail! Requirement source: Customer Request (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: [....]
ARTD-98951	Bug	[AE]Fix limitation when config on both EB and CT Detailed description (how to reproduce it): When config on EB and CT, the error will raised if we don't have ref link in node AePlatformSourceName. !image-2023-11-03-10-13-58-752.png! !image-2023-11-03-10-12-28-162.png! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: The error raised if won't choose source name platform. Expected behavior: No error raised if won't choose source name platform. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-99476	Bug	Port: Macro "PORT_CONFIG_EXT" generated in Precompile mode Detailed description (how to reproduce it): Create new project (DS and CT) Add port component, add pins, config variant = Precompile Generate and check Port_Cfg.h Preconditions: Config variant = Precompile Test Case ID (internal TC that caught the defect) optional: Port_TS_014 Observed behavior:

ID	Subtype	Headline and Description
		Macro "PORT_CONFIG_EXT" generated in Precompile mode !image-2023-09-13-17-28-33-901.png|width=803,height=800! !image-2023-09-13-17-30-40-623.png! Expected behavior: Macro "PORT_CONFIG_EXT" should not be used in PreCompile mode Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-99511	Bug	[WDG] Wdg in EB configuration does not report error when Initial Timer smaller than Wdg Timeout Detailed description (how to reproduce it): In EB Configure Initial timeout: 1s Configure Wdg Timeout Period: 2s Preconditions: Test Case ID (internal TC that caught the defect) optional: Wdg_TC_FCT_0301 Observed behavior: EB configuration does not report error when Initial timeout < Wdg Timeout Period. Expected behavior: EB configuration report error when Initial timeout < Wdg Timeout Period. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-99851	Bug	[Mem_InFls] MemWritePageSize dost not work as in Autosar Spec description Detailed description (how to reproduce it): MemWritePageSize dost not work as in Autosar Spec description => it reduces running performance. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Each write corresponds to this value eg: when user config MemWritePageSize = 8 and want to write 1024 btye => So it takes 1024/8 loops to write all 1024 bytes. Expected behavior: This value is just the smallest unit that the mem can handle Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-100063	Bug	[WDG] Disable WdgClockReferencePoint container from EB and DS interface for AeWdog instance Detailed description (how to reproduce it): The container WdgClockReferencePoint is not used for AeWdog instance. It should be disabled from CT and EB interface Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: WdgClockReferencePoint can be set but it is not used in any code generate file. Expected behavior: WdgClockReferencePoint is disabled Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:

ID	Subtype	Headline and Description
		NA
ARTD-100125	Bug	[PORT]: Port_Ipw_SetGpioPadOutput and Port_Ipw_SetGpioDirChangeability should be static Detailed description (how to reproduce it): These functions do not represent the API of driver. It should be static. Preconditions: Test Case ID (internal TC that caught the defect) optional: Observed behavior: Expected behavior: Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:
ARTD-100218	Bug	[MCU] Missing implementations of macros notifying power error detections Detailed description (how to reproduce it): 1. POWER_IP_ISR_VOLTAGE_HVD_VDDINT_DETECT POWER_IP_ISR_VOLTAGE_HVD_15_DETECT POWER_IP_ISR_VOLTAGE_HVD_VDD_DETECT POWER_IP_ISR_VOLTAGE_LVD_VDDC_DETECT POWER_IP_ISR_VOLTAGE_LVD_VLS_DETECT are missing in Mcu_Ipw_ReportPowerErrorsCallback !image-2023-10-30-09-35-24-126.png! 2. POWER_IP_ERROR_ISR_NOTIFICATION POWER_IP_DETECT_ISR_NOTIFICATION are not implemented to be able to be generated 3. Redundant parameters of POWER_IP_DETECT_ISR_NOTIFICATION POWER_IP_DETECT_ISR_NOTIFICATION is using 2 parameters but a normal NOTIFICATION only uses 1 parameter !image-2023-10-30-09-43-29-431.png! !image-2023-10-30-09-43-49-168.png! 4. Power_Ip_PMC_AE_VoltageDetectLvdOnVddcIsr and Power_Ip_PMC_AE_VoltageDetectLvdOnVlsIsr are using wrong macros !image-2023-10-30-09-46-33-793.png! !image-2023-10-30-09-46-49-673.png! they should be POWER_IP_ISR_VOLTAGE_LVD_VDDC_DETECT and POWER_IP_ISR_VOLTAGE_LVD_VLS_DETECT Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Missing implementations of macros notifying power error detections Expected behavior: Correct implementations of macros notifying power error detections Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-100254	Bug	[Base] Copyright issues Detailed description (how to reproduce it): Please review the attached log and correct in driver template the copyright issues Preconditions: Plugin built Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: wrong copyright template Expected behavior: no issues on copyright Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)

<table>
<tr><th>ID</th><th>Subtype</th><th>Headline and Description</th></tr>
<tr><td></td><td></td><td>Proposed solution optional:
NA</td></tr>
<tr><td>ARTD-100613</td><td>New Feature</td><td>[WDG] Update documentation with note about fix for Multiple instances of Wdg cannot generate in parallel

NewWorkDescription:
An issue will be occurred with multiple Wdg instances generation in parallel. The reason for this issue is, only a single plugin folder for Wdg is used to supports multiple instances, and the folders like "generate", "generate_PC", "generate_PB", "generate_swcd" are common. They are all accessed from plugin xml for each instance from the TemplateBasedCodeGenerator. We should warn users about Multiple instances of Wdg cannot generate in parallel.

Requirement source:
Customer Request

Proposed solution optional:
Add the following setting in TemplateBasedCodeGenerator of WDG instances:

!image-2022-12-01-18-51-26-654.png!

2. Add a note in UM/IM about this limitation.</td></tr>
<tr><td>ARTD-101855</td><td>Bug</td><td>[MCU] PMC_AE_MONITOR[LVDCF] flag is not cleared by Mcu driver after AE reset

As stated by RM, PMC_AE_MONITOR[LVDCF] is always set after AE reset

!image-2023-11-24-17-51-48-768.png!

Mcu doesn't clear flags anymore but In order to work properly, CanPHY IP expects the flag must be cleared.

other than that. the flags should be cleared/handled after reset in order to not loss event after that.

!image-2023-11-24-17-54-59-912.png!

please help to investigate it in Mcu driver if it should clear the flags or not, now when using CanTrcv driver, we must clear the flag manually in our tests.</td></tr>
<tr><td>ARTD-105159</td><td>New Feature</td><td>[IMPLEMENTATION] Fast APIs for ADC readings

CR description:

There was an analysis regarding motor control application for S32K1 and S32M2 using RTDs. The effort started migrating the motor control software from SDK to RTD and then report the findings/limitations founded in the process.

One important thing we noticed when migrating MCAL motor control application is that PWM update API and ADC read APIs takes much more time to execute comparing with the LLD API. Please check below the table with comparison.
 |*PWM Update*|*Execution time
RTD Low Level*|Ftm_Pwm_Ip_FastUpdatePwmDuty (includes 3 differential pair update and Sync update)|*~1.91 us
 **
RTD MCAL*|Pwm_SetDutyCycle_NoUpdate(PwmChannel_0, (uint16)pwmDutyA)
Pwm_SetDutyCycle_NoUpdate(PwmChannel_1, (uint16)pwmDutyB);
Pwm_SetDutyCycle_NoUpdate(PwmChannel_2, (uint16)pwmDutyC);
Pwm_SyncUpdate(INST_FTM3);|*~52.13 us
Baremetal*|FTM3->CONTROLS[0].CnV
FTM3->CONTROLS[2].CnV
FTM3->CONTROLS[4].CnV
FTM3->SYNC \|= FTM_SYNC_SWSYNC_MASK;|*~1.4375 us

 |*ADC Update*|*Execution time
RTD Low Level*|Phase A current Adc_Ip_GetConvData(INST_ADC0, 0)
Phase B current Adc_Ip_GetConvData(INST_ADC1, 1)
DC Bus Voltage Adc_Ip_GetConvData(INST_ADC1, 0)|*~1.81 us
 **
RTD MCAL*|Adc_ReadRawData(0, &ADC0CH0_array, 1, &PhaseA_Current);
Adc_ReadRawData(1, &ADC1CH1_array, 1, &PhaseB_Current);
Adc_ReadRawData(1, &ADC1CH0_array, 1, &DCBus_Voltage);|*~14.08 us
Baremetal*|Phase A current ADC0->R[0]
Phase B current ADC1->R[1]
DC Bus Voltage ADC1->R[0]|*~1.225 us
As you can see the execution time is quite higher when measuring the MCAL APIs.

The proposal is to keep existing functions for users which prefer universal functions with higher level of abstraction and add new functions for users who need fast PWM update, and ADC read.

Adding new functions for drivers was discussed in a meeting with Ioana-Andreea Pocea and Tomas Fedor on July 10th.
Reason for this change:

These new functions will help us to provide motor control software using AUTOSAR for customers interested in S32K1 and S32M24x. For S32M2 is particularly special as this device is about to be release next year and is targeted specifically for motor control applications.
Benefit:

Wider free space for user calculations/algorithms will put S32K1 and S32M24 into better position.

Also these fast APIs will help us run motor control application using AUTOSAR.
Use-case:

Motor control area.
HW/Application Engineer contact (as applies):</td></tr>
</table>

ID	Subtype	Headline and Description
		Felipe Garcia felipe.garcia_1@nxp.com Note: relevant documents to be attached to the ticket.
ARTD-102167	Bug	[crc] Validation of partition reference should be OS version independent Detailed description (how to reproduce it): Some RTD drivers (Adc, Crc, Dio, Icu, Platform, Port, Pwm, Serdes) have the following check when [drv]MulticoreEnabled = 'true': <a:tst expr="count(text:grep(node:refs('ASPathDataOfSchema:/TS_T40D11M40I2R0/Os')/OsApplication/*/OsAppEcucPartitionRef, .)) = 0" true="The ECUC partition must be defined in OS."/> RTD shall not validate references against a certain OS version. There can be any OS version, either in a Tresos compatible plugin format or not. Creating a dependency to a certain version will cause the following validation error: !https://jira.sw.nxp.com/secure/attachment/1295507/image-2023-11-27-16-04-31-321.png! Preconditions: [drv]MulticoreEnabled is set to 'true' and a correct configuration of the partition is done in OS. Test Case ID (internal TC that caught the defect) optional: none Observed behavior: The following drivers Adc, Crc, Dio, Icu, Platform, Port, Pwm, Serdes show validation error is Tresos even when a correct reference to a partition is configured. Expected behavior: No error should be caused if different OSes are used. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Option 1) Change node:refs('ASPathDataOfSchema:/TS_T40D11M40I2R0/Os')/OsApplication/*/OsAppEcucPartitionRef To something like: as:modconf('Os')[1]/OsApplication/*/OsAppEcucPartitionRef as found in CAN. Option 2) Alternatively this check can be removed altogether since drivers should not be aware of Os partition configuration. The reason is simple: partition configuration is part of high level components like: EcuM, BSWM and RTE, all above driver level.
ARTD-102182	Bug	[dio] Validation of partition reference should be OS version independent Detailed description (how to reproduce it): Some RTD drivers (Adc, Crc, Dio, Icu, Platform, Port, Pwm, Serdes) have the following check when [drv]MulticoreEnabled = 'true': <a:tst expr="count(text:grep(node:refs('ASPathDataOfSchema:/TS_T40D11M40I2R0/Os')/OsApplication/*/OsAppEcucPartitionRef, .)) = 0" true="The ECUC partition must be defined in OS."/> RTD shall not validate references against a certain OS version. There can be any OS version, either in a Tresos compatible plugin format or not. Creating a dependency to a certain version will cause the following validation error: !https://jira.sw.nxp.com/secure/attachment/1295507/image-2023-11-27-16-04-31-321.png! Preconditions: [drv]MulticoreEnabled is set to 'true' and a correct configuration of the partition is done in OS. Test Case ID (internal TC that caught the defect) optional: none Observed behavior: The following drivers Adc, Crc, Dio, Icu, Platform, Port, Pwm, Serdes show validation error is Tresos even when a correct reference to a partition is configured. Expected behavior: No error should be caused if different OSes are used.

ID	Subtype	Headline and Description
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Option 1) Change node:refs('ASPathDataOfSchema:/TS_T40D11M40I2R0/Os')/OsApplication/*/OsAppEcucPartitionRef To something like: as:modconf('Os')[1]/OsApplication/*/OsAppEcucPartitionRef as found in CAN. Option 2) Alternatively this check can be removed altogether since drivers should not be aware of Os partition configuration. The reason is simple: partition configuration is part of high level components like: EcuM, BSWM and RTE, all above driver level.
ARTD-102242	Bug	[memacc] Validation of partition reference should be OS version independent Detailed description (how to reproduce it): Some RTD drivers (Adc, Crc, Dio, Icu, Platform, Port, Pwm, Serdes) have the following check when [drv]MulticoreEnabled = 'true': <a:tst expr="count(text:grep(node:refs('ASPathDataOfSchema:/TS_T40D11M40I2R0/Os')/OsApplication/*/OsAppEcucPartitionRef, .)) = 0" true="The ECUC partition must be defined in OS."/> RTD shall not validate references against a certain OS version. There can be any OS version, either in a Tresos compatible plugin format or not. Creating a dependency to a certain version will cause the following validation error: !https://jira.sw.nxp.com/secure/attachment/1295507/image-2023-11-27-16-04-31-321.png! Preconditions: [drv]MulticoreEnabled is set to 'true' and a correct configuration of the partition is done in OS. Test Case ID (internal TC that caught the defect) optional: none Observed behavior: The following drivers Adc, Crc, Dio, Icu, Platform, Port, Pwm, Serdes show validation error is Tresos even when a correct reference to a partition is configured. Expected behavior: No error should be caused if different OSes are used. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Option 1) Change node:refs('ASPathDataOfSchema:/TS_T40D11M40I2R0/Os')/OsApplication/*/OsAppEcucPartitionRef To something like: as:modconf('Os')[1]/OsApplication/*/OsAppEcucPartitionRef as found in CAN. Option 2) Alternatively this check can be removed altogether since drivers should not be aware of Os partition configuration. The reason is simple: partition configuration is part of high level components like: EcuM, BSWM and RTE, all above driver level.
ARTD-102275	Bug	[port] Validation of partition reference should be OS version independent Detailed description (how to reproduce it): Some RTD drivers (Adc, Crc, Dio, Icu, Platform, Port, Pwm, Serdes) have the following check when [drv]MulticoreEnabled = 'true': <a:tst expr="count(text:grep(node:refs('ASPathDataOfSchema:/TS_T40D11M40I2R0/Os')/OsApplication/*/OsAppEcucPartitionRef, .)) = 0" true="The ECUC partition must be defined in OS."/> RTD shall not validate references against a certain OS version. There can be any OS version, either in a Tresos compatible plugin format or not. Creating a dependency to a certain version will cause the following validation error: !https://jira.sw.nxp.com/secure/attachment/1295507/image-2023-11-27-16-04-31-321.png! Preconditions: [drv]MulticoreEnabled is set to 'true' and a correct configuration of the partition is done in OS.

ID	Subtype	Headline and Description
		Test Case ID (internal TC that caught the defect) optional: none Observed behavior: The following drivers Adc, Crc, Dio, Icu, Platform, Port, Pwm, Serdes show validation error is Tresos even when a correct reference to a partition is configured. Expected behavior: No error should be caused if different OSes are used. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Option 1) Change node:refs('ASPathDataOfSchema:/TS_T40D11M40I2R0/Os')/OsApplication/*/OsAppEcucPartitionRef To something like: as:modconf('Os')[1]/OsApplication/*/OsAppEcucPartitionRef as found in CAN. Option 2) Alternatively this check can be removed altogether since drivers should not be aware of Os partition configuration. The reason is simple: partition configuration is part of high level components like: EcuM, BSWM and RTE, all above driver level.
ARTD-102314	Bug	[wdg] Validation of partition reference should be OS version independent Detailed description (how to reproduce it): Some RTD drivers (Adc, Crc, Dio, Icu, Platform, Port, Pwm, Serdes) have the following check when [drv]MulticoreEnabled = 'true': <a:tst expr="count(text:grep(node:refs('ASPathDataOfSchema:/TS_T40D11M40I2R0/Os')/OsApplication/*/ OsAppEcucPartitionRef, .)) = 0" true="The ECUC partition must be defined in OS."/> RTD shall not validate references against a certain OS version. There can be any OS version, either in a Tresos compatible plugin format or not. Creating a dependency to a certain version will cause the following validation error: !image-2023-11-27-16-04-31-321.png! Preconditions: [drv]MulticoreEnabled is set to 'true' and a correct configuration of the partition is done in OS. Test Case ID (internal TC that caught the defect) optional: none Observed behavior: The following drivers Adc, Crc, Dio, Icu, Platform, Port, Pwm, Serdes show validation error is Tresos even when a correct reference to a partition is configured. Expected behavior: No error should be caused if different OSes are used. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Option 1) Change node:refs('ASPathDataOfSchema:/TS_T40D11M40I2R0/Os')/OsApplication/*/OsAppEcucPartitionRef To something like: as:modconf('Os')[1]/OsApplication/*/OsAppEcucPartitionRef as found in CAN. Option 2) Alternatively this check can be removed altogether since drivers should not be aware of Os partition configuration. The reason is simple: partition configuration is part of high level components like: EcuM, BSWM and RTE, all above driver level.
ARTD-102484	Bug	[Mem_InFls] Generated files got missing names when nodename too long Detailed description (how to reproduce it): Put name of MemInstance longer (such as MemInstance_Local_Config_Bank_A) !image-2023-11-30-14-58-44-463.png!

ID	Subtype	Headline and Description
		Generate project and check Mem_43_INFLS_Cfg.c file Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Mem_43_INFLS_MemInstances definition has wrong SectorBatch's name: Mem_43_INFLS_MemInstance_Local_Config_Bank_A_SectorB cause errors !image-2023-11-30-15-00-57-087.png! Expected behavior: Correct name as Mem_43_INFLS_MemInstance_Local_Config_Bank_A_SectorBatch, Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-102661	Bug	[Uart] Uart header file shall follows the naming convention as CDD_Uart.h Detailed description (how to reproduce it): For below CDD modules, include files are not prefixed with CDD in *Uart* plugin, CDD_Uart.h i added with below macro inclusion #ifndef UART_H #define UART_H instead, should have been as #ifndef CDD_UART_H #define CDD_UART_H Preconditions: #ifndef CDD_UART_H #define CDD_UART_H Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: as Detailed description Expected behavior: as Preconditions Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: as Preconditions
ARTD-104259	Bug	[MemAcc]: Fee swap error when the read & write burst size configured to 128. Detailed description (how to reproduce it): Swap error when the read & write burst size configured to 128. when decrease the burst read size to 1, and the burst write size to 8, it works normally. The issue come from MemAcc, when Fee request a length =128, and the requested area is on 2 sub address area (see the picture for easier to understand). MemAcc should split 2 request for each SubAddress area (Mem Instance). But currently, MemAcc only request the first subaddress with length =128. > The mem driver will not accept the job because the length is over the size of the mem instance > job fail. Please refer to the attached project which based on the default example. [^Fee_Example_S32K344.7z] Preconditions: read & write burst size configured to 128 and trigger swap. Test Case ID (internal TC that caught the defect) optional: None Observed behavior: FEE return error, can't swap as expected. Expected behavior: None Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update code to split request for 2 SubAddress when the length is bigger than remaining length of the first SubAddress
ARTD-104792	Bug	[gpt] Ftm_Gpt_Ip does not start after it is stopped

<table>
<tr><th>ID</th><th>Subtype</th><th>Headline and Description</th></tr>
<tr><td></td><td></td><td>Detailed description (how to reproduce it):
When Ftm_Gpt is stopped by Ftm_Gpt_Ip_StopTimer function, it cannot be started again by Ftm_Gpt_Ip_StartTimer function. The reason is Ftm_Gpt_Ip_StopTimer_ disables the clock to the Ftm but Ftm_Gpt_Ip_StartTimer_ does not re-enable it. In the previous version the function Ftm_Gpt_Ip_StartTimer enabled the clock.

!FtmGpt2.PNG|width=603,height=254!

!FtmGpt1.PNG|width=622,height=170!

Preconditions:
As described above

Test Case ID (internal TC that caught the defect) optional:
N/A

Observed behavior:
As described above

Expected behavior:
Ftm_Gpt_StartTimer to start the timer after it was stopped as in the previous version clock to the module need to be enabled/ selected

Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)

Proposed solution optional:
[....]</td></tr>
<tr><td>ARTD-104966</td><td>Bug</td><td>I2C driver needs added protection against potential race conditions

Detailed description (how to reproduce it):

There is the following code in the Lpi2c_Ip_MasterSendDataBlocking:

if(Master->I2cIdle)
{
Master->BufferSize = TxSize;
Master->DataBuffer = TxBuff;
Master->Direction = LPI2C_IP_SEND;
Master->SendStop = SendStop;
Master->I2cIdle = FALSE;
Master->Status = LPI2C_IP_BUSY_STATUS;

Means, first there is a check if the bus is idle and then it is set to busy. It is possible that a task executes the if clause, but doesn't reach the bus busy setting, yet, because it does get interrupted by another task. That other task could also issue an i2c transfer, happily seeing that the bus is still idle.
A "test-and-set" mechanisms needs to be implemented to protection against race-conditions, i.e. adding task suspend/resume for atomically checking and setting Master->I2cidle.

Preconditions:
NA

Test Case ID (internal TC that caught the defect) optional:
NA

Observed behavior:
See description

Expected behavior:
Race conditions should not occur

Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)

Proposed solution optional:
[....]</td></tr>
<tr><td>ARTD-105162</td><td>New Feature</td><td>[pwm] Fast APIs for PWM update

CR description:

There was an analysis regarding motor control application for S32K1 and S32M2 using RTDs. The effort started migrating the motor control software from SDK to RTD and then report the findings/limitations founded in the process.

One important thing we noticed when migrating MCAL motor control application is that PWM update API and ADC read APIs takes much more time to execute comparing with the LLD API. Please check below the table with comparison.
 |*PWM Update*|*Execution time
RTD Low Level*|Ftm_Pwm_Ip_FastUpdatePwmDuty (includes 3 differential pair update and Sync update)|*~1.91 us
 **
RTD MCAL*|Pwm_SetDutyCycle_NoUpdate(PwmChannel_0, (uint16)pwmDutyA)
Pwm_SetDutyCycle_NoUpdate(PwmChannel_1, (uint16)pwmDutyB);
Pwm_SetDutyCycle_NoUpdate(PwmChannel_2, (uint16)pwmDutyC);
Pwm_SyncUpdate(INST_FTM3);|*~52.13 us
Baremetal*|FTM3->CONTROLS[0].CnV
FTM3->CONTROLS[2].CnV
FTM3->CONTROLS[4].CnV
FTM3->SYNC \|= FTM_SYNC_SWSYNC_MASK;|*~1.4375 us</td></tr>
</table>

ID	Subtype	Headline and Description
		|*ADC Update*|*Execution time RTD Low Level*|Phase A current Adc_Ip_GetConvData(INST_ADC0, 0) Phase B current Adc_Ip_GetConvData(INST_ADC1, 1) DC Bus Voltage Adc_Ip_GetConvData(INST_ADC1, 0)|*~1.81 us ** RTD MCAL*|Adc_ReadRawData(0, &ADC0CH0_array, 1, &PhaseA_Current); Adc_ReadRawData(1, &ADC1CH1_array, 1, &PhaseB_Current); Adc_ReadRawData(1, &ADC1CH0_array, 1, &DCBus_Voltage);|*~14.08 us Baremetal*|Phase A current ADC0->R[0] Phase B current ADC1->R[1] DC Bus Voltage ADC1->R[0]|*~1.225 us As you can see the execution time is quite higher when measuring the MCAL APIs. The proposal is to keep existing functions for users which prefer universal functions with higher level of abstraction and add new functions for users who need fast PWM update, and ADC read. Adding new functions for drivers was discussed in a meeting with Ioana-Andreea Pocea and Tomas Fedor on July 10th. Reason for this change: These new functions will help us to provide motor control software using AUTOSAR for customers interested in S32K1 and S32M24x. For S32M2 is particularly special as this device is about to be release next year and is targeted specifically for motor control applications. Benefit: Wider free space for user calculations/algorithms will put S32K1 and S32M24 into better position. Also these fast APIs will help us run motor control application using AUTOSAR. Use-case: Motor control area. HW/Application Engineer contact (as applies): Felipe Garcia felipe.garcia_1@nxp.com Note: relevant documents to be attached to the ticket.
ARTD-105226	Bug	[Integration] Wrong Build Version Detailed description (how to reproduce it): Step1: In S32DS choose File > New > S32DS Application project Step2: Fulfil project name and Choose Family from the Test Data column-> Next Step3: Select the required toolchain plugin from the toolchain tab x number of tools chains Step4: Type a project name(e.g NewProject). Step5: Select SDK version from Test Data column Step6: Select only core from Test Data column Step7: Click Finish Step8: Build project (Debug_FLASH) Step9: Click on Update code Step10: Build project (Debug_FLASH Step11: Debug project, the program jumps to "main" function Step 12: On C/C perspective, right click to current project, select "Clean Project" Step 13: Repeat from Step 8 to Step 11 with Debug_RAM option Step 14: Check "SW version" and "Build version" in project header file Build version is found wrong at step 14 for Debug_RAM and Debug_FLASH option for all derivatives Debug_FLASH !image-2023-12-14-09-50-05-276.png! Debug_RAM !image-2023-12-14-09-50-22-142.png! Preconditions: S32K1XX_RTD_4_7_RTM_P02_2_0_0_DS_updatesite_2312 Test Case ID (internal TC that caught the defect) optional: INT_DS_00007 Observed behavior: Wrong Build Version Expected behavior: Correct build version Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-105357	New Feature	[memacc] Remove Duplicated MemSectorBatch check NewWorkDescription: Memacc config forbids 2 sub-areas to use the same sector batch, issuing this error: "Duplicated MemSectorBatch between sub address areas"

ID	Subtype	Headline and Description
		I think this should be removed, it should be allowed to refer the same sector batch twice, for example if an application wants to use sectors 1, 2, 10, 11 of the same sector batch. Then it will need two sub-areas, 2 sectors each, pointing to the same batch but different sector offset Requirement source: Proposed solution optional:
ARTD-105454	Bug	[Spi][ASR 4.7][ASR 4.4] Meaning of parameter Length in Spi_SetupEB() Detailed description (how to reproduce it): In our implementation, Function Spi_SetupEB() has parameter Length looks like mean that the number of byte of SrcDataBuffer and DestDataBuffer. However, our customer said that it should be the number of element that defined under Framesize (SpiDataWidth) as per Autosar Spec: !image-2023-12-15-19-26-02-545.png! For example: Config Channel's SpiDataWitdh = 16bit Spi_DataBufferType TxChBuf[2] = \{0xaa,0xbb}; Spi_DataBufferType RxChBuf[2]; Spi_SetupEB(Channel, TxChBuf, RxChBuf, 1} {color}>* {color:#172b4d}In their opinion, the Length in this case should be 1 (means 16bits as per SpiDataWidth).* But in our current implementation, this value in this case must be 2*{color}Seems that the Autosar spec is not clear in this case, so it make some confusions for us. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: The customer needs to set the length in bytes for the SPI transfer in {*}SPI_SetupEB{*}() function. Parameter SpiEbMaxLength is also affected Expected behavior: The customer needs to set the length in data elements as per SPI 00180 requirement for the SPI transfer in SPI_SetupEB() function SWS_Spi_00180 SWS_Spi_00376 Also check following document for clarifications within autosar forum on this topic: [^AUTOSAR+JIRA +2023-12-20T10_06_30+0100.doc]_ Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Treat the Length parameter as number of data elements which can be in uint8, uint16 and uint32 as defined by {*}SpiDataWidth{*}. For SpiDataWidth uint 16 the length must be multiplied by 2 For SpiDataWidth uint32 the length must be multiplied by 4 So the same IP functions can be used.
ARTD-106295	Bug	RTD for S32M2XX (AE10 Driver) Issue 1: RTD Driver shall fix bug related to Frame Counter Fault processing SPI comm has GHS in which Frame Counter occupies bits from 14:8, which gives it range 0 127. Variable u32FrameCounter is never checked and never limited to max value of 127. On AE side, if counter goes above, it starts from 0 again, this needs to be implemented on RTD side Issue 2: This is for S32M27X platform, NMI flag is never cleared in WKPU (it should be done by RTD side) Issue 3: This is for S32M27X platform, NMI can be only edge sensitive. This causes problem as, we only read fault status once. In case another fault happens during handling of initial fault, this new fault is not handled. There has to be logic done at the end of fault processing, to read back Fault/Event status and if anything is pending to call again AEC_IRQEventFaultHandler(); Issue 4: Current implementation uses Lpspi_Ip_Cancel function in MNI. This fixes problems with SPI, but creates new one. Previous interrupted communication is not aware of this problem, and LPSPI timeout occurs. But due to this, interrupt for handling of AE faults is postponed for this timeout interval (default is 50ms). Please fix it, so if Lpspi_Ip_Cancel is called, ongoing comm is notified and doesnt get stuck in that timeout loop. Issue 5: Reading of Faults and Events when injected through IRQ_SET register is not working properly. Faults/Event is recognized, but GHS is checked only for Event, and if not event is pending, fault is not handled. (function AEC_IRQEventFaultHandler)

ID	Subtype	Headline and Description
ARTD-106823	Bug	[SAF8X R41 2.0.0] MemAcc: Can't generate when enable multicore type 1 Detailed description (how to reproduce it): Generate fail when enable multicore type 1 Preconditions: enable multicore type 1 Test Case ID (internal TC that caught the defect) optional: MemAcc_TS_090 Observed behavior: !image-2023-12-26-14-29-58-876.png|thumbnail! [...] Expected behavior: Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-106969	Bug	[Det] Det error get overwritten Detailed description (how to reproduce it) Det_ErrorId[DET_NO_ECU_CORES] may get overwritten by multiple error report. Preconditions Inject error by call Fee_Read with wrong parameter(BlockNumber, BlockOffset,...). Test Case ID (internal TC that caught the defect) optional: Fee_TC_FCT_1004 Observed behavior: Det_ErrorId[DET_NO_ECU_CORES] may get overwritten by multiple error report and return only last error Expected behavior Det_TestLastReportError return the injected error. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Add support for internal ram log for det module. Structure of ram log should fulfill requirement of fan-out mechanism. (static linked list should be prefered) Det_LastReportError and other function for det_stub which mainly use for testing should be keep the same functionalities
ARTD-106955	Bug	[S32K1-S32M24x][pwm] Validation of partition reference should be OS version independent Detailed description (how to reproduce it): Some RTD drivers (Adc, Crc, Dio, Icu, Platform, Port, Pwm, Serdes) have the following check when [drv]MulticoreEnabled = 'true': <a:tst expr="count(text:grep(node:refs('ASPathDataOfSchema:/TS_T40D11M40I2R0/Os')/OsApplication/*/ OsAppEcucPartitionRef, .)) = 0" true="The ECUC partition must be defined in OS."/> RTD shall not validate references against a certain OS version. There can be any OS version, either in a Tresos compatible plugin format or not. Creating a dependency to a certain version will cause the following validation error: !https://jira.sw.nxp.com/secure/attachment/1295507/image-2023-11-27-16-04-31-321.png! Preconditions: [drv]MulticoreEnabled is set to 'true' and a correct configuration of the partition is done in OS. Test Case ID (internal TC that caught the defect) optional: none Observed behavior: The following drivers Adc, Crc, Dio, Icu, Platform, Port, Pwm, Serdes show validation error is Tresos even when a correct reference to a partition is configured. Expected behavior: No error should be caused if different OSes are used. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Option 1) Change node:refs('ASPathDataOfSchema:/TS_T40D11M40I2R0/Os')/OsApplication/*/OsAppEcucPartitionRef To something like:

ID	Subtype	Headline and Description
		as:modconf('Os')[1]/OsApplication/*/OsAppEcucPartitionRef as found in CAN. Option 2) Alternatively this check can be removed altogether since drivers should not be aware of Os partition configuration. The reason is simple: partition configuration is part of high level components like: EcuM, BSWM and RTE, all above driver level.
ARTD-107325	Bug	[ETH] Wrong range in MainFunction loop Detailed description (how to reproduce it): The range of the loop that goes through all controllers in the MainFunction is wrong Preconditions: More than 1 controller is configured, and there are different numbers of controllers configured in different variants Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: The loop may loop through unconfigured controllers Expected behavior: The loop should only go through configured controllers Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Change upper limit of the loop in MainFunction to use the number of configured controllers for that specific configuration !image-2024-01-03-09-35-59-902.png!
ARTD-107359	Bug	Error when verify the default project of Dio_Example_S32K148 in SW32K1_S32M24x_RTD_4.4_R21-11_2.0.0 Customer met the issue when import and verify the default project of {color:#ff0000}Dio_Example_S32K148{color} in {color:#ff0000}SW32K1_S32M24x_RTD_4.4_R21-11_2.0.0{color} through {color:#ff0000}Tresos_rev. This could be duplicated on NXP's environment. I also test my site, the error is the same as the customer Please help to resolve the bug. !Error_Information.png! {_}*Note: in the “{_}{_}Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.){_} Proposed solution optional: [....][link title|http://example.com]
ARTD-107788	Bug	[crc] Driver makefile needs to be updated with the correct version of AUTOSAR SWS specification Detailed description (how to reproduce it): !image-2023-12-29-14-56-48-393.png|width=645,height=253! In <driver>.mak file, the information of AUTOSAR module SWS specification version are incorrect. E.g.: some drivers contains 4.3 references Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Wrong information from: # When importing project to EBT !image-2023-12-29-14-59-58-330.png|width=470,height=393! # plugin.xml from plugins !image-2023-12-29-15-01-33-443.png|width=310,height=306! Expected behavior: The information needs to be updated. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update the information from <driver>.mak based on the supported version of ASR for drivers that are delivering ASR R2x-11: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_7_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 7 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000

ID	Subtype	Headline and Description

		endif
		only for drivers that are delivering ASR 4.4:
		
		ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_4_REV_0000)
		SWS_SPEC_VERSION_MAJOR = 4
		SWS_SPEC_VERSION_MINOR = 4
		SWS_SPEC_VERSION_PATCH = 0
		SWS_SPEC_VERSION_SUFFIX = Rev_0000
		endif

ARTD-107809	Bug	[dio] Driver makefile needs to be updated with the correct version of AUTOSAR SWS specification Detailed description (how to reproduce it): !image-2023-12-29-14-56-48-393.png|width=645,height=253! In <driver>.mak file, the information of AUTOSAR module SWS specification version are incorrect. E.g.: some drivers contains 4.3 references Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Wrong information from: # When importing project to EBT !image-2023-12-29-14-59-58-330.png|width=470,height=393! # plugin.xml from plugins !image-2023-12-29-15-01-33-443.png|width=310,height=306! Expected behavior: The information needs to be updated. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update the information from <driver>.mak based on the supported version of ASR for drivers that are delivering ASR R2x-11: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_7_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 7 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif only for drivers that are delivering ASR 4.4: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_4_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 4 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif
ARTD-107812	Bug	[eth] Driver makefile needs to be updated with the correct version of AUTOSAR SWS specification Detailed description (how to reproduce it): !image-2023-12-29-14-56-48-393.png|width=645,height=253! In <driver>.mak file, the information of AUTOSAR module SWS specification version are incorrect. E.g.: some drivers contains 4.3 references Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Wrong information from: # When importing project to EBT !image-2023-12-29-14-59-58-330.png|width=470,height=393! # plugin.xml from plugins !image-2023-12-29-15-01-33-443.png|width=310,height=306! Expected behavior: The information needs to be updated. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update the information from <driver>.mak based on the supported version of ASR

ID	Subtype	Headline and Description

		for drivers that are delivering ASR R2x-11:
		
		ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_7_REV_0000)
		SWS_SPEC_VERSION_MAJOR = 4
		SWS_SPEC_VERSION_MINOR = 7
		SWS_SPEC_VERSION_PATCH = 0
		SWS_SPEC_VERSION_SUFFIX = Rev_0000
		endif
		only for drivers that are delivering ASR 4.4:
		
		ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_4_REV_0000)
		SWS_SPEC_VERSION_MAJOR = 4
		SWS_SPEC_VERSION_MINOR = 4
		SWS_SPEC_VERSION_PATCH = 0
		SWS_SPEC_VERSION_SUFFIX = Rev_0000
		endif

ARTD-107815	Bug	[fee] Driver makefile needs to be updated with the correct version of AUTOSAR SWS specification Detailed description (how to reproduce it): !image-2023-12-29-14-56-48-393.png|width=645,height=253! In <driver>.mak file, the information of AUTOSAR module SWS specification version are incorrect. E.g.: some drivers contains 4.3 references Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Wrong information from: # When importing project to EBT !image-2023-12-29-14-59-58-330.png|width=470,height=393! # plugin.xml from plugins !image-2023-12-29-15-01-33-443.png|width=310,height=306! Expected behavior: The information needs to be updated. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update the information from <driver>.mak based on the supported version of ASR for drivers that are delivering ASR R2x-11: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_7_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 7 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif only for drivers that are delivering ASR 4.4: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_4_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 4 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif
ARTD-107842	Bug	[mcl] Driver makefile needs to be updated with the correct version of AUTOSAR SWS specification Detailed description (how to reproduce it): !image-2023-12-29-14-56-48-393.png|width=645,height=253! In <driver>.mak file, the information of AUTOSAR module SWS specification version are incorrect. E.g.: some drivers contains 4.3 references Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Wrong information from: # When importing project to EBT !image-2023-12-29-14-59-58-330.png|width=484,height=405! # plugin.xml from plugins !image-2023-12-29-15-01-33-443.png|width=310,height=306!

ID	Subtype	Headline and Description

		Expected behavior:
		The information needs to be updated.
		
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)
		
		Proposed solution optional:
		Update the information from <driver>.mak based on the supported version of ASR
		
		for drivers that are delivering ASR R2x-11:
		
		ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_7_REV_0000)
		SWS_SPEC_VERSION_MAJOR = 4
		SWS_SPEC_VERSION_MINOR = 7
		SWS_SPEC_VERSION_PATCH = 0
		SWS_SPEC_VERSION_SUFFIX = Rev_0000
		endif
		only for drivers that are delivering ASR 4.4:
		
		ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_4_REV_0000)
		SWS_SPEC_VERSION_MAJOR = 4
		SWS_SPEC_VERSION_MINOR = 4
		SWS_SPEC_VERSION_PATCH = 0
		SWS_SPEC_VERSION_SUFFIX = Rev_0000
		endif

ARTD-107857	Bug	[port] Driver makefile needs to be updated with the correct version of AUTOSAR SWS specification Detailed description (how to reproduce it): !image-2023-12-29-14-56-48-393.png|width=645,height=253! In <driver>.mak file, the information of AUTOSAR module SWS specification version are incorrect. E.g.: some drivers contains 4.3 references Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Wrong information from: # When importing project to EBT !image-2023-12-29-14-59-58-330.png|width=470,height=393! # plugin.xml from plugins !image-2023-12-29-15-01-33-443.png|width=310,height=306! Expected behavior: The information needs to be updated. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update the information from <driver>.mak based on the supported version of ASR for drivers that are delivering ASR R2x-11: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_7_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 7 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif only for drivers that are delivering ASR 4.4: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_4_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 4 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif
ARTD-107875	Bug	[spi] Driver makefile needs to be updated with the correct version of AUTOSAR SWS specification Detailed description (how to reproduce it): !image-2023-12-29-14-56-48-393.png|width=645,height=253! In <driver>.mak file, the information of AUTOSAR module SWS specification version are incorrect. E.g.: some drivers contains 4.3 references Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA

ID	Subtype	Headline and Description
		Observed behavior: Wrong information from: # When importing project to EBT !image-2023-12-29-14-59-58-330.png|width=470,height=393! # plugin.xml from plugins !image-2023-12-29-15-01-33-443.png|width=310,height=306! Expected behavior: The information needs to be updated. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update the information from <driver>.mak based on the supported version of ASR for drivers that are delivering ASR R2x-11: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_7_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 7 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif only for drivers that are delivering ASR 4.4: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_4_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 4 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif
ARTD-107878	Bug	[uart] Driver makefile needs to be updated with the correct version of AUTOSAR SWS specification Detailed description (how to reproduce it): !image-2023-12-29-14-56-48-393.png|width=645,height=253! In <driver>.mak file, the information of AUTOSAR module SWS specification version are incorrect. E.g.: some drivers contains 4.3 references Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Wrong information from: # When importing project to EBT !image-2023-12-29-14-59-58-330.png|width=470,height=393! # plugin.xml from plugins !image-2023-12-29-15-01-33-443.png|width=310,height=306! Expected behavior: The information needs to be updated. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update the information from <driver>.mak based on the supported version of ASR for drivers that are delivering ASR R2x-11: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_7_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 7 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif only for drivers that are delivering ASR 4.4: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_4_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 4 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif
ARTD-107887	Bug	[qd] Driver makefile needs to be updated with the correct version of AUTOSAR SWS specification Detailed description (how to reproduce it): !image-2023-12-29-14-56-48-393.png|width=645,height=253!

ID	Subtype	Headline and Description
		In <driver>.mak file, the information of AUTOSAR module SWS specification version are incorrect. E.g.: some drivers contains 4.3 references Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Wrong information from: # When importing project to EBT !image-2023-12-29-14-59-58-330.png|width=470,height=393! # plugin.xml from plugins !image-2023-12-29-15-01-33-443.png|width=310,height=306! Expected behavior: The information needs to be updated. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update the information from <driver>.mak based on the supported version of ASR for drivers that are delivering ASR R2x-11: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_7_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 7 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif only for drivers that are delivering ASR 4.4: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_4_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 4 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif
ARTD-107896	Bug	[ae] Driver makefile needs to be updated with the correct version of AUTOSAR SWS specification Detailed description (how to reproduce it): !image-2023-12-29-14-56-48-393.png|width=645,height=253! In <driver>.mak file, the information of AUTOSAR module SWS specification version are incorrect. E.g.: some drivers contains 4.3 references Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Wrong information from: # When importing project to EBT !image-2023-12-29-14-59-58-330.png|width=470,height=393! # plugin.xml from plugins !image-2023-12-29-15-01-33-443.png|width=310,height=306! Expected behavior: The information needs to be updated. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update the information from <driver>.mak based on the supported version of ASR for drivers that are delivering ASR R2x-11: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_7_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 7 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif only for drivers that are delivering ASR 4.4: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_4_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 4 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000

ID	Subtype	Headline and Description
		endif
ARTD-107899	Bug	[CanTrcv] Driver makefile needs to be updated with the correct version of AUTOSAR SWS specification Detailed description (how to reproduce it): !image-2023-12-29-14-56-48-393.png|width=645,height=253! In <driver>.mak file, the information of AUTOSAR module SWS specification version are incorrect. E.g.: some drivers contains 4.3 references Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Wrong information from: # When importing project to EBT !image-2023-12-29-14-59-58-330.png|width=470,height=393! # plugin.xml from plugins !image-2023-12-29-15-01-33-443.png|width=310,height=306! Expected behavior: The information needs to be updated. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update the information from <driver>.mak based on the supported version of ASR for drivers that are delivering ASR R2x-11: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_7_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 7 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif only for drivers that are delivering ASR 4.4: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_4_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 4 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif
ARTD-107923	Bug	[LinTrcv] Driver makefile needs to be updated with the correct version of AUTOSAR SWS specification Detailed description (how to reproduce it): !image-2023-12-29-14-56-48-393.png|width=645,height=253! In <driver>.mak file, the information of AUTOSAR module SWS specification version are incorrect. E.g.: some drivers contains 4.3 references Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Wrong information from: # When importing project to EBT !image-2023-12-29-14-59-58-330.png|width=470,height=393! # plugin.xml from plugins !image-2023-12-29-15-01-33-443.png|width=310,height=306! Expected behavior: The information needs to be updated. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update the information from <driver>.mak based on the supported version of ASR for drivers that are delivering ASR R2x-11: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_7_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 7 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000

ID	Subtype	Headline and Description

		endif
		only for drivers that are delivering ASR 4.4:
		
		ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_4_REV_0000)
		SWS_SPEC_VERSION_MAJOR = 4
		SWS_SPEC_VERSION_MINOR = 4
		SWS_SPEC_VERSION_PATCH = 0
		SWS_SPEC_VERSION_SUFFIX = Rev_0000
		endif

ARTD-108527	Bug	[Mem_ExFls] Mem_JobResultType deviates from AUTOSAR Detailed description (how to reproduce it): the Mem-drivers of the mentioned MCAL. Mem_JobResultType is deviates from AUTOSAR. According to SWS_Mem_10019, there should be e.g. : MEM_43_EXFLS_INCONSISTENT MEM_43_EXFLS_ECC_UNCORRECTED MEM_43_EXFLS_ECC_CORRECTED. But it is defined in the MCAL: MEM_43_EXFLS*_JOB*_INCONSISTENT MEM_43_EXFLS*_JOB*_ECC_UNCORRECTED MEM_43_EXFLS*_JOB*_ECC_CORRECTED. Same is valid for INFLS and EEP Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: the Mem-drivers of the mentioned MCAL. Mem_JobResultType is deviates from AUTOSAR Expected behavior: the Mem-drivers of the mentioned MCAL. Mem_JobResultType follows from AUTOSAR Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-108778	Bug	[pwm][S32K1XX] Driver makefile needs to be updated with the correct version of AUTOSAR SWS specification Detailed description (how to reproduce it): !image-2023-12-29-14-56-48-393.png|width=645,height=253! In <driver>.mak file, the information of AUTOSAR module SWS specification version are incorrect. E.g.: some drivers contains 4.3 references Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Wrong information from: # When importing project to EBT !image-2023-12-29-14-59-58-330.png|width=470,height=393! # plugin.xml from plugins !image-2023-12-29-15-01-33-443.png|width=310,height=306! Expected behavior: The information needs to be updated. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update the information from <driver>.mak based on the supported version of ASR for drivers that are delivering ASR R2x-11: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_7_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 7 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif only for drivers that are delivering ASR 4.4: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_4_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 4

ID	Subtype	Headline and Description

		SWS_SPEC_VERSION_PATCH = 0
		SWS_SPEC_VERSION_SUFFIX = Rev_0000
		endif

ARTD-109056	New Feature	[SPI][k1_M24x] For baudrate and delay time calculation, add a node to switch between the current automation mode and a new manual mode NewWorkDescription: For Spi baudrate and delay time calculation, add a node to switch between the current automation mode and a new manual mode. Automation mode: inputs are desired baudrate and delay times; outputs are register generated values as PRESCALE, SCKDIV, SCKPCS, PCSSCK, DBT. SCK baud rate = (fP/PBR) x [(1+DBR)/BR] currently calculated with smallest error compared to requested input. Manual mode: inputs are register values as PRESCALE, SCKDIV, SCKPCS, PCSSCK, DBT; outputs are the calculated baudrate and delay times for the inputs. Show automatic calculation for SCK baud rate = (fP/PBR) x [(1+DBR)/BR] in the user interfaces as the customer understands the output of the configuration Explain in User manual the switch between Automation mode and Manual mode for SCK calculation Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: N/A
ARTD-109084	Bug	[Qdec] Inconsistent driver naming used across all CDD driver source/include files Detailed description (how to reproduce it): The naming convention for header files we implement for CDD drivers follows the following mention in AUTOSAR specification: !image-2023-12-04-14-07-57-719.png! Preconditions: Qdec.h need to change name to CDD_Qdec.h and the macros of this file should change to: #ifndef CDD_QDEC_H #define CDD_QDEC_H Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: as Preconditions Expected behavior: as Preconditions Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: as Preconditions
ARTD-109108	Bug	[LIN]: Lin driver not enable overrun interrupt in LPUART instance Detailed description (how to reproduce it): Lin driver doesn't enable overrun interrupt (CTRL[ORIE]) in LPUART instances Preconditions: Lin driver doesn't report overrun error, some bits misbehave when an overrun error occurs Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Lin driver not enable overrun interrupt (CTRL[ORIE]) in LPUART instances Expected behavior: Enable overrunt interrupt in LPUART instances Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-110318	Bug	[I2c] Inconsistent driver naming used across all CDD driver source/include files Detailed description (how to reproduce it): The naming convention for header files we implement for CDD drivers follows the following mention in AUTOSAR specification: Link spec: [Complex Driver design and integration guideline (autosar.org)|https://www.autosar.org/fileadmin/standards/R22-11/CP/AUTOSAR_EXP_CDDDesignAndIntegrationGuideline.pdf] !image-2023-12-04-14-07-57-719.png!

ID	Subtype	Headline and Description
		Preconditions: The header file needs to be updated with the correct name according to the spec mentioned. !image-2024-01-26-16-12-54-134.png! Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: as Preconditions Expected behavior: as Preconditions Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: as Preconditions
ARTD-110425	Bug	[SAF8X R41 2.0.0] Fee: Compare code gen tests failed Detailed description (how to reproduce it): There is differences between code generated by EB and S32DS Driver tag: PVT_FEE_SAF85_SAF86_S32R41_RTM2.0.0_V07 Test tag: PVT_TEST_FEE_SAF85_SAF86_S32R41_2.0.0_023 Test suite id: Fee_TS_101 Derivative: SAF8544 Lines swap: !image-2024-01-31-17-52-41-268.png|width=1161,height=57! Preconditions: There is differences between code generated by EB and S32DS Test Case ID (internal TC that caught the defect) optional: Observed behavior: Expected behavior: Code generated from S32DS and EB tresos must be the same Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:
ARTD-110643	Bug	[I2s]Polling transfer type not working on Flexio Detailed description (how to reproduce it): I2S Flexio2Flexio communication with polling Preconditions: Flexio master send data to flexio sllave Test Case ID (internal TC that caught the defect) optional: TS0102 Observed behavior: DevAssert on channel validation Expected behavior: No DevAssert Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Replace (uint8)Master->FlexioCommon.ResourceIndex/2U with [(uint8)Master->FlexioCommon.ResourceCount/2U
ARTD-111176	Bug	[SPI]Updated solution - Driver can be buildable when SpiDMAFastTransfer is not enabled but the SpiMaxFastTransfer is not empty Detailed description (how to reproduce it):

ID	Subtype	Headline and Description
		[S32K1 S32N S32XX S32ZSE SAF SJA] Driver should be buildable when SpiDMAFastTransfer is not enable but the SpiMaxFastTransfer have the value. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: !image-2024-02-05-11-46-45-615.png! Expected behavior: NA Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-111317	Bug	Power_Ip LVDF Detailed description (how to reproduce it): [Power_Ip_PMC_VoltageErrorIsr() in Power_Ip_PMC.c clears LVDF and LVWI incorrectly For example: POWER_IP_PMC->LVDSC1 = (VoltageDetectStatus & POWER_IP_PMC_LVDSC1_LVDF_MASK); The Flag should be cleared with a write to VDSC1_LVDACK.] Preconditions: [none] Test Case ID (internal TC that caught the defect) optional: [594659] Observed behavior: [The flag does not get cleared even though the VDD voltage is above the threshold again.] Expected behavior: [ ] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]

ARTD-111320	Bug	Power_Ip_PMC_GetInterruptStatus, same status code for LVD and LVWI
		
		Detailed description (how to reproduce it):
		[
		uint8 Power_Ip_PMC_GetInterruptStatus(void)
		{
		uint8 VoltageStatus = 0U;
		#if(STD_ON == POWER_IP_PMC_LVDSC1_SUPPORT)
		/ Read Low Voltage Detect Flag /
		VoltageStatus = ((POWER_IP_PMC->LVDSC1) & POWER_IP_PMC_LVDSC1_LVDF_MASK);
		#endif
		/ Read Low Voltage Warning Flag /
		VoltageStatus = ((POWER_IP_PMC->LVDSC2) & POWER_IP_PMC_LVDSC2_LVWF_MASK);
		
		return VoltageStatus;
		}
		
		The above function return the same status code.
		
		]
		
		Preconditions:
		[none]
		
		Test Case ID (internal TC that caught the defect) optional:
		[none]
		
		Observed behavior:
		[We can't distinguish between LVD and LVWI]
		
		Expected behavior:
		[callback should return either LVD or LVWI]
		
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)

ID	Subtype	Headline and Description
		Proposed solution optional: [there should be an ENUM with different low voltage errors]
ARTD-111453	Bug	[Base] Incorrect SuspendInterrupts abstraction over FreeRTOS Detailed description (how to reproduce it): Create a S32DS project that also integrates FreeRTOS. Add a periodic interrupt in the GPT. Preconditions: None Test Case ID (internal TC that caught the defect) optional: None Observed behavior: Upon suspending all interrupts within a GPT ISR, with OsIf_SuspendAllInterrupts, from OsIf_Internal.h, when using FreeRTOS, will call taskEXIT_CRITICAL(), which will result in an assertion failure, since the function is not meant to be called from an ISR context. Expected behavior: Successfully suspend interrupts Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: The OsIf should rather call taskEXIT_CRITICAL_FROM_ISR(), when in an interrupt context Some more information from FreeRTOS forum: https://forums.freertos.org/t/why-vportentercritical-is-not-safe-called-from-an-interrupt-context/12912/4
ARTD-111789	Bug	[AE]: DET error should be raised if initialized failed Detailed description (how to reproduce it): According to the current implementation of Ae_init(), there is no DET error if the initialization is failed: !image-2024-02-16-15-24-03-615.png! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: No DET error if initialization failed Expected behavior: a DET error raised if initialization failed Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-112210	Bug	[ADC] AdcPdbCounterPeriod value not used in HLD Detailed description (how to reproduce it): Value set to the AdcPdbCounterPeriod node is not used by ADC HLD at runtime. Preconditions: Use ADC HLD component, configure AdcPdbCounterPeriod node with a value, enable and set channel delays in the group configuration and then start a conversion at runtime with said group. Test Case ID (internal TC that caught the defect) optional: Adc_TS_062 Observed behavior: PDB counter modulus will be set at runtime to a calculated value based on channel delays rather than the value set at configuration time. Expected behavior: Use the configured value or not have the option to set this value. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Investigate if there are use cases that require this node or remove it if it is considered redundant.
ARTD-112248	Bug	[Mcl] No_init variable was put in the VAR_INIT region Detailed description (how to reproduce it): Mcl_au8EmiosLogicToHwInstance[eMIOS_INSTANCE_COUNT] and Mcl_au8FlexioLogicToHwInstance[FLEXIO_INSTANCE_COUNT] are no_init variable but they are put in the VAR_INIT region MCL_START_SEC_VAR_INIT_UNSPECIFIED_NO_CACHEABLE.

ID	Subtype	Headline and Description
		They should be put in No_init region MCL_START_SEC_VAR_CLEARED_UNSPECIFIED_NO_CACHEABLE !image-2024-02-20-10-46-09-793.png! Preconditions: No_init variable was put in the VAR_INIT region MCL_START_SEC_VAR_INIT_UNSPECIFIED_NO_CACHEABLE. Observed behavior: Variables are in the Incorrect region. Expected behavior: Variables are in the correct region. Follow Coding_NXPDrivers_CompilerAbstraction_MemMap: [https://nxp1.sharepoint.com/:p:/s/Zebra/ETwcdn9fq2lPlKiRHHfyz2IBQ65w7dVyMNplIFilWboF7A?e=JpRwPbBasic] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Put them in No_init region MCL_START_SEC_VAR_CLEARED_UNSPECIFIED_NO_CACHEABLE
ARTD-112522	Bug	[lin] [S32K3_S32M27x] Problem configuring the Lin Frame Timeout Detailed description (how to reproduce it): The response timeout value is wrong, no need to divide by 10 !image-2024-02-22-17-57-08-317.png! Preconditions: The response timeout value is wrong, no need to divide by 10 Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: The expression to calculate response time contain "{*}div 10{*}" Expected behavior: The expression to calculate response time shouldn't contain "{*}div 10{*}" Note:* in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-112669	Bug	[ALL_GENERAL] modules.h file <module> always STD_OFF Detailed description (how to reproduce it): GPT module is enabled in EB tresos configuration tool(ver 29.0), but macro USE_GPT_MODULE in module.h file is always STD_OFF.* !image-2024-02-23-17-48-13-906.png! The issues is happen in another modules, like SPI, DIO, PORT.* !image-2024-02-23-18-02-33-413.png! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: The defined macros for the modules are consistently set to STD_OFF. Expected behavior: The module's define macro is set to STD_ON during usage. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-113366	Bug	[SPI] SpiCsSelection disabled leads to PCS0 being used as fixed value Detailed description (how to reproduce it): SpiCsSelection needs to be enabled to switch Peripheral chip select. If it is not enabled, PCS0 will be used as fix value For Slave devices this can be confusing as no warning is issued. Preconditions: SpiCsSelection disabled

ID	Subtype	Headline and Description
		SPICsIdentifier is not PCS0 !image-2024-02-28-09-42-09-853.png! Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: CS is generated with PCS0 value Expected behavior: CS shall be generated with value chosen by user OR Issue warning that SpiCsSelection needs to be enabled Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A Also, GPIO is used, see if CsIdentifier can be greyed out !image-2024-03-01-13-36-48-049.png!
ARTD-113584	Bug	PMC AE causes an error during compilation if selected(ticked) in POWER LLD gui Detailed description (how to reproduce it): While selecting PMC AE in POWER LLD GUI, it generates the configuration structure(constants) correctly which is right but during compilation process an error is generated !PMC_AE_POWERLLD.png! Preconditions: no special preconditions just PMC_AE api needs to be selected in POWER LLD GUI Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: As described above Expected behavior: It is needed to be able to select PMC_AE api in POWER LLD GUI and to get error-free compilation. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: A working workaround is NOT to select(tick) PMC_AE api in POWER LLD GUI. It generates the configuration structure for PMC_AE anyway and the compiler doesn't give an error. It seems it would be enough not to include Ae.h if LLD drivers(non Autosar) are used in the application
ARTD-113745	New Feature	[MCU]update sequence with function disable Spll,FIRC,SOSC NewWorkDescription: when disable the Pll,SOSC,... Driver missing check status of register .it maybe raise unknown behavior. !image-2023-12-21-10-36-23-120.png|thumbnail! use case: https://community.nxp.com/t5/S32K/s32k-disabling-the-PLL/td-p/1189285 Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: add check the status after disable the feature
ARTD-114173	Bug	[SPI]Fix Misra findings Rule 8.4 on IP Driver code level Detailed description (how to reproduce it): MISRA justifications not used according to agreed deviation on k3 (mainly) and on others platforms. !image-2024-03-06-16-21-34-609.png! Preconditions: Runtime configuration is applicable for post build file where the MISRA deviation can be justified. For other cases they shall be justified. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: MISRA deviation for rule 8.4 applied where should be fixed

ID	Subtype	Headline and Description
		Expected behavior: MISRA deviations fixed for rule 8.4 in IP files Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-114772	Bug	[Mem_InFls] Mem_JobResultType deviates from AUTOSAR Detailed description (how to reproduce it): the Mem-drivers of the mentioned MCAL. Mem_JobResultType is deviates from AUTOSAR. According to SWS_Mem_10019, there should be e.g. : MEM_43_EXFLS_INCONSISTENT MEM_43_EXFLS_ECC_UNCORRECTED MEM_43_EXFLS_ECC_CORRECTED. But it is defined in the MCAL: MEM_43_EXFLS*_JOB*_INCONSISTENT MEM_43_EXFLS*_JOB*_ECC_UNCORRECTED MEM_43_EXFLS*_JOB*_ECC_CORRECTED. Same is valid for INFLS and EEP Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: the Mem-drivers of the mentioned MCAL. Mem_JobResultType is deviates from AUTOSAR Expected behavior: the Mem-drivers of the mentioned MCAL. Mem_JobResultType follows from AUTOSAR Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-115965	New Feature	[ADC] Support DMA error notification NewWorkDescription: Currently, DMA error notification has not been supported Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Support DMA error notification for ADC
ARTD-116694	Bug	[MCU] Driver is not implementing SWS_Mcu_00133 completely Detailed description (how to reproduce it): If McuDevErrorDetect is disabled and Mcu_GetResetReason is called before Mcu_Init Mcu_GetResetReason will not equal MCU_RESET_UNDEFINED !image-2024-03-21-17-52-17-995.png! It does not obey SWS_Mcu_00133 !image-2024-03-21-17-53-25-590.png! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Mcu_TC_FCT_0089 Observed behavior: Driver is not implementing SWS_Mcu_00133 completely Expected behavior: Implement SWS_Mcu_00133 completely and all other requirements related to UNDEFINED results Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Move line #if (MCU_VALIDATE_GLOBAL_CALL == STD_ON) from Mcu_GetResetReason into Mcu_HLDChecksEntry and Mcu_HLDChecksExit !image-2024-03-21-18-03-59-155.png! !image-2024-03-21-18-04-12-829.png!
ARTD-117777	Bug	[gpt] SRtc does not compile.

ID	Subtype	Headline and Description
		Detailed description (how to reproduce it): When using or integrating SRtc to project there is a problem with generation of SRtc_Ip_Types file. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Program does not compile. Expected behavior: Build the program. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-117866	New Feature	[RM] Add option for global RM Init function or separate per each IP at HLD level Original request: Can we consider splitting the RM module to implement different modules separately ? At present, the RM module includes many peripherals, such as XRDC, SEMA42, XBIC, DMA MUX, MSCM, etc. At present, we have encountered some problems when integrating functional safety software, as XRDC is not configured in the boot header yet and it often needs to be initialized earlier. But other modules such as XBIC may need to be initialized later. However, currently these modules are placed within the same RM module, making it difficult to achieve the above requirement by calling the RM-Init function: initializing different modules at different stages. NewWorkDescription: In order to allow for greater flexibility and to cover use case when integrator needs to have a customer order of init, an option in configuration must allow a global init or an init per IP level. Requirement source: FAE, CPRT (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Create node "Global RM driver Init" with default value True to allow for same functionality in future releases When option is disabled, in CDD_Rm the init functions must be per IP: RM_XRDC_Init(), RM_SEMA4_Init(), RM_DMAMUX_Init() etc. For similar use case, see Fss_Rem_Pm_IrmInit and Fss_Rem_Pm_InitEscm in [Source of CDD_Fss_Rem_Pm.c fss_rem_pm NXP Bitbucket|https://bitbucket.sw.nxp.com/projects/ARTD/repos/fss_rem_pm/browse/generic/src/CDD_Fss_Rem_Pm.c]
ARTD-123147	New Feature	[NVM] Add support for S32K1_M24x Add Nvm support for S32K1_M24x platform.
ARTD-123396	Bug	[S32K1XX_S32M24x_2.0.0_P04][WDG] Wrong valid range for HIGHCOMPARE register value in EWM ip {_}*Detailed description (how to reproduce it):*{_}{*}{{*}} The max value of Compare High Register is 0xFE, but on EB it's 255 !image-2024-04-04-16-35-08-951.png|width=536,height=145! !image-2024-04-04-16-32-14-762.png|width=440,height=227! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: maximum of Compare High Register Value is 254 Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-123437	Bug	[Mem_InFls] Update Service ID value [hex]. Detailed description (how to reproduce it):

ID	Subtype	Headline and Description
		Update Service ID. Bellow is example from the Mem_InFls driver. !image-2024-03-26-08-45-18-122.png! Preconditions: Test Case ID (internal TC that caught the defect) optional: Observed behavior: Expected behavior: Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:
ARTD-123664	Bug	[Base] BaseNXP module should contain BaseNXP.h file Detailed description (how to reproduce it): N/A Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: BaseNXP module does not contain BaseNXP.h header file As per the Autosar document for General Specification of Basic Software Module, Bsw module is required to have an implementation header containing Module Implementation Prefix directly mentioning module name. For BaseNXP module this is vioalted by not providing BaseNXP.h and respective header is provided with the name OsIf.h !image-2024-04-08-14-06-09-092.png|width=542,height=455! Expected behavior: BaseNXP.h shall be available Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-123706	New Feature	[wdg] [S32K1_S32M24x] Implement requirement AEWDOG_IP_003_001 NewWorkDescription: Implement requirement AEWDOG_IP_003_001 Requirement source: AEWDOG_IP_003_001 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: NA
ARTD-123766	Bug	[LIN] Incorrect copyright Detailed description (how to reproduce it): Wrong copyright !image-2024-04-09-15-52-46-479.png! Preconditions: Build plugin on bamboo Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Wrong copyright Expected behavior: Update copyright Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)

ID	Subtype	Headline and Description
		Proposed solution optional: [....]
ARTD-124084	Bug	[I2S] Flexio DMA I2s can't trigger next transfer using callback function. Detailed description (how to reproduce it): The Flexio want to trigger next transmission using callback function. But after the end of the first transmission, we can receive anything Preconditions: Using Flexio DMA Using callback function Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Flexio I2s can't trigger next transfer using callback function. Expected behavior: Flexio I2s trigger next transfer using callback function and transfer exatly. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-124134	New Feature	[BASE][IMPLEMENTATION] R21-11 Multicore Concept changes - multipartition vs multicore CR description: Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrlE69] Implementation requested for BASE: Implement the api {_}OsIf_GetUserID(){_}, which shall return, based on the configuration used, the following IDs: 1. The *ECUC partition ID* mapped to the OS app. Implementation will be available only for AUTOSAR OS. 2. The *Core ID* (backwards compatibility maintained for other OSes and type I drivers ex: MCU) 3. Drop down selection in base and usage of a callback mechanism that assure returning a custom ID (defined by the user in his implementation) default of selection shall maintain backwards compatibility selecting OsIf_GetCoreID() as default. Note: # Single core constraints which don't allow multiple partitions on one core must be removed from Base. reference implementation(s) : ARTD-97002
ARTD-124297	New Feature	[adc] [IMPLEMENTATION] R21-11 Multicore Concept changes - multipartition vs multicore CR description: Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrlE69] Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of OsIf_GetCoreID to OsIfGetUserID. # Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing core_id check to partition_id check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multicore (multipartition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_} reference implementation(s) : ARTD-97002
ARTD-124359	New Feature	[dio] [IMPLEMENTATION] R21-11 Multicore Concept changes - multipartition vs multicore CR description: Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrlE69] Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of OsIf_GetCoreID to OsIfGetUserID. # Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing core_id check to partition_id check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multicore (multipartition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_} reference implementation(s) : ARTD-97002
ARTD-124395	New Feature	[i2c] [IMPLEMENTATION] R21-11 Multicore Concept changes - multipartition vs multicore CR description:

ID	Subtype	Headline and Description
		Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrlE69] Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of OsIf_GetCoreID to OsIfGetUserID. # Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing core_id check to partition_id check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multicore (multipartition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_} reference implementation(s) : ARTD-97002
ARTD-124398	New Feature	[i2s] [IMPLEMENTATION] R21-11 Multicore Concept changes - multipartition vs multicore CR description: Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrlE69] Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of OsIf_GetCoreID to OsIfGetUserID. # Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing core_id check to partition_id check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multicore (multipartition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_} reference implementation(s) : ARTD-97002
ARTD-124425	New Feature	[mem_infls] [IMPLEMENTATION] R21-11 Multicore Concept changes - multipartition vs multicore CR description: Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrlE69] Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of OsIf_GetCoreID to OsIfGetUserID. # Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing core_id check to partition_id check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multicore (multipartition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_} reference implementation(s) : ARTD-97002
ARTD-124458	New Feature	[port] [IMPLEMENTATION] R21-11 Multicore Concept changes - multipartition vs multicore CR description: Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrlE69] Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of OsIf_GetCoreID to OsIfGetUserID. # Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing core_id check to partition_id check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multicore (multipartition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_} reference implementation(s) : ARTD-97002
ARTD-124485	New Feature	[spi] [IMPLEMENTATION] R21-11 Multicore Concept changes - multipartition vs multicore CR description: Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrlE69] Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of OsIf_GetCoreID to OsIfGetUserID. # Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing core_id check to partition_id check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multicore (multipartition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_}

ID	Subtype	Headline and Description
		reference implementation(s) : ARTD-97002
ARTD-124494	New Feature	[wdg] [IMPLEMENTATION] R21-11 Multicore Concept changes - multipartition vs multicore CR description: Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrIE69] Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of OsIf_GetCoreID to OsIfGetUserID. # Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing core_id check to partition_id check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multicore (multipartition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_} reference implementation(s) : ARTD-97002
ARTD-124882	New Feature	[PORT] Missing implement requirement SWS_Port_00043, SWS_Port_00055, SWS_Port_00082 NewWorkDescription: Validation change from review to test case. To check and ensure the implementation for these requirements: SWS_Port_00043 : The function Port_Init shall avoid glitches and spikes on the affected port pins. Hint: make sure the OBE/IBE as the last steps. SWS_Port_00055 : The function Port_Init shall set the port pin output latch to a default level (defined during configuration) before setting the port pin direction to output. SWS_Port_00082 : The PORT Driver module shall not provide the facility to configure pin level inversion. The default value shall be set (i.e. not inverted) Requirement source: SWS_Port_00082, SWS_Port_00055, SWS_Port_00043 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional:
ARTD-124914	New Feature	[pwm] R21-11 Multicore Concept changes - multipartition vs multicore CR description: Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrIE69] Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of OsIf_GetCoreID to OsIfGetUserID. # Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing core_id check to partition_id check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multicore (multipartition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_} reference implementation(s) : ARTD-97002
ARTD-125095	Bug	[GPT] Srtc_Ip_EnableInterrupt function cannot enable alarm interrupt when alarm flag already set Detailed description (how to reproduce it): SRTC cannot enable alarm interrupt when timer alarm flag (TAF) already set: !image-2024-04-17-13-38-41-959.png! Please refer requirement CPR_RTD_01182.gpt !image-2024-12-31-16-27-59-108.png! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: cannot set interrupt when status flag is set Expected behavior: Allow enable alarm interrupts even in cases where the status flag is set Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]

ID	Subtype	Headline and Description
ARTD-125160	Bug	[GPT] FTM ISR does not fulfill 'CPR_RTD_00011' requirement Detailed description (how to reproduce it): GPT ISRs do not fulfill '{_}CPR_RTD_00011{_}' requirement: !image-2024-04-03-18-32-24-205.png! Observed behavior: According to the driver code showed below, the Interrupt Status Flag is cleared only if both interrupt status and interrupt enable flags are set If the driver isn't initialized and the interrupt enable flag is not set, the ISR return immediately and does not clear the interrupt status flag. For STM: !image-2024-04-03-18-03-51-099.png! For FTM: !image-2024-04-11-13-59-35-778.png! For PIT: !image-2024-04-11-14-04-22-258.png! For Stimer: !image-2024-04-11-14-07-03-672.png|width=900,height=329! For STM: !image-2024-04-03-18-22-31-607.png|width=870,height=650! For FTM: !image-2024-04-11-14-35-27-714.png|width=866,height=592! For PIT: !image-2024-04-11-14-50-27-919.png! For Stimer: !image-2024-04-11-14-53-55-576.png|width=844,height=494! Expected behavior: ISR shall satisfy the CPR_RTD_00011 requirement: "{_}ISR shall check whether its respective driver is initialized. If the driver is not initialized, the ISR shall only clear interrupt status flag and return immediately.{_}"
ARTD-125361	Bug	[GPT] LPIT ISR does not fulfill 'CPR_RTD_00011' requirement Detailed description (how to reproduce it): GPT ISRs do not fulfill '{_}CPR_RTD_00011{_}' requirement: !image-2024-04-03-18-32-24-205.png! Observed behavior: According to the driver code showed below, the Interrupt Status Flag is cleared only if both interrupt status and interrupt enable flags are set If the driver isn't initialized and the interrupt enable flag is not set, the ISR return immediately and does not clear the interrupt status flag. For STM: !image-2024-04-03-18-03-51-099.png! For FTM: !image-2024-04-11-13-59-35-778.png! For PIT: !image-2024-04-11-14-04-22-258.png! For Stimer: !image-2024-04-11-14-07-03-672.png|width=900,height=329! For STM: !image-2024-04-03-18-22-31-607.png|width=870,height=650! For FTM: !image-2024-04-11-14-35-27-714.png|width=866,height=592! For PIT: !image-2024-04-11-14-50-27-919.png! For Stimer: !image-2024-04-11-14-53-55-576.png|width=844,height=494! Expected behavior: ISR shall satisfy the CPR_RTD_00011 requirement: "{_}ISR shall check whether its respective driver is initialized. If the driver is not initialized, the ISR shall only clear interrupt status flag and return immediately.{_}"
ARTD-125557	Bug	[AE] Ae header file shall follows the naming convention as CDD_Ae.h Detailed description (how to reproduce it): For Ae CDD module,files are not prefixed with CDD

ID	Subtype	Headline and Description
		Ae.c should be {*}CDD_Ae{*}.c Ae.h should be {*}CDD_Ae{*}.h Inside Ae.h: #ifndef AE_H #define AE_H should be #ifndef CDD_AE_H #define CDD_AE_H Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: CDD naming convention is wrong Expected behavior: CDD naming convention is correct Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: !image-2024-04-22-09-34-17-140.png! Reference: AUTOSAR_EXP_CDDDesignAndIntegrationGuideline.pdf
ARTD-125566	Bug	[Uart] Define macro in Uart header file should be CDD_UART_H Detailed description (how to reproduce it): For below CDD modules, include files are not prefixed with CDD in *Uart* plugin, CDD_Uart.h is added with below macro inclusion #ifndef UART_H #define UART_H instead, should have been as #ifndef CDD_UART_H #define CDD_UART_H Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: define macro in header file should have CDD in the prefix Expected behavior: define macro in header file should have CDD in the prefix Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-125694	Bug	[icu] Issue with generated macro definition when configuring multiple variant Detailed description (how to reproduce it): When configuration with multiple variant VS0 and VS1 VS_0 : Configuring FTM instance: 1, 2 and 3 VS_1 : Configuring FTM instance: 0, 1 and 2 But in the generation file Ftm_Icu_Ip_Defines.h only generate FTM 0, 1 and 2 / Macros that indicate FTM instances used by ICU. / #ifndef FTM_0_USED #define FTM_0_USED #else #error "FTM_0 instance cannot be used by ICU. Instance locked by another driver!" #endif / brief FTM_0_CH_0 ISR enable/disable / #define FTM_ICU_0_CH_0_ISR_USED (STD_ON) / brief FTM_0_CH_1 ISR enable/disable / #define FTM_ICU_0_CH_1_ISR_USED (STD_ON) / brief FTM 0 OVF interrupt service enable/disable / #define FTM_ICU_0_OVF_ISR_USED (STD_ON)#ifndef FTM_1_USED #define FTM_1_USED #else #error "FTM_1 instance cannot be used by ICU. Instance locked by another driver!" #endif

ID	Subtype	Headline and Description
		/ brief FTM_1_CH_0 ISR enable/disable / #define FTM_ICU_1_CH_0_ISR_USED (STD_ON) / brief FTM_1_CH_1 ISR enable/disable / #define FTM_ICU_1_CH_1_ISR_USED (STD_ON) / brief FTM_1_CH_2 ISR enable/disable / #define FTM_ICU_1_CH_2_ISR_USED (STD_ON) / brief FTM 1 OVF interrupt service enable/disable / #define FTM_ICU_1_OVF_ISR_USED (STD_ON)#ifndef FTM_2_USED #define FTM_2_USED #else #error "FTM_2 instance cannot be used by ICU. Instance locked by another driver!" #endif / brief FTM_2_CH_0 ISR enable/disable / #define FTM_ICU_2_CH_0_ISR_USED (STD_ON) / brief FTM 2 OVF interrupt service enable/disable / #define FTM_ICU_2_OVF_ISR_USED (STD_ON) Preconditions: Configuring multiple variant Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: Generate the instances which are configured Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-125817	New Feature	[ocu] [IMPLEMENTATION] R21-11 Multicore Concept changes - multipartition vs multicore (remaining platforms) CR description: Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrlE69] Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of OsIf_GetCoreID to OsIfGetUserID. # Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing core_id check to partition_id check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multicore (multipartition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_} reference implementation(s) : ARTD-97002
ARTD-125838	Bug	[S32NZ5X 1.0.0] Fee: FEE_CLUSTER_OVERHEAD macro generated by EB and S32DS mismatch Detailed description (how to reproduce it): # Turn FeeSubAddressAreaRetirement on # Generate code with EBtresos # Take the EPC output files as input for S32DS # Generate code with S32DS # Compare the two generated code Preconditions: FeeSubAddressAreaRetirement turned on. Test Case ID (internal TC that caught the defect) optional: Fee_TC_FCT_0705 Observed behavior: The macro on the left was generated by EBtresos, the other by S32DS !image-2024-04-23-14-14-03-870.png|width=982,height=13! Expected behavior: The macros generated by EBtresos and S32DS must have the same value with the same input config files. For example: In the case above, 112 is the correct value. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:
ARTD-126023	Bug	[ICU] Incorrect version checking and duplicate include Detailed description (how to reproduce it): 1 Not include Std_Types.h in Ftm_Icu_Ip_Types.h file but driver still checking version of this file !image-2023-11-30-16-48-09-948.png|thumbnail!

ID	Subtype	Headline and Description
		Solution: Remove checking Std_Types.h version in Ftm_Icu_Ip_Types.h file 2 Dulicate incldue " Wkpu_Ip_Types.h" in Wkpu_Ip.h file. because Wkpu_Ip_Types.h already incldued in Wkpu_Ip_Cfg.h !image-2023-11-30-16-48-20-626.png|thumbnail! !image-2023-11-30-16-48-31-322.png|thumbnail! !image-2023-11-30-16-48-50-598.png|thumbnail! Solution: Remove #include "Wkpu_Ip_Types.h" in Wkpu_Ip.h file Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: N/A Expected behavior: N/A Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-126275	New Feature	[icu] [IMPLEMENTATION] R21-11 Multicore Concept changes - multipartition vs multicore CR description: Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrIE69] Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of OsIf_GetCoreID to OsIfGetUserID. # Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing core_id check to partition_id check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multicore (multipartition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_} reference implementation(s) : ARTD-97002
ARTD-126336	Bug	[FEE]: Some variables' definition conflict with the section attribute in memmap Detailed description (how to reproduce it): Some variables' definition conflict with the section attribute in memmap. For example,: In the Wdg module. The variable: Wdg_aePreviousMode[] which with a default initial value when definition. but it was located in the .mcal_bss section. This will case compiling error or unexpected binary file content. !image-2024-02-22-09-56-49-054.png! In the Gpt module. The variable: Gpt_Ipw_HwInstanceConfig_PB[] which is a normal variable, but it was located to .mcal_const_cfg section, which means it shouldn't be changed. Variable: Gpt_Ipw_ChannelConfig_PB and Fee_JobScheduleLookupTable also have the similar issue. !image-2024-02-22-09-57-07-701.png! Preconditions: Always Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Compiling error or unexpected contents in binary file. Expected behavior: Source code should be following the coding rule more strictly. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Had a code review carefully for all the module code by each module owner.
ARTD-126358	Bug	A bug of system tick calculation if the OS is enabled Detailed description (how to reproduce it): Background: Due to Jira ticket ARTD-61688, the maximum value in the OsIf_Timer_System_Internal_GetElapsed() function has been changed from S32_SysTick->RVR to SYSTICK_MAX. It's totally fine if you use OsIfBaremetalType. However, if the customer add the operating system, this change is a bug because SYSTICK_MAX will not be equal to S32_SysTick->RVR.

ID	Subtype	Headline and Description
		OsIf_Timer_System_Internal_Init() in RTD will not be called(pls check Figure 1), and the value in S32_SysTick->RVR will not be initialized to 0xFFFFFFu, which is equal to SYSTICK_MAX. The maximum value is not a real value as expected. So the value of dif will be affected when using system tick (pls check Figure 2). !image-2024-04-28-10-14-22-947.png|width=571,height=501! !image-2024-04-28-10-19-49-256.png|width=376,height=242! Preconditions: OS is enabled Observed behavior: The calculated dif is greater than the real dif Expected behavior: The maximum value is the actual value from S32_SysTick->RVR instead of SYSTICK_MAX Proposed solution optional: !image-2024-04-30-15-29-12-230.png|width=658,height=256!
ARTD-126556	Bug	[S32NZ5X 1.0.0] Fee: Functional operations failed when enable SubAddressArea Retirement Detailed description (how to reproduce it): # Fee Virtual Page Size is set to 32(maybe other numbers will do as well). # Write page size in Mem_43_Exfls is set equal to Fee Virtual Page Size(32). # Enable SubAddressArea Retirement. Preconditions: As mentioned above. Test Case ID (internal TC that caught the defect) optional: Fee_TC_FCT_0705.c Observed behavior: Fee_eJobResult = MEMIF_JOB_FAILED whenever performs Fee_Init, Fee_Write,... Expected behavior: Fee_eJobResult = MEMIF_JOB_OK; Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Hint: FEE_CLUSTER_OVERHEAD should be calculated so that it is divisible by Write page size or Fee Virtual Page Size
ARTD-126574	Bug	[Mem_InFls] implement requirement CPR_RTD_00444.mem_infls Detailed description (how to reproduce it): !screenshot-1.png|thumbnail! update requirement CPR_RTD_00444.mem_exfls Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Expected behavior: Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: update requirement CPR_RTD_00444.mem_exfls
ARTD-127144	Bug	[BaseNxp][Os] Redefinition of macros errors Detailed description (how to reproduce it): Compile the RTD with GHS Preconditions: Use RTD's OS. Test Case ID (internal TC that caught the defect) optional: Bundle_TC_001 Observed behavior: Following errors raised at compile time: !image-2024-03-07-11-28-20-667.png! Expected behavior: No errors

ID	Subtype	Headline and Description
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-127586	Bug	[FEE] EB tool does not give error mesage when config block with size is too large Detailed description (how to reproduce it): EB tool does not give error mesage when config block with size is too large ex: cluster size: 4096 byte block size : 4000 byte => EB must give error message to user Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: EB tool does not give error mesage when config block with size is too large Expected behavior: EB tool will give error mesage to user when config block with size is too large Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-127680	Bug	[MemAcc] Investigate issue on CT config when selecting to use Read Burst Detailed description (how to reproduce it): Issue appears when generating MemAcc_PBcfg.c when MemAccUseReadBurst. The exact error is in the "MemAcc generation error" attachment Preconditions: MemAccUseReadBurst is checked in the CT config Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Generation error, see attached picture Expected behavior: No issues Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Investigate and see if this issue can be reproduced.
ARTD-127749	Bug	[i2c] Using the sint8 data type for variables 'I2c_as8ChannelHardwareMap' and 'I2c_as8PartitionHardwareMap' would violate the CertC standard Detailed description (how to reproduce it): Value of array I2c_as8ChannelHardwareMap[Channel] will be used to get element of I2c_aeChannelStatus array. So. it cannot be a negative number Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: NA Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-127792	Bug	[S32N5 1.0.0] MEM_EXFLS: Compare generated code between EB and DS is not same when inport EB to DS Detailed description (how to reproduce it): MEM_EXFLS: Compare generated code between EB and DS is not same when inport EB to DS. Detail see the picture below. !image-2024-05-16-17-01-58-316.png|thumbnail! Preconditions:

ID	Subtype	Headline and Description
		N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Compare generated code between EB and DS is not same when inport EB to DS. Expected behavior: Compare generated code between EB and DS is same when inport EB to DS. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-127925	Bug	[FEE-ASRr21.11] fix violations for Cert-c report Detailed description (how to reproduce it): fix violations for Cert-c report as in excel file Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: violations for Cert-c report as in excel file Expected behavior: no violations for Cert-c report as in excel file Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-128136	Bug	[ICU] Build fail when IcuWakeupFunctionalityApi = ON and IcuReportWakeupSource = OFF Detailed description (how to reproduce it): Build fail when IcuWakeupFunctionalityApi = ON and IcuReportWakeupSource = OFF both are RTD notes Preconditions: when IcuWakeupFunctionalityApi = ON and IcuReportWakeupSource = OFF Test Case ID (internal TC that caught the defect) optional: Icu_TS_COT_900 Observed behaviour: N/a Expected behaviour: N/a Note: in the “Expected behaviour” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:N/a
ARTD-128826	Bug	[I2C] EB tresos not raise an error when baudrate out of range operation mode Detailed description (how to reproduce it): Add I2c component into EB tresos Chose Standard operating mode Config values to change the baudrate higher than 400kps Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: The warning does not appear if the baud rate value does not match the mode Expected behavior: Need condition with specific operating mode Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Refer S32DS. The baudrate should be: standard: >0 & <= 100k fast: >100k & <= 400k fast plus: >400k & <=1M
ARTD-129192	Bug	The contents of modules.h files generated by S32DS and EB are inconsistent

ID	Subtype	Headline and Description
		Detailed description (how to reproduce it): Customers found that when they used the same configuration to generate files for Dio_Example_S32G399A_M7 in RTD 4.0.2 P08, the content in modules.h generated using EB and using S32DS was inconsistent. The details are shown in the figure below. !image-2024-05-28-15-07-08-740.png|width=563,height=151! Preconditions: Import Dio_Example_S32G399A_M7 into EB and S32DS Test Case ID (internal TC that caught the defect) optional: Observed behavior: USE_DIO_MODULE value is different in EB and S32DS Expected behavior: USE_DIO_MODULE value is the same in EB and S32DS Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-129198	Bug	[FEE-ASRr21.11] fix violations for CWE, Misra report Detailed description (how to reproduce it): fix violations for CWE, MISRA report as in excel file Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: violations for Misra, CWE report as in excel file Expected behavior: no violations for Misra, CWE report as in excel file Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update code to fix Cert-C violation
ARTD-129297	Bug	[Port] Fix SA Finding: Using invalid direction to set pin direction Detailed description (how to reproduce it): No measure are applied to check invalid parameter "direction" for function Port_SetPinDirection Preconditions: Call Port_SetPinDirection with invalid value (valid value is PORT_PIN_IN = 0, PORT_PIN_OUT = 1, PORT_PIN_INOUT = 2, PORT_PIN_HIGH_Z = 3) Test Case ID (internal TC that caught the defect) optional: None Observed behavior: If the invalid direction passed to Port_SetPinDirection, this function will set the direction as PORT_PIN_HIGH_Z. This is unwanted behavior Expected behavior: If development error detection is enabled, DET must have raise when passed invalid parameter. For example: PORT_E_PARAM_INVALID_DIRECTION reported when calling Port_SetPinDirection(0, PORT_PIN_OUT); Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Check the validation of parameter "direction". If invalid, the function must skipped and return without any action
ARTD-131154	Bug	[Port][ECPD] The "Pin tool functional group" doesn't work properly when using at least 2 container Detailed description (how to reproduce it): Step 1: Create new project on DS Step2: Add Port component, configure with 2 container (PortContainer_0, PortContainer_1) and one "Pin tool functional Group" !image-2023-04-18-09-40-10-833.png|width=499,height=374! !image-2023-04-18-09-40-33-743.png|width=512,height=351! !image-2023-04-18-09-40-50-950.png|width=522,height=330!

ID	Subtype	Headline and Description
		Step3: Generate ECVD file Step4: Import ECVD file to EB, generate EPC file Step5: Import EPC file to S32DS (as new one) Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Port_TS_COT_019 Observed behavior: in the PortContainer_1, "Pin tool functional group" doesn't work properly, it selected PortContainer_0_BOARD_InitPeripherals instead of PortContainer_1_BOARD_InitPeripherals !image-2023-04-18-09-49-39-362.png|width=486,height=346! !image-2023-04-18-09-49-53-955.png|width=494,height=342! !image-2023-04-18-09-50-30-882.png|width=495,height=315! !image-2023-04-18-09-50-54-755.png|width=512,height=312! Expected behavior: "Pin tool functional group" in all containers works properly Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-131430	Bug	[MCU] change the macro contains platform name to DERIVATIVE_XXX Detailed description (how to reproduce it): 1,in driver S32K1-S32M24x have many macro have information platform name as #if (defined(POWER_IP_S32K148) defined(POWER_IP_S32K146) defined(POWER_IP_S32K144) defined(POWER_IP_S32K142)). it should be avoid (coding guidelines rule 6) 2,Correct macro name with the format in S32N (coding guidelines rule 20: All macros used in preprocessor directives shall be in uppercase, with multiple words separated by an underscore following the format) in dev\drivers\AutoSAR\mcu\ip\Clock_S32N \specific\S32N\src\Clock_Ip_Specific.c Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: in driver S32K1-S32M24x have many macro have information platform name and S32N have macro not follow coding guideline Expected behavior: not have macro contain platform name in driver code Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-131501	Bug	[MCU] S32CT Clock files are generated when Mcu component and Clocks tool are disabled Detailed description (how to reproduce it): Step 1: create new project for S32N Step 2: make sure enable clocks tool and add Mcu component Step 3: disable Clocks tool and remove Mcu component !image-2024-06-06-17-49-11-565.png|width=544,height=154! After this step clock file are still generated: !image-2024-06-06-17-49-55-786.png|width=293,height=221! Preconditions: S32CT Test Case ID (internal TC that caught the defect) optional: Test manually Observed behavior: From pull request: [https://bitbucket.sw.nxp.com/projects/ARTD/repos/base/pull-requests/919/overview] If Mcu component is removed from project, it will generate Clock file for Clock IP. But if Clocks tool (Clock Ip) is removed also, the script in pull request still run, so Clock files are still generated.

ID	Subtype	Headline and Description
		Expected behavior: Clock file will not be generated if Clocks tool and Mcu component Clock_ip component are not enabled Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update script in system component to check if Clocks tool is enabled or not via getClockFunctionalGroups();
ARTD-131642	Bug	[S32K1-S32M24X][MCU]Mcu driver do not implement correct step when switching system clock Detailed description (how to reproduce it): Mcu driver not implement correct step by step when switch system clock . !image-2024-06-07-10-05-33-062.png|thumbnail! RM version : S32K1xxRM_Rev14.pdf Preconditions: S32K1xxRM_Rev14.pdf Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Mcu driver not implement correct step by step when switch system clock Expected behavior: Mcu driver implement correct step by step when switch system clock Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-131722	New Feature	[dpga][S32M24x] [IMPLEMENTATION] R21-11 Multicore Concept changes - multipartition vs multicore CR description: Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrIE69] Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of OsIf_GetCoreID to OsIfGetUserID. # Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing core_id check to partition_id check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multicore (multipartition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_} reference implementation(s) : ARTD-97002
ARTD-132710	Bug	[Base] Wrong inclusion of MemAcc_GeneralTypes.h In memory drivers the file MemAcc_GeneralTypes.h is included for E_MEM_SERVICE_NOT_AVAIL but as per AUTOSAR R21-11, MCAL should not include MemAcc_GeneralTypes.h, the define of macro should come from Std_Types.h
ARTD-133731	Bug	[S32ZSE 2.0.0] MemAcc: multicore type 1 run with data fauld when config with one Mem driver and one channel sema4 Detailed description (how to reproduce it): multicore type 1 detect a data fauld whenrun through the function MemAcc_Init Preconditions: Multicore type 1 Use one Mem IP driver Use one channel sema4 Test Case ID (internal TC that caught the defect) optional: MemAcc_TS_090 Observed behavior: Expected behavior: No error when run multicore Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-133773	New Feature	[CAN] Add support for TX Low\High Msg Buff First NewWorkDescription: The new S32N79 (Luxor) for FLEXCAN add support in hardware for new feature as which message buffer first. Lowest-number message buffer first

ID	Subtype	Headline and Description
		If CTRL1[LBUF] is 1, the first (lowest number) active transmission message buffer found is the arbitration winner. MCR[LPRIOEN] has no effect when CTRL1[LBUF] is 1. Highest-priority message buffer first If CTRL1[LBUF] is 0, the arbitration process searches for the active transmission message buffer with the highest priority. The frame of this message buffer has a higher probability of winning the arbitration on the CAN bus when multiple external nodes compete for the bus simultaneously. The sequence of bits considered for this arbitration is called the arbitration value of the message buffer. The transmission message buffer with the lowest arbitration value among all transmission message buffers has the highest priority. If two or more message buffers have equivalent arbitration values, the message buffer with the lowest number is the arbitration winner. The composition of the arbitration value depends on MCR[LPRIOEN]. High Attention to the Arbitration Process in order to don't create priority inversion. Requirement source: Update only if added a new field in configurator for HLD. (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: [....]
ARTD-133783	Bug	[MCL] Issue about Mcl_Deinit function on multicore Detailed description (how to reproduce it): Now, Mcl_Deinit function only work 1 core because the Mcl_pxConfig variable is cleared to NULL_PTR after deinit so when calling Mcl_Deinit function on other core it will don't work and report DET error. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Mcl_TC_FCT_MULTI_001 Mcl_TC_FCT_MULTI_002 Observed behavior: As detailed description Expected behavior: Deinit function work on both two core Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-133829	Bug	[I2C] High speed support Target only Detailed description (how to reproduce it): Create a EBtresos/S32DS and add CDD_I2c component Config Lpi2c channel with master mode Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: I2c_TS_COT_EQ01 Observed behavior: In Reference Manual, Lpi2c support high speed mode for Target only. But we have high speed config field in Lpi2c_Master section Expected behavior: High speed config field should be remove in Lpi2c_Master section if not supported Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-133873	Bug	[MemAcc] Inconsistency between gen tools: EB and S32CT Detailed description (how to reproduce it): some minor differences in generated codes (attached in the ticket) between EB and S32ct: !image-2024-06-23-09-50-29-888.png|thumbnail! Preconditions: Derivative: S32ZSE Multicore Type 1: Enable Multicore Type 3: Enable Test Case ID (internal TC that caught the defect) optional: MemAcc_TS_110

ID	Subtype	Headline and Description
		Observed behavior: Expected behavior: Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-134037	Bug	The condition of HW channel in configuration of LIN and UART driver is incorrect Detailed description (how to reproduce it): Case 1: When config channel LPUART_MSC for UART and LIN driver, no error is occurred Case 2: When config channel LPUART_1 for UART driver and LPUART_IP_11 for LIN driver, the error is occurred All of those cases are incorrect. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: The error is triggered incorrectly when config HW channel Expected behavior: The error is triggered correctly when config HW channel Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update the condition in the configuration code
ARTD-134147	Bug	The condition of HW channel in configuration of LIN and UART driver is incorrect Detailed description (how to reproduce it): The condition of HW channel in configuration of LIN and UART driver is incorrect. The HW channel of LIN will report error when same HW channel in UART, but it don't Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: The error is triggered incorrectly when config HW channel Expected behavior: The error is triggered correctly when config HW channel Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update the condition in the configuration code
ARTD-134243	Bug	[I2C] Can not use Lpi2c instace macro when use Lpi2c2 for first channel Detailed description (how to reproduce it): Create a S32DS project and add Lpi2c_Ip component Add Lpi2c2 for first channel Generate code Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Ip_Lpi2c_TS_200 Observed behavior: LPI2C_CHANNEL_0 defined is 2 Expected behavior: With Lpi2c Ip module we have only 2 instances, so if use Lpi2c1 generated code should be generate to 0 and Lpi2c2 is 1 Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)

ID	Subtype	Headline and Description
		Proposed solution optional: [....]
ARTD-134259	Bug	[BASE][MEM_43_EEP][MEM_43_INFLS][MEM_43_EXFLS] Missing declaration Detailed description (how to reproduce it): Compiling the RTD SW32K1_S32M24x_RTD_R21-11_2.0.0_P05 leads to following errors: !image-2024-06-27-09-32-43-555.png! Same for MEM_43_EXFLS and MEM_43_EEP Preconditions: Compile the RTD Test Case ID (internal TC that caught the defect) optional: Bundle_TC_001 Observed behavior: Errors at compile time Expected behavior: No errors Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-134748	New Feature	[S32K3XX][Base] Add software semaphores for K3 derivatives without the SEMA4 IP NewWorkDescription: There is no way to sync between the HSE core and application core on S32K3XX platforms that don't have hardware semaphores (see attached picture) Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: add Multicore support on K310/K311/K312 derivatives that don't have Sema4 support (from resource files) add Software semaphores support on the previously mentioned derivatives. investigate how these SW semaphores can be assigned in Ram to be also visible for the HSE Core.
ARTD-136540	Bug	[S32Kxx_M2xx] The sequence to enable Pull Up and Pull Down of UCFG(x) should be improved Detailed description (how to reproduce it): With the current sequence to configure Pull Up and Pull Down in HVM (High Voltage Module), we do not disable Pull Down before enabling Pull Up and opposite. !image-2024-07-05-16-13-39-520.png|thumbnail! !screenshot-1.png|thumbnail! This happens when we use multi-variants: in VS_0: Pull Up enabled in VS_1: Pull Down enabled Then Ae_Init is called twice with 2 variants above, this can cause short circuit or failure to configure Pull Up and Pull Down Preconditions: Use multi variants Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: The sequence is not correct Expected behavior: Pull Down must be disabled before enabling Pull Up and opposite Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-136756	Bug	RTD Mcu_Ipw_SetMode() function implementation Customer Kostal is asking whether the issue from *__* https://jira.sw.nxp.com/browse/MCAL-28553 is applicable to S32K1 RTD 4.4 2.0.0 ?
ARTD-136869	Bug	Frame Counter Bug

ID	Subtype	Headline and Description
		RTD Driver shall fix bug related to Frame Counter Fault processing SPI comm has GHS in which Frame Counter occupies bits from 14:8, which gives it range 0 127. Variable u32FrameCounter is never checked and never limited to max value of 127. On AE side, if counter goes above, it starts from 0 again, this needs to be implemented on RTD side
ARTD-136878	Bug	LPSPI timeout fix Current implementation uses Lpspi_Ip_Cancel function in MNI. This fixes problems with SPI, but creates new one. Previous interrupted communication is not aware of this problem, and LPSPI timeout occurs. But due to this, interrupt for handling of AE faults is postponed for this timeout interval (default is 50ms). Please fix it, so if Lpspi_Ip_Cancel is called, ongoing comm is notified and doesnt get stuck in that timeout loop.
ARTD-136882	Bug	Injection of fault not working correctly Reading of Faults and Events when injected through IRQ_SET register is not working properly. Faults/Event is recognized, but GHS is checked only for Event, and if not event is pending, fault is not handled. (function AEC_IRQEventFaultHandler)
ARTD-137041	Bug	[S32K1XX_S32M24X][Port] S32DS Pins tool generate code incorrectly when add PowerAndGround pins Detailed description (how to reproduce it): 1. Create project in S32DS In Pins tool: 2. Add 1 PowerAndGround pins first 3. Add 1 other peripheral pin (e.g GPIO pin) 4. Generate code !image-2023-11-27-14-59-14-258.png! Step 2 and step 3 MUST follow above order Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Code generate for GPIO pin is incorrect, it becomes NOT AVAILABLE almost attribute: !image-2023-11-27-14-57-47-970.png! Expected behavior: PowerAndGround is routed as default, so it should not be added or affect to other pins if user add it into Pins list Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-137418	Bug	[Mem_Infls] The container "MemInstance" and node "MemInstanceId" need to improve to follow requirement [ECUC_Mem_00003] + [ECUC_Mem_00007] Detailed description (how to reproduce it): The current value of the node "MemInstanceId" in the configuration interface does not make sense, Following the requirement "ECUC_Mem_00007", it should be "ID of the related memory driver instance". So "MemInstance" will be one container for each Mem driver instance. for example: The S32K3XX platform has only 1 memory driver instance QSPI_0, so the container "MemInstance" will be one container only for one Mem driver instance (QSPI_0). The value of the node "MemInstanceId" should be 0x0U. The S32ZE platform has 2 memory driver instances QSPI_0 and QSPI_1, so the container "MemInstance" will be 2 containers maximum for 2 Mem driver instances (QSPI_0 QSPI_1). The node "MemInstanceId" value should be 0x0U and 0x01 in order. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: The current node is no reference to the mem driver instance: !image-2024-07-12-22-13-49-860.png! Expected behavior: "MemInstance" will be one container for each Mem driver instance. The node "MemInstanceId" value should be "ID of the related memory driver instance". Requirement source: [ECUC_Mem_00003] [ECUC_Mem_00007] Proposed solution optional: N/A
ARTD-137821	Bug	[platform] Remove adhoc CCOPT values and use generated values based on resource Detailed description (how to reproduce it):

ID	Subtype	Headline and Description
		Remove custom CCOPT and use generated values based on resource used see [PR comment|https://bitbucket.sw.nxp.com/projects/ARTD/repos/platform/pull-requests/751/overview?commentId=2165839] Remove the mentions from UM/IM as well Preconditions: None Test Case ID (internal TC that caught the defect) optional: None Observed behavior: Custom CCOPT are requested to be used by customer to build the code for a specific NPI these values should be generated based on configuration and not forced in CCOPT in makefiles Expected behavior: Use generated values Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-137950	Bug	[Mem_Eep] The container "MemInstance" and node "MemInstanceId" need to improve to follow requirement [ECUC_Mem_00003] + [ECUC_Mem_00007] Detailed description (how to reproduce it): The current value of the node "MemInstanceId" in the configuration interface does not make sense, Following the requirement "ECUC_Mem_00007", it should be "ID of the related memory driver instance". So "MemInstance" will be one container for each Mem driver instance. for example: The S32K3XX platform has only 1 memory driver instance QSPI_0, so the container "MemInstance" will be one container only for one Mem driver instance (QSPI_0). The value of the node "MemInstanceId" should be 0x0U. The S32ZE platform has 2 memory driver instances QSPI_0 and QSPI_1, so the container "MemInstance" will be 2 containers maximum for 2 Mem driver instances (QSPI_0 QSPI_1). The node "MemInstanceId" value should be 0x0U and 0x01 in order. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: The current node is no reference to the mem driver instance: !image-2024-07-12-22-13-49-860.png! Expected behavior: "MemInstance" will be one container for each Mem driver instance. The node "MemInstanceId" value should be "ID of the related memory driver instance". Requirement source: [ECUC_Mem_00003] [ECUC_Mem_00007] Proposed solution optional: N/A
ARTD-137971	Bug	[PWM][S32DS] Cannot generate channel when config channel Id equal total number channel Detailed description (how to reproduce it): [Change channel ID from equal Channel index (value of current Channel ID is 2) to total number (3 channel). In this configuration, we have 3 channel pwm is PwmChannel_0, PwmChannel_1 and PwmChannel_2. After generate with changed Channel Id of PwmChannel_2 configuration, Pwm_VS_0_PBcfg.c loose the changed channel.] Preconditions: [Change value of Channel ID of any channel to total number channel will not raise any error.] Observed behavior: [Change channel ID from 2 to 3 but s32ds did not raise any error !image-2024-07-17-11-21-20-185.png|width=849,height=448! Code generation with this changed configuration missed PwmChannel_2 !image-2024-07-17-11-23-40-825.png|width=847,height=803!] Expected behavior: [Change value of channel ID equal or exceed number of total channel is not valid] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [Channel ID must mapping with channel index]
ARTD-140316	Bug	[LIN] Code not generated when enable alternate clock reference

ID	Subtype	Headline and Description
		Detailed description (how to reproduce it): An error raised when enable node LinClockRef_Alternate as below: !image-2024-07-18-14-12-09-865.png! Preconditions: Test Case ID (internal TC that caught the defect) optional: Lin_TS_009 Observed behavior: Expected behavior: No error raised Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:
ARTD-140821	Bug	[S32K3XX][Mem_Infls] The LoadAc operation still occurs when JobBlockSector =C40_IP_BLOCK_INVALID, BlockAc = C40_IP_BLOCK_INVALID Detailed description (how to reproduce it): The LoadAc operation still occurs when JobBlockSector =C40_IP_BLOCK_INVALID, BlockAc = C40_IP_BLOCK_INVALID in IPW_CheckLoadAc function Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: The LoadAc operation still occurs when JobBlockSector =C40_IP_BLOCK_INVALID, BlockAc = C40_IP_BLOCK_INVALID in IPW_CheckLoadAc function Expected behavior: The LoadAc operation does not doccur when JobBlockSector =C40_IP_BLOCK_INVALID, BlockAc = C40_IP_BLOCK_INVALID in IPW_CheckLoadAc function Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-141479	Bug	[UART] Uart_Init always returns UART_DRV_INIT whether init is successful or not Detailed description (how to reproduce it): When a error occurs. Uart cannot be initialized UartStatePtr->IsDriverInitialized = FALSE !image-2024-07-25-11-12-18-040.png|width=718,height=166! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Uart_au8DriverStatus[PartitionId] = UART_DRV_INIT !image-2024-07-25-11-13-12-209.png|width=480,height=223! Expected behavior: Uart_au8DriverStatus[PartitionId] = UART_DRV_UNINIT Proposed solution optional: N/A
ARTD-141488	Bug	[SPI] In RTD_SPI_UM.pdf file, guideline DMA Mux configuration is incorrect Detailed description (how to reproduce it): !image-2024-01-05-15-52-36-720.png! Following the UM file, DMA MUX Source was configured at the Mcl module instead of RM module as actually. Preconditions: Guideline for DMA MUX configuration must be noted correct in UM file Test Case ID (internal TC that caught the defect) optional:

ID	Subtype	Headline and Description
		N/A Observed behavior: as Detailed description Expected behavior: as Preconditions Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-141482	Bug	[LIN] Lin_Init function always returns LIN_INIT whether init is successful or not Detailed description (how to reproduce it): When a error occurs. Lin cannot be initialized if at least One case in function Lin_InitImplementation below fail, Driver will be not initialized.((TRUE == Lin_apxConfigPtr[PartitionID]_>{-}Lin_ChannelPtr[Lin_ChLoop]_>{-}AllocatedPartition) && (PartitionID == Lin_FLEXIO_apxConfigPtr[PartitionID]{-}>Lin_ChannelPtr[Lin_ChLoop]{-}>ChannelPartitionID)) but actually, driver always return Lin_Init that to notify Driver was initialized !image-2024-07-25-11-22-13-651.png! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Lin_au8LinDrvStatus[PartitionID] = LIN_43_LPUART_FLEXIO_INIT; Expected behavior: Lin_au8LinDrvStatus[PartitionID] = LIN_43_LPUART_FLEXIO_UNINIT; if an error occured. Proposed solution optional: N/A
ARTD-141574	Bug	[LIN] Fix wrong generation for Multicore on S32DS Detailed description (how to reproduce it): Fail generate when config multicore feature on S32DS !image-2024-07-25-18-30-20-632.png|width=644,height=267! !image-2024-07-25-18-30-50-994.png|width=684,height=333! !image-2024-07-25-18-31-25-217.png|width=557,height=244! !image-2024-07-25-18-31-45-476.png! Preconditions: Generate code when config multicore on DS Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: There is error when config multicore on DS Expected behavior: No error when generate code Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-141687	Bug	[Mem_Infls] Async mode is not completely asynchronous Detailed description (how to reproduce it): Currently, write/erase is not completely asynchronous. on each mainfunction call, the job is complete done on hardware. > it is not Async actually. The driver should be return immediate after set job on hardware. And will check the status on the next mainfunction call. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: The driver set job on hardware and waits status on the mainfunction

ID	Subtype	Headline and Description
		Expected behavior: The driver set job on hardware and return immediately, and check status on the next main function. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: implement the async mode correctly
ARTD-141887	Bug	[SJA11XX][BLN_RTD_4.4_SJA11xx_1.0.1] Dio_Example_SJA1110 not working. Detailed description (how to reproduce it): Create new project from example: Dio_Example_SJA1110 Open CT and click "Update Code" Build (Debug_RAM) Reset board. Run(Debug) "Dio_Example_SJA1110_Debug_RAM_PNE" Preconditions: SW2.10 and SW2.11 OFF (so NOT booting from flash) Observed behavior: D47 not blinking; D84 turns off. Expected behavior: D47 blinking; Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Additional information optional: Running Gpio_Sw_Dio_Ip_Example_SJA1110:is Ok. RGB blinks blue. Running Dio_Example_SJA1110 after first running Gpio_Sw_Dio_Ip_Example_SJA1110 (without power-cycle, without reset button) is Ok. RGB blinks blue. Running Dio_Example_SJA1110 after after power-cycle of after reset button does not work.
ARTD-141909	Bug	[MEMACC][S32N 1.0.0] Memory section for _SEC_VAR_INIT_UNSPECIFIED is missing from MemAcc_Bswmd.arxml Detailed description (how to reproduce it): in the file ..\MemAcc_TS_T31D62M10I0R0\generate_PB\src\MemAcc_PBcfg.c below mention section is define, [!IF "node:value(AutosarExt/MemAccMulticoreType3Support)='true'"!][!// [!VAR "MemAccSection" = "'_SEC_VAR_SHARED_INIT_UNSPECIFIED_NO_CACHEABLE'"!][!// [!ELSE!][!// [!VAR "MemAccSection" = "'_SEC_VAR_INIT_UNSPECIFIED'"!][!// [!ENDIF!][!// Observed behavior: Memory Section for "_SEC_VAR_INIT_UNSPECIFIED" is missing from MemAcc_Bswmd.arxml. Expected behavior: Memory Section for "_SEC_VAR_INIT_UNSPECIFIED" should be present in MemAcc_Bswmd.arxml. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update MemAcc_Bswmd.arxml to contain memory Section for "_SEC_VAR_INIT_UNSPECIFIED.
ARTD-142116	Bug	[i2s] Generated headers cannot be built in case sensitive file systems Detailed description (how to reproduce it): Create S32DS Project from Example, configure the project with any valid configuration and generate the code. When building the project there will be errors since the path is invalid. In conclusion, the case sensitive issues are found in RTD drivers. See the slide attached. Preconditions: The host file system being used to build the sources is case-sensitive.

ID	Subtype	Headline and Description
		Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: When you build the project there will be errors since the path is invalid. Expected behavior: Generated sources can be build in case-sensitive systems. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Assume always that sources are building in a case-sensitive file system, therefore use the filename path in the C preprocessor includes directives with the same case as the actual filename path. Steps: # Create new folder "C:\NXPCaseSensitive" (or use the existing ones) now we call it as "working folder". # Run the below command on Windows PowerShell opened in administrator mode:"{*}Enable-WindowsOptionalFeature Online FeatureName Microsoft-Windows-Subsystem-Linux{*}". It takes few seconds and will prompt you to restart your computer. # After restarting, open the command prompt in administrator mode and run: "{*}fsutil.exe file SetCaseSensitiveInfo <YourDestinationFolder> enable{*}" where <YourDestinationFolder> is the folder to enable the case sensitive. # Install S32DS and RTD test environment into the above working folders (from step 1). # Ensure all subfolders inside working folder (from step 1) are renew (re-created, re-layout, re-generated,...) after step 3. Otherwise, we can use this command to enable case sensitive for all created subfolders and no need to renew them: "(Get-ChildItem foldername Recurse Directory).Fullname ForEach-Object {fsutil file setCaseSensitiveInfo "$_" enable}" # Smoke check by building a development test/examples. Hint: # To check the status of case sensitive for the folder: "fsutil.exe file {*}query{*}CaseSensitiveInfo <YourDestinationFolder>". # To disable the case sensitive: "fsutil.exe file {*}Set{*}CaseSensitiveInfo <YourDestinationFolder> {*}disable{*}". # We can enable the case sensitive only the folders that contain the information released to the customers: "../output/eclipse/plugins" folder and "../PlatformSDK_<NPI>" folders
ARTD-142431	Bug	[S32K3XX_S32M27x_5.0.0][SPI]: Can not update the parameters in SpiBaudrateConfig on EB with VariantPostBuild Detailed description (how to reproduce it): Users can not update these nodes with VariantPostBuild. !image-2024-07-31-17-57-18-355.png! !image-2024-07-31-15-32-11-325.png! Code generated: !image-2024-07-31-17-04-29-373.png! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Spi_TC_FCT_1001 Observed behavior: Can not update the parametes in SpiBaudrateConfig on EB with VariantPostBuild. Expected behavior: The parameters in SpiBaudrateConfig must be update correctly on EB with VariantPostBuild. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-142537	Bug	[MEM_INFLS] Generation error in MEM_43_INFLS module if using MEMACC not provided by NXP Detailed description (how to reproduce it): When user add MemAcc config which is not provided by NXP, into Mem_Infls project, they cannot generate code and face below error: !Error.png! MemAccMulticoreType3Support parameter is not present in User MemAcc. In NXP MemAcc this parameter is present under AutosarExt(under non AUTOSAR parameter) Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Cannot generate code in EB Tresos Expected behavior: User can use their MemAcc, as long as it comply Autosar spec

ID	Subtype	Headline and Description
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Need to check if the node path exists before get value of MemAccMulticoreType3Support
ARTD-142540	Bug	[MEM_EXFLS] Generation error in MEM_43_INFLS module if using MEMACC not provided by NXP Detailed description (how to reproduce it): When user add MemAcc config which is not provided by NXP, into Mem_Exfls project, they cannot generate code and face below error: MemAccMulticoreType3Support parameter is not present in User MemAcc. In NXP MemAcc this parameter is present under AutosarExt(under non AUTOSAR parameter) Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Cannot generate code in EB Tresos Expected behavior: User can use their MemAcc, as long as it comply Autosar spec Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Need to check if the node path exists before get value of MemAccMulticoreType3Support
ARTD-142547	Bug	[MEM_EEP]Cloned: Generation error in MEM_43_INFLS module if using MEMACC not provided by NXP Detailed description (how to reproduce it): When user add MemAcc config which is not provided by NXP, into Mem_Infls project, they cannot generate code and face below error: !Error.png! MemAccMulticoreType3Support parameter is not present in User MemAcc. In NXP MemAcc this parameter is present under AutosarExt(under non AUTOSAR parameter) Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Cannot generate code in EB Tresos Expected behavior: User can use their MemAcc, as long as it comply Autosar spec Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Need to check if the node path exists before get value of MemAccMulticoreType3Support
ARTD-142806	Bug	[Mem_InFls] Can not access the functions Ftfc_Ip_ValidAddress when running CCOV Detailed description (how to reproduce it): Functions C40_Ip_ValidAddress, C40_Ip_ValidRangeAddress is used as parameter of C40_IP_DEV_ASSERT(); When running ccov this function do nothing so I can not access C40_Ip_ValidAddress . Preconditions: Functions C40_Ip_ValidAddress, C40_Ip_ValidRangeAddress is used as parameter of C40_IP_DEV_ASSERT(); When running ccov Test Case ID (internal TC that caught the defect) optional: IP_C40_TC_0001 Observed behavior: !image-2024-08-02-17-32-13-502.png!!image-2024-08-02-17-33-46-055.png!!image-2024-08-02-17-34-38-669.png! Expected behavior: Access functions C40_Ip_ValidAddress, C40_Ip_ValidRangeAddress successfully when running ccov Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:
ARTD-142896	Bug	[S32K3XX_S32M27x_5.0.0] Mising VendorApiInfix in ECVD file

ID	Subtype	Headline and Description
		Detailed description (how to reproduce it): Checking the ECVD file on the S32DS config and the ECVD file when importing EPC from EB to S32DS Preconditions: DIO: PVT_DIO_S32K3XX_S32M27X_RTM_5.0.0_010 PORT: PVT_PORT_S32K3XX_S32M27X_RTM_5.0.0_011 DIO_ITG: PVT_TEST_DIO_S32K3XX_M27X_500_V34 PORT_ITG: PVT_TEST_PORT_S32K3XX_M27X_500_V30 Test Case ID (internal TC that caught the defect) optional: Dio: Dio_TS_ECPD_001 Observed behavior: Issue 1: Dio component Mising VendorApiInfix in ECVD file !image-2024-08-05-15-01-04-360.png! Issue 2: Port component SYNC_SIZE and SYNC_VALUE options appear together in VendorApiInfix (Port.component) !image-2024-08-05-14-44-37-988.png! Expected behavior: full VendorApiInfix information in ECVD file Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-143359	Bug	[ICU]Generated headers cannot be built in case sensitive file systems Detailed description (how to reproduce it): Create S32DS Project from Example, configure the project with any valid configuration and generate the code. When building the project there will be errors since the path is invalid. In conclusion, the case sensitive issues are found in RTD drivers. See the slide attached. Preconditions: The host file system being used to build the sources is case-sensitive. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: When you build the project there will be errors since the path is invalid. Expected behavior: Generated sources can be build in case-sensitive systems. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Assume always that sources are building in a case-sensitive file system, therefore use the filename path in the C preprocessor includes directives with the same case as the actual filename path. Steps: # Create new folder "C:\NXPCaseSensitive" (or use the existing ones) now we call it as "working folder". # Run the below command on Windows PowerShell opened in administrator mode:"{*}Enable-WindowsOptionalFeature Online FeatureName Microsoft-Windows-Subsystem-Linux{*}". It takes few seconds and will prompt you to restart your computer. # After restarting, open the command prompt in administrator mode and run: "{*}fsutil.exe file SetCaseSensitiveInfo <YourDestinationFolder> enable{*}" where <YourDestinationFolder> is the folder to enable the case sensitive. # Install S32DS and RTD test environment into the above working folders (from step 1). # Ensure all subfolders inside working folder (from step 1) are renew (re-created, re-layout, re-generated,...) after step 3. Otherwise, we can use this command to enable case sensitive for all created subfolders and no need to renew them: "(Get-ChildItem foldername Recurse Directory).Fullname ForEach-Object {fsutil file setCaseSensitiveInfo "$_" enable}" # Smoke check by building a development test/examples. Hint: # To check the status of case sensitive for the folder: "fsutil.exe file {*}query{*}CaseSensitiveInfo <YourDestinationFolder>". # To disable the case sensitive: "fsutil.exe file {*}Set{*}CaseSensitiveInfo <YourDestinationFolder> {*}disable{*}". # We can enable the case sensitive only the folders that contain the information released to the customers: "../output/eclipse/plugins" folder and "../PlatformSDK_<NPI>" folders
ARTD-143423	Bug	[MemAcc] Fix violations with Plugin Check Detailed description (how to reproduce it): There are duplicate UUID tags in xdm file: !image-2024-08-08-08-36-12-361.png|thumbnail! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA

ID	Subtype	Headline and Description
		Observed behavior: NA Expected behavior: Fixed all violations Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Change UUID
ARTD-143734	Bug	[icu] Validation of partition reference should be OS version independent Detailed description (how to reproduce it): Some RTD drivers (Adc, Crc, Dio, Icu, Platform, Port, Pwm, Serdes) have the following check when [drv]MulticoreEnabled = 'true': <a:tst expr="count(text:grep(node:refs('ASPathDataOfSchema:/TS_T40D11M40I2R0/Os')/OsApplication/*/ OsAppEcucPartitionRef, .)) = 0" true="The ECUC partition must be defined in OS."/> RTD shall not validate references against a certain OS version. There can be any OS version, either in a Tresos compatible plugin format or not. Creating a dependency to a certain version will cause the following validation error: !https://jira.sw.nxp.com/secure/attachment/1295507/image-2023-11-27-16-04-31-321.png! Preconditions: [drv]MulticoreEnabled is set to 'true' and a correct configuration of the partition is done in OS. Test Case ID (internal TC that caught the defect) optional: none Observed behavior: The following drivers Adc, Crc, Dio, Icu, Platform, Port, Pwm, Serdes show validation error is Tresos even when a correct reference to a partition is configured. Expected behavior: No error should be caused if different OSes are used. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Option 1) Change node:refs('ASPathDataOfSchema:/TS_T40D11M40I2R0/Os')/OsApplication/*/OsAppEcucPartitionRef To something like: as:modconf('Os')[1]/OsApplication/*/OsAppEcucPartitionRef as found in CAN. Option 2) Alternatively this check can be removed altogether since drivers should not be aware of Os partition configuration. The reason is simple: partition configuration is part of high level components like: EcuM, BSWM and RTE, all above driver level.
ARTD-143714	Bug	[gpt] Incorrect section memory for the variables Detailed description (how to reproduce it): As in [Driver]_Memmap.h, section memory xx_START_SEC_CONFIG_DATA_xx refers to the .mcal_const_cfg section, which is for constant value. However, in Gpt driver some variables are defined as a variable which is not a const type. for example in Gpt_Ipw_PBcfg.c #define GPT_START_SEC_CONFIG_DATA_UNSPECIFIED #include "Gpt_MemMap.h" Gpt_Ipw_HwInstanceConfigType Gpt_Ipw_HwInstanceConfig_PB_VS_0_P_EcucPartition_0[5U]= {... } Gpt_Ipw_HwChannelConfigType Gpt_Ipw_ChannelConfig_PB_VS_0_P_EcucPartition_0[8U]= {... } #define GPT_STOP_SEC_CONFIG_DATA_UNSPECIFIED #include "Gpt_MemMap.h"

ID	Subtype	Headline and Description
		Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: Correct section memory Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Need to add const for the variable for data configuration in generation file
ARTD-143792	Bug	[WDG] Code generation fails when WdgEnableMultiPartitionSupport node is enabled on S32CT Detailed description (how to reproduce it): Step 1: Create new project on S32DS with S32Z270/S32E278 derivative Step 2: Add Wdg and some dependencies to the project Step 3: Enable WdgEnableMultiCoreSupport node Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Code generation fails when WdgEnableMultiCoreSupport node is enabled on S32CT !image-2023-08-16-14-41-28-664.png|width=544,height=516! Expected behavior: Generate code successfully Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-144106	Bug	[BaseNXP][General] Incompatible data type definitions Detailed description (how to reproduce it): Using SW32ZE_RTD_R21-11_2.0.0 Preconditions: Observed behavior: BaseNXP causing compilation errors from GHS when there isn't NO_STD_INT_H defined. Definition of uint32 and uint32_t differs between stdint.h and Platform_Types.h. typedef uint32_t uint32; was removed from Platform_Types.h if using stdint.h. [Change introduced here|https://bitbucket.sw.nxp.com/projects/ARTD/repos/base/pull-requests/2004/diff#specific%2FS32ZSE%2Fwizard_data%2FPlatform_Types.h?f=190] There wasn't any communication from RTD team to dependent products (SAF) about the change. Dependency among software products is already in place, but not working. Now there are headers using uint32_t and other source code using uint32 which are not compatible causing errors even in RTD itself. Details: GHS stdint.h defines uint32_t as unsigned int Platform_Types.h defines uint32 as unsigned long (CPU_TYPE_32) Can't mix long and int without errors. [Explicit casting as introduced in the release|https://bitbucket.sw.nxp.com/projects/ARTD/repos/adc/commits/12e963e16b54bb3912b63cc609f27c8036c90ee9#ip%2FIP_ADC_SAR%2Finclude%2FAdc_Sar_Ip_HwAccess.h] to mitigate this error is not solution rather than workaround. Expected behavior: Compilation for all supported compilers works without errors. Proposed solution optional: There are several possibilities: Revert autosar data types definitions using compiler data types as before. ({+}typedef uint32_t uint32;{+}) Use only autosar data types across all source files and header files. Change definition of autosar data types to match compiler definitions. Compiler-dependent definitions.
ARTD-144186	Bug	[S32K3XX_S32M27x_5.0.0][I2S]: Mismatch between ebt and ds configuration Detailed description (how to reproduce it):

ID	Subtype	Headline and Description
		There are mismatch between ebt and ds configuration Preconditions: SaiSampleRate = 0 in EBT and SaiSampleRate = 0 in DS !image-2024-08-14-18-34-38-120.png|thumbnail! [...] Test Case ID (internal TC that caught the defect) optional: I2s_TS_COT_004 Observed behavior: There is not mismatch between ebt and ds configuration Expected behavior: There are mismatch between ebt and ds configuration Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-144548	New Feature	[S32K1][gpt] [IMPLEMENTATION] R21-11 Multicore Concept changes - multipartition vs multicore CR description: Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrlE69] Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of '{_}OsIf_GetCoreID(){_}' by '{_}OsIf_GetUserId(){_}'. # Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing '{_}coreID{_}' check to '{_}userId{_}' check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multi-core (multi-partition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_} reference implementation(s) : ARTD-97002 {_}*Proposed Solution*{_}: The issue was partially integrated by [ARTD-154091|https://jira.sw.nxp.com/browse/ARTD-154091]; The calls of '{_}OsIf_GetCoreID(){_}' function should be also replaced by '{_}OsIf_GetUserId(){_}'
ARTD-144825	Bug	[otp] Generated headers cannot be built in case sensitive file systems Detailed description (how to reproduce it): Create S32DS Project from Example, configure the project with any valid configuration and generate the code. When building the project there will be errors since the path is invalid. In conclusion, the case sensitive issues are found in RTD drivers. See the slide attached. Preconditions: The host file system being used to build the sources is case-sensitive. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: When you build the project there will be errors since the path is invalid. Expected behavior: Generated sources can be build in case-sensitive systems. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Assume always that sources are building in a case-sensitive file system, therefore use the filename path in the C preprocessor includes directives with the same case as the actual filename path. Steps: # Create new folder "C:\NXPCaseSensitive" (or use the existing ones) now we call it as "working folder". # Run the below command on Windows PowerShell opened in administrator mode:"{*}Enable-WindowsOptionalFeature Online FeatureName Microsoft-Windows-Subsystem-Linux{*}". It takes few seconds and will prompt you to restart your computer. # After restarting, open the command prompt in administrator mode and run: "{*}fsutil.exe file SetCaseSensitiveInfo <YourDestinationFolder> enable{*}" where <YourDestinationFolder> is the folder to enable the case sensitive. # Install S32DS and RTD test environment into the above working folders (from step 1). # Ensure all subfolders inside working folder (from step 1) are renew (re-created, re-layout, re-generated,...) after step 3. Otherwise, we can use this command to enable case sensitive for all created subfolders and no need to renew them: "(Get-ChildItem foldername Recurse Directory).Fullname ForEach-Object {fsutil file setCaseSensitiveInfo "$_" enable}" # Smoke check by building a development test/examples. Hint:

ID	Subtype	Headline and Description
		# To check the status of case sensitive for the folder: "fsutil.exe file {*}query{*}CaseSensitiveInfo <YourDestinationFolder>". # To disable the case sensitive: "fsutil.exe file {*}Set{*}CaseSensitiveInfo <YourDestinationFolder> {*}disable{*}". # We can enable the case sensitive only the folders that contain the information released to the customers: "../output/eclipse/plugins" folder and "../PlatformSDK_<NPI>" folders
ARTD-145349	Bug	[UART] Return value of LPUART_Ip AbortReceivingData API is incorrect Detailed description (how to reproduce it): When the transmission data is completed, the receive data buffer or fifo is empty. After that, we call the abort API, the status need to return is abort, not timeout (the example is in the ticket ARTD-136885) Preconditions: The data size for TX is small than the data size for RX Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Return value of LPUART_Ip AbortReceivingData API is incorrect (timeout) in this case Expected behavior: Return value of LPUART_Ip AbortReceivingData API is abort in this case Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update condition of the LPUART_Ip AbortReceivingData API to return value correctly
ARTD-145372	New Feature	[PLATFORM] System_Ip_DeviceRegisters static code should be replaced by generated code NewWorkDescription: Currently, System_Ip_DeviceRegisters is static code which define FPU features for each specific NPI. That definition should be generated base on derivative in resource. To make code more optimal , we should not use specific NPI conditional compilation in source code (ex: #if defined(S32K116)) . !image-2024-08-26-15-30-35-379.png|thumbnail! Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Remove System_Ip_DeviceRegisters.h Add FPU feature definition in System_Ip_CfgDefines.h Use M4 to exclude other platforms in System_Ip_CfgDefines.h file
ARTD-145634	Bug	[AE] [S32K3_M27x 5.0.0]: Frame counter issues Detailed description (how to reproduce it): Issue 1: if a fault occurs, the Framecount in GHS will be reset but with the current implementation, the variable Aec_Ip_u32FrameCounter is not reset. Frame Count Error will be raised and the program will jump to Frame Count callback (if enabled) in every SPI read or write. In my point of view, this is not an expected behavior or we shall provide an option on configuration interface (EB/CT) for users to decide to reset Aec_Ip_u32FrameCounter or not Issue 2: After a Frame Count Fault is raised and handled, Aec_Ip_u32FrameCounter and Frame Count in GHS should be the same (The same as clearing the interrupt flag). If not, the program will jump to callback in every SPI read/write and the next fault can not be handled. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: [Ae_TC_FCT_0501|https://bitbucket.sw.nxp.com/projects/ARTD/repos/test_ae/browse/generic/src/Ae_TC_FCT_0501.c] Observed behavior: Aec_Ip_u32FrameCounter is not reset after faults occur Expected behavior: Aec_Ip_u32FrameCounter is reset after faults occur Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-145686	Bug	[MemAcc] Can't release resource when cancel a job that in suspending state Detailed description (how to reproduce it): when a job in state suspending, i call Memacc cancel. after that, i call other job, but job state just in starting, i can't change it to processing. The program can't continue to operate. Preconditions: job in suspending Test Case ID (internal TC that caught the defect) optional: MemAcc_TC_313 Observed behavior: job call after cancel not working

ID	Subtype	Headline and Description
		Expected behavior: job call after cancel can working normal Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-145689	Bug	[MemAcc]: Job with higher priority executed after lower priority job in state retry Detailed description (how to reproduce it): job with lower priority in state retry call job with higher priority call main func Preconditions: job with lower priority in state retry Test Case ID (internal TC that caught the defect) optional: MemAcc_TC_334 Observed behavior: job with higher priority executed after lower priority job Expected behavior: job with lower priority executed after higher priority job Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-145727	Bug	[MCL] Fix 8.4 misra violations rule Detailed description (how to reproduce it): Fix 8.4 misra violations rule because Mcl is not in comment case Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: N/A Expected behavior: N/A Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Add extern for all 8.4 violations
ARTD-145792	Bug	[BASE] Osif_Software_Semaphore.c build fail with ARM_A64_ARCH Detailed description (how to reproduce it): Build project used BaseNxp with Resource is ARM_A64_ARCH and compiler is: gcc-9.2-arm64-eabi Preconditions: Test Case ID (internal TC that caught the defect) optional: Base_TS_COT_106 Observed behavior: Project build fail with error as following figure: !screenshot-1.png|thumbnail! Also note that base has requirement related to ARM_A64_ARCH CPR_RTD_00576.base Base shall support ARMv8 architecture. Rationale:Base is a dependency for NXP SW products., therefore support for 32bit and 64bit is required. Expected behavior: BaseNxp can be built pass with ARM_A64_ARCH Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:
ARTD-145817	Bug	[MEMACC] Memory section for _SEC_VAR_INIT_UNSPECIFIED is missing from MemAcc_Bswmd.arxml - Follow up Detailed description (how to reproduce it):

ID	Subtype	Headline and Description
		In the file ..\MemAcc_TS_T31D62M10I0R0\generate_PB\src\MemAcc_PBcfg.c below mention section is define, [!IF "node:value(AutosarExt/MemAccMulticoreType3Support)='true'"!][!// [!VAR "MemAccSection" = "'_SEC_VAR_SHARED_INIT_UNSPECIFIED_NO_CACHEABLE'"!][!// [!ELSE!][!// [!VAR "MemAccSection" = "'_SEC_VAR_INIT_UNSPECIFIED'"!][!// [!ENDIF!][!// Consider merging all changes from this pull request into the develop branch: [Pull Request #300: [ARTD-141909] Memory section for SEC_VAR_INIT_UNSPECIFIED is missing from MemAcc_Bswmd.arxml NXP Bitbucket|https://bitbucket.sw.nxp.com/projects/ARTD/repos/memacc/pull-requests/300/overview] Observed behavior: Memory Section for "_SEC_VAR_INIT_UNSPECIFIED" is missing from MemAcc_Bswmd.arxml. Expected behavior: Memory Section for "_SEC_VAR_INIT_UNSPECIFIED" should be present in MemAcc_Bswmd.arxml. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update MemAcc_Bswmd.arxml to contain memory Section for "_SEC_VAR_INIT_UNSPECIFIED.
ARTD-146586	New Feature	[IMPLEMENTATION] S32K3: Async vs Sync Mode on Spi Driver The S32K3 SPI driver that we provide is supposed to be Level 2 AUTOSAR compliant !MicrosoftTeams-image.png|width=532,height=141! Level 2 means that the controller can handle both asynchronous and synchronous transmissions. However, this is partially true with our S32K3 SPI driver, as we can handle asynchronous and synchronous transmissions as long as we set two different SPI channels. Don't be confused with the mechanism. One thing is the mode (asynchronous or synchronization) and another thing is the mechanism (polling or interrupt). The problem is with the mode, not in the mechanism. In summary, SPI driver can handle 3 modes: Sync Mode Async Mode Interrupt Async Mode Polling However, with the current driver implementation: If SPI0 is configured in Sync Mode... we cannot use the AsyncTransmit() function If SPI0 is configured in Async Mode... we can do both interrupt and polling in runtime. But we cannot use SyncTransmit() function The reason why... is because of the SpiPhyUnitSync parameter in Tresos/ConfigTools. When the user calls the function Spi_AsyncTransmit() or Spi_SyncTransmit().... there is a check in the functions to see if the channel is configured as Sync or Async. This is preventing the customer for using Async and Sync calls on the same SPI channel !image.png|width=538,height=197! The request/update is to remove the SpiPhyUnitSync parameter as this is obsolete since Autosar 4.3 !image (1).png|width=479,height=341!
ARTD-148103	Bug	[S32K1] ADC Interleave: PTB13 when configure ADC signals always configure interleave feature Detailed description (how to reproduce it): In situation that configure with PTC0 mode adc0_se8 and PTB13 mode adc1_se8 (SIM_CHIPCTL = 0), mean NO Interleave: !image-2024-09-20-20-02-07-505.png|width=408,height=302! But in the Pins Tool was throw an error when configure both PTC0 and PTB13: !image-2024-09-20-20-03-27-946.png! Preconditions: Test Case ID (internal TC that caught the defect) optional: None Observed behavior: Error with the configuration tool Expected behavior: Can configure between use Interleave feature or not.

ID	Subtype	Headline and Description
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: None
ARTD-148429	Bug	S32K1xx RTD drivers - Lptmr_Gpt_Ip.h header file bug Detailed description (how to reproduce it): [Lptmr_Gpt_Ip.h header file can not work during C implementation.] Preconditions: [C implementation and LPTMR_GPT_IP_USED is STD_OFF] Test Case ID (internal TC that caught the defect) optional: [None] Observed behavior: [build ha[^Lptmr_Gpt_Ip header file bug RTD200.pptx]s error] Expected behavior: [None] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Complete the extern "C" \{ ... } block properly according to the description from the attached file
ARTD-148524	New Feature	[platform] [IMPLEMENTATION] R21-11 Multicore Concept changes - multipartition vs multicore CR description: Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrlE69] Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of OsIf_GetCoreID to OsIfGetUserID. # Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing core_id check to partition_id check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multicore (multipartition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_} reference implementation(s) : ARTD-97002
ARTD-148861	Bug	[base] Fix remaining code violation rule 29 Detailed description (how to reproduce it): All violations and analysis in ARTD-147064 Code review Rule||Issue||Effort||Implementation Rule 29|Local variables and function parameters do not follow the coding convention (almost)|8h|SAF85-SAF86-S32R41 next release Preconditions: All violations and analysis in ARTD-147064 Test Case ID (internal TC that caught the defect) optional: ARTD-147064 Observed behavior: Some violations in code design checklist Expected behavior: All violations should be fixed Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Fix violations
ARTD-149548	Bug	[GPT] Missing SRtc_Ip_Types.h header file when using SRtc in IP layer Detailed description (how to reproduce it): Customer add SRtc in IP layer Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: When compile the project, error of missing SRtc_Ip_Types.h is reported.

ID	Subtype	Headline and Description
		!image-2024-10-03-14-43-25-077.png! This file should be copied from RTD package source to the folder name "RTD/Include" in DS project, but i can't see it. When i add GPT in MCAL layer, then the header files got copied. !image-2024-10-03-14-45-15-823.png! But customer don't wanna use GPT in MCAL. Expected behavior: Adding SRtc and be able to use it in IP layer, the SRtc IP header files be copied correctly. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-149585	Bug	[mcu] Generated headers cannot be built in case sensitive file systems Detailed description (how to reproduce it): Create S32DS Project from Example, configure the project with any valid configuration and generate the code. When building the project there will be errors since the path is invalid. In conclusion, the case sensitive issues are found in RTD drivers. See the slide attached. Preconditions: The host file system being used to build the sources is case-sensitive. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: When you build the project there will be errors since the path is invalid. Expected behavior: Generated sources can be build in case-sensitive systems. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Assume always that sources are building in a case-sensitive file system, therefore use the filename path in the C preprocessor includes directives with the same case as the actual filename path. Steps: # Create new folder "C:\NXPCaseSensitive" (or use the existing ones) now we call it as "working folder". # Run the below command on Windows PowerShell opened in administrator mode:"{*}Enable-WindowsOptionalFeature Online FeatureName Microsoft-Windows-Subsystem-Linux{*}". It takes few seconds and will prompt you to restart your computer. # After restarting, open the command prompt in administrator mode and run: "{*}fsutil.exe file SetCaseSensitiveInfo <YourDestinationFolder> enable{*}" where <YourDestinationFolder> is the folder to enable the case sensitive. # Install S32DS and RTD test environment into the above working folders (from step 1). # Ensure all subfolders inside working folder (from step 1) are renew (re-created, re-layout, re-generated,...) after step 3. Otherwise, we can use this command to enable case sensitive for all created subfolders and no need to renew them: "(Get-ChildItem foldername Recurse Directory).Fullname ForEach-Object {fsutil file setCaseSensitiveInfo "$_" enable}" # Smoke check by building a development test/examples. Hint: # To check the status of case sensitive for the folder: "fsutil.exe file {*}query{*}CaseSensitiveInfo <YourDestinationFolder>". # To disable the case sensitive: "fsutil.exe file {*}Set{*}CaseSensitiveInfo <YourDestinationFolder> {*}disable{*}". # We can enable the case sensitive only the folders that contain the information released to the customers: "../output/eclipse/plugins" folder and "../PlatformSDK_<NPI>" folders
ARTD-149656	Bug	[spi] Generated headers cannot be built in case sensitive file systems Detailed description (how to reproduce it): Create S32DS Project from Example, configure the project with any valid configuration and generate the code. When building the project there will be errors since the path is invalid. In conclusion, the case sensitive issues are found in RTD drivers. See the slide attached. Preconditions: The host file system being used to build the sources is case-sensitive. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: When you build the project there will be errors since the path is invalid. Expected behavior: Generated sources can be build in case-sensitive systems.

ID	Subtype	Headline and Description
		Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Assume always that sources are building in a case-sensitive file system, therefore use the filename path in the C preprocessor includes directives with the same case as the actual filename path. Steps: # Create new folder "C:\NXPCaseSensitive" (or use the existing ones) now we call it as "working folder". # Run the below command on Windows PowerShell opened in administrator mode:"{*}Enable-WindowsOptionalFeature Online FeatureName Microsoft-Windows-Subsystem-Linux{*}". It takes few seconds and will prompt you to restart your computer. # After restarting, open the command prompt in administrator mode and run: "{*}fsutil.exe file SetCaseSensitiveInfo <YourDestinationFolder> enable{*}" where <YourDestinationFolder> is the folder to enable the case sensitive. # Install S32DS and RTD test environment into the above working folders (from step 1). # Ensure all subfolders inside working folder (from step 1) are renew (re-created, re-layout, re-generated,...) after step 3. Otherwise, we can use this command to enable case sensitive for all created subfolders and no need to renew them: "(Get-ChildItem foldername Recurse Directory).Fullname ForEach-Object {fsutil file setCaseSensitiveInfo "$_" enable}" # Smoke check by building a development test/examples. Hint: # To check the status of case sensitive for the folder: "fsutil.exe file {*}query{*}CaseSensitiveInfo <YourDestinationFolder>". # To disable the case sensitive: "fsutil.exe file {*}Set{*}CaseSensitiveInfo <YourDestinationFolder> {*}disable{*}". # We can enable the case sensitive only the folders that contain the information released to the customers: "../output/eclipse/plugins" folder and "../PlatformSDK_<NPI>" folders
ARTD-151012	Bug	[I2c] Missing macro I2C_CTRL_HS_MODE_AVAILABLE on HLD layer Detailed description (how to reproduce it): Run test I2C_TS_0999 Preconditions: Macro use to guard High Speed Mode on HLD layer are guarding by IPV macro. But with this implement it will impact to another platform if it doesn't use this IPV. It should replace by macro on HLD layer. According SAF platform it already updates in the PR: [Pull Request #968: [ARTD-147562] [SAF8X R41 3.0.0][I2C]: Tresos and DS Examples part 2 NXP Bitbucket|https://bitbucket.sw.nxp.com/projects/ARTD/repos/i2c/pull-requests/968/overview]. Test Case ID (internal TC that caught the defect) optional: I2C_TS_0999 and test case from testing side Observed behavior: Macro use to guard High Speed Mode on HLD layer are guarding by IPV macro. Expected behavior: Using macro on HLD layer to guard High Speed mode on HLD layer. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-151300	Bug	[ADC] Generation of some defines in precompile files is based on variant-aware nodes Detailed description (how to reproduce it): Need to check all defines which generated in precompile files but value obtained from variant-aware nodes (E.g: CTU_IP_FIFO_DMA_RAW_EN or AdcGroup_0_AdcChannel_1 symbolic names) Preconditions: Using test with VARIANT_NO > 1 Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: the CTU_IP_FIFO_DMA_RAW_EN might be = OFF for all variants (if last variant in EB or first variant in CT = false) Expected behavior: defines which generated in precompile file shouldn't obtain value from variant-aware node Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-151605	New Feature	[IMPLEMENTATION] [adc] PDB should be used in conjunction only with ADC for triggering conversions and interrupt generation CR description:_* In the current RTD architecture design for S32M24x hardware parts, PDB will not be able to generate an interrupt delay and an ADC trigger conversion for the same PDB instance. In motor control applications, the PDB should be able to generate triggers for ADC in order to measure currents required for motor control algorithms and also to generate an interrupt at the value specified by interrupt delay register(PDB-IDLY). In general, this interrupt is used to update at runtime the PDB pretrigger delay values for ADC conversion to acquire new data for 3 phase currents reconstruction measurement. Also, the PDB peripheral is not an independent component; it can be used just in conjunction with ADC peripheral to trigger ADC conversions at specific timing.

ID	Subtype	Headline and Description
		Reason for this change: Flexibility of PDB configuration to trigger ADC conversion for motor control applications. The customers working on motor control applications should be able to: Enable/Disable Interrupt generation for a PDB instance used to trigger ADC conversions; Configure the PDB interrupt callback; Set the Interrupt delay value when the PDB interrupt is generated. Use-case: The most important use-case is the single-shunt motor control application which can be implemented using S32M24X-L/C064EVB. In the following picture is the timing diagram for this kind of motor control use-case. !image-2024-04-02-09-56-55-980.png! It can be seen that the PDB interrupt is used to enable FTM3 init trigger for ADC conversions and also to update the PDB pretrigger delay values that are dinamically set to acquire data for 3 phase current reconstruction algorithm.
ARTD-151853	Bug	[Platform] Trust function is not include in header file Detailed description (how to reproduce it): There are 3 errors: No1: This funtion is in trusted header but don't found in other file trusted_header trusted_call: NOK Mpu_M7_Ip_GetErrorDetails No2: This list function have in driver code but don't found in trusted header file trusted_call trusted_header: NOK Mpu_M7_Ip_GetErrorRegisters System_Ip_ConfigIrq System_Ip_SetAhbSlavePriority !image-2024-10-23-14-48-37-085.png! No3: This funtions is declare with scope static Mpu_M7_Ip.c: Mpu_M7_Ip_GetErrorRegisters, scope: static Preconditions: MPU M7 IP System_Ip IP Test Case ID (internal TC that caught the defect) optional: Platform_TS_COT_TRUST_002 Observed behavior: function have in driver but not found in trusted header Expected behavior: Trusted functions in the plugin have been included in the trusted header file Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Mpu_M7_Ip_GetErrorRegisters() should be declare in the trusted header file
ARTD-152215	Bug	S32K1XX - Period for TimingEvent_MainFunction should be higher than 0 Detailed description (how to reproduce it): In MemAcc_Bswmd.arxml, Mem_43_EEP_Bswmd.arxml, Mem_43_EXFLS_Bswmd.arxml and Mem_43_INFLS_Bswmd.arxml PERIOD for TimingEvent_MainFunction is set to 0. According to Autosar's requirements value should be higher than 0. !image-2024-10-21-13-42-09-907.png! Preconditions: [constr_2031] Period of TimingEvent shall be greater than 0 d The value of the attribute period of TimingEvent shall be greater than 0. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: as Detailed description Expected behavior: as Preconditions Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)

ID	Subtype	Headline and Description
		Proposed solution optional: N/A
ARTD-152231	Bug	[UART] The path reference to MCL is incorrect in configuration file (epd) Detailed description (how to reproduce it): Customers (using epd) cannot connect to the DMA channel when the path reference to MCL (right in capture) is incorrect as below: !image-2024-10-21-14-42-08-725.png! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: DMA channel can not be configured Expected behavior: DMA channel can be configured Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-152550	New Feature	S32M24xC doesn't support in S32DS Pins tool NewWorkDescription: Package S32K1_S32M24x Real-Time Drivers Version 2.0.0 doesn't support S32M24x{color:#de350b}*C* {color:#172b4d}in S32DS Pins tool. There're 2 series S32M24XL and S32M24xC described in S32M24X_IOMUX.xlsx, and it has some slight difference in Pinout: !image-2024-10-23-14-44-09-347.png! !image-2024-10-23-14-44-34-247.png! !https://community.nxp.com/t5/image/serverpage/image-id/303989i34F5D177C9C037E4/image-dimensions/688x643?v=v2! Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: In S32DS Pins tool, user can able to select S32M24XL or S32M24XC in switch processor feature
ARTD-152542	Bug	[Base] Trust function is not include in trusted header file Detailed description (how to reproduce it): {*}1. OsIf_Interrupts_ResumeAllInterrupts{*}: function is in driver file but not found in trusted header file (currently included in OsIf_Internal.h) {*}2. OsIf_Interrupts_SuspendAllInterrupts{*}: function is in driver file but not found in trusted header file !image-2024-10-23-13-57-10-224.png! {*}3. OsIf_Timer_System_Internal_GetCounter{*}: function is in driver file but not found in trusted header file (currently included in OsIf_Timer_System_Internal_GenericTimer.h and OsIf_Timer_System_Internal_Systick.h) {*}4. OsIf_Timer_System_Internal_GetElapsed{*}: function is in driver file but not found in trusted header file (currently included in OsIf_Timer_System_Internal_GenericTimer.h and OsIf_Timer_System_Internal_Systick.h) {*}5. OsIf_Timer_System_Internal_Init{*}: function is in driver file but not found in trusted header file (currently included in OsIf_Timer_System_Internal_GenericTimer.h and OsIf_Timer_System_Internal_Systick.h) !image-2024-10-23-14-03-16-998.png! !image-2024-10-23-14-03-38-927.png! {*}6. Sys_EL1ResumeInterrupts{*}: function is declared in OsIf_Internal.h and and trusted function call here, not found in trusted header file {*}7. Sys_EL1SuspendInterrupts{*}: function is declared in OsIf_Internal.h and and trusted function call here, not found in trusted header file {*}8. Sys_GetCoreID{*}: function is declared in OsIf_Internal.h and and trusted function call here, not found in trusted header file !image-2024-10-23-14-09-54-697.png! !image-2024-10-23-14-13-33-153.png! !image-2024-10-23-14-14-35-777.png! {*}9. k_cycle_get_32{*}: is define in OsIf_Timer_System.c and call buy trusted function in here, not found in trusted header file !image-2024-10-23-14-18-41-536.png! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: Base_TS_COT_TRUST_001 Observed behavior: Trust function is not include in trusted header file Expected behavior: Trust function is include in trusted header file

ID	Subtype	Headline and Description
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Include Trust function in trusted header file
ARTD-152839	New Feature	I2S Flexio clock polarity must be configurable NewWorkDescription: For Flexio i2s, the clock polarity is fixed in driver code. We received a complaint that our driver does not communicate with certain devices with positive clock polarity in i2s and there was nothing to do about it. Requirement source: Customer (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Add option in Flexio configuration to reverse clock polarity
ARTD-153134	Bug	[MCU] Warning and Build failures Detailed description (how to reproduce it): The errors are detected on K3XX and they impact on other platforms !image-2024-10-29-10-25-58-165.png!!image-2024-10-29-16-00-35-649.png! Preconditions: EB,CT Test Case ID (internal TC that caught the defect) optional: MCU_TS_056 Observed behavior: Build failed Expected behavior: Build passes Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-154308	Bug	[RTD_TRAIN_PILOT][LIN] Different number arguments of Flexio_Lin_Ip_IrqHandler function at source and header Detailed description (how to reproduce it): [By tag LIN_144, missing argument uint8 HwInstance at header file Flexio_Lin_Ip.c: !image-2024-11-07-14-41-19-288.png! Flexio_Lin_Ip_Irq.h: !image-2024-11-07-14-41-37-984.png! Beside that, different type of variable {*}Base{*}[] ({*}FLEXIO_Type }}) when assign to Flexio_Lin_Ip_apxBases (const FLEXIO_Type) !image-2024-11-07-15-10-56-451.png!] Preconditions: [Use FlexIO] Test Case ID (internal TC that caught the defect) optional: [Lin_TS_100] Observed behavior: [Build fail due to incompatible function !image-2024-11-07-14-44-47-186.png!] Expected behavior: [N/A] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [Update argument at header file: void Flexio_Lin_Ip_IrqHandler(uint8 ShifterIndex, uint8 ShifterMaskFlag, uint8 ShifterErrorMaskFlag); > void Flexio_Lin_Ip_IrqHandler({*}uint8 HwInstance{*}, uint16 ShifterIndex, uint16 ShifterMaskFlag, uint16 ShifterErrorMaskFlag)]
ARTD-154369	New Feature	[mcu] Add support for CMU variation and latency (code + tresos).

ID	Subtype	Headline and Description
		NewWorkDescription: Add support for CMU variation and latency. Add support in code and tresos. Requirement source: Proposed solution optional:
ARTD-154541	Bug	[PORT] VSMD error comment for rule TpsEcuc_08033 must be updated following the Safety findings Detailed description (how to reproduce it): Regarding the comment from SA inside this ticket: https://jira.sw.nxp.com/browse/ARTD-154402 There was a discussion with Mihai M. and SA, we agreed that the justification/comment now have to be updated: From: Requirement ECUC_PORT_00130 was rejected (on AAI-192) because it states that the PortPinMode attribute in the Port.xdm has multiplicity 1..*, meaning that it should have been implemented as a list, where the user can select more than one mode for a given pin. In RTD's Port driver plugin implementation, the PortPinMode is defined as a enumeration, allowing the user to select only one value for the pin's mode. To: Requirement ECUC_PORT_00130 was rejected (on AAI-192) because it states that the PortPinMode attribute in the Port.xdm has multiplicity 1..*, meaning that it should have been implemented as a list, where the user can select more than one mode for a given pin. In RTD's Port driver plugin implementation, the PortPinMode is defined as a enumeration, allowing the user to select only one value for the pin's mode. Post build variant multiplicity can be set to true, as requested by AUTOSAR, only for lists or optional fields. Since PortPinMode is not implemented as a list, as explained, the post build variant multiplicity has to be set to false. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: The justification/comment is not covering all the cases from error message (missing covering for Post build variant multiplicy) Expected behavior: The justification/comment will have to cover all cases. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update VSMD comments, regenerate VSMD reports for all platforms
ARTD-154681	Bug	[PWM][TRAINs] Generate failed due to missing some attributes in resources Detailed description (how to reproduce it): Missing resource for S32K1XX, S32M244 When generate test on derivative S32K148,S32M244 !image-2024-11-11-13-51-26-320.png|width=550,height=486! Preconditions: Derivative S32K148, S32M244 Release RTD_Trains Test Case ID (internal TC that caught the defect) optional: Pwm_TS_000 Observed behavior: Test fail at gen Expected behavior: Test gen build pass Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-154753	New Feature	[ADC] Implement of the configurable user gain and offset requirements NewWorkDescription: The HDL function Adc_SetUserGainAndOffset and the IPL counterpart functions should be added to K1 for changing the user gain and offset. Requirement source: CPR_RTD_01124.adc CPR_RTD_01123.adc (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional:

ID	Subtype	Headline and Description
		[....]
ARTD-155051	Bug	[S32M2] [2.0.0 P04] CanTrcv Example has errors in S32CT and maybe others(DPGA) In the P04, the examples delivered with the RTD targeting CAN, DPGA have errors in the S32CT configurations, as you can see in the images below, like for the CanTrcv Example. The AE component has some naming issues. And if we solve the naming errors in the Ae component for the CT when we generate the code and build the code, we get errors in 'Aec_Ip_Cfg.h'. The same happens for the DPGA example.
ARTD-155478	Bug	[Mem_INFLS] [S32K148] Hardfault when disable MemDevErrorDetect variable Detailed description (how to reproduce it): When using callout function, we did not config it yet but still call > hardfault !screenshot-1.png|thumbnail! Preconditions: Disable MemDevErrorDetect Disable MemStartFlashAccessNotif Test Case ID (internal TC that caught the defect) optional: Mem_InFls_TS_005 Observed behavior: Hardfault Expected behavior: Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: !screenshot-2.png|thumbnail! move line Ftfc_Mem_InFls_Ip_pConfigPtr = Ftfc_Mem_InFls_Ip_pInitConfig; out of #if (FTFC_MEM_INFLS_IP_DEV_ERROR_DETECT == STD_ON)
ARTD-157177	Bug	[Mem_INFLS] Cannot access Mem_43_INFLS_IPW_EraseResume function Detailed description (how to reproduce it): Step 1: Call Mem_43_INFLS_Erase. Step 2: Call Mem_43_INFLS_Suspend through Mem_43_INFLS_MainFunction. Step 3: Call Mem_43_INFLS_GetJobResult and this function must return MEM_43_INFLS_JOB_PENDING. But it return MEM_43_INFLS_JOB_OK. Step 4: Call Mem_43_INFLS_Resume and this function must return E_OK but it return E_NOT_OK. Because JobCurrentInfo->JobResult is MEM_43_INFLS_JOB_OK not MEM_43_INFLS_JOB_PENDING so cannot access Mem_43_INFLS_IPW_EraseResume. !image-2024-11-18-14-06-45-422.png!!image-2024-11-18-14-07-55-632.png! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Mem_InFls_TC_FCT_0012 in Mem_InFls_TS_001 Observed behavior: Mem_43_INFLS_IPW_Erase will return MEM_43_INFLS_JOB_OK when run with syn mode. !image-2024-11-18-14-10-37-255.png! Expected behavior: Access Mem_43_INFLS_IPW_EraseResume succed. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional
ARTD-157337	Bug	[MCL] Got bus error when using dma_crc and scatter gather feature Detailed description (how to reproduce it): Got bus error when using dma_crc and scatter gather feature: Reproduce: Passing Logic channel to Mcl_SetDmaChannelCrcList Mcl_SetDmaChannelScatterGatherConfig Mcl_GetDmaChannelParam and Mcl_SetDmaChannelScatterGatherList*, Which only config for transfers list not for CRC and Scatter/Gather list. so address of CRC and Scatter/Gather list is NULL_PTR (0x0) At this address access is not allowed. !image-2024-11-19-17-06-26-147.png|thumbnail! The program got hardfault when access to this address. So driver need to protect when CRC and Scatter/Gather list pointer is 0x00. The issue also happen when using all function API with Multi partition enabled and mcl_init() fail. !screenshot-1.png|thumbnail!

ID	Subtype	Headline and Description
		Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Mcl_TC_FCT_7609 Mcl_TC_FCT_7501 Observed behavior: As detailed description Expected behavior: N/A Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Checking configuration address before access. For scatter/gather: Mcl_SetDmaChannelScatterGatherList !image-2024-11-19-17-10-34-120.png|thumbnail! For crc: Mcl_SetDmaChannelCrcList !image-2024-11-19-17-13-32-859.png|thumbnail! Same update for Mcl_SetDmaChannelScatterGatherConfig Mcl_GetDmaChannelParam
ARTD-157833	Bug	[PWM] Can configuare HalfCycle Loading Point is larger than period value at the half cycle in Reload Interrupt Detailed description (how to reproduce it): Now, when i configure use reload interrupt at the half cycle, i can configuare HalfCycle Loading Point is larger than period value. this is wrong. !image-2024-05-23-14-05-28-103.png! !image-2024-05-23-14-05-39-756.png! Preconditions: PwmReloadNotificationSupported is enabled !image-2024-05-23-13-59-13-822.png! HalfCycle Loading Point is enabled and value of HalfCycle Loading Point is larger than period. !image-2024-05-23-14-00-28-055.png! !image-2024-05-23-14-06-01-485.png! Test Case ID (internal TC that caught the defect) optional: Pwm_TC_FCT_0255 Observed behavior: Can configuare HalfCycle Loading Point is larger than period value Expected behavior: Cannot configuare HalfCycle Loading Point is larger than period value Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-157915	Bug	[Mem_INFLS] Cannot generate code on S32DS Detailed description (how to reproduce it): Issue: [CODEGEN] Failed to generate file "Mem_43_INFLS_Cfg.c" Level: Error Type: Tool problem Tool: Peripherals Origin: Peripherals Resource: Sources Information: [CODEGEN] Failed to generate file "Mem_43_INFLS_Cfg.c" Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Ip_Ftfc_TC_FCT_0001 Observed behavior: !image-2024-11-22-10-14-05-566.png! Expected behavior: Generate successfully Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: !screenshot-1.png|thumbnail! missing !] syntax for line 522

ID	Subtype	Headline and Description
ARTD-158153	Bug	[S32K1XX][MEM_EEP] Fix compare code gen report Fail Detailed description (how to reproduce it): Fix compare code gen (the code generated from EB and S32DS) fail for K1XX, M244 Preconditions: Compare code gen report fail, Some node difffirent when gen EB toools end S32DS toools. !image-2024-11-24-13-51-24-374.png! !image-2024-11-24-13-51-52-336.png! !image-2024-11-24-13-52-00-726.png! Test Case ID (internal TC that caught the defect) optional: Mem_Eep_TS_001.mak Mem_Eep_TS_002.mak Mem_Eep_TS_003.mak Observed behavior: Test compare code gen report fail Expected behavior: Report comapre code gen Successfull Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-158163	Bug	[PLATFORM][S32ZSE Train] Build failed MPU M33, MPU R52, MPU M7 Detailed description (how to reproduce it): Platform_Ipw_Cfg.c:115:5: error: expected expression before ',' token 182 ,&Mpu_R52_Config / Mpu R52 Configuration / !image-2024-11-25-08-52-34-796.png|width=1035,height=182! MPU M7 on K1 !image-2024-11-25-15-36-32-107.png|width=916,height=245! !image-2024-11-26-08-57-49-108.png! Preconditions: Build MPU M33, r52; on S32ZSE Build MPU M7: S32K1 PLATFORM_293 Test Case ID (internal TC that caught the defect) optional: TC ID: Platform_TS_COT_001 Platform_Mpu_R52_TS_01 Platform_TS_COT_001: CFG 1 Qdec_TS_104 Observed behravior: Build fail. redundant symbol "," Expected behavior: Update generate code and build pass Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update generate code, remove redundant symbol
ARTD-158278	Bug	[pwm][S32K1-S32M24x] Generated headers cannot be built in case sensitive file systems Detailed description (how to reproduce it): Create S32DS Project from Example, configure the project with any valid configuration and generate the code. When building the project there will be errors since the path is invalid. In conclusion, the case sensitive issues are found in RTD drivers.

ID	Subtype	Headline and Description
		See the slide attached. Preconditions: The host file system being used to build the sources is case-sensitive. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: When you build the project there will be errors since the path is invalid. Expected behavior: Generated sources can be build in case-sensitive systems. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Assume always that sources are building in a case-sensitive file system, therefore use the filename path in the C preprocessor includes directives with the same case as the actual filename path. Steps: # Create new folder "C:\NXPCaseSensitive" (or use the existing ones) now we call it as "working folder". # Run the below command on Windows PowerShell opened in administrator mode:"{*}Enable-WindowsOptionalFeature Online FeatureName Microsoft-Windows-Subsystem-Linux{*}". It takes few seconds and will prompt you to restart your computer. # After restarting, open the command prompt in administrator mode and run: "{*}fsutil.exe file SetCaseSensitiveInfo <YourDestinationFolder> enable{*}" where <YourDestinationFolder> is the folder to enable the case sensitive. # Install S32DS and RTD test environment into the above working folders (from step 1). # Ensure all subfolders inside working folder (from step 1) are renew (re-created, re-layout, re-generated,...) after step 3. Otherwise, we can use this command to enable case sensitive for all created subfolders and no need to renew them: "(Get-ChildItem foldername Recurse Directory).Fullname ForEach-Object {fsutil file setCaseSensitiveInfo "$_" enable}" # Smoke check by building a development test/examples. Hint: # To check the status of case sensitive for the folder: "fsutil.exe file {*}query{*}CaseSensitiveInfo <YourDestinationFolder>". # To disable the case sensitive: "fsutil.exe file {*}Set{*}CaseSensitiveInfo <YourDestinationFolder> {*}disable{*}". # We can enable the case sensitive only the folders that contain the information released to the customers: "../output/eclipse/plugins" folder and "../PlatformSDK_<NPI>" folders
ARTD-158287	Bug	[PWM][S32K1-S32M24x] Missing macros generated when using multiple-variants Detailed description (how to reproduce it): (This ticket is created to review the generation of macros with nodes used in files Cfg.h, CfgDefines.h to prevent from generating macros insufficiently in these files on platforms) Here's an example about generating macros insufficiently encountered on S32DS of S32XX in file Ftm_Pwm_Ip_Cfg.h when users configure multiple variants: VS_0: FTM_1 used VS_1: FTM_0 used Macros that indicate FTM instances used by Pwm are only generated for FTM_1 (VS_0): / Macros that indicate FTM instances used by Pwm / #ifndef FTM_1_USED #define FTM_1_USED #else #error "FTM_1 instance cannot be used by Pwm. Instance locked by another driver!" #endif > This will be an issue when users use VS_1 Preconditions: Configure 2 variants: VS_0, VS_1 VS_0: FTM_1 used VS_1: FTM_0 used Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Macros that indicate FTM instances used by Pwm are only generated for VS_0 Expected behavior: Macros that indicate FTM instances used by Pwm can be generated correctly for all variants. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-158344	New Feature	[mem_eep][S32K1] [IMPLEMENTATION] R21-11 Multicore Concept changes - multipartition vs multicore CR description: Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrIE69]

ID	Subtype	Headline and Description
		Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of OsIf_GetCoreID to OsIfGetUserID. # Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing core_id check to partition_id check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multicore (multipartition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_} reference implementation(s) : ARTD-97002
ARTD-158498	Bug	[pwm] Generate fail IP layer node user mode support when add component Ftm_Pwm in S32K1xx Detailed description (how to reproduce it): Fail generate when add component Ftm_Pwm in ip layer due missing config for node user mode support: !image-2024-11-27-14-13-17-102.png! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Ip_Ftm_TS_001 Observed behavior: Test fail generate Expected behavior: Test pass Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-158503	Bug	[Platform] Mem_Infls Load Access Code fail on IAR Detailed description (how to reproduce it): Build fail on IAR !image-2024-11-27-14-31-35-665.png|thumbnail! [...] Preconditions: Enable Load Access Code Test Case ID (internal TC that caught the defect) optional: Mem_InFls_TC_0001 Observed behavior: Build fail on IAR because uppercase and lowercase do not correspond linker file and memmap !screenshot-1.png|thumbnail! Expected behavior: Build success on IAR Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: update linker file mapping with memmap
ARTD-158538	Bug	[QDEC] Variant names are in all upper when they are not macros Detailed description (how to reproduce it): Variant names are in all upper when they are not macros: !image-2023-12-28-21-43-53-859.png! Preconditions: config in S32DS Test Case ID (internal TC that caught the defect) optional: Qdec_TS_003 Observed behavior: There is toUpperCase() function while get variant name !image-2023-12-28-21-49-27-866.png! Expected behavior: !image-2023-12-28-21-51-16-097.png! Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA

ID	Subtype	Headline and Description
ARTD-158545	Bug	[QDEC] Difference between of code generated by EB and S32CT Detailed description (how to reproduce it): Difference between of code generated by EB and S32CT !image-2024-11-27-16-42-45-817.png! Config file for each test suites Qdec_TS_002 Preconditions: Comparison between code generated by DS and EB Test Case ID (internal TC that caught the defect) optional: Qdec_TS_001 Observed behavior: NA Expected behavior: The code generated by DS and EB needs to be the same Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-158558	Bug	[WDG][RTD_TRAIN_PILOT] Build code fail on S32M244 Detailed description (how to reproduce it): Step 1: clean generate Wdg_TS_007 on GHS compiler Step 2: Build Wdg_TS_007 on GHS compiler Preconditions: NA Test Case ID (internal TC that caught the defect) optional: Wdg_TC_FCT_0008 Observed behavior: Build code fail on S32M244 !image-2024-11-27-17-43-20-713.png|width=1170,height=202! Expected behavior: All tests build code successfully. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-158616	Bug	[Mem_InFls] Fix Bswmd issues Detailed description (how to reproduce it): ERROR: Build of Mem_InFls_TS_GOT_001_cfg1_INFLS has failed. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Mem_InFls_TS_GOT_001_cfg1_INFLS Observed behavior: !image-2024-11-28-09-32-50-613.png! Expected behavior: Test Mem_InFls_TS_GOT_001_cfg1_INFLS pass Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-158795	Bug	[S32K1XX][RTD_TRAIN_PILOT][DIO] Cannot be used on S32DS Detailed description (how to reproduce it): Dio can layout resources on S32DS but no components found in the interface to use Detail: !image-2024-11-29-09-41-34-465.png! Preconditions:

ID	Subtype	Headline and Description
		Dio: PVT_DIO_RTD_TRAIN_25V1_003 PL: S32K1_S32M24 Derivatives: All Test Case ID (internal TC that caught the defect) optional: Dio_TS_ECPD_001 Observed behavior: Not available in S32DS component interface Expected behavior: Can be used in S32DS Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-159280	Bug	[BASE] Osif_Software_Semaphore.c build fail with ArmCortexM0Plus Detailed description (how to reproduce it): Build project used BaseNxp with Resource is Platform.ArmCortexM0Plus : STD_ON and compiler is: gcc-10.2-arm64-eabi Preconditions: Derivative 118,116 (ArmCortexM0Plus enable) Test Case ID (internal TC that caught the defect) optional: Platform_TS_COT_004 Observed behavior: Project build fail with error as following figure: !image-2024-12-03-10-25-55-974.png|width=921,height=279! Expected behavior: BaseNxp can be built pass with ArmCortexM0Plus Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Add barrier for M0 Plus core !image-2024-12-05-14-52-56-944.png|width=861,height=145!
ARTD-161235	Bug	[Mem_InFls] Check controller is idle before calling the function Fls_Flash_InvalidPrefetchBuff_Ram Detailed description (how to reproduce it): As RM mentioned as below: !image-2024-12-03-16-05-57-584.png|thumbnail! But this checking is missing in driver. For example, the sequence for Write operation in Sync Mode is: Wait controller is idle Write to Fls memory (need to wait Fls controller is idle) > missing in driver Verify Write (include the function InvalidPrefetchBuff for Read) Because of this, the customer encountered the issue with memory fault. This issue just occurred suddenly. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: See memory fault. Expected behavior: Not encounter fault. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-159430	Bug	[S32K1_S32M24][dio] Generated headers cannot be built in case sensitive file systems Detailed description (how to reproduce it): Create S32DS Project from Example, configure the project with any valid configuration and generate the code. When building the project there will be errors since the path is invalid. In conclusion, the case sensitive issues are found in RTD drivers. See the slide attached.

ID	Subtype	Headline and Description
		Preconditions: The host file system being used to build the sources is case-sensitive. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: When you build the project there will be errors since the path is invalid. Expected behavior: Generated sources can be build in case-sensitive systems. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Assume always that sources are building in a case-sensitive file system, therefore use the filename path in the C preprocessor includes directives with the same case as the actual filename path. Steps: # Create new folder "C:\NXPCaseSensitive" (or use the existing ones) now we call it as "working folder". # Run the below command on Windows PowerShell opened in administrator mode:"{*}Enable-WindowsOptionalFeature Online FeatureName Microsoft-Windows-Subsystem-Linux{*}". It takes few seconds and will prompt you to restart your computer. # After restarting, open the command prompt in administrator mode and run: "{*}fsutil.exe file SetCaseSensitiveInfo <YourDestinationFolder> enable{*}" where <YourDestinationFolder> is the folder to enable the case sensitive. # Install S32DS and RTD test environment into the above working folders (from step 1). # Ensure all subfolders inside working folder (from step 1) are renew (re-created, re-layout, re-generated,...) after step 3. Otherwise, we can use this command to enable case sensitive for all created subfolders and no need to renew them: "(Get-ChildItem foldername Recurse Directory).Fullname ForEach-Object {fsutil file setCaseSensitiveInfo "$_" enable}" # Smoke check by building a development test/examples. Hint: # To check the status of case sensitive for the folder: "fsutil.exe file {*}query{*}CaseSensitiveInfo <YourDestinationFolder>". # To disable the case sensitive: "fsutil.exe file {*}Set{*}CaseSensitiveInfo <YourDestinationFolder> {*}disable{*}". # We can enable the case sensitive only the folders that contain the information released to the customers: "../output/eclipse/plugins" folder and "../PlatformSDK_<NPI>" folders
ARTD-159603	Bug	[Mem_INFLS] [S32K148] Get block number from address incorrect Detailed description (how to reproduce it): STEP1: config following these variable: MemSyncModeEnable_CFG1 := false MemAcLoadOnJobStart_CFG1 := true MemPhysicalSector_CFG1 := FLS_CODE_ARRAY_0_BLOCK_1_S255 MemStartAddress_CFG1 := 0xff000 STEP2: clean generate and build test with option TEST_PARAMS=@LoadTo=flash STEP3: run test with debugapp=ON and receive the error Preconditions: MemSyncModeEnable_CFG1 := false MemAcLoadOnJobStart_CFG1 := true MemPhysicalSector_CFG1 := FLS_CODE_ARRAY_0_BLOCK_1_S255 MemStartAddress_CFG1 := 0xff000 Test Case ID (internal TC that caught the defect) optional: Mem_InFls_TC_FCT_0026 from test suite Mem_InFls_TS_012_cfg2 Observed behavior: currently all address from CODE_BLOCK is recognize as FLS_CODE_BLOCK_0 because FTFC_MEM_INFLS_IP_P_BLOCK_SIZE is maximum of code block size !image-2024-12-04-13-25-28-073.png! !image-2024-12-04-13-29-50-403.png! Expected behavior: return correctly block number from input address FLS_CODE_BLOCK_0 < 0x80000 0x80000 < FLS_CODE_BLOCK_1 < 0x100000 0x100000 < FLS_CODE_BLOCK_2 < 0x180000 Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: n/a
ARTD-159674	Bug	[Mem_Infls] Erase opration will be missing sector when user config sectorbust > sector size Detailed description (how to reproduce it): Erase opration will be missing sector when user config sectorbust > sector size

ID	Subtype	Headline and Description
		Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: IP_C40_TC_0001 Observed behavior: Erase opration will be missing sector when user config sectorbust > sector size Expected behavior: Erase operation will erase all Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-159840	Bug	[RTD-TRAIN][S32K148][PORT][s32ct]: Build fail on S32DS Detailed description (how to reproduce it): Run compare code gen test Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Qdec_TS_107 Observed behavior: build fail from Port_Cfg.h [link AF : [ARTD][S32K148] RTD_TRAIN_PILOT_25V1|http://zcz01-3158.ea.freescale.net/0/project/ar_go_compare_code_gen_qdec_ci/details] !image-2024-12-05-10-46-00-560.png|width=1047,height=346! Expected behavior: build successfully Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-160191	Bug	[S32ZSE][Mem_INFLS] Gen VSMD fail Detailed description (how to reproduce it): Gen VSMD fail on MemUtestBlockToCheckIntegrity Preconditions: check VSMD for plugin mem_infls Test Case ID (internal TC that caught the defect) optional: Mem_InFls_TC_0001 (dev-test) Observed behavior: An ENUMERATION-PARAM-DEF must define at least one ENUMERATION-LITERAL-DEF. !image-2024-12-09-15-49-03-310.png|thumbnail! !image-2024-12-09-15-50-05-261.png|thumbnail! Expected behavior: Gen VSMD success Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-160676	New Feature	[CRC][CRC32C]Update resource files to remaining platforms for support CRC32C polynomial NewWorkDescription: Update resource files to remaining platforms for support CRC32C polynomial with software mode, table mode, hardware mode (hardware mode only support on some platforms) Requirement source: RM for checking hardware mode support (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Adding/Updating these resources: Crc.Hardware.PolygExt.IsAvailable Crc.Hardware.Support32CastagnoliCalculation Crc.Hardware.Support64HardwareCalculation Crc.AutosarType.List

ID	Subtype	Headline and Description
		Crc.NonAutosar.Crc32CMode.List Crc.Protocols.AllTypes.List Crc.Protocols.OnlyHardwareExtSupport.List
ARTD-161107	Bug	[S32K1][gpt] Driver makefile needs to be updated with the correct version of AUTOSAR SWS specification Detailed description (how to reproduce it): !image-2023-12-29-14-56-48-393.png|width=645,height=253! In <driver>.mak file, the information of AUTOSAR module SWS specification version are incorrect. E.g.: some drivers contains 4.3 references Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Wrong information from: # When importing project to EBT !image-2023-12-29-14-59-58-330.png|width=470,height=393! # plugin.xml from plugins !image-2023-12-29-15-01-33-443.png|width=310,height=306! Expected behavior: The information needs to be updated. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update the information from <driver>.mak based on the supported version of ASR for drivers that are delivering ASR R2x-11: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_7_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 7 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif only for drivers that are delivering ASR 4.4: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_4_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 4 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif
ARTD-161225	Bug	[OCU] Incomplete resource files for K1 Detailed description (how to reproduce it): !image-2024-12-12-12-33-02-625.png! For S32K148, there are missing channels from resource. for example, here are missing FTM_7_CH_4 and FTM_7_CH_6. _**_ Preconditions: Conifigure the driver in Tresos Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Have channels 4, 6 unavailable for FTM_7 instance in tresos config. Expected behavior: Have channels 0-7 available for FTM_7 instance in tresos config, as stated in Reference manual. !image-2024-12-12-12-35-52-365.png! Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Check and update resource files for all derivatives.
ARTD-161363	New Feature	[S32K1_M24X 3.0.0] Update driver for BASE Drivers/IPs list in SOW: [see SOW for details|https://nxp1.sharepoint.com/:w:/r/sites/Zebra/Shared%20Documents/Project%20Management/Projects%20Planning/SW32K1_M24x/SW32K1_S32M24x_RTD_4.4_R21-11_3.0.0/schedule/SW32K1_S32M24x%20RTD%20ASR%20R21-11%203.0.0%20SOW.docx?d=wd624c1f156734716ad62ab951d60d9fc&csf=1&web=1&e=28pQnM]
ARTD-161557	Bug	[S32K1_S32M24x 3.0.0][wdg] Driver makefile needs to be updated with the correct version of AUTOSAR SWS specification Detailed description (how to reproduce it): !image-2023-12-29-14-56-48-393.png|width=645,height=253!

ID	Subtype	Headline and Description
		In <driver>.mak file, the information of AUTOSAR module SWS specification version are incorrect. E.g.: some drivers contains 4.3 references Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Wrong information from: # When importing project to EBT !image-2023-12-29-14-59-58-330.png|width=470,height=393! # plugin.xml from plugins !image-2023-12-29-15-01-33-443.png|width=310,height=306! Expected behavior: The information needs to be updated. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update the information from <driver>.mak based on the supported version of ASR for drivers that are delivering ASR R2x-11: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_7_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 7 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif only for drivers that are delivering ASR 4.4: ifeq ($(AR_RELEASE_REVISION),ASR_REL_4_4_REV_0000) SWS_SPEC_VERSION_MAJOR = 4 SWS_SPEC_VERSION_MINOR = 4 SWS_SPEC_VERSION_PATCH = 0 SWS_SPEC_VERSION_SUFFIX = Rev_0000 endif
ARTD-161563	Bug	[S32K1_S32M24x 3.0.0][wdg] Generated headers cannot be built in case sensitive file systems Detailed description (how to reproduce it): Create S32DS Project from Example, configure the project with any valid configuration and generate the code. When building the project there will be errors since the path is invalid. In conclusion, the case sensitive issues are found in RTD drivers. See the slide attached. Preconditions: The host file system being used to build the sources is case-sensitive. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: When you build the project there will be errors since the path is invalid. Expected behavior: Generated sources can be build in case-sensitive systems. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Assume always that sources are building in a case-sensitive file system, therefore use the filename path in the C preprocessor includes directives with the same case as the actual filename path. Steps: # Create new folder "C:\NXPCaseSensitive" (or use the existing ones) now we call it as "working folder". # Run the below command on Windows PowerShell opened in administrator mode:"{*}Enable-WindowsOptionalFeature Online FeatureName Microsoft-Windows-Subsystem-Linux{*}". It takes few seconds and will prompt you to restart your computer. # After restarting, open the command prompt in administrator mode and run: "{*}fsutil.exe file SetCaseSensitiveInfo <YourDestinationFolder> enable{*}" where <YourDestinationFolder> is the folder to enable the case sensitive. # Install S32DS and RTD test environment into the above working folders (from step 1). # Ensure all subfolders inside working folder (from step 1) are renew (re-created, re-layout, re-generated,...) after step 3. Otherwise, we can use this command to enable case sensitive for all created subfolders and no need to renew them: "(Get-ChildItem foldername Recurse Directory).Fullname ForEach-Object {fsutil file setCaseSensitiveInfo "$_" enable}" # Smoke check by building a development test/examples. Hint: # To check the status of case sensitive for the folder: "fsutil.exe file {*}query{*}CaseSensitiveInfo <YourDestinationFolder>". # To disable the case sensitive: "fsutil.exe file {*}Set{*}CaseSensitiveInfo <YourDestinationFolder> {*}disable{*}".

ID	Subtype	Headline and Description
		# We can enable the case sensitive only the folders that contain the information released to the customers: "../output/eclipse/plugins" folder and "../PlatformSDK_<NPI>" folders
ARTD-161619	Bug	[Mem_INFLS] [S32K148] Configuration of the same hardware instance between HL configuration and IP configuration incorrected Detailed description (how to reproduce it): For checking requirement {*}CPR_RTD_00543{*}, on s32ds interface, select component Mem_43_INFLS for MCAL and Ftfc_Mem_InFls_Ip for Drivers !screenshot-1.png|thumbnail! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: Ip_Ftfc_TC_COT_0001 Observed behavior: no error occurs Expected behavior: [CPR_RTD_00543]: The driver shall prevent configuration of the same hardware instance between HL configuration and standalone IP configuration. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: n/a
ARTD-161649	Bug	Issue in Trgmux_Ip.c with User Mode Execution Detailed description (how to reproduce it): In the *TrigMux* source file, {{{}Trgmux_Ip.c{}}}, the customer encountered an issue that arises when enabling User Mode Execution. !image-2024-12-17-10-16-15-096.png! !image-2024-12-17-10-43-57-409.png! !image-2024-12-17-10-46-19-493.png! Preconditions: No issue Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: as Detailed description Expected behavior: No issue Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: !image-2024-12-17-10-55-45-196.png!
ARTD-161679	New Feature	[mcu] [IMPLEMENTATION] Add support multipartition CR description: Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrlE69] Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of OsIf_GetCoreID to OsIfGetUserID. # Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing core_id check to partition_id check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multicore (multipartition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_} reference implementation(s) : ARTD-97002
ARTD-161682	New Feature	[ae] [IMPLEMENTATION] add support multipartition CR description: Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrlE69] Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of OsIf_GetCoreID to OsIfGetUserID.

ID	Subtype	Headline and Description
		# Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing core_id check to partition_id check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multicore (multipartition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_} reference implementation(s) : ARTD-97002
ARTD-161689	New Feature	[mcl] [IMPLEMENTATION] R21-11 Add support multipartition for K1 platform CR description: Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrlE69] Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of OsIf_GetCoreID to OsIfGetUserID. # Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing core_id check to partition_id check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multicore (multipartition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_} reference implementation(s) : ARTD-97002
ARTD-162312	New Feature	[crc] [IMPLEMENTATION] R21-11 Add support multipartition for K1 platform CR description: Add support for multi-partition support according to [AUTOSAR_EXP_BSWDistributionGuide.pdf R21-11|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared%20Documents/Documents/Standards/AutosarR21-11/BSWGeneral/AUTOSAR_EXP_BSWDistributionGuide.pdf?csf=1&web=1&e=OrlE69] Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of OsIf_GetCoreID to OsIfGetUserID. # Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing core_id check to partition_id check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multicore (multipartition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_} reference implementation(s) : ARTD-97002
ARTD-162315	Bug	[S32K1_M24X 3.0.0][Mem_INFLS] Update Mem_InFls_HwSpecificService to return E_NOT_AVAIL when trying to call hw cancel Detailed description (how to reproduce it): On S32K3, S32E, hardware support "clearing EHV while DONE is low will abort the current program/erase high voltage operation". But on S32K1 after trigger start new job Erase/Write hardware does not support set bit to cancel (abort) operation{color}. Preconditions: Call Mem_43_INFLS_Cancel Function Test Case ID (internal TC that caught the defect) optional: Mem_InFls_TC_0001 (dev-test) Observed behavior: Mem_43_INFLS_Cancel Function implemented follow wait job execute done => incorrect with requirement request by customer Expected behavior: should return E_NOT_Available on HLD when call Mem_43_INFLS_Cancel Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-162318	Bug	[Mem_INFLS] update infix Detailed description (how to reproduce it): infix in driver incorrect Preconditions: infix check Test Case ID (internal TC that caught the defect) optional: Mem_InFls_TC_0001 (dev-test) Observed behavior: infix on driver incorrect !image-2024-12-18-09-51-35-708.png! Expected behavior: 43_INFLS incorrect

ID	Subtype	Headline and Description
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: should update 43_INFLS to "##M4_SRC_INFIX)!!##"
ARTD-162467	Bug	[BASE][platform] DeviceDefinition.h is generated incorrectly for S32K144W and S32K142W Detailed description (how to reproduce it): After adding: [https://bitbucket.sw.nxp.com/projects/ARTD/repos/base/commits/32aa522222d826ce0fe8629fd8bec6e7ec997609#specific%2FS32K1XX%2Fgenerate%2FEBT%2Finclude%2FDeviceDefinition.h] DeviceDefinition.h is generated for k144 and k142 in the case of k144w and k142w Below build is from a k144w example !image-2024-12-18-16-37-10-973.png! __ Also make file has similar error !screenshot-1.png|thumbnail! Preconditions: Generate and build examples for k144w/k142w Test Case ID (internal TC that caught the defect) optional: None Observed behavior: the DeviceDefinition is generated for k144 instead of k144w Expected behavior: Generate correct DeviceDefinition Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: for derivatives that have more characters (k144w and k142w) the logic checks first if the string contains k144 and generates for k144. The verification should be either with equals or it should check first if it contains k144w or k142w then check if it has k144/k142
ARTD-162653	Bug	[mem_exfls] mem init failed because of crc calculation Detailed description (how to reproduce it): Failed at init function Preconditions: crc enabled Test Case ID (internal TC that caught the defect) optional: Mem_TC_FCT_0002 at Mem_ExFls_TS_001 Observed behavior: as description Expected behavior: no error Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-162696	New Feature	[IMPLEMENTATION] [ADC] Add async mode support for Adc_Calibrate Context: The Adc_Calibrate() API will take 10.5ms per ADC instance, and 21ms for two ADC instances, which will add 21ms to the bootup time. Customers want to reduce bootup {*}time{*}. They tried to configure the NRSMPL bit of the SMR register, but we can't guarantee the accuracy of this way, because we have checked with the HW team, and we don't have some test data under this configuration. Idea: From the code of {_}Adc_Sar_Ip_DoCalibration(){_}, the while() loop that polls the CALBISTREG takes about 10ms per instance to wait for the calibration to complete. Customer wants to add async mode support. Asynchronous support means moving this polling loop to another API, then customer code can call the "another API" after 10ms to check if ADC calibration is complete. Avoid the M7 core being busy waiting for this 10ms to reduce the bootup time. Question: Customer wants to know whether API Adc_Calibrate() supports asynchronous mode, will it be feasible?
ARTD-162745	Bug	[i2s] [S32K1_S32M24x]ERR050476: Set FCONT to 0 NewWorkDescription: ERR050476: Set FCONT to 0

ID	Subtype	Headline and Description
		Requirement source: |[S32K148_0N20V, Rev. 15/Jan/2024, 1/2024, S32K148_0N20V.pdf|https://nxp1.sharepoint.com/:b:/r/sites/Zebra/Shared %20Documents/Documents/Platforms/S32K1xx/Documents/Errata/S32K148_0N20V.pdf?csf=1&web=1&e=CBIOql] Proposed solution optional: ERR050476: Set FCONT to 0
ARTD-162950	New Feature	[S32K1] NotUsedPortPin: Consider new mode option to don't touch all of non-configured signals NewWorkDescription: If a pin wasn't configure in PortContainer, it will be set by Port_Init with the configuration in NotUsedPortPin. Right now, NotUsedPortPin has 2 mode options: GPIO and ALT0. Please consider a use-case that user won't touch to this pin, leave this pin with default configuration at Power-Of-Reset Requirement source: None (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: None

ARTD-162953	New Feature	[S32K1][PORT][DIO] Add constraint when adding both Port and Siul2_Port component in S32DS
		
		NewWorkDescription:
		Starting from the inquiry here: https://jira.sw.nxp.com/browse/ARTD-142138 there was a compile error when using both Port and Siul2_Port component but the error was not clear.
		
		We can improve this to have a better message in configuration saying what exactly the user needs to do.
		
		Requirement source:
		FAE on behalf of customer request
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		
		Proposed solution optional:
		In Port_Cfg.h or any generated files in S32DS:
		
		for each (var componentInstanceConfig in ComponentConfigurationsList)
		{
		    if (componentInstanceConfig.getComponent().getId() == "Siul2_Port")
		    {
		        scriptApi.logError("Please remove either Port or Siul2_Port component ")
		    }
		}

ARTD-162981	Bug	S32K1: Port: Some APIs need to be check DET before access data Detailed description (how to reproduce it): In function Port_Ipw_SetPinDirection, has access pUsedPadConfig array use PinIndex before checking out-of-range. It can be happened when user pass value out-of-range, caused crash when run !image-2024-12-23-14-41-31-504.png|width=1025,height=533! Preconditions: PVT_PORT_S32J100_0.4.1_CD_001 Test Case ID (internal TC that caught the defect) optional: None Observed behavior: Access out-of-range array Expected behavior: Should be check range before access data element Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: None
ARTD-163118	Bug	[MCL] Compare code generate failure for S32K148 - follow up ticket Detailed description (how to reproduce it): Compare code generate between eb and s32ct is failed for S32K148 . Please see the detail error logs at the attached file (24_12_2025_S32K1_300_RTD_MCL_CompareCodeGenMergeReport) Preconditions: compare code, s32k148 Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Comapre code generate between eb and s32ct is failed for s32k148 Expected behavior:

ID	Subtype	Headline and Description
		No compare error between eb and ct Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-163236	Bug	[OCU] [S32N55][S32K1XX] Issue related to OcuEmiosBusSelect when check ECPD test Detailed description (how to reproduce it): In S32N55 and S32K1xx , when I test with EPCD test, I get this error. Now, I know that Emios ipv is not supported so I think we need to set OcuEmiosBusSelect which can be set by default to a value in range. !image-2024-12-25-10-40-35-741.png! !image-2024-12-25-14-09-54-742.png! !image-2024-12-25-10-26-35-289.png! Preconditions: Test Case ID (internal TC that caught the defect) optional: Ip_Ftm_TS_ECPD_001 and Ocu_TS_ECPD_001 Observed behavior: OcuEmiosBusSelect is not in range Expected behavior: OcuEmiosBusSelect which can be set by default to a value in range. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:
ARTD-163359	New Feature	[MCU] Support Low Power Oscillator Trim Register (LPOTRIM) NewWorkDescription: LPOTRIM should be supported due to the req !image-2024-12-27-13-56-32-102.png!!image-2024-12-27-13-57-46-075.png! !image-2024-12-27-13-58-04-616.png! Requirement source: CPR_RTD_00112.mcu (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: N/A
ARTD-163370	Bug	[S32K1_M24X 3.0.0][PWM] Phase shift cannot be set for channels in 2 different instance Detailed description (how to reproduce it): When configuring phase shift for channels in 2 different instances, cannot set phase shift of channels in 2 different instances. Use phase shift for FTM1_CH4 and FTM2_CH2 with phaseshift value =0 when init. Steps: Step-1 : Call Pwm_Init Signals of FTM1_CH4(A22) and FTM2_CH2(A20) after init. !image-2024-12-27-14-17-52-025.png! Step-2 : Call Mcl_SelectCommonTimebase(1,6) to synchronize counters between 2 instances FTM1 and FTM2. Signals of FTM1_CH4(A22) and FTM2_CH2(A20) after synchronizing counters between 2 instances. !image-2024-12-27-14-18-50-519.png! Step-3 : Call Pwm_SetDutyPhaseShift for 2 channels (FTM1_CH4 and FTM2_CH2) with PhaseShift = 0, Duty value > 0. Signal of FTM1_CH4(A22) and FTM2_CH2(A20) after setting duty PhaseShift = 0 for both channels. ! image-2024-12-27-14-23-25-071.png! When calling Pwm_SetDutyPhaseShift with PhaseShift = 0, there is a loss of counter synchronization between the 2 instances when calling Ftm_Pwm_Ip_SetSoftwareTriggerCmd => Ftm_Pwm_Ip_SetPhaseShift. Preconditions: User function Pwm_SetDutyPhaseShift for channel in 2 instances Test Case ID (internal TC that caught the defect) optional: Pwm_TS_118 Observed behavior: Phase shift cannot be set for channels in 2 different instance.

ID	Subtype	Headline and Description
		Expected behavior: Phase shift can be set for channels in 2 different instance. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-163508	Bug	[BASE] Single core constraints which don't allow multiple partitions on one core must be removed from Base Detailed description (how to reproduce it): In ticket ARTD-124134, there is a note: # Single core constraints which don't allow multiple partitions on one core must be removed from Base. But when checking with M244 EBT this condition is not removed: !image-2024-12-31-19-54-04-615.png|width=914,height=337! Please also check for other derivatives and platforms Preconditions: OsifUseGetUserID = GET_PARTITION_ID Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Using GET_PARTITION_ID requires turn on Multi core feature Expected behavior: Can use GET_PARTITION_ID without Multi core feature Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-163565	Bug	[UART] Incorrect description in FMEA file Detailed description (how to reproduce it): Uart_Deinit function has no return type following the requirement CDD_UART_00007 but in FMEA file No 3.1 line 21 and No 3.4 line 29 , Uart_Deinit function still require a return E_NOT_OK in case timeout error occured. !image-2025-01-02-15-08-02-814.png! !image-2025-01-02-15-08-54-484.png! Preconditions: Test Case ID (internal TC that caught the defect) optional: Uart_TC_WBT_0009 Observed behavior: Expected behavior: Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:
ARTD-163603	Bug	[S32K1XX][MEM_EEP] Missing clearing RDCOLERR status bit Detailed description (how to reproduce it): When a collision error is detected, the RDCOLERR status bit is automatically set. The driver is missing clearing this bit after reading it and before executing other commands. !image-2025-01-02-20-41-56-464.png! !image-2025-01-02-20-42-03-455.png! !image-2025-01-02-20-42-14-118.png! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA

ID	Subtype	Headline and Description
		Observed behavior: the Driver returns error even though it can actually read and write to FlexRAM Expected behavior: Clear the RDCOLERR status bit after reading and before executing other commands. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-163633	Bug	[S32K1_M24X 3.0.0][MEMACC]: MemAcc_RequestLock function always return fail when run with IAR compiler Detailed description (how to reproduce it): MemAcc_RequestLock function always return fail when run with IAR compiler Preconditions: Compiler is IAR Test Case ID (internal TC that caught the defect) optional: MemAcc_TS_0370 Observed behavior: Expected behavior: MemAcc_RequestLock can return E_OK with correct input parameter Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-163671	New Feature	[IMPLEMENTATION] [ADC] Implement CPR_RTD_01182 for K1 NewWorkDescription: Pending interrupt flags shall be cleared before enabling interrupts. And the driver for K1 has not implemented yet Requirement source: CPR_RTD_01182 Proposed solution optional: Clear pending interrupt flags
ARTD-163709	Bug	[S32K1_M24X 3.0.0] PLATFORM: Compare generated code between generators NewWorkDescription: Report all test in the attachment RTD_PLATFORM_CompareCodeGenMergeReport.xlsx{*}{{*}} IntCtrl_Ip_CfgDefines.h !image-2025-01-06-10-43-57-041.png! Details infor for Mpu_Ip_Cfg.c !image-2025-01-06-10-44-17-379.png! List file .h are different IntCtrl_Ip_CfgDefines.h List file .c are different Mpu_Ip_Cfg.c Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Codes gen in EB are the same in CT
ARTD-163866	Bug	[RTD_TRAIN_PILOT][S32K1_M24X 3.0.0][S32DS][LIN] Miss match lower case variant generation code Detailed description (how to reproduce it): [Using variant named is "Variant", detail is picture below: !image-2025-01-06-16-37-04-905.png! The name of config structure generated by tresos has sub-fix as right part is Lin_43_LPUART_FLEXIO_xConfig_{*}Variant{*}_0 but by DS is Lin_43_LPUART_FLEXIO_xConfig_{*}VARIANT{*}_0 !image-2025-01-06-16-39-31-862.png!]

ID	Subtype	Headline and Description
		Preconditions: [Use class compare code gen] Test Case ID (internal TC that caught the defect) optional: [Lin_TS_016] Observed behavior: [Use class compare code gen] Expected behavior: [N/A] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [N/A]
ARTD-163897	New Feature	[S32K1_M24X 3.0.0][WDG] Update fileStruct in AeWdog, WDOG, EWM IPs NewWorkDescription: Currently, when use IPL layer, user must include "..._Ip_PBcfg.h" > Need to improve this to include "AeWdog_Ip.h", "Wdog_Ip.h", "EWM_Ip.h" only. Requirement source: (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional:
ARTD-164114	Bug	[MCU] Wrong calculation of RefCount in Clock_Ip_SetCmuFcFceRefCntLfrefHfref Detailed description (how to reproduce it): The calculation in RM is mentioned as the below !image-2025-01-09-14-28-05-309.png! It finds the max value first and then return the ceiling value of max while our driver is ceiling each value first and then find the max of them !image-2025-01-09-14-29-37-019.png! Which lead to some wrong values in some cases First ceiling is defined as the below !image-2025-01-09-14-31-52-998.png! => ceiling of integer = the original value for exp !image-2025-01-09-14-32-52-782.png! => Issues can occur if Cmp1 and Cmp2 are integer number => currently plus 1 in the formulas for Cmp1 and Cmp2 is wrong {*}UPDATE{*} RM version: S32K1xx Series Reference Manual, Rev. 14, 09/2021 The values which are underlined in red are replaced by different values in the driver which is wrong !image-2025-01-23-16-26-42-435.png! !image-2025-01-23-16-26-08-464.png! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Mcu_TC_WBT_0049 Observed behavior: Wrong calculation of RefCount in Clock_Ip_SetCmuFcFceRefCntLfrefHfref Expected behavior: Correct calculation of RefCount in Clock_Ip_SetCmuFcFceRefCntLfrefHfref Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-164141	New Feature	[S32K1_M24X 3.0.0][QDEC]: Reduce some APIs aren't supported on K1XX platform NewWorkDescription: Some APIs are not supported on K1 platform, and It is not used so it affects the display of the code exposure. !image-2025-01-09-17-09-06-724.png! Requirement source: CCOV documents, code exposure. (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)

ID	Subtype	Headline and Description
		Proposed solution optional: > Use the M4 variable to hide these unused APIs.
ARTD-164150	Bug	[S32K1][port] Generated headers cannot be built in case sensitive file systems Detailed description (how to reproduce it): Create S32DS Project from Example, configure the project with any valid configuration and generate the code. When building the project there will be errors since the path is invalid. In conclusion, the case sensitive issues are found in RTD drivers. See the slide attached. Preconditions: The host file system being used to build the sources is case-sensitive. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: When you build the project there will be errors since the path is invalid. Expected behavior: Generated sources can be build in case-sensitive systems. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Assume always that sources are building in a case-sensitive file system, therefore use the filename path in the C preprocessor includes directives with the same case as the actual filename path. Steps: # Create new folder "C:\NXPCaseSensitive" (or use the existing ones) now we call it as "working folder". # Run the below command on Windows PowerShell opened in administrator mode:"{*}Enable-WindowsOptionalFeature Online FeatureName Microsoft-Windows-Subsystem-Linux{*}". It takes few seconds and will prompt you to restart your computer. # After restarting, open the command prompt in administrator mode and run: "{*}fsutil.exe file SetCaseSensitiveInfo <YourDestinationFolder> enable{*}" where <YourDestinationFolder> is the folder to enable the case sensitive. # Install S32DS and RTD test environment into the above working folders (from step 1). # Ensure all subfolders inside working folder (from step 1) are renew (re-created, re-layout, re-generated,...) after step 3. Otherwise, we can use this command to enable case sensitive for all created subfolders and no need to renew them: "(Get-ChildItem foldername Recurse Directory).Fullname ForEach-Object {fsutil file setCaseSensitiveInfo "$_" enable}" # Smoke check by building a development test/examples. Hint: # To check the status of case sensitive for the folder: "fsutil.exe file {*}query{*}CaseSensitiveInfo <YourDestinationFolder>". # To disable the case sensitive: "fsutil.exe file {*}Set{*}CaseSensitiveInfo <YourDestinationFolder> {*}disable{*}". # We can enable the case sensitive only the folders that contain the information released to the customers: "../output/eclipse/plugins" folder and "../PlatformSDK_<NPI>" folders
ARTD-164260	Bug	[Resource] Update modules with resources Detailed description (how to reproduce it): Resource.cfg shall be updated according to SOW(Eep driver (which is for ASR 4.4) appears in the list of RES_MODULES) !image-2025-01-10-11-53-52-291.png! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: ASR 4.4 module present Expected behavior: Resource.cfg shall updated according to SOW Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Check SOW and update Resource.cfg.
ARTD-164287	New Feature	[IMPLEMENTATION] Add support for tri-stated outputs by configuring MISCTRL0 register CR description:_* [The PWM module does not allow changing SIM_MISCTRL0[FTMx_OBE_CTRL],]_ Reason for this change: [The PWM cannot be used with tristated outputs]_

ID	Subtype	Headline and Description
		Benefit: [Customer needs the output to be tristated, not retained]_ Onetime CR/Strategic CR: [text placeholder]_ Use-case: [text placeholder] HW documentation reference (as applies): [S32K1xx Series Reference Manual, Rev. 14, Section 11.1.1.2 Miscellaneous control register 0 (MISCTRL0)]_ HW/Application Engineer contact (as applies): [text placeholder]_ Note: relevant documents to be attached to the ticket.
ARTD-164397	Bug	[OCU][S32K1_M24X 3.0.0] Duplicate macro define name between 2 modules Ocu and Qdec module Detailed description (how to reproduce it): When I configure Ocu and Qdec in 1 configuration, when building test, I get this error !image-2025-01-13-17-13-20-952.png! I check and see that we duplicate macro define name between 2 modules Ocu and Qdec. Please help me check !image-2025-01-13-17-13-41-376.png! !image-2025-01-13-17-15-56-327.png! !image-2025-01-13-17-17-02-853.png! Preconditions: configure Ocu and Qdec in 1 configuration Test Case ID (internal TC that caught the defect) optional: Ocu_TS_001 Observed behavior: Duplicate macro define name between 2 modules Ocu and Qdec module Expected behavior: Macro define name between 2 modules Ocu and Qdec are different Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-164511	New Feature	[rm] [IMPLEMENTATION] R21-11 Multicore Concept changes - CPR_RTD_00420 CR description: Related to multipartiton, RM driver is type 1 (can only be assigned to one core or a single partition). The RM driver needs to know the id of the partition to check in its APIs if it was addressed from the proper partition. If APIs need to be available from other cores/partitions, the verification must be done with the maximum IDs of the cores/partitions (CPR_RTD_00420) Implementation requested on driver : # Drivers must have the configuration structures implemented based on the number of partitions and not the number of cores. # Drivers must replace the existing call of OsIf_GetCoreID to OsIfGetUserID. # Drivers must have in the configuration structures the referenced Ecuc Partition. # Drivers must replace existing core_id check to partition_id check. # Example(s) must be updated and this shall be seen as an input for future definition on Platform Base Bundle configuration. # Single core constraints which don't allow enabling the multicore (multipartition) on one core must be removed from driver. # Update label and description of "{_}Multicore Support Enable"{_} node to "{_}MultiPartition Support Enable"{_} reference implementation(s) : ARTD-97002 Requirement CPR_RTD_00420 to be interpreted as: !image-2025-01-14-08-41-53-775.png!

ID	Subtype	Headline and Description
		Definition of done: Feature is confirmed by testing Generated all code violation reports and solved all issues
ARTD-164516	Bug	[S32K1XX_M24X][I2c] Functions in trusted header file but not called Detailed description (how to reproduce it): [Functions in trusted header file but not called Flexio_I2c_Ip_MasterDmaTransferErrorHandler Flexio_I2c_Ip_MasterEndDmaTransfer Lpi2c_Ip_MasterCompleteDMATransfer Lpi2c_Ip_MasterDMATransferErrorHandler] Preconditions: [None] Test Case ID (internal TC that caught the defect) optional: [TC_002: report by tool] Observed behavior: [None] Expected behavior: [N/A] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [None]
ARTD-164781	New Feature	[WDG][S32K1_M24X 3.0.0] Add fault injection point for IP driver to cover FMEA NewWorkDescription: Add fault injection point for IP driver to cover FMEA [^AeWdog_Ip.c] Requirement source: Add fault injection point for all IPs on S32K1XX (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: [....]
ARTD-164900	Bug	[PLATFORM][S32K1_M24X 3.0.0] Bug of driver in code interface Detailed description (how to reproduce it): List function in "ReqExport.txt" file but does not include in "*.h" files of driver code: boolean Platform_Mpu_GetErrorDetails(Platform_Mpu_ErrorDetailsType pErrorDetails) 1/ in driver code is: Std_ReturnType in req export is: boolean Preconditions: Test Case ID (internal TC that caught the defect) optional: Run custom_checkinterface Observed behavior: Have some functions is different in driver code and reqs export Expected behavior: List function in "ReqExport.txt" file but does not include in "*.h" files of driver code is EMPTY Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update function in the ReqExport.
ARTD-164964	New Feature	[rte] Add exclusive areas SchM_Enter_Gpt_GPT_EXCLUSIVE_AREA_19 for GPT NewWorkDescription: The GPT driver needs to have exclusive areas SchM_Enter_Gpt_GPT_EXCLUSIVE_AREA_19 to be used by the source code. SchM_Gpt.c/h must be added to RTE. Requirement source: NA

ID	Subtype	Headline and Description
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: NA
ARTD-165212	Bug	[PLATFORM] Fix compiler warning on file Platform_Ipw.c Detailed description (how to reproduce it): No1: Compiler warning on gcc in file: Platform_Ipw.c !image-2025-01-20-17-33-58-421.png! Preconditions: No1: gcc compiler Test Case ID (internal TC that caught the defect) optional: Platform_TS_COT_004 Observed behavior: unused parameter 'pConfig' [-Wunused-parameter] Expected behavior: Not have warning Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Remove unused parameter 'pConfig' [-Wunused-parameter]
ARTD-165399	Bug	[FEE R21.11] EB,DS will report an error when the node does not exist Detailed description (how to reproduce it): EB,DS will report an error when the " FeeBufferAlignmentValue" node does not exist or may not be referenced Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: FEE_TC_001 Observed behavior: EB,DS does not report an error when the " FeeBufferAlignmentValue" node does not exist or may not be referenced Expected behavior: EB,DS will report an error when the " FeeBufferAlignmentValue" node does not exist or may not be referencedEB, DS will report an error when the " FeeBufferAlignmentValue" node does not exist or may not be referenced Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-165918	Bug	[icu]The interrupt flag and the interrupt enable bit are cleared in the Lptmr_Icu_Ip_Init function when the LPTMR_ICU_STANDBY_WAKEUP_SUPPORT feature is enabled. Detailed description (how to reproduce it): The STANDBY_WAKEUP_SUPPORT feature is a non-autosar feature, it is necessary to keep the interrupt flag and interrupt enable bit value unchanged in the Icu_init function The Icu driver shall provide an optional configuration parameter to support wakeup IP operation across STANDBY. Per default this optional functionality and configuration parameters shall be disabled,. If the configuration parameter is enabled, the following shall be respected: The driver shall NOT CLEAR the interrupt flag or the interrupt enable bit, after a wakeup event. The driver shall make sure it will correctly service the wakeup event, if this event occurred before init. This is done to support the hardware functionality of IP's that may have flags set after the occurred wakeup event that leads to a memory clear." Notes: This might clash with Autosar requirements: [SWS_Icu_00061] The function Icu_Init shall disable all notifications. [SWS_Icu_00121] The function Icu_Init shall disable the wakeup-capability of all channels. [SWS_Icu_00054] The function Icu_Init shall only set the resources that are configured in the configuration file (including clearing of pending interrupt flags

ID	Subtype	Headline and Description
		Req ID: CPR_RTD_00403.icu however in the Lptmr_init function, when interrupt flag=1 and interrupt enable flag=1, this is not being done correctly interrupt flag and interrupt enable flag need to be kept as is: !image-2025-01-21-15-23-46-914.png! Preconditions: STD_ON ==LPTMR_ICU_STANDBY_WAKEUP_SUPPORT Test Case ID (internal TC that caught the defect) optional: Icu_TC_WBT_0040(Icu_TS_034) Observed behavior: The interrupt flag and the interrupt enable bit are cleared in the Lptmr_Icu_Ip_Init function when the LPTMR_ICU_STANDBY_WAKEUP_SUPPORT feature is enabled. Expected behavior: The interrupt flag and interrupt enable bit must be kept set in the Lptmr_Icu_Ip_Init function when the LPTMR_ICU_STANDBY_WAKEUP_SUPPORT feature is enabled. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: need to correct the condition so that the interrupt flag and interrupt enable flag remain the same in non-autosar stand by mode: !image-2025-01-21-15-27-33-356.png!
ARTD-165961	Bug	[S32K1_M24X 3.0.0][Port][develop] pau32Port_PinToPartitionMap generator different between EB and CT Detailed description (how to reproduce it): The pau32Port_PinToPartitionMap index value in array generator different between EB and CT. On EB: the index in pau32Port_PinToPartitionMap array shall loop and checking with " PortPinId" On CT: The index in pau32Port_PinToPartitionMap array shall loop and checking with " PortPin" Preconditions: PVT_PORT_S32XX_RTM_4.0.2_010 Test Case ID (internal TC that caught the defect) optional: Dio_TS_200 Observed behavior: on EB: Ex with portPin_L7 configuration with "{*}PortPin Id{*}" equal "{*}1{*}". The pau32Port_PinToPartitionMap value of portPin_L7 will be at position 0 in the pau32Port_PinToPartitionMap array. !image-2023-06-13-17-15-58-319.png! The same config portPin_L7 on CT the The pau32Port_PinToPartitionMap value of portPin_L7 will be at position 7 in the pau32Port_PinToPartitionMap array. !image-2023-06-13-17-17-53-448.png! Expected behavior: On CT: The pau32Port_PinToPartitionMap index in array mapping to "PortPin Id" value Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A

ARTD-166157	Bug	[CAN] MainFunction_Read Buffer Allocation issue
		Detailed description (how to reproduce it):
		The customer use Driver in Polling mode with an Os that allocate the Stack dinamically and have a 5 Ms Task where he calls the Can_43_FLEXCAN_MainFunction_Read which > Can_43_FLEXCAN_Ipw_MainFunction_Read which allocates a buffer and pass the to the > FlexCAN_Ip_Receive and set the MB Status in RX if the receive will not happen at second call will call again this sequence but the message buffer could be alocated to other address and the FlexCAN_Ip_Receive will return status busy but will not set new address of the message buffer and keeps the older value from previous call.
		Preconditions:
		This describe the extracted flow of the sequences :
		Can_43_FLEXCAN_MainFunction_Read ( );
		{
		Can_43_FLEXCAN_Ipw_MainFunction_Read ( );
		{
		Flexcan_Ip_MsgBuffType ReceivedDataBuffer;
		//-> This changes every time we call the function,
		// because of the OS, the stack is dynamically allocated

ID	Subtype	Headline and Description

		FlexCAN_Ip_Receive (ReceivedDataBuffer);
		{
		FlexCAN_StartRxMessageBufferData (ReceivedDataBuffer);
		{
		if (state->mbs[mb_idx].state != FLEXCAN_MB_IDLE)
		{
		result = FLEXCAN_STATUS_BUSY;
		// If I don't receive, I will keep entering here
		// The ReceivedDataBuffer may change its location every time we call the function
		}
		else
		{
		//We will enter here only if state == IDLE
		state->mbs[mb_idx].state = FLEXCAN_MB_RX_BUSY; // In the 1st call
		state->mbs[mb_idx].pMBmessage = ReceivedDataBuffer;
		}
		}
		}
		Test Case ID (internal TC that caught the defect) optional:
		N?A
		
		Observed behavior:
		This is happening in the OS context where the Tasks can have different allocation on the stack, bettwen multiple calls.
		
		Expected behavior:
		Correct behavior to allocate each time call the correct buffer address.
		
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)
		
		Proposed solution optional:
		[....]

ARTD-166261	Bug	[Pwm] All trusted functions should not defined as static or inline functions Detailed description (how to reproduce it): All trusted functions {*}should NOT defined as static or inline functions{*}, so OS Application can be able to call them from outside RTD driver. But Ftm_Pwm_Ip_SetSimOutputBufferEn defined as static or inline functions Ensure these functions are defined without "static" or "inline" keywords, so OS Application can use "extern" keyword to declare and call them outside RTD drivers. !image-2025-01-23-17-25-59-817.png|width=728,height=322! Preconditions: use USING_OS_AUTOSAROS and Ftm_Pwm_Ip_SetSimOutputBufferEn function. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Ftm_Pwm_Ip_SetSimOutputBufferEn defined as static or inline functions. Expected behavior: External function Ftm_Pwm_Ip_SetSimOutputBufferEn for user to use. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-167239	Bug	[PLATFORM] System_Ip_TrustedFunctions.h is not included to plugin Detailed description (how to reproduce it): Missing System_Ip_TrustedFunctions.h in S32K1 and S32K5 plugin Preconditions: Using System_Ip_TrustedFunctions.h to extern privileged functions Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Missing System_Ip_TrustedFunctions.h in S32K1 and S32K5 plugin Expected behavior: System_Ip_TrustedFunctions.h must be added into plugin Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: update Platform.mak to add System_Ip_TrustedFunctions.h to plugin
ARTD-167830	New Feature	[S32K1_M24X 3.0.0][WDG] Fix Misra rule 8.5 NewWorkDescription:

ID	Subtype	Headline and Description
		!image-2025-02-13-10-18-19-616.png! !image-2025-02-13-10-18-55-063.png!!image-2025-02-13-10-24-37-471.png!!image-2025-02-13-10-23-23-782.png! Requirement source: (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: 12351214: Create Wdg_Ipw_Irq.h and move "ISR(Wdg_Ipw_Isr);" from Wdg_Ipw_Irq.c to Wdg_Ipw_Irq.h Remove extern ISR(Wdg_Ipw_Isr); in Wdg_TestSetup.c & Wdg_TestSetup.h Add #include "Wdg_Ipw_Irq.h" in Wdg_Ipw_Cfg_Defines.h}} 12351191: Move void Wdog_Ip_IrqHandler(uint8 Instance); from Wdog_Ip.c to Wdog_Ip.h Add #include "Wdog_Ip.h"}} & remove extern void Wdog_Ip_IrqHandler(uint8 Instance);}} in Wdg_Ipw_Irq.c Add #include "Wdog_Ip.h"}} & remove extern void Wdog_Ip_IrqHandler(uint8 Instance);}} in Wdog_Ip_Irq.c 15650752: Remove extern void Wdg_CallBackNotification0(void); in Wdg_TestSetup.h}}
ARTD-168364	Bug	[BLN_RTD_4.7_S32K1XX_S32M24x_3.0.0][S32K1xx] Multiple problem have occurred when adding LinTrcv_43_AE Detailed description (how to reproduce it): 1. Create a project for S32K1xx attached RTD 2. Open Peripheral tool 3. Click 'Manage SDK component' option on toolbar 4. Add LinTrcv_43_AE Test Case ID (internal TC that caught the defect) optional: TL30_00380_01190 Observed behavior: Multiple problems have occurred Expected behavior: If S32K1xx isn't support LinTrcv_43_AE, LinTrcv_43_AE should be removed or disabled
ARTD-168768	Bug	[MCU] Build failed due to cycling include when build with OsIfAutosarOsType Detailed description (how to reproduce it): M244: Build test Int_TS_COT_001 with OsIfOperatingSystemType = OsIfAutosarOsType Error: !image-2025-02-18-17-45-29-353.png|width=1106,height=292! The root cause is when compiling with OsIfOperatingSystemType = OsIfAutosarOsType; Power_Ip_AEC_Types.h > include > Mcal.h > include > Os.h > include > Os_TrustedFunctions.h > Power_Ip_Types.h > Power_Ip_AEC_Types.h It is cycling include > that causes this error Preconditions: S32M244 Test Case ID (internal TC that caught the defect) optional: Int_TS_COT_001 Observed behavior: Compile error Expected behavior: Compile pass Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Remove Mcal.h from Power_Ip_AEC_Types.h
ARTD-172615	Bug	[Mem_Eep][S32K1 3.0.0] Update the description of Mem_Read() in UM Detailed description (how to reproduce it): Currently, the description of function Mem_Read() is incorrectly. The requirements relates to ECC can not implement as ECC correction can not detected. !image-2025-03-04-16-43-11-761.png! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: the description of function Mem_Read() is incorrectly. Expected behavior: the description of function Mem_Read() is correctly.

ID	Subtype	Headline and Description
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: update UM
ARTD-172776	Bug	[S32K1_M24X 3.0.0 FSRA Finding] MCL MISRA - subsequent code fixes NewWorkDescription: [ARTD-172493 code fixes: this is created to fully fix in code the FS findings in ARTD-172493] the function: hwv2AccInlineDmaCh_GetErrorStatus(uint32 LocHwCh, const Dma_Ip_Hwv2InstRegType ptCh, uint32 const pValue) has inside an if causing MISRA issues 14.4 Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: [the function: hwv2AccInlineDmaCh_GetErrorStatus(uint32 LocHwCh, const Dma_Ip_Hwv2InstRegType ptCh, uint32 const pValue) has inside an if causing MISRA issues " if(ptCh->reg_ERR >> LocHwCh)}} " which will be changed to: " if((ptCh->regErr >>LowCh) != FALSE)}} "]
ARTD-173248	Bug	[S32K1-S32M27x][MemAcc] Update code fix HIS_CALL, HIS_LEVEL Detailed description (how to reproduce it): HIS_LEVEL, HIS_CALL violations need to fixed Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: there are some HIS_CALL, HIS_LEVEL have not fix Expected behavior: They need to fixed Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update driver code

4.2 Change List for 2.0.0 P01

ID	Subtype	Headline and Description
ARTD-26302	New Feature	[eep] Add support for multiple configurations in S32CT NewWorkDescription: Previously, we limited the number of functional groups (multiple configurations) to 1. Now we want to add support for multiple configurations (in the form of post-build variants and classic functional groups) so that we can also replicate in S32CT the module configurations from EB Tresos in which there are two or more post-build variants. Requirement source: SRS_BSW_00405 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Read the newly introduced section ("13. Multiple Configurations") Make sure you have added the SYNC_VALUE and SYNC_SIZE options to each component setting that requires it Apply the changes shown in the source code references. The main points are: {noformat} [!IF "var:defined('postBuildVariant')"!]{noformat} translates to {noformat} if (useFgInfix()){noformat}

ID	Subtype	Headline and Description
		The file names and configuration structures should not contain the variant infix when there's no selected post-build variant (actually, most of the changes you'll see in the references are done just to support this use-case) Sanity Check: Check that your driver can be successfully built in all three scenarios: ## ComponentGenerationMethod = EcucPostBuildVariants and VARIANT_NO = 0 ## ComponentGenerationMethod = EcucPostBuildVariants and VARIANT_NO > 1 (VARIANT_NO == 1 is the base case from which you started) ## ComponentGenerationMethod = FunctionalGroups (with at least 2 FGs configured in the project) Extensive Testing: By default, EB Tresos cannot generate variant aware EPC files without throwing a null pointer exception. To get around this limitation, the dreisoft.tresos.autosar2 plugin must be patched. To do this, download one of the two attached archives, depending on your EBT version, and extract it in ebt_root_dir/plugins. # Go to your driver's plugin, open {_}plugin.xml{_}, search for {_}id="EPCGenerator"{_}, and set the value of parameter "allVariants" to {_}"true"{_}. This will enable variant aware EPC file generation for your module. # Create a configuration with at least two post-build variants. The project will need EcuC for post-build variants to be enabled. The configuration should ideally use different values for nodes with postBuildVariantValue=TRUE and different list elements for nodes with postBuildVariantMultiplicity=TRUE. This will ensure that errors in variant handling can be easily caught later on while doing the diffs. # Generate the EPC files by right clicking on your project's configuration and select "Generate Project". If the 1st step has been carried out correctly, there will be a "PostBuildSelectableVariants.arxml" file in project_root_dir/output/output and your module's EPC file, lying inside the same directory, will contain lots of "VARIATION-POINT" elements. # Open S32DS and select File > Import > S32 Configuration Tools > Import ECU Configuration. Browse to the location where your EPC files are located within the EBT project and select at least your module's EPC file, EcuC's EPC file and PostBuildSelectableVariants.arxml. Check the "Import the configuration as a new one" radio button and select the processor / package / sdk version accordingly. # Go to the "Global settings" panel and make sure that ComponentGenerationMethod=EcucPostBuildVariants in the System component. # Select "Update Code" to generate your S32DS project files. Perform a comparison/diff of the generated files between S32DS and EBT. These files should be pretty much identical (sans whitespace or formatting) # Go to the "Global settings" panel again to generate the ECVD files (EcvdGenerationMethod=INDIVIDUAL, browse a location for EcvdOutputPath and click on the Generate Configuration button). Compare/diff your module's EPC file with your module's ECVD file. These files should be pretty much identical (sans whitespace or formatting) # The above steps tested the EBT->S32DS compatibility. To also test the S32DS->EBT compatibility, do the above steps in reverse. That is, start from an S32DS project with post-build variants, generate the ECVD files, import them into EBT, compare again the generated files and the configuration files.
ARTD-58596	New Feature	[EEP] [S32K1XX] Ftfc_Eep_Ip_Write propose improvement to prevent potential bug NewWorkDescription: Customer used Ftfc_Eep_Ip_Write() to write data from RAM to FlexRam 1.pu8SrcAddress input parameter is 0x2000015f (unaligned with 4byte) 2.PageSize is 4 byte 3.FTFC_EEP_IP_ALIGNED_RAM_ACCESS option is STD_OFF When debug until function Ftfc_Eep_Ip_WriteIntoFlexram(): FTFC_EEP_IP_ALIGNED_RAM_ACCESS is off so RTD's EEPROM driver does not care for unaligned addresses of pu8SrcAddress and causes a HardFault on write: !image-2023-02-28-16-33-14-384.png! HardFault occurs when the LDR instruction accesses an address that is not aligned with 4 bytes of R1 (0x2000015F) As the S32K118 ARM Coretex-M0 Reference Manual states that access to unaligned addresses causes a HardFault !image-2023-02-28-16-34-23-197.png! Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Ftfc_Eep_Ip_Write should return Fail if parameter is invalid for pu8SrcAddress, or we should document this point about parameter input for this function must be aligned with 4byte (or 2byte) address if PageSize is 4Byte (or 2byte) in the User Manual For further information, please check the propose workaround from customer in attachment[^Request check the workaround.pdf]
ARTD-75388	Bug	[Fls] Wrong Interface QSPI in DS Detailed description (how to reproduce it): Wrong Interface QSPI in DS (IPL, HLD), different from EB. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: TS_001 Observed behavior: !image-2023-07-06-17-06-16-864.png! Expected behavior: Correct interface like EB: !image-2023-07-06-17-10-24-873.png!

ID	Subtype	Headline and Description
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-79740	Bug	[Fls]Example K148 run fail because init mcu for K148 Detailed description (how to reproduce it): Init Mcu fail because configuretion mcu fail Preconditions: !image-2023-07-26-18-16-15-632.png! Test Case ID (internal TC that caught the defect) optional: Observed behavior: Expected behavior: Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-23417	Bug	[fee] [S32CT] Incorrect enable attributes for ConfigTimeSupport fields Detailed description (how to reproduce it): Manually check the ConfigTimeSupport fields of the S32CT component / template Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: POST_BUILD_VARIANT_USED has enable="false" IMPLEMENTATION_CONFIG_VARIANT has enable="false" Expected behavior: POST_BUILD_VARIANT_USED has enable="false" if and only if postBuildVariantSupport="false" in the config set's options_expr. IMPLEMENTATION_CONFIG_VARIANT has enable="true" or no enable attribute (which is equivalent to enable="true"). Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: POST_BUILD_VARIANT_USED has enable="false" if and only if postBuildVariantSupport="false" in the config set's options_expr. IMPLEMENTATION_CONFIG_VARIANT has enable="true" or no enable attribute (which is equivalent to enable="true").
ARTD-25882	Bug	[FLS] Write data error when the data size will go over the sector boundary Detailed description (how to reproduce it): when writing data on a sector in asynchronous mode and if this data length causes spanning, it will causes errors. After code analyze, this is a bug which the low level driver process a wrong sector index when the write data need to cross sectors. And this issue exist in both K1 and K3 RTD. Preconditions: write data in async mode, and the write data size will across the sector edge. Test Case ID (internal TC that caught the defect) optional: None Observed behavior: write data error Expected behavior: Write data normally Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: add the index variable when the case happened in the low level driver, can refer to the MCAL code.
ARTD-25946	Bug	[FEE] Warnings of zero-initialized variables with complier IAR Detailed description (how to reproduce it): When building Fee driver with IAR compiler, there are some warnings appear, because of zero-initialized variables still have explicit zero-initializers Root cause: there was an incorrect fixing for{color}{color:#172b4d} compiler warning, it did not follow the rule of zero-initialized variables in "CLEARED" memory sections [https://bitbucket.sw.nxp.com/projects/ARTD/repos/fee/pull-requests/117/diff#generic/src/Fee.c]

ID	Subtype	Headline and Description
		Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: There are compiler warning on IAR Expected behavior: There are no warnings Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Remove all explicit zero-initializers in the "CLEARED" memory sections. Following the guidline in the ticket: https://jira.sw.nxp.com/browse/ARTD-12239
ARTD-26477	Bug	[FLS] [QSPI] SEQID is treated as a LUT index Detailed description (how to reproduce it): In the functions mentioned below the lut parameter is actually a sequence ID, and not a LUT index: Qspi_Ip_IpWrite}} Qspi_Ip_IpRead}} Qspi_Ip_IpCommand}} Besides the confusion, the functional defect is that the parameter is checked against the maximum number of LUT registers: DEV_ASSERT_QSPI(lut < QuadSPI_LUT_COUNT);}} Preconditions: none Test Case ID (internal TC that caught the defect) optional: no TC Observed behavior: the check is too permissive Expected behavior: we should differentiate between the id of the sequence and the index of the LUT register Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Besides the functional change proposed below, a better naming of the parameters would be suited: DEV_ASSERT_QSPI(SeqId < QuadSPI_LUT_COUNT / FEATURE_QSPI_LUT_SEQUENCE_SIZE);}}
ARTD-26603	Bug	[Fls][S32ZE_EAR_080] Remove include "Os.h" in Fls.h for avoiding build failed re-declared function Detailed description (how to reproduce it): Since Os.h has been updated as https://bitbucket.sw.nxp.com/projects/ARTD/repos/os/pull-requests/35/diff#generic/include/Os.h: Build failed occure (re-declare ResumeAllInterrupts() and SuspendAllInterrupts()) on GHS if USER_MODE enable . Preconditions: Osif_Internal.h defines the above APIs already Test Case ID (internal TC that caught the defect) optional: All test build on GHS with USER_MODE enable Observed behavior: Build failed on GHS. Expected behavior: Fls.h shall be updated Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA

ARTD-41522	New Feature	[FLS][QSPI] Add support to escape from being stuck in BUSY state
		
		The customer's code will be stuck in the BUSY state after several read/write operations, until timeout, and loopback again and then still BUSY and timeout.
		do
		{
		/ Add Fault Injection point for FR_ILLINE flag /
		MCAL_FAULT_INJECTION_POINT(FLS_FIP_FR_ERROR_ABORTSUSPEND);
		
		status = Qspi_Ip_ControllerGetStatus(controllerInstance);
		Fls_Qspi_u32ElapsedTicks = OsIf_GetElapsed(and Fls_Qspi_u32CurrentTicks, (OsIf_CounterType)QSPI_IP_TIMEOUT_TYPE);
		if ((STATUS_QSPI_IP_BUSY == status) and and (Fls_Qspi_u32ElapsedTicks >= Fls_Qspi_u32TimeoutTicks))
		{
		(void) Det_ReportRuntimeError((uint16)FLS_MODULE_ID, FLS_INSTANCE_ID, FLS_MAINFUNCTION_ID, FLS_E_TIMEOUT);

ID	Subtype	Headline and Description

		status = STATUS_QSPI_IP_TIMEOUT;
		}
		}
		while (STATUS_QSPI_IP_BUSY == status);
		
		Customer's general boot flow:
		1. BootROM(QSPI AHB): using QSPI reconfig parameter 133MHz DDR mode
		2. bootloader(QSPI AHB)-> reuse BootROM settings
		3. M7 AutoSRA(QSPI AHB)-> will call Fls_Init function, after do some write and read, using AHB read via DMA to load uboot code
		4. U-boot
		
		Questions:
		1. The API to reset QSPI after the timeout invoked by monitoring the 'BUSY'. Now, we're trying with Fls_IPW_InitControllers. Please see the attached mail to get the detailed modification. This seems can fix their currently BUSY issue. Is it reasonable ?
		
		2. If the AHB reading is invoked during the Fls operation, is it possible to get the QSPI stuck at 'BUSY' state ?
		3. The API to provide the status of flash/qspi which could tell that the AHB reading by the other software module is safe. Currently, below APIs are being used.
		/ Start DMA transfer /
		if(1==Fls_GetStatus() and and (2==Fls_GetJob()))
		{
		/*AHB read code*/
		}
		4. The recommended setting of the timeout to monitor the 'BUSY'.
		Please see the attached mail to get the details. Do we have some suggestions about those timeout values?

ARTD-55453	Bug	[fls] [GENERAL] PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Detailed description (how to reproduce it): In readme.txt file, the following sentence: PLUGINS_DIR The path to the Tresos plugins directory !john_floros_0-1653582301493.png|width=696,height=261! In fact, PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: PLUGINS_DIR points to EB Tresos plugin folder in readme.txt, which is incorrect. Expected behavior: PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt. PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: PLUGINS_DIR should point to RTD plugin folder : PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins
ARTD-74367	Bug	[EEP] Redundant macro in config file in precompile mode Detailed description (how to reproduce it): Preconditions: Variant with precompile Test Case ID (internal TC that caught the defect) - optional: Eep_TS_001 Observed behavior: Redundant macro in config file eep_cfg.h in precompile mode Expected behavior: Remove this marco. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution - optional: Update config file
ARTD-78743	Bug	[Fls] HardFault_Handler when enable User mode support Detailed description (how to reproduce it): HardFault_Handler when enable User mode support. Preconditions: N/A

ID	Subtype	Headline and Description
		Test Case ID (internal TC that caught the defect) - optional: TS_001 Observed behavior: Expected behavior: Run pass when enable user mode support Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution - optional: [....]
ARTD-79240	Bug	[Fls] HardFault_Handler when enable User mode support Detailed description (how to reproduce it): Example K144W run fail for Init MCU Preconditions: MCu no PLL not enable Test Case ID (internal TC that caught the defect) - optional: Observed behavior: Expected behavior: Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution - optional: [....]
ARTD-75388	Bug	[Fls] Wrong Interface QSPI in DS Detailed description (how to reproduce it): Wrong Interface QSPI in DS (IPL, HLD), different from EB. Preconditions: N/A Test Case ID (internal TC that caught the defect) - optional: TS_001 Observed behavior: Expected behavior: Correct interface like EB: Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution - optional: [....]
ARTD-55132	Bug	[FLS]There is no any Mcal interface to enable clock gate for Qspi in RTD software of S32K148 Detailed description (how to reproduce it): [When user want to enable the Qspi based on the RTD_4.4_S43K1XX_1.0.1_HF01 software, the timeout error was detected which can lead to Fls initialization failure. The process for duplication as following. 1st. Step: Create new project of S32K148 processor in EB Tresos Studio 2nd. Step: Configure the Fls and other essential modules in the EB Tresos Studio(Refer to the example of Fls) 3rd. Step: Integrate the generated code and RTD source file in a project based on S32DS environment. 4th. Step: Initial the Qspi driver through Fls_Init interface 5th. Step: Debugging with S32K148 Evb board and the Qspi timeout error happened during Fls initialization phase.] Preconditions: [Software environment need to be installed Processor: S32K148 (Based on hardware 700-29644 REV X1) EB Tresos Studio version: 27.1.0

ID	Subtype	Headline and Description
		RTD (Real Time Driver) version: 1.0.1 Hot Fix 01] Test Case ID (internal TC that caught the defect) - optional: [None] Observed behavior: [After research from my side, the root cause should be there is no any interface for enable Qspi clock gate. For AB testing, adding code for enable Qspi clock gate. After that, the Qspi is initialized successfully and the data can be read correctly from Qspi flash chip.] Expected behavior: [There shall be an interface for enable clock gate of Qspi in RTD software.] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution - optional: [....]

4.3 Change List for 2.0.0

ID	Subtype	Headline and Description
ARTD-14554	New Feature	[S32DS 3.5][Regression] Board folder should be excluded after detach SDK Scenario 1: [S32DS 3.5][Regression] Board folder should be excluded after detach SDK Precondition:* Open S32DS B210723 which installed package *SW32G2_RTD_4.4_2.0.0_HF03_DS_updatesite_D2106.zip Test Case: # Create project enable SDK S32CT for any processors (ex: S32G274A Rev2) # Right-click on project → SDKs → Detach SDK # Check board and SDK folder Observed behavior: SDK folder is excluded but board folder is still *included Expected behavior: SDK and board folder should be excluded after detach SDK Note: This issue does not occur in S32DS 3.4 Update 1 Release SW32G2_RTD_4.4_2.0.0_HF03_DS_updatesite_D2106.zip! Scenario 2: [S32DS 3.5] RTD folder should be excluded after detach SDK Precondition:* Open S32DS 3.5 release which installed package S32R_RTD_4_4_BETA_0_9_0_DS_updatesite_2207_signed_RC01.zip Test Case: # Create project enable RTD S32CT for S32R418AA # Right-click on project → SDKs → Detach SDK # Check board and RTD folder Observed behavior: Board folder is excluded but the RTD folder is still included Expected behavior: RTD and board folder should be excluded after detaching SDK. Scenario 3: [S32DS 3.5] SDK folder should be excluded after detaching the SDK Precondition:* Open S32DS 3.5 U1 which installed package S32S247TV SDK 0.8.1 # Create project enable RTD S32CT for S32S247TV SDK # Right-click on project → SDKs → Detach SDK # Check board and RTD folder Observed behavior: Board folder is excluded but the SDK folder is still included Expected behavior: SDK and board folder should be excluded after detaching SDK.
ARTD-15063	New Feature	RTD MCAL Example project makefile optimization

ID	Subtype	Headline and Description
		For the example projects included in the RTD MCAL, the GCC compiler will rebuild all files even if only one source file is modified a little bit. Is it possible to optimize the make file or the build script file so that only the modified source files are built when we run the "make build" command?
ARTD-18753	Bug	[mcu] Analysis the consistent between functionalities requirement and driver implementation Situation: During the last minutes of S32K3 1.0.0, we used an owner tool to check the consistent between requirement and source code implementation and found some drivers that have the inconsistent between the functions' requirements and drivers implementation. Something is really to fix in the below scenarios: Miss or redundant parameters. Platform is declared mark for this situation Wrong function name Functions are not expose to Users (in header files) We already fixed in Platform, LIN, MCL, MCU, LIN, SENT, FLS, ADC, SAI drivers. The rests should be analysis and dig into the inconsistent if it is wrong or not. The following 3 categories were found : The functions are declared in "*.h" files of driver code (exported to user) but not included in "ReqExport.txt" file (no requirement are found in Doors > no traceability > no test) The functions are found in "ReqExport.txt" (requirement exists) but the functions are not declared in "*.h" files of driver code (are not present for user) The function name is found in both "ReqExport.txt" and in "*.h" file but there is at least one mismatch on data type, variable name, ... Proposal: With the remaining findings, please to: This ticket should perform after requirement analyis ticket Export all requirement and take a review to conclude the consistent between requirement and implementation. Please take a review on Crucible, the checklist at attachment. In the case there is the needed to update driver or to update requirement, please create a ticket to implement and link to this ticket For each driver, analysis and resolution should be completed with relevant information Reference: Findings from S32K3 1.0.0 release (attachment), as optional Contact SW Testers to provide the latest results for each drivers (it's mandatory input)
ARTD-18762	Bug	[Base] Driver and requirement are inconsistent Detailed description (how to reproduce it): There are 3 requirements that are inconsistent between Requirements and Driver code !image-2021-10-26-16-17-07-413.png|thumbnail! Preconditions: Requirement baseline: 26.3 INTREQ_BASE_RTD_4.4_SJA11XX_0.9.0_I03 Base tag: BASE_258 Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: See detail description Expected behavior: The requirement and driver should be consistent Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Create ITWG ticket to update internal requirements
ARTD-19058	New Feature	[ETH][NETC] Implement scatter gather on transmission NewWorkDescription: Allow sending multiple buffer pointers using a single Send command. For example: buffer1 = headers of the frame (dest mac, src mac, eth type, tcp header, etc.)

ID	Subtype	Headline and Description
		buffer2 = payload One single send command will bring the 2 buffers together in one single frame Requirement source: Internal requirement (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Use the buffer descriptor chaining feature to do this
ARTD-19289	New Feature	[UART] - Support Loopback mode on Uart driver NewWorkDescription: Support Loopback mode transfer Requirement source: RM.pdf Proposed solution optional: Add a check in configurator in order to put the driver in loopback or not Configure the driver accordingly Add testcase to validate the feature.
ARTD-22020	New Feature	[ETH]Add infix support in the Eth driver NewWorkDescription: Add infix support in the Eth driver Requirement source: Planned activity (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: For having separate and independent namespaces, drivers with the defined multiplicity greater than 1 need to have their name extended with an infix. Steps: Outside of the driver: Add in Base a new MemMap Eth_43_NETC_MemMap.h generated for the new naming Add in Rte new SchM files Schm_Eth_43_NETC,c and Schm_Eth_43_NETC.h generated for the new naming In the driver: Add in the HLD and IPW M4 tags that will be replaced with the infix The files need to be renamed, the types, the functions, etc. Do not rename the types and defines that are specified in Eth_GeneralTypes. Those need to keep their name, as they will be used as defined by all Eth drivers (if more drivers are present in a project) Update the xdm file to change the package name and use the short_name of the driver instead of MODULE_NAME where needed Update the mak file of the driver to rename all files and to propagate the m4_infix_value in all needed files In the tests: Change the xdm configuration for the tests to use the new format Change the mak file of the tests to compile the correct plugin folder Create a wrapper file Eth.h which includes Eth_43_NETC.h and redefines all needed macros, typedefs and functions to point to the newly named entities
ARTD-23108	Bug	[I2s] Flexio supports only 1 Master or Slave channel On K1, Flexio has 4 shifters/timers. So it can support max 2 masters or 1 slave per derivatives. For K3 it will be higher since it has 8 shifters/timers But configurator support max 1 master or 1 slave for K1. This requires some changes in codegen Fix Master and Slave cannt work if the first resource index number is odd Add devtest to verify the new feature with multiple flexio configuration on K3 Remove I2sNumLogicChn node from configurator
ARTD-23359	Bug	[I2C] Wrong variable of I2c_ErrorCallback Detailed description (how to reproduce it): Project: C:\NXP\SW32_RTD_4.4_3.0.0\eclipse\plugins\I2c_TS_T40D11M30I0R0\examples\EBT \I2c_HLD_Transfer_S32G274A_M7 EB treos: DMA was in disable state. Disable I2c_Callback, Enable I2c_ErrorCallback. !image-2022-01-24-11-15-02-761.png|width=515,height=330! I2c_Ipw.c : !image-2022-01-24-11-16-52-969.png|width=878,height=103!

ID	Subtype	Headline and Description
		I2C_EVENT_DMA_TRANSFER_ERROR_MASTER could not be found in I2c_Ipw.c* file result to building failed. But I found I2C_MASTER_EVENT_DMA_TRANSFER_ERROR in I2c_Ip_Callbacks.h. I2c_Ip_Callbacks.h : !image-2022-01-24-11-42-53-783.png|width=510,height=131! So, I doubt it should be "I2C_MASTER_EVENT_DMA_TRANSFER_ERROR" instead of "I2C_EVENT_DMA_TRANSFER_ERROR_MASTER" in I2c_Ipw.c. Please help to check it. Thanks. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: N/A Expected behavior: N/A Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-23384	Bug	[WDG]There are some data types mismatch between EBT and CT There are some value mismatches between EBT and CT. Please see for example the attached file. [VALIDATION] Mismatch between the types of EPD:WdgClockValue (ECUC-INTEGER-PARAM-DEF) and CT:WdgClockValue (info) [VALIDATION] EPD:WdgEcucPartitionRef has multiplicity, but CT:WdgEcucPartitionRef is not an array [VALIDATION] EPD:WdgExternalConfiguration has multiplicity, but CT:WdgExternalConfiguration is not an array [VALIDATION] EPD:WdgExternalContainerRef has multiplicity, but CT:WdgExternalContainerRef is not an array Verify EPD/EPC support for HLD and IPLD. Please refer to [this post on Teams https://teams.microsoft.com/l/message/19:e451c595d3ba410c8f9c1f5efc19a8a3@thread.tacv2/1606904026692?tenantId=686ea1d3-bc2b-4c6f-a92c-d99c5c301635and groupId=f9decd68-7b74-4101-a4d2-ad54e86f727fand parentMessageId=1606904026692and teamName=Zebraand channelName=Configuration%20(code%2C%20DS%2C%20EB%2C%20etc)and createdTime=1606904026692] for additional details. From the presentation, the main focus should be on the following sections: Mapping XDM to Component EPD Importer EPD Generation EPC Importer EPC Generation
ARTD-23425	Bug	[lin] [S32CT] Incorrect enable attributes for ConfigTimeSupport fields Detailed description (how to reproduce it): Manually check the ConfigTimeSupport fields of the S32CT component / template Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: POST_BUILD_VARIANT_USED has enable="false" IMPLEMENTATION_CONFIG_VARIANT has enable="false" Expected behavior: POST_BUILD_VARIANT_USED has enable="false" if and only if postBuildVariantSupport="false" in the config set's options_expr. IMPLEMENTATION_CONFIG_VARIANT has enable="true" or no enable attribute (which is equivalent to enable="true"). Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: POST_BUILD_VARIANT_USED has enable="false" if and only if postBuildVariantSupport="false" in the config set's options_expr. IMPLEMENTATION_CONFIG_VARIANT has enable="true" or no enable attribute (which is equivalent to enable="true").
ARTD-23433	Bug	[port] [S32CT] Incorrect enable attributes for ConfigTimeSupport fields Detailed description (how to reproduce it): Manually check the ConfigTimeSupport fields of the S32CT component / template Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A

ID	Subtype	Headline and Description
		Observed behavior: POST_BUILD_VARIANT_USED has enable="false" IMPLEMENTATION_CONFIG_VARIANT has enable="false" Expected behavior: POST_BUILD_VARIANT_USED has enable="false" if and only if postBuildVariantSupport="false" in the config set's options_expr. IMPLEMENTATION_CONFIG_VARIANT has enable="true" or no enable attribute (which is equivalent to enable="true"). Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: POST_BUILD_VARIANT_USED has enable="false" if and only if postBuildVariantSupport="false" in the config set's options_expr. IMPLEMENTATION_CONFIG_VARIANT has enable="true" or no enable attribute (which is equivalent to enable="true").
ARTD-23442	Bug	[uart] [S32CT] Incorrect enable attributes for ConfigTimeSupport fields Detailed description (how to reproduce it): Manually check the ConfigTimeSupport fields of the S32CT component / template Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: POST_BUILD_VARIANT_USED has enable="false" IMPLEMENTATION_CONFIG_VARIANT has enable="false" Expected behavior: POST_BUILD_VARIANT_USED has enable="false" if and only if postBuildVariantSupport="false" in the config set's options_expr. IMPLEMENTATION_CONFIG_VARIANT has enable="true" or no enable attribute (which is equivalent to enable="true"). Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: POST_BUILD_VARIANT_USED has enable="false" if and only if postBuildVariantSupport="false" in the config set's options_expr. IMPLEMENTATION_CONFIG_VARIANT has enable="true" or no enable attribute (which is equivalent to enable="true").

ARTD-23563	Bug	[MCU] Power mode change notification not work without Mcu_InitClock
		
		{_}*Detailed description (how to reproduce it):*{_}{*}{{*}}
		
		Test case is to check power change notification feature without Mcu_InitClock
		
		System clock before VLPR mode (picture 1) is FIRC
		
		System clock after back to Run mode from VLPR mode (picture 2) is SIRC
		
		Issue: power moder change notification feature can not make system clock back to FIRC
		
		Preconditions:
		[...]
		
		Test Case ID (internal TC that caught the defect) optional:
		Mcu_TC_0593
		
		Observed behavior:
		System clock sourse change after exit from VLPR mode
		
		Expected behavior:
		System clock sourse not change before and after exit from VLPR mode
		
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)
		
		Proposed solution optional:
		Fix this funtion like this :
		
		void Clock_Ip_PowerModeChangeNotification(Clock_Ip_PowerModesType PowerMode,Clock_Ip_PowerNotificationType Notification)
		{
		    static boolean copy = FALSE;
		    if(copy == FALSE)
		
		    {
		
		        copy = TRUE;
		
		        Clock_Ip_Command(Clock_Ip_pxConfig, CLOCK_IP_INITIALIZE_CLOCK_OBJECTS_COMMAND);
		
		     }
		
		    Clock_Ip_ClockPowerModeChangeNotification(PowerMode, Notification);
		}

ID	Subtype	Headline and Description
ARTD-23614	Bug	[Base] -fdump-ipa-all shall not be used in project setting by default Detailed description (how to reproduce it): See the report from Mike and an internal user below for DS3.4 Seems the makefile generator for GCC ARM is enabling a GCC developer option, fdump-ipa-all. This option dumps internal GCC optimization information – not useful to the user. In this case, a user is getting an error. I suppose in most cases, the dump file would be created and destroyed without the user ever knowing it was created. Since this is a GCC developer option, no user would ever have any use for it, so it is a bit mysterious why it should show up in projects. I searched some of my 3.4 projects and found the flag in the some of the projects' arg files. … I"C:/Users/nxa10550/workspaceS32DS.3.4/project.zip_expanded/Crypto_Aes_Enc_Dec/generate/src" I"C:/Users/nxa10550/workspaceS32DS.3.4/project.zip_expanded/Crypto_Aes_Enc_Dec/generate/include" Os fshort-enums funsigned-char fstack-usage fdump-ipa-all fomit-frame-pointer Looking a bit closer – could this have come from an RTD project imported from somewhere else because the "Other optimization flags" string field is apparently set. ./.cproject: <option id="gnu.c.compiler.option.optimization.flags.1148268502" name="Other optimization flags" superClass="gnu.c.compiler.optio n.optimization.flags" useByScannerDiscovery="false" value="-fshort-enums funsigned-char fstack-usage fdump-ipa-all fomit-frame-pointer fno-zero-initialized-in-bss" valueType="string"/> This appears also in a new application project see attached email. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: see above Expected behavior: see above Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-23610	Bug	[SPI][S32K1XX] Half Duplex Slave Transmitter 1bit CPOL_CPHA=HIGH_LEADING run fail Detailed description (how to reproduce it): Half Duplex Slave Transmitter 1bit CPOL_CPHA=HIGH_LEADING run fail. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: Spi_TS_122 cfg_sets=3

ID	Subtype	Headline and Description
		Observed behavior: Half Duplex Slave Transmitter 1bit CPOL_CPHA=HIGH_LEADING run fail. Expected behavior: Half Duplex Slave Transmitter 1bit CPOL_CPHA=HIGH_LEADING run pass. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-23630	Bug	[CAN] Driver is not compiling when using INFIX check on S32CT Detailed description (how to reproduce it): Driver is not compiling when using INFIX. There are multiple variables that are infixed in some of the places, not in all places. There are 2 methods of adding infix using M4, it is almost impossible to keep track of the infix method based on the file. There is no INFIX in the IP DRIVER. Flex LLCE will use the IP driver data structures in the interface, so it can't compile along FlexLLCE driver on the host side. Preconditions: Generate the driver using infix Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Not compiling {_}*Expected behavior:*{_}To compile Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Fix Compile error. Stick to only one way of adding infix. Add INFIX to IP driver.
ARTD-23643	Bug	[PORT] Inconsistent file name and component references Detailed description (how to reproduce it): Inconsistent casing between the driver file names and the references from CT components. This makes the build/generate stage fail on Linux environments which are case sensitive. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: NA Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Check and ensure all file names and the places where they are referenced, will use the exact same case. See the attached email for more details.
ARTD-23682	New Feature	[PORT] Change the define names to not contain derivatives name NewWorkDescription: [https://bitbucket.sw.nxp.com/projects/ARTD/repos/port/pull-requests/410/overview?commentId=1104467] [Radu-Andrei Brasoveanu|https://bitbucket.sw.nxp.com/users/nxa19269] i would advise in creating a follow up ticket for entire code and changing the define names to not contain derivatives name. more suitable will be to use something like: FEATURE_PORT_CI_PORT_IP_PCR_MUXING}} Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Change the define names to not contain derivatives name.

ID	Subtype	Headline and Description
ARTD-23702	Bug	[Wdg] When WdgDevErrorDetect disabled, some generated defines are not used Detailed description (how to reproduce it): When WdgDevErrorDetect disabled, some generated defines are not used MISRA C-2012 Rule 2.5 1164 Advisory Low Quality Macro "WDG_TIMEOUT_VALUE_ARRAY" is defined but never used. /ARTD-CIW55-9/sources/SJA11XX_4.4/output/SJA11XX_SJA1110/wdg/Wdg_TS_Shared_000_cfgPB_CORE0/generate_tresos/include/Wdg_Cfg_Defines.h 228 Intentional Dismissed This violation requires information for indirect service mode MISRA C-2012 Rule 2.5 1164 Advisory Low Quality Macro "WDG_IPW_TIMEOUT_VALUE_ARRAY" is defined but never used. /ARTD-CIW55-9/sources/SJA11XX_4.4/output/SJA11XX_SJA1110/wdg/Wdg_TS_Shared_000_cfgPB_CORE0/generate_tresos/include/Wdg_Ipw_Cfg_Defines.h Preconditions: Direct service mode disabled Test Case ID (internal TC that caught the defect) optional: N.A. Observed behavior: See description Expected behavior: Defines are not generated if not used Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Only generate when WdgDevErrorDetect enabled in both S32CT and EBT
ARTD-23845	Bug	[MCL] Remove DMA Enable Start option for DMA Transfer configuration Changing Enable Start in the configuration tool does not change the generated structure !image-2022-02-09-10-02-10-091.png!
ARTD-23850	Bug	[CAN] Driver is not compiling when using INFIX Detailed description (how to reproduce it): Driver is not compiling when using INFIX. There are multiple variables that are infixed in some of the places, not in all places. There are 2 methods of adding infix using M4, it is almost impossible to keep track of the infix method based on the file. There is no INFIX in the IP DRIVER. Flex LLCE will use the IP driver data structures in the interface, so it can't compile along FlexLLCE driver on the host side. Preconditions: Generate the driver using infix Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Not compiling {_}*Expected behavior:*{_}To compile Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Fix Compile error. Stick to only one way of adding infix. Add INFIX to IP driver.
ARTD-23858	Bug	Customer reported issue adding driver Flexio_Uart to S32CT Peripherals on S32K148 project Customer experience is reported in S32DS Community site: [https://community.nxp.com/t5/S32-Design-Studio/S32Design-Studio-Error-when-adding-Flexio-Uart-driver/m-p/1418438#M8249] Description below is from S32DS Support Team's reproduction of the issue. Detailed description (how to reproduce it): Issue is easily reproduced 1. Create New Project from Example 'Gpio_Dio_ToggleLed_S32K148' 2. Open S32CT Peripherals tool. 3. Add new driver Flexio_Uart

ID	Subtype	Headline and Description
		Preconditions: 1. S32 Design Studio v3.4 installed 2. S32K1xx Development Package installed 3. RTD S32K1 1.0.0 installed Observed behavior: 1. Error message: Execution of javascript function hasExclusiveOwnership failed An error has occurred. See error log for more details. PeripheralExpressionException[Description: Execution of javascript function hasExclusiveOwnership failed Function: execScriptFunction(valid, valid, valid, valid, valid) Contexts: WrappedContext[source_context = ExpressionContext[child_context = ChildContext[$this = Flexio_Uart.GeneralConfiguration.UartDevErrorDetect $instance = Flexio_Uart_1 $group = BOARD_InitPeripherals]]] ExpressionContext[child_context = ChildContext[$this = Flexio_Uart.GeneralConfiguration.UartDevErrorDetect $instance = Flexio_Uart_1 $group = BOARD_InitPeripherals]] ExpressionContext[child_context = ChildContext[$this = Flexio_Uart.GeneralConfiguration.UartDevErrorDetect $instance = Flexio_Uart_1 $group = BOARD_InitPeripherals]]] Expected behavior: No error should be reported
ARTD-23864	Bug	[BASE][S32CT] Code generation failed during RTD - LLCE integration Detailed description (how to reproduce it): While creating a project with multiple drivers, including LLCE modules and Dio, the code generation was failed. Preconditions: Integrating Dio from RTD_3.0.0 and LLCE modules from LLCE_1.0.3 Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: The following error was shown: !image-2022-02-25-09-39-48-908.png! Expected behavior: The project can be generated without any error. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: A temporary solution is to add a requireScript in dio_codegenerator.js file. It is needed because "index" was somewhere initiated, and requireScript reimport index function to make sure that Dio can get the correct "index" !image-2022-02-25-09-42-58-349.png!
ARTD-23906	Bug	[WDG] Generated code by 3rd party tool is incorrect when /ActiveEcuC/Wdg is defined Detailed description (how to reproduce it): Generated code by 3rd party tool is incorrect when /ActiveEcuC/Wdg is defined in the configuration file instead of /Wdg/Wdg. Preconditions: Generated code by 3rd party tool, /ActiveEcuC/Wdg is defined in the configuration file Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Generated code by 3rd party tool is incorrect when /ActiveEcuC/Wdg is defined in the configuration file instead of /Wdg/Wdg. Expected behavior: Generated code should be correct regardless the node definitions. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: There was a known issue from MCAL (e.g. MCAL-2875 in which:

ID	Subtype	Headline and Description
		The autosar path Mod/ELEMENTS/Mod (where Mod is module name, for example Adc, Mcu, Resource etc.) must not be used in code generation templates (Mod_PBcfg.c, Mod_Cfg.c) to reference other nodes. The reason is that this path does not exist if there is used third party configuration tool. This leads to generation errors as node is not found. The solution would be replace the ./../../../Wdg/ELEMENTS/Wdg by: node:refs('ASPathDataOfSchema:/AUTOSAR/Wdg')
ARTD-23915	New Feature	[adc] Adjust range of integer node in EB due to limitation of java NewWorkDescription: Previously, we implement integer node has maximum range from [TPS_GST_00008]: 9223372036854775808 .. 18446744073709551615 but in latest release node of EB28, it state that: Deviating from the AUTOSAR traceable [TPS_GST_00008], the formula language in terpreter of EB tresos Studio supports only the signed 64-bit range for integer values, i. e. from 0x8000000000000000 (-9223372036854775808) to 0x7fffffffffffffff (9223372036854775807). The formula language parser of EB tresos Studio is implemented in Java. It uses the long primitive type for integer values and therefore can only support the restricted range. Requirement source: Please try to generate vsmd report using EB 28.2 if you got no error, please ignore this ticket. If you got some errors, you might see this warning in log prompt from EB on terminal (eg: ADC): This is a restriction of tresos Studio. The value is automatically restricted to 9223372036854775807. WARNING 22-03-08,11:28:58 (23016) The MAX attribute (describing the maximum allowed value for the parameter AdcPowerState with current value 18446744073709551615 defined in file D:\Git_rtd\tools\tresos_plugin_builder\make\..\..\..\output \eclipse\plugins\Adc_TS_T31D53M8I0R0\autosar\Adc_s32e27x_bga975.epd at line 4981) is greater than 64 bit signed integer. This is a restriction of tresos Studio. The value is automatically restricted to 9223372036854775807. {_}*Error EcucSws_1007: For Integer and Float Parameters the MIN values must be >= and the MAX values <= as in the StMD. (see example in VSMD report of ADC in attachment*{_}{*}{{*}} Proposed solution optional: To elimilate warning and error on vsmd report, we need to implement a fix to change the range of integer node adapt with EB example: Before* <v:var name="AdcChannelConvTime" type="INTEGER"> <a:da name="INVALID" type="Range"> <a:tst expr="and lt;=18446744073709551615"/> <a:tst expr="and gt;=0"/> </a:da> </v:var> Example: After fixed <v:var name="AdcChannelConvTime" type="INTEGER"> <a:da name="INVALID" type="Range"> <a:tst expr="and lt;=9223372036854775807"/> <a:tst expr="and gt;=0"/> </a:da> </v:var> The range of all integer nodes should be limit from 0x8000000000000000 (-9223372036854775808) to 0x7fffffffffffffff (9223372036854775807).
ARTD-23977	New Feature	[can] [S32CT] Update code generation to work in JS strict mode NewWorkDescription: Note: Only Peripherals Tool's components are affected{color}*. There has been an update to the System module: System scripts now run in strict mode and contain include guards to protect against multiple inclusion System utilities / functions have been made read-only Code templates are now evaluated in strict mode so that developers can be aware of overwriting existing variables or using undeclared variables*. Requirement source: Consequence of https://jira.sw.nxp.com/browse/ARTD-23864 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: What you have to do: ## Open S32DS's logging file (swtools.log) or launch the program with console logging activated (e.g. run ".\s32ds.exe consoleLog in CMD/PowerShell) ## Create a new project and instantiate your high-level / low-level component.

ID	Subtype	Headline and Description
		## Fix the encountered errors related to your component. Most of them will be along the lines of "ReferenceError: <variable_name> is not defined" or "TypeError: <variable_name> is not a writable property of [object global]"_. Both can be resolved by simply defining the offending variable using "var". ## Remove inclusions of "refsSolver.js" in your code templates or code generator scripts. If there's an API you need from that file which is not provided in codegeneratorUtils.js, please let me know.
ARTD-23983	New Feature	[dio] [S32CT] Update code generation to work in JS strict mode NewWorkDescription: Note: Only Peripherals Tool's components are affected{color}*. There has been an update to the System module: System scripts now run in strict mode and contain include guards to protect against multiple inclusion System utilities / functions have been made read-only Code templates are now evaluated in strict mode so that developers can be aware of overwriting existing variables or using undeclared variables*. Requirement source: Consequence of https://jira.sw.nxp.com/browse/ARTD-23864 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: What you have to do: ## Open S32DS's logging file (swtools.log) or launch the program with console logging activated (e.g. run ".\s32ds.exe consoleLog in CMD/PowerShell) ## Create a new project and instantiate your high-level / low-level component. ## Fix the encountered errors related to your component. Most of them will be along the lines of "ReferenceError: <variable_name> is not defined" or "TypeError: <variable_name> is not a writable property of [object global]"_. Both can be resolved by simply defining the offending variable using "var". ## Remove inclusions of "refsSolver.js" in your code templates or code generator scripts. If there's an API you need from that file which is not provided in codegeneratorUtils.js, please let me know.
ARTD-23994	New Feature	[i2c] [S32CT] Update code generation to work in JS strict mode NewWorkDescription: Note: Only Peripherals Tool's components are affected{color}*. There has been an update to the System module: System scripts now run in strict mode and contain include guards to protect against multiple inclusion System utilities / functions have been made read-only Code templates are now evaluated in strict mode so that developers can be aware of overwriting existing variables or using undeclared variables*. Requirement source: Consequence of https://jira.sw.nxp.com/browse/ARTD-23864 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: What you have to do: ## Open S32DS's logging file (swtools.log) or launch the program with console logging activated (e.g. run ".\s32ds.exe consoleLog in CMD/PowerShell) ## Create a new project and instantiate your high-level / low-level component. ## Fix the encountered errors related to your component. Most of them will be along the lines of "ReferenceError: <variable_name> is not defined" or "TypeError: <variable_name> is not a writable property of [object global]"_. Both can be resolved by simply defining the offending variable using "var". ## Remove inclusions of "refsSolver.js" in your code templates or code generator scripts. If there's an API you need from that file which is not provided in codegeneratorUtils.js, please let me know.
ARTD-24005	New Feature	[port] [S32CT] Update code generation to work in JS strict mode NewWorkDescription: Note: Only Peripherals Tool's components are affected{color}*. There has been an update to the System module: System scripts now run in strict mode and contain include guards to protect against multiple inclusion System utilities / functions have been made read-only Code templates are now evaluated in strict mode so that developers can be aware of overwriting existing variables or using undeclared variables*. Requirement source: Consequence of https://jira.sw.nxp.com/browse/ARTD-23864 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: What you have to do: ## Open S32DS's logging file (swtools.log) or launch the program with console logging activated (e.g. run ".\s32ds.exe consoleLog in CMD/PowerShell) ## Create a new project and instantiate your high-level / low-level component. ## Fix the encountered errors related to your component. Most of them will be along the lines of "ReferenceError: <variable_name> is not defined" or "TypeError: <variable_name> is not a writable property of [object global]"_. Both can be resolved by simply defining the offending variable using "var". ## Remove inclusions of "refsSolver.js" in your code templates or code generator scripts. If there's an API you need from that file which is not provided in codegeneratorUtils.js, please let me know.
ARTD-24006	New Feature	[pwm] [S32CT] Update code generation to work in JS strict mode

ID	Subtype	Headline and Description
		NewWorkDescription: Note: Only Peripherals Tool's components are affected{color}*. There has been an update to the System module: System scripts now run in strict mode and contain include guards to protect against multiple inclusion System utilities / functions have been made read-only Code templates are now evaluated in strict mode so that developers can be aware of overwriting existing variables or using undeclared variables*. Requirement source: Consequence of https://jira.sw.nxp.com/browse/ARTD-23864 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: What you have to do: ## Open S32DS's logging file (swtools.log) or launch the program with console logging activated (e.g. run ".\s32ds.exe consoleLog in CMD/PowerShell) ## Create a new project and instantiate your high-level / low-level component. ## Fix the encountered errors related to your component. Most of them will be along the lines of "ReferenceError: <variable_name> is not defined" or "TypeError: <variable_name> is not a writable property of [object global]"_. Both can be resolved by simply defining the offending variable using "var". ## Remove inclusions of "refsSolver.js" in your code templates or code generator scripts. If there's an API you need from that file which is not provided in codegeneratorUtils.js, please let me know.
ARTD-24008	New Feature	[rm] [S32CT] Update code generation to work in JS strict mode NewWorkDescription: Note: Only Peripherals Tool's components are affected{color}*. There has been an update to the System module: System scripts now run in strict mode and contain include guards to protect against multiple inclusion System utilities / functions have been made read-only Code templates are now evaluated in strict mode so that developers can be aware of overwriting existing variables or using undeclared variables*. Requirement source: Consequence of https://jira.sw.nxp.com/browse/ARTD-23864 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: What you have to do: ## Open S32DS's logging file (swtools.log) or launch the program with console logging activated (e.g. run ".\s32ds.exe consoleLog in CMD/PowerShell) ## Create a new project and instantiate your high-level / low-level component. ## Fix the encountered errors related to your component. Most of them will be along the lines of "ReferenceError: <variable_name> is not defined" or "TypeError: <variable_name> is not a writable property of [object global]"_. Both can be resolved by simply defining the offending variable using "var". ## Remove inclusions of "refsSolver.js" in your code templates or code generator scripts. If there's an API you need from that file which is not provided in codegeneratorUtils.js, please let me know.
ARTD-24012	New Feature	[spi] [S32CT] Update code generation to work in JS strict mode NewWorkDescription: Note: Only Peripherals Tool's components are affected{color}*. There has been an update to the System module: System scripts now run in strict mode and contain include guards to protect against multiple inclusion System utilities / functions have been made read-only Code templates are now evaluated in strict mode so that developers can be aware of overwriting existing variables or using undeclared variables*. Requirement source: Consequence of https://jira.sw.nxp.com/browse/ARTD-23864 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: What you have to do: ## Open S32DS's logging file (swtools.log) or launch the program with console logging activated (e.g. run ".\s32ds.exe consoleLog in CMD/PowerShell) ## Create a new project and instantiate your high-level / low-level component. ## Fix the encountered errors related to your component. Most of them will be along the lines of "ReferenceError: <variable_name> is not defined" or "TypeError: <variable_name> is not a writable property of [object global]"_. Both can be resolved by simply defining the offending variable using "var". ## Remove inclusions of "refsSolver.js" in your code templates or code generator scripts. If there's an API you need from that file which is not provided in codegeneratorUtils.js, please let me know.
ARTD-24015	New Feature	[wdg] [S32CT] Update code generation to work in JS strict mode NewWorkDescription: Note: Only Peripherals Tool's components are affected{color}*. There has been an update to the System module: System scripts now run in strict mode and contain include guards to protect against multiple inclusion System utilities / functions have been made read-only

ID	Subtype	Headline and Description
		Code templates are now evaluated in strict mode so that developers can be aware of overwriting existing variables or using undeclared variables*. Requirement source: Consequence of https://jira.sw.nxp.com/browse/ARTD-23864 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: What you have to do: ## Open S32DS's logging file (swtools.log) or launch the program with console logging activated (e.g. run ".\s32ds.exe consoleLog in CMD/PowerShell) ## Create a new project and instantiate your high-level / low-level component. ## Fix the encountered errors related to your component. Most of them will be along the lines of "ReferenceError: <variable_name> is not defined" or "TypeError: <variable_name> is not a writable property of [object global]"_. Both can be resolved by simply defining the offending variable using "var". ## Remove inclusions of "refsSolver.js" in your code templates or code generator scripts. If there's an API you need from that file which is not provided in codegeneratorUtils.js, please let me know.
ARTD-24018	New Feature	[DIO] Change the define names to not contain derivatives name NewWorkDescription: [https://bitbucket.sw.nxp.com/projects/ARTD/repos/port/pull-requests/410/overview?commentId=1104467] [Radu-Andrei Brasoveanu|https://bitbucket.sw.nxp.com/users/nxa19269] i would advise in creating a follow up ticket for entire code and changing the define names to not contain derivatives name. more suitable will be to use something like: FEATURE_PORT_CI_PORT_IP_PCR_MUXING}} Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Change the define names to not contain derivatives name.
ARTD-24111	New Feature	[adc] [S32CT] Update code generation to work in JS strict mode NewWorkDescription: Note: Only Peripherals Tool's components are affected{color}*. There has been an update to the System module: System scripts now run in strict mode and contain include guards to protect against multiple inclusion System utilities / functions have been made read-only Code templates are now evaluated in strict mode so that developers can be aware of overwriting existing variables or using undeclared variables*. Requirement source: Consequence of https://jira.sw.nxp.com/browse/ARTD-23864 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: What you have to do: ## Open S32DS's logging file (swtools.log) or launch the program with console logging activated (e.g. run ".\s32ds.exe consoleLog in CMD/PowerShell) ## Create a new project and instantiate your high-level / low-level component. ## Fix the encountered errors related to your component. Most of them will be along the lines of "ReferenceError: <variable_name> is not defined" or "TypeError: <variable_name> is not a writable property of [object global]"_. Both can be resolved by simply defining the offending variable using "var". ## Remove inclusions of "refsSolver.js" in your code templates or code generator scripts. If there's an API you need from that file which is not provided in codegeneratorUtils.js, please let me know.
ARTD-24454	Bug	[MCU] When setting Mcu Register Values Optimization is enable, mcu can not init Detailed description (how to reproduce it): When setting Mcu Register Values Optimization is enable, MCU can not init !screenshot-1.png|thumbnail! Issue2: Improvement generation script: 2 list MCU.CMU_FC.Address.List}} and {{MCU.CMU_FC.List}} contain same information: cmuInstance and cmuMonitorname.-> using string operation and add function macro to use 1 list only try "constrains" to avoid using multiple #IF to check McuCmuName Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: hardfault when call MCU_initclock() Expected behavior: MCU_initclock() run fine Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc. Proposed solution optional: [...]
ARTD-24479	New Feature	[base] add Can_TimestampType to Can_GeneralType.h NewWorkDescription:

ID	Subtype	Headline and Description
		as defined by EA_RTD_00109 !image-2022-03-16-18-21-24-594.png! Can_TimeStampType*_ should be available via Can_GeneralType.h Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: add typedef struct { uint32 nanoseconds; /**< brief Nanoseconds part of the time. / uint32 seconds; /**< brief Seconds part of the time. / }Can_TimeStampType;
ARTD-24504	New Feature	[S32CT] Prepare environment for multiple configurations support NewWorkDescription: Prepare environment for multiple configurations support Requirement source: [SRS_BSW_00405] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Prepare environment for multiple configurations support
ARTD-25461	Bug	[I2S] SAI DMA transfer counts limited by 32767, but no assert/warning by RTD Detailed description (how to reproduce it): [Invoke "void Sai_Ip_Send(const uint8 u8Instance, const uint8 const aData[], const uint32 u32Count)" with more than 32767 counts] Preconditions: [SAI work in DMA transfer mode] Test Case ID (internal TC that caught the defect) optional: [SFDC CN 00456190] Observed behavior: [Only part of the u32Count can be transfered, the DMA TCD major loop counts CITTER[0-14] which will cut lower 15-bits width of the total "u32Count"] Expected behavior: [all of the u32Count data can be transfered, or a warning/assert to indicate the input param out of range] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [Software divided the source data to multiple slices, and ensure each of the slice data counts under the "DMA mode, 32767 counts" limitation]
ARTD-25626	Bug	[ADC] Adc physical channel name gives error when duplicated in channel configuration array Detailed description (how to reproduce it): When configuring the channels configuration array, setting more than one physical channel with the same value results in an error, even if the control channel value is different. On K1, multiple control channels can be associated with a single physical channel so this is a valid configuration. Preconditions: Project with Adc_Ip component. Observed behavior: Error is given with valid configuration. Expected behavior: No errors. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-25640	New Feature	[BASE] Update compatibility.xml for S3K1XX, S32ZSE NewWorkDescription:

ID	Subtype	Headline and Description
		Update compatible.xml for S3K1XX, S32ZSE and SJA11XX. Requirement source: (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional:
ARTD-25732	New Feature	[adc] Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25733	New Feature	[build_env] Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25734	New Feature	[can] Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25735	New Feature	[crc] Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets)

ID	Subtype	Headline and Description
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25736	New Feature	[crypto] Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25741	New Feature	[dio] Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25744	New Feature	[eep] Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25745	New Feature	[eth] Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)

ID	Subtype	Headline and Description
		Proposed solution optional: Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25750	New Feature	[gpt] Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25751	New Feature	[i2c] Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25752	New Feature	[i2s] Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25755	New Feature	[lin] Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)

ID	Subtype	Headline and Description
		Proposed solution optional: Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25756	New Feature	[mcl] Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25757	New Feature	[mcu] Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25763	New Feature	[port] Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25765	New Feature	[pwm] Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional:

ID	Subtype	Headline and Description
		Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25766	New Feature	[qd] Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25767	New Feature	[rm] Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25873	New Feature	[eth] Rename OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25902	Bug	[S32K1][PORT]Declare variables are not constants in PORT_START_SEC_CONST_32 memory Detailed description (how to reproduce it): In file Port_Ci_Port_Ip.h, Port_au32PortCiPortBaseAddr[5] must declare is a constant. #define PORT_START_SEC_CONST_32 #include "Port_MemMap.h" extern PORT_Type Port_au32PortCiPortBaseAddr[5]; extern GPIO_Type Port_au32PortCiGpioBaseAddr[5]; #define PORT_STOP_SEC_CONST_32 Preconditions: Declare variable must be constant in PORT_START_SEC_CONST_32 memory Test Case ID (internal TC that caught the defect) optional: None

ID	Subtype	Headline and Description
		Observed behavior: Declare variable isn't constant in PORT_START_SEC_CONST_32 memory Expected behavior: Port_au32PortCiPortBaseAddr[5] and Port_au32PortCiGpioBaseAddr[5] must be the constants Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Add the "const" keyword.
ARTD-25914	New Feature	[base] Rename Base to BaseNXP in EBT and OsIf to BaseNXP in S32CT NewWorkDescription: Actions that need to be taken: All schema files have to be updated (.xdm, .component) Effort mitigated by attached script Testing makefiles (with dependency on Base) have to be updated (in theory, only in common.mak) Effort mitigated by attached script All test configuration files for Base have to be updated: For XDM configuration files Effort mitigated by attached script For MEX configuration files Manually performed (the script can't help here since the structure of OsIf has been drastically changed in S32CT to match the one from EBT) Have to update input.json files if they are referencing Base Effort mitigated by attached script Requirement source: EB Request (refer to linked tickets) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Copy the attached script in the root directory of your driver's repository and execute it from there in Cygwin (./RenameBase.sh) Re-configure OsIf in the MEX configuration files. Regenerate examples to check that everything is working fine again.
ARTD-25940	New Feature	[RM] Move MPU Requirements to Platform NewWorkDescription: Move Mpu, Mpu_M7 and Smpu IP requirements from RM to Platform. Remove Mpu HLD requirements from RM Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Move Mpu, Mpu_M7 and Smpu IP requirements from RM to Platform. Remove Mpu HLD requirements from RM
ARTD-25997	Bug	[CAN] fail to check constraint for CAN_1 when enabled multicore / S32CT Detailed description (how to reproduce it): A> Create S32Z project / s32ct Enable multicores Add CAN_0, then fix constraints Add CAN_1, => expectation: constraint log instead of anonymous code-generate error B> Add EcucPartition_0 to CanEcucPartitionRef CAN_0 refer to EcucPartition_One (instead of EcucPartition_0) => contraint log should be displayed C> Missing constraint for checking mcl channel and dma_callback_name. Please add constraint for this as in EB. For 24 instances this is necessary feature Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: A> !image-2022-04-19-18-21-47-015.png|thumbnail! [...] B> No contraint log was displayed Expected behavior: A> constraint log instead of anonymous code-generate error B> contraint log should be displayed Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-26056	Bug	[can] constraint for dma channel check should be support / IpFlexCAN

ID	Subtype	Headline and Description
		Detailed description (how to reproduce it): Create s32z project / IPL layer Add Flexcan_0 and enable dma Add Flexcan_1 and enable dma By default the values of dma channel is the same as 0 value => no error indicate when all controllers have same dma_channel value Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: !image-2022-04-22-11-25-00-962.png|thumbnail! Expected behavior: for fields that need distinct value, constraint should be implemented. For example as the implementation for "FlexCan Configuration Name" Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-26084	Bug	[Port]: the memory section don't match the variable type, it will cause the compile result abnormal when compile with IAR Detailed description (how to reproduce it): when I compiling the code with IAR, the binary file include a block which the address belongs to RAM area. it will cause download failed. after analysis the code, it caused by the code in Port.c. After I remove the initial value which showed in the picture, the compile result will not show the block 3. !image-2022-04-26-14-29-44-797.png! !image-2022-04-26-14-30-48-463.png! Preconditions: compile with IAR Test Case ID (internal TC that caught the defect) optional: None Observed behavior: the compile binary file abnormal Expected behavior: without data which belongs to RAM areas Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: optimize the initialization value or change the section

ARTD-26206	New Feature	S32K3xx, RTD LPSPI Async Transmit
		
		I have modified the LPSPI_Ip_HalfDuplexTransfer_S32K344 example to have Async transfer from the Master.
		
		/ Master transmits data in half-duplex mode by sync method /
		//Lpspi_Ip_SyncTransmitHalfDuplex(and MASTER_EXTERNAL_DEVICE, TxMasterBuffer, NUMBER_OF_BYTES, LPSPI_IP_HALF_DUPLEX_TRANSMIT, TIMEOUT);
		Lpspi_Ip_AsyncTransmitHalfDuplex(and MASTER_EXTERNAL_DEVICE, TxMasterBuffer, NUMBER_OF_BYTES, LPSPI_IP_HALF_DUPLEX_TRANSMIT, NULL_PTR);
		
		do
		{
		status_master = Lpspi_Ip_GetStatus(MASTER_EXTERNAL_DEVICE.Instance);
		}
		while(status_master != LPSPI_IP_IDLE);
		
		While the transmit function returns SUCCESS, the GetStatus function keeps returning LPSPI_IP_BUSY, and I don’t see anything on the bus (clock is idle).

ARTD-26218	New Feature	[ETH] Rework Eth_ReportTransmission to clean-up multiple frames instead of one by one NewWorkDescription: Currently the Eth_ReportTransmission checks the sent frames one by one. For some IPs like NETC this affects the performance, as for each frame checked the consumer index needs to be read from a register. To improve the performance the consumer index should be read once, and if multiple packets were sent we don't have to read it for each descriptor we handle Requirement source: Performance measurement for NETC

ID	Subtype	Headline and Description
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Rework Eth_ReportTransmission to allow handling multiple frames
ARTD-26246	Bug	[Canif] IgnoreComponentSuffix behavior is not as expectation Detailed description (how to reproduce it): create s32z project enable IgnoreComponentSuffix check the name of added components (CAN and CANIF) => the component name should not be suffixed automatically Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: !image-2022-05-05-09-44-18-175.png|thumbnail! [...] Expected behavior: !image-2022-05-05-09-45-17-270.png|thumbnail! [...] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-26253	Bug	[LIN-S32ZSE] Can't generate code - undefine variable "LinMasterNodeUsed" in EB tresos Detailed description (how to reproduce it): when generate code be used SLAVE node for module LIN. In EB tresos variable "LinMasterNodeUsed " undefine in file Lin_Defines !image-2022-05-05-11-20-17-475.png! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional Lin_TC_FCT_0215,Lin_TC_FCT_0216,Lin_TC_FCT_0217,Lin_TC_FCT_0218,Lin_TC_FCT_0219 Observed behavior: N/A Expected behavior: define variable "LinMasterNodeUsed " Proposed solution optional: N/A
ARTD-26271	Bug	Spi_SetupEB should follow AUTOSAR spec Detailed description (how to reproduce it): In current spi driver, the API is Std_ReturnType Spi_SetupEB (Spi_ChannelType Channel, Spi_DataBufferType SrcDataBufferPtr, Spi_DataBufferType DesDataBufferPtr, Spi_NumberOfDataType Length); Spi_DataBufferType SrcDataBufferPtr should be const Spi_DataBufferType SrcDataBufferPtr to follow AUTOSAR spec !image-2022-05-05-17-18-49-897.png|width=504,height=239! Preconditions: None Test Case ID (internal TC that caught the defect) optional: None Observed behavior: wrong API prototype Expected behavior: correct API prototype Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]

ID	Subtype	Headline and Description
ARTD-26299	New Feature	[dio] Add support for multiple configurations in S32CT NewWorkDescription: Previously, we limited the number of functional groups (multiple configurations) to 1. Now we want to add support for multiple configurations (in the form of post-build variants and classic functional groups) so that we can also replicate in S32CT the module configurations from EB Tresos in which there are two or more post-build variants. Requirement source: SRS_BSW_00405 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Read the newly introduced section ("13. Multiple Configurations") in the following presentation: [NXP_RTD_AUTOSAR_S32CT_V2|https://nxp1.sharepoint.com/:p:/r/sites/Zebra/Shared%20Documents/Configuration%20(code,%20DS,%20EB,%20etc)/NXP_RTD_AUTOSAR_S32CT_V2.pptx?d=w28a51ba3c45240f5abb6a6933671dcd3and csf=1and web=1and e=r7YbFB] Make sure you have added the SYNC_VALUE and SYNC_SIZE options to each component setting that requires it (as suggested in the presentation, you can use [epdimp2ct|https://nxp1.sharepoint.com/:f:/s/Zebra/EprWldG7fTNPgdoXtWTgmIsBn1Xo56J_qJQGC37m92ax9A?e=ivi58c] to automate this task as long as you have a matching EPD for your component) Apply the changes shown in the source code references. The main points are: {noformat} [!IF "var:defined('postBuildVariant')"!]{noformat} translates to {noformat} if (useFgInfix()){noformat} The file names and configuration structures should not contain the variant infix when there's no selected post-build variant (actually, most of the changes you'll see in the references are done just to support this use-case) Sanity Check: Check that your driver can be successfully built in all three scenarios: ## ComponentGenerationMethod = EcucPostBuildVariants and VARIANT_NO = 0 ## ComponentGenerationMethod = EcucPostBuildVariants and VARIANT_NO > 1 (VARIANT_NO == 1 is the base case from which you started) ## ComponentGenerationMethod = FunctionalGroups (with at least 2 FGs configured in the project) Extensive Testing: By default, EB Tresos cannot generate variant aware EPC files without throwing a null pointer exception. To get around this limitation, the dreisoft.tresos.autosar2 plugin must be patched. To do this, download one of the two attached archives, depending on your EBT version, and extract it in <ebt_root_dir>/plugins. # Go to your driver's plugin, open {_}plugin.xml{_}, search for {_}id="EPCGenerator"{_}, and set the value of parameter "allVariants" to {_}"true"{_}. This will enable variant aware EPC file generation for your module. # Create a configuration with at least two post-build variants. The project will need EcuC for post-build variants to be enabled. The configuration should ideally use different values for nodes with postBuildVariantValue=TRUE and different list elements for nodes with postBuildVariantMultiplicity=TRUE. This will ensure that errors in variant handling can be easily caught later on while doing the diffs. # Generate the EPC files by right clicking on your project's configuration and select "Generate Project". If the 1st step has been carried out correctly, there will be a "PostBuildSelectableVariants.arxml" file in <project_root_dir>/output/output and your module's EPC file, lying inside the same directory, will contain lots of "VARIATION-POINT" elements. # Open S32DS and select File > Import > S32 Configuration Tools > Import ECU Configuration. Browse to the location where your EPC files are located within the EBT project and select at least your module's EPC file, EcuC's EPC file and PostBuildSelectableVariants.arxml. Check the "Import the configuration as a new one" radio button and select the processor / package / sdk version accordingly. # Go to the "Global settings" panel and make sure that ComponentGenerationMethod=EcucPostBuildVariants in the System component. # Select "Update Code" to generate your S32DS project files. Perform a comparison/diff of the generated files between S32DS and EBT. These files should be pretty much identical (sans whitespace or formatting) # Go to the "Global settings" panel again to generate the ECVD files (EcvdGenerationMethod=INDIVIDUAL, browse a location for EcvdOutputPath and click on the Generate Configuration button). Compare/diff your module's EPC file with your module's ECVD file. These files should be pretty much identical (sans whitespace or formatting) # The above steps tested the EBT->S32DS compatibility. To also test the S32DS->EBT compatibility, do the above steps in reverse. That is, start from an S32DS project with post-build variants, generate the ECVD files, import them into EBT, compare again the generated files and the configuration files.
ARTD-26309	New Feature	[i2c] Add support for multiple configurations in S32CT NewWorkDescription: Previously, we limited the number of functional groups (multiple configurations) to 1. Now we want to add support for multiple configurations (in the form of post-build variants and classic functional groups) so that we can also replicate in S32CT the module configurations from EB Tresos in which there are two or more post-build variants. Requirement source: SRS_BSW_00405 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Read the newly introduced section ("13. Multiple Configurations") in the following presentation: [NXP_RTD_AUTOSAR_S32CT_V2|https://nxp1.sharepoint.com/:p:/r/sites/Zebra/Shared%20Documents/Configuration%20(code,%20DS,%20EB,%20etc)/NXP_RTD_AUTOSAR_S32CT_V2.pptx?d=w28a51ba3c45240f5abb6a6933671dcd3and csf=1and web=1and e=r7YbFB] Make sure you have added the SYNC_VALUE and SYNC_SIZE options to each component setting that requires it (as suggested in the presentation, you can use [epdimp2ct|https://nxp1.sharepoint.com/:f:/s/Zebra/EprWldG7fTNPgdoXtWTgmIsBn1Xo56J_qJQGC37m92ax9A?e=ivi58c] to automate this task as long as you have a matching EPD for your component) Apply the changes shown in the source code references. The main points are:

ID	Subtype	Headline and Description
		{noformat} [!IF "var:defined('postBuildVariant')"!]{noformat} translates to {noformat} if (useFgInfix()){noformat} The file names and configuration structures should not contain the variant infix when there's no selected post-build variant (actually, most of the changes you'll see in the references are done just to support this use-case) Sanity Check: Check that your driver can be successfully built in all three scenarios: ## ComponentGenerationMethod = EcucPostBuildVariants and VARIANT_NO = 0 ## ComponentGenerationMethod = EcucPostBuildVariants and VARIANT_NO > 1 (VARIANT_NO == 1 is the base case from which you started) ## ComponentGenerationMethod = FunctionalGroups (with at least 2 FGs configured in the project) Extensive Testing: By default, EB Tresos cannot generate variant aware EPC files without throwing a null pointer exception. To get around this limitation, the dreisoft.tresos.autosar2 plugin must be patched. To do this, download one of the two attached archives, depending on your EBT version, and extract it in <ebt_root_dir>/plugins. # Go to your driver's plugin, open {_}plugin.xml{_}, search for {_}id="EPCGenerator"{_}, and set the value of parameter "allVariants" to {_}"true"{_}. This will enable variant aware EPC file generation for your module. # Create a configuration with at least two post-build variants. The project will need EcuC for post-build variants to be enabled. The configuration should ideally use different values for nodes with postBuildVariantValue=TRUE and different list elements for nodes with postBuildVariantMultiplicity=TRUE. This will ensure that errors in variant handling can be easily caught later on while doing the diffs. # Generate the EPC files by right clicking on your project's configuration and select "Generate Project". If the 1st step has been carried out correctly, there will be a "PostBuildSelectableVariants.arxml" file in <project_root_dir>/output/output and your module's EPC file, lying inside the same directory, will contain lots of "VARIATION-POINT" elements. # Open S32DS and select File > Import > S32 Configuration Tools > Import ECU Configuration. Browse to the location where your EPC files are located within the EBT project and select at least your module's EPC file, EcuC's EPC file and PostBuildSelectableVariants.arxml. Check the "Import the configuration as a new one" radio button and select the processor / package / sdk version accordingly. # Go to the "Global settings" panel and make sure that ComponentGenerationMethod=EcucPostBuildVariants in the System component. # Select "Update Code" to generate your S32DS project files. Perform a comparison/diff of the generated files between S32DS and EBT. These files should be pretty much identical (sans whitespace or formatting) # Go to the "Global settings" panel again to generate the ECVD files (EcvdGenerationMethod=INDIVIDUAL, browse a location for EcvdOutputPath and click on the Generate Configuration button). Compare/diff your module's EPC file with your module's ECVD file. These files should be pretty much identical (sans whitespace or formatting) # The above steps tested the EBT->S32DS compatibility. To also test the S32DS->EBT compatibility, do the above steps in reverse. That is, start from an S32DS project with post-build variants, generate the ECVD files, import them into EBT, compare again the generated files and the configuration files.
ARTD-26310	New Feature	[i2s] Add support for multiple configurations in S32CT NewWorkDescription: Previously, we limited the number of functional groups (multiple configurations) to 1. Now we want to add support for multiple configurations (in the form of post-build variants and classic functional groups) so that we can also replicate in S32CT the module configurations from EB Tresos in which there are two or more post-build variants. Requirement source: SRS_BSW_00405 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Read the newly introduced section ("13. Multiple Configurations") in the following presentation: [NXP_RTD_AUTOSAR_S32CT_V2|https://nxp1.sharepoint.com/:p:/r/sites/Zebra/Shared%20Documents/Configuration%20(code,%20DS,%20EB,%20etc)/NXP_RTD_AUTOSAR_S32CT_V2.pptx?d=w28a51ba3c45240f5abb6a6933671dcd3and csf=1and web=1and e=r7YbFB] Make sure you have added the SYNC_VALUE and SYNC_SIZE options to each component setting that requires it (as suggested in the presentation, you can use [epdimp2ct|https://nxp1.sharepoint.com/:f:/s/Zebra/EprWldG7fTNPgdoXtWTgmIsBn1Xo56J_qJQGC37m92ax9A?e=ivi58c] to automate this task as long as you have a matching EPD for your component) Apply the changes shown in the source code references. The main points are: {noformat} [!IF "var:defined('postBuildVariant')"!]{noformat} translates to {noformat} if (useFgInfix()){noformat} The file names and configuration structures should not contain the variant infix when there's no selected post-build variant (actually, most of the changes you'll see in the references are done just to support this use-case) Sanity Check: Check that your driver can be successfully built in all three scenarios: ## ComponentGenerationMethod = EcucPostBuildVariants and VARIANT_NO = 0 ## ComponentGenerationMethod = EcucPostBuildVariants and VARIANT_NO > 1 (VARIANT_NO == 1 is the base case from which you started) ## ComponentGenerationMethod = FunctionalGroups (with at least 2 FGs configured in the project) Extensive Testing: By default, EB Tresos cannot generate variant aware EPC files without throwing a null pointer exception. To get around this limitation, the dreisoft.tresos.autosar2 plugin must be patched. To do this, download one of the two attached archives, depending on your EBT version, and extract it in <ebt_root_dir>/plugins.

ID	Subtype	Headline and Description
		# Go to your driver's plugin, open {_}plugin.xml{_}, search for {_}id="EPCGenerator"{_}, and set the value of parameter "allVariants" to {_}"true"{_}. This will enable variant aware EPC file generation for your module. # Create a configuration with at least two post-build variants. The project will need EcuC for post-build variants to be enabled. The configuration should ideally use different values for nodes with postBuildVariantValue=TRUE and different list elements for nodes with postBuildVariantMultiplicity=TRUE. This will ensure that errors in variant handling can be easily caught later on while doing the diffs. # Generate the EPC files by right clicking on your project's configuration and select "Generate Project". If the 1st step has been carried out correctly, there will be a "PostBuildSelectableVariants.arxml" file in <project_root_dir>/output/output and your module's EPC file, lying inside the same directory, will contain lots of "VARIATION-POINT" elements. # Open S32DS and select File > Import > S32 Configuration Tools > Import ECU Configuration. Browse to the location where your EPC files are located within the EBT project and select at least your module's EPC file, EcuC's EPC file and PostBuildSelectableVariants.arxml. Check the "Import the configuration as a new one" radio button and select the processor / package / sdk version accordingly. # Go to the "Global settings" panel and make sure that ComponentGenerationMethod=EcucPostBuildVariants in the System component. # Select "Update Code" to generate your S32DS project files. Perform a comparison/diff of the generated files between S32DS and EBT. These files should be pretty much identical (sans whitespace or formatting) # Go to the "Global settings" panel again to generate the ECVD files (EcvdGenerationMethod=INDIVIDUAL, browse a location for EcvdOutputPath and click on the Generate Configuration button). Compare/diff your module's EPC file with your module's ECVD file. These files should be pretty much identical (sans whitespace or formatting) # The above steps tested the EBT->S32DS compatibility. To also test the S32DS->EBT compatibility, do the above steps in reverse. That is, start from an S32DS project with post-build variants, generate the ECVD files, import them into EBT, compare again the generated files and the configuration files.
ARTD-26326	New Feature	[qd] Add support for multiple configurations in S32CT NewWorkDescription: Previously, we limited the number of functional groups (multiple configurations) to 1. Now we want to add support for multiple configurations (in the form of post-build variants and classic functional groups) so that we can also replicate in S32CT the module configurations from EB Tresos in which there are two or more post-build variants. Requirement source: SRS_BSW_00405 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Read the newly introduced section ("13. Multiple Configurations") in the following presentation: [NXP_RTD_AUTOSAR_S32CT_V2|https://nxp1.sharepoint.com/:p:/r/sites/Zebra/Shared%20Documents/Configuration%20(code,%20DS,%20EB,%20etc)/NXP_RTD_AUTOSAR_S32CT_V2.pptx?d=w28a51ba3c45240f5abb6a6933671dcd3and csf=1and web=1and e=r7YbFB] Make sure you have added the SYNC_VALUE and SYNC_SIZE options to each component setting that requires it (as suggested in the presentation, you can use [epdimp2ct|https://nxp1.sharepoint.com/:f:/s/Zebra/EprWldG7fTNPgdoXtWTgmIsBn1Xo56J_qJQGC37m92ax9A?e=ivi58c] to automate this task as long as you have a matching EPD for your component) Apply the changes shown in the source code references. The main points are: {noformat} [!IF "var:defined('postBuildVariant')"!]{noformat} translates to {noformat} if (useFgInfix()){noformat} The file names and configuration structures should not contain the variant infix when there's no selected post-build variant (actually, most of the changes you'll see in the references are done just to support this use-case) Sanity Check: Check that your driver can be successfully built in all three scenarios: ## ComponentGenerationMethod = EcucPostBuildVariants and VARIANT_NO = 0 ## ComponentGenerationMethod = EcucPostBuildVariants and VARIANT_NO > 1 (VARIANT_NO == 1 is the base case from which you started) ## ComponentGenerationMethod = FunctionalGroups (with at least 2 FGs configured in the project) Extensive Testing: By default, EB Tresos cannot generate variant aware EPC files without throwing a null pointer exception. To get around this limitation, the dreisoft.tresos.autosar2 plugin must be patched. To do this, download one of the two attached archives, depending on your EBT version, and extract it in <ebt_root_dir>/plugins. # Go to your driver's plugin, open {_}plugin.xml{_}, search for {_}id="EPCGenerator"{_}, and set the value of parameter "allVariants" to {_}"true"{_}. This will enable variant aware EPC file generation for your module. # Create a configuration with at least two post-build variants. The project will need EcuC for post-build variants to be enabled. The configuration should ideally use different values for nodes with postBuildVariantValue=TRUE and different list elements for nodes with postBuildVariantMultiplicity=TRUE. This will ensure that errors in variant handling can be easily caught later on while doing the diffs. # Generate the EPC files by right clicking on your project's configuration and select "Generate Project". If the 1st step has been carried out correctly, there will be a "PostBuildSelectableVariants.arxml" file in <project_root_dir>/output/output and your module's EPC file, lying inside the same directory, will contain lots of "VARIATION-POINT" elements. # Open S32DS and select File > Import > S32 Configuration Tools > Import ECU Configuration. Browse to the location where your EPC files are located within the EBT project and select at least your module's EPC file, EcuC's EPC file and PostBuildSelectableVariants.arxml. Check the "Import the configuration as a new one" radio button and select the processor / package / sdk version accordingly. # Go to the "Global settings" panel and make sure that ComponentGenerationMethod=EcucPostBuildVariants in the System component. # Select "Update Code" to generate your S32DS project files. Perform a comparison/diff of the generated files between S32DS and EBT. These files should be pretty much identical (sans whitespace or formatting) # Go to the "Global settings" panel again to generate the ECVD files (EcvdGenerationMethod=INDIVIDUAL, browse a location for EcvdOutputPath and click on the Generate Configuration button). Compare/diff your module's EPC file with your module's ECVD file. These files should be pretty much identical (sans whitespace or formatting)

ID	Subtype	Headline and Description
		# The above steps tested the EBT->S32DS compatibility. To also test the S32DS->EBT compatibility, do the above steps in reverse. That is, start from an S32DS project with post-build variants, generate the ECVD files, import them into EBT, compare again the generated files and the configuration files.
ARTD-26330	New Feature	[spi] Add support for multiple configurations in S32CT NewWorkDescription: Previously, we limited the number of functional groups (multiple configurations) to 1. Now we want to add support for multiple configurations (in the form of post-build variants and classic functional groups) so that we can also replicate in S32CT the module configurations from EB Tresos in which there are two or more post-build variants. Requirement source: SRS_BSW_00405 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Read the newly introduced section ("13. Multiple Configurations") in the following presentation: [NXP_RTD_AUTOSAR_S32CT_V2|https://nxp1.sharepoint.com/:p:/r/sites/Zebra/Shared%20Documents/Configuration%20(code,%20DS,%20EB,%20etc)/NXP_RTD_AUTOSAR_S32CT_V2.pptx?d=w28a51ba3c45240f5abb6a6933671dcd3and csf=1and web=1and e=r7YbFB] Make sure you have added the SYNC_VALUE and SYNC_SIZE options to each component setting that requires it (as suggested in the presentation, you can use [epdimp2ct|https://nxp1.sharepoint.com/:f:/s/Zebra/EprWldG7fTNPgdoXtWTgmIsBn1Xo56J_qJQGC37m92ax9A?e=ivi58c] to automate this task as long as you have a matching EPD for your component) Apply the changes shown in the source code references. The main points are: {noformat} [!IF "var:defined('postBuildVariant')"!]{noformat} translates to {noformat} if (useFgInfix()){noformat} The file names and configuration structures should not contain the variant infix when there's no selected post-build variant (actually, most of the changes you'll see in the references are done just to support this use-case) Sanity Check: Check that your driver can be successfully built in all three scenarios: ## ComponentGenerationMethod = EcucPostBuildVariants and VARIANT_NO = 0 ## ComponentGenerationMethod = EcucPostBuildVariants and VARIANT_NO > 1 (VARIANT_NO == 1 is the base case from which you started) ## ComponentGenerationMethod = FunctionalGroups (with at least 2 FGs configured in the project) Extensive Testing: By default, EB Tresos cannot generate variant aware EPC files without throwing a null pointer exception. To get around this limitation, the dreisoft.tresos.autosar2 plugin must be patched. To do this, download one of the two attached archives, depending on your EBT version, and extract it in <ebt_root_dir>/plugins. # Go to your driver's plugin, open {_}plugin.xml{_}, search for {_}id="EPCGenerator"{_}, and set the value of parameter "allVariants" to {_}"true"{_}. This will enable variant aware EPC file generation for your module. # Create a configuration with at least two post-build variants. The project will need EcuC for post-build variants to be enabled. The configuration should ideally use different values for nodes with postBuildVariantValue=TRUE and different list elements for nodes with postBuildVariantMultiplicity=TRUE. This will ensure that errors in variant handling can be easily caught later on while doing the diffs. # Generate the EPC files by right clicking on your project's configuration and select "Generate Project". If the 1st step has been carried out correctly, there will be a "PostBuildSelectableVariants.arxml" file in <project_root_dir>/output/output and your module's EPC file, lying inside the same directory, will contain lots of "VARIATION-POINT" elements. # Open S32DS and select File > Import > S32 Configuration Tools > Import ECU Configuration. Browse to the location where your EPC files are located within the EBT project and select at least your module's EPC file, EcuC's EPC file and PostBuildSelectableVariants.arxml. Check the "Import the configuration as a new one" radio button and select the processor / package / sdk version accordingly. # Go to the "Global settings" panel and make sure that ComponentGenerationMethod=EcucPostBuildVariants in the System component. # Select "Update Code" to generate your S32DS project files. Perform a comparison/diff of the generated files between S32DS and EBT. These files should be pretty much identical (sans whitespace or formatting) # Go to the "Global settings" panel again to generate the ECVD files (EcvdGenerationMethod=INDIVIDUAL, browse a location for EcvdOutputPath and click on the Generate Configuration button). Compare/diff your module's EPC file with your module's ECVD file. These files should be pretty much identical (sans whitespace or formatting) # The above steps tested the EBT->S32DS compatibility. To also test the S32DS->EBT compatibility, do the above steps in reverse. That is, start from an S32DS project with post-build variants, generate the ECVD files, import them into EBT, compare again the generated files and the configuration files.
ARTD-26332	New Feature	[uart] Add support for multiple configurations in S32CT NewWorkDescription: Previously, we limited the number of functional groups (multiple configurations) to 1. Now we want to add support for multiple configurations (in the form of post-build variants and classic functional groups) so that we can also replicate in S32CT the module configurations from EB Tresos in which there are two or more post-build variants. Requirement source: SRS_BSW_00405 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Read the newly introduced section ("13. Multiple Configurations") in the following presentation: [NXP_RTD_AUTOSAR_S32CT_V2|https://nxp1.sharepoint.com/:p:/r/sites/Zebra/Shared%20Documents/Configuration%20(code,%20DS,%20EB,%20etc)/NXP_RTD_AUTOSAR_S32CT_V2.pptx?d=w28a51ba3c45240f5abb6a6933671dcd3and csf=1and web=1and e=r7YbFB] Make sure you have added the SYNC_VALUE and SYNC_SIZE options to each component setting that requires it (as suggested in the presentation, you can use [epdimp2ct|https://nxp1.sharepoint.com/:f:/s/Zebra/

ID	Subtype	Headline and Description
		EprWldG7fTNPgdoXtWTgmIsBn1Xo56J_qJQGC37m92ax9A?e=ivi58c] to automate this task as long as you have a matching EPD for your component) Apply the changes shown in the source code references. The main points are: {noformat} [!IF "var:defined('postBuildVariant')"!]{noformat} translates to {noformat} if (useFgInfix()){noformat} The file names and configuration structures should not contain the variant infix when there's no selected post-build variant (actually, most of the changes you'll see in the references are done just to support this use-case) Sanity Check: Check that your driver can be successfully built in all three scenarios: ## ComponentGenerationMethod = EcucPostBuildVariants and VARIANT_NO = 0 ## ComponentGenerationMethod = EcucPostBuildVariants and VARIANT_NO > 1 (VARIANT_NO == 1 is the base case from which you started) ## ComponentGenerationMethod = FunctionalGroups (with at least 2 FGs configured in the project) Extensive Testing: By default, EB Tresos cannot generate variant aware EPC files without throwing a null pointer exception. To get around this limitation, the dreisoft.tresos.autosar2 plugin must be patched. To do this, download one of the two attached archives, depending on your EBT version, and extract it in <ebt_root_dir>/plugins. # Go to your driver's plugin, open {_}plugin.xml{_}, search for {_}id="EPCGenerator"{_}, and set the value of parameter "allVariants" to {_}"true"{_}. This will enable variant aware EPC file generation for your module. # Create a configuration with at least two post-build variants. The project will need EcuC for post-build variants to be enabled. The configuration should ideally use different values for nodes with postBuildVariantValue=TRUE and different list elements for nodes with postBuildVariantMultiplicity=TRUE. This will ensure that errors in variant handling can be easily caught later on while doing the diffs. # Generate the EPC files by right clicking on your project's configuration and select "Generate Project". If the 1st step has been carried out correctly, there will be a "PostBuildSelectableVariants.arxml" file in <project_root_dir>/output/output and your module's EPC file, lying inside the same directory, will contain lots of "VARIATION-POINT" elements. # Open S32DS and select File > Import > S32 Configuration Tools > Import ECU Configuration. Browse to the location where your EPC files are located within the EBT project and select at least your module's EPC file, EcuC's EPC file and PostBuildSelectableVariants.arxml. Check the "Import the configuration as a new one" radio button and select the processor / package / sdk version accordingly. # Go to the "Global settings" panel and make sure that ComponentGenerationMethod=EcucPostBuildVariants in the System component. # Select "Update Code" to generate your S32DS project files. Perform a comparison/diff of the generated files between S32DS and EBT. These files should be pretty much identical (sans whitespace or formatting) # Go to the "Global settings" panel again to generate the ECVD files (EcvdGenerationMethod=INDIVIDUAL, browse a location for EcvdOutputPath and click on the Generate Configuration button). Compare/diff your module's EPC file with your module's ECVD file. These files should be pretty much identical (sans whitespace or formatting) # The above steps tested the EBT->S32DS compatibility. To also test the S32DS->EBT compatibility, do the above steps in reverse. That is, start from an S32DS project with post-build variants, generate the ECVD files, import them into EBT, compare again the generated files and the configuration files.
ARTD-26350	New Feature	[SPI] Add a note in UM to guide users configure correctly the priority of DMA ISR NewWorkDescription: In DMA mode, the priority of DMA ISR which is configured by Platform should be follow the order of priority as below: DMA ISR of TX channel is more important than DMA ISR of RX channel For more details, see the email in attachment Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Add a note in UM to guide users configure correctly the priority of DMA ISR (Driver usage and configuration tips)
ARTD-26385	Bug	[PORT] The dedicated signals routed to the same pin at once cause an hardfault Detailed description (how to reproduce it): Configuring the dedicated signals in the same pin. The warning was throwed as the capture below: !image-2022-05-10-17-30-10-708.png|width=1249,height=201! Continue to use the configuration to compile. Because of the NUM_OF_CONFIGURED_PINS0 is 2 but the array of pins in Siul2_Port_Ip_Cfg.c have only 1 signals, the hardfault will be thrown when parsing the array. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: The warning was thrown, but user can still using the configuration and in runtime, they will get the hardfault state. Expected behavior: No hardfault or blocking user to continue the configuration.

ID	Subtype	Headline and Description
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: There are 2 options as below: Raising an error in this situation to avoid using the configuration. Do the same like K3, user can configure the dedicated in same pins, but there is no generation code was generated (NUM_OF_CONFIGURED_PINS0 = 0, there's no arrays of pins in Siul2_Port_Ip_Cfg.c)
ARTD-26448	New Feature	[mcu] CPR_RTD_00028 is replaced by CPR_RTD_00664 NewWorkDescription: Requirement CPR_RTD_00028 is rejected and replaced by CPR_RTD_00664: "ISR shall check whether its respective interrupt status flag and interrupt enable flag are set; i.e. whether the interrupt is not spurious. If at least one of the flags is not set, then the interrupt is spurious and the ISR shall not treat the event (return immediately). Rationale: A spurious interrupt is considered when there is a glitch in the hardware mechanisms that trigger the CPU exception. Drivers must ensure only relevant events are treated and spurious ones are discarded." Check and update (if needed) the driver implementation to be according to CPR_RTD_00664. Update FMEA and include also Adrian Herea (Safety Manager) to the review. Requirement source: cPRT {*}Proposed solution{*}: The rationale behind the update is to not consider as spurious the events that may otherwise be treated in a different context (e.g. polling mode) and for which the interrupt enable bit is not set. For interrupt lines shared by multiple events, it should be possible for some of the events to be treated by the ISR, while other to be left uncleared so that they can be treated in the thread (as long as they have the interrupt enabled bit cleared, meaning the interrupt is disabled). The update required here would be to only clear the flags for the events that are enabled and leave the others untouched, as long as they are managed by the driver in a different context. The consequence of that would be to allow multiple events that have separate interrupt enable bits but share a single interrupt line to function both in "interrupt" and "polling" modes simultaneously. Hence, the clearing of the flag should only happen if the condition "(interrupt_enable and and interrupt_flag)" is true, otherwise it is assumed that the flag is handled outside the ISR. The rationale applies only for drivers that handle hardware resources that share interrupt lines, but the correct implementation of the ISR logic should be checked for all drivers.
ARTD-26639	Bug	[Mcu] [S32DS] Name of McuModeSettingConf is generated as hardcode Detailed description (how to reproduce it): If user renamed of Functional groups in ClockTool (equivalent with node Name of McuModeSettingConf), it will not generated as expectation. !image-2022-05-20-16-56-43-735.png! This name will be used as parameter of Mcu_InitClock Api. Currently, driver is generating the hard code as below (in Mcu_Cfg.h) !image-2022-05-20-16-59-34-235.png! Preconditions: Renamed McuClockSettingConfig and try to use that name in Mcu_InitClock Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Driver is hardcoding the name of McuClockSettingConfig Expected behavior: Should use name of node instead hardcoding Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-26809	Bug	[LIN][S32K1XX/S32K3XX] Lin driver is still waking up with signal < 150 us Detailed description (how to reproduce it): In Lin Specification Package, Both Master and Slave node shall detect the wake up signal (a dominant pulse longer than 150 us) But the Lin driver is still waking up with signal < 150 us. Preconditions: First, Call Lin_GoToSleep or Lin_GoToSleepInternal function.

ID	Subtype	Headline and Description
		Then, Send the wake up pulse < 150 us Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Lin driver is still waking up with signal < 150 us Expected behavior: Lin driver is only waking up with signal > 150 us Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Solution for this issue is to use timer and UART edge detection interrupts to detect and measure the wake up signal.
ARTD-26817	Bug	[MCU] When setting Mcu Register Values Optimization is enable, mcu can not init Detailed description (how to reproduce it): When setting Mcu Register Values Optimization is enable, MCU can not init !screenshot-1.png|thumbnail! Issue2: Improvement generation script: 2 list MCU.CMU_FC.Address.List}} and {{MCU.CMU_FC.List}} contain same information: cmuInstance and cmuMonitorname.-> using string operation and add function macro to use 1 list only try "constrains" to avoid using multiple #IF to check McuCmuName Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: hardfault when call MCU_initclock() Expected behavior: MCU_initclock() run fine Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc. Proposed solution optional: [...]
ARTD-26849	Bug	[Platform] The exception entry in linker files for R52 and A53 should be aligned with 2^11 Detailed description (how to reproduce it): The exception entry in linker files for R52 and A53 should be aligned with 2^11 !image-2022-05-30-15-33-26-310.png! It should be pushed at top of a memory region to avoid wasting memory Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: NA Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-27105	Bug	[BASE] Generate fail in Soc_Ips.h in S32CT Detailed description (how to reproduce it): Create new project in S32CT for S32K148 There is an error in Code Preview window relate to Soc_Ips.h Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Please see attach image. Expected behavior:

ID	Subtype	Headline and Description
		Generate pass Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-27723	Bug	[S32K1][ICU] The buffer of timestamp is possible to overflow if the DMA is used. Detailed description (how to reproduce it): From Icu_TimestampDmaProcessing(), we can see that the DMA is controlled by software. For timestamp mode of ICU, if the DMA is used, the DMA will keep transfer captured value to the buffer until the software take some actions, e.g., stopping DMA in case of LINEAR_BUFFER or reconfiguring DMA in case of CIRCULAR_BUFFER. This method will lead to buffer overflow if the input is a high frequency signal because the DMA continually transfer the data to out of buffer until the software stop DMA or reconfigure DMA. It takes time from ISR is responded to the software take actions to deal DMA, and during this gap, input signal is active and DMA still works. As tested, just 5 kHz will lead overflow. Customer FuTe reported this issue. Preconditions: Set a ICU channel working timestamp mode and enable DMA. Then input a high frequency signal greater than 5 kHz. Test Case ID (internal TC that caught the defect) optional: None Observed behavior: Buffer overflow Expected behavior: Buffer don't overflow Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: See ARTD-27721 for more details.
ARTD-27729	Bug	[PORT]Symbolic name of PortPin generated wrong Detailed description (how to reproduce it): The issue with the symbolic names of PortPinId according to TPS_ECUC_02108 in AUTOSAR_TPS_ECUConfiguration.pdf. The sample port configuratiuon this leads to the generation result: #define PortConfigSet_PortContainer_0_PortPin_0 0 But according to TPS_ECUC_02108, it must be like this: #define PortConf_PortPin_PortPin_0 0 The short name EcucParamConfContainerDef of the declaring module is PortPin and the short name of the EcuCContainerValue container, which holds the symbolicNameValue configuration parameter value, is PortPin_0 Preconditions: the name generated of the Port module must be: #define PortConf_PortPin_PortPin_0 0 Test Case ID (internal TC that caught the defect) optional: None Observed behavior: #define PortConfigSet_PortContainer_0_PortPin_0 0 Expected behavior: #define PortConf_PortPin_PortPin_0 0 Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: The current name is a deviation from Autosar and must be marked if it shall be used. Add the constraint for duplication of Port Pin Name
ARTD-27899	New Feature	[ICU] Create test case in case interrupt second channel raised while first channel is processing interrupt NewWorkDescription: In the function interrupt of Port_Ci harware, there is mistake in processing interrupt in case many channel occurs interrupt at the same time. The details is in ticket https://jira.sw.nxp.com/browse/ARTD-27004. Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Create testcase for 2 case : # Generating pulse for the two or more channel Port_Ci in test case.

ID	Subtype	Headline and Description
		#create test case multi-channel interrupt flag simultaneously set to 1 after interrupt handling function ISR, interrupt flag of some channels (interrupt status flag = 1, interrupt enable flag = 0)is still kept
ARTD-27904	New Feature	[platform] [S32CT] Update code generation to work in JS strict mode NewWorkDescription: Note: Only Peripherals Tool's components are affected{color}*. There has been an update to the System module: System scripts now run in strict mode and contain include guards to protect against multiple inclusion System utilities / functions have been made read-only Code templates are now evaluated in strict mode so that developers can be aware of overwriting existing variables or using undeclared variables*. Requirement source: Consequence of https://jira.sw.nxp.com/browse/ARTD-23864 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: What you have to do: ## Open S32DS's logging file (swtools.log) or launch the program with console logging activated (e.g. run ".\s32ds.exe consoleLog in CMD/PowerShell) ## Create a new project and instantiate your high-level / low-level component. ## Fix the encountered errors related to your component. Most of them will be along the lines of "ReferenceError: <variable_name> is not defined" or "TypeError: <variable_name> is not a writable property of [object global]"_. Both can be resolved by simply defining the offending variable using "var". ## Remove inclusions of "refsSolver.js" in your code templates or code generator scripts. If there's an API you need from that file which is not provided in codegeneratorUtils.js, please let me know.
ARTD-28056	Bug	[WDG] ServiceIdType is not match with AUTOSAR specification. Detailed description (how to reproduce it): [Wrong the service ID with function Wdg_SetTriggerCondition !image-2022-06-23-14-37-32-924.png|thumbnail! and Autosar spec !image-2022-06-23-14-51-09-227.png|thumbnail!] Preconditions: [Autosar specification link https://www.autosar.org/fileadmin/user_upload/standards/classic/19-11/AUTOSAR_SWS_WatchdogDriver.pdf] Test Case ID (internal TC that caught the defect) optional: [Review] Observed behavior: [Wrong WDG service ID] Expected behavior: [Correct the service ID. This should impact to the bswmd report, dox.... So you need to check report again.] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-28103	Bug	[DIO]Symbolic name of DioChannelGroup generated wrong Detailed description (how to reproduce it): The issue with the symbolic names of PortPinId according to TPS_ECUC_02108 in AUTOSAR_TPS_ECUConfiguration.pdf. The sample port configuratiuon this leads to the generation result: #define DioConf_DioChannelGroupIdentification_DioChannelGroup_0* 0 But according to TPS_ECUC_02108, it must be like this: #define DioConf_DioChannelGroup_DioChannelGroup_0* 0 The short name EcucParamConfContainerDef of the declaring module is PortPin and the short name of the EcuCContainerValue container, which holds the symbolicNameValue configuration parameter value, is DioChannelGroup_0 Preconditions: the name generated of the Port module must be: #define DioConf_DioChannelGroup_DioChannelGroup_0 0 Test Case ID (internal TC that caught the defect) optional:

ID	Subtype	Headline and Description
		None Observed behavior: #define DioConf_DioChannelGroup_DioChannelGroup_0 0 Expected behavior: #define DioConf_DioChannelGroup_DioChannelGroup_0 0 Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: The current name is a deviation from Autosar and must be marked if it shall be used.
ARTD-28205	New Feature	[adc] Update copyright template NewWorkDescription: Replace : “(c) Copyright \{year range\} NXP Semiconductors All Rights Reserved.” with : Copyright \{year range\} NXP !screenshot-2.png|thumbnail! to !screenshot-3.png|thumbnail! Requirement source: as described in “Updating Copyright Years” topic on this guide https://confluence.sw.nxp.com/display/OSS/File+Headers+-+Copyright+and+License+Information). (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: as above
ARTD-28210	New Feature	[i2s] Update copyright template NewWorkDescription: Replace : “(c) Copyright \{year range\} NXP Semiconductors All Rights Reserved.” with : Copyright \{year range\} NXP !screenshot-2.png|thumbnail! to !screenshot-3.png|thumbnail! Requirement source: as described in “Updating Copyright Years” topic on this guide https://confluence.sw.nxp.com/display/OSS/File+Headers+-+Copyright+and+License+Information). (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: as above
ARTD-28240	New Feature	[icu] Update copyright template NewWorkDescription: Replace : “(c) Copyright \{year range\} NXP Semiconductors All Rights Reserved.” with : Copyright \{year range\} NXP !screenshot-2.png|thumbnail! to !screenshot-3.png|thumbnail! Requirement source: as described in “Updating Copyright Years” topic on this guide https://confluence.sw.nxp.com/display/OSS/File+Headers+-+Copyright+and+License+Information). (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: as above
ARTD-28760	Bug	[WDG] Add M4 to remove IACR functions and fix traceability matrix warnings Detailed description (how to reproduce it):

ID	Subtype	Headline and Description
		Wdg have some warning relate to IACR, some function only support on SJA1XX platform. So wdg need to add M4 into driver to delete the code relate to IACR Preconditions: [NA] Test Case ID (internal TC that caught the defect) optional: [NA] Observed behavior: [NA] Expected behavior: Traceability report no warning Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Add M4 to remove code relate to IACR
ARTD-41707	Bug	[ETH] EthIf controller index shall be passed into EthIf callbacks in Eth driver NewWorkDescription: With current implementation, Eth controller index is passed into EthIf callbacks (e.g. EthIf_RxIndication, EthIf_TxConfirmation,...) in Eth driver. However, according to the following requirement: !image-2022-07-07-15-18-27-834.png! The EthIf controller index shall be passed into EthIf callbacks in Eth driver, instead of Eth controller index Requirement source: SWS_EthIf_00091, SWS_EthIf_00085, SWS_EthIf_00231 Proposed solution optional: Add a new field in the configuration of the Eth controller that will allow setting the ID of that specific controller in the context of EthIf. The name of the field should be EthCtrlEthIfIdx and it should be placed in the EthCtrlVendorSpecific container to have a consistent approach with PFE Use the new index when calling EthIf functions from the Eth driver context
ARTD-28967	Bug	[Spi] Build faild on HLD with DS due to Spi Configuration gen wrong name Detailed description (how to reproduce it): !image-2022-07-13-16-05-11-141.png|thumbnail! Build faild on HLD with DS due to Spi Configuration gen wrong name: Spi_Config__BOARD_InitPeripherals !image-2022-07-13-16-07-04-158.png|thumbnail! !image-2022-07-13-16-06-09-481.png|thumbnail! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: Spi_TS_D02 cfg set = 1 Observed behavior: NA Expected behavior: NA Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-29287	Bug	[WDG] S32M24X: Fix misra violations Detailed description (how to reproduce it): 22653943|MISRA C-2012 Rule 2.3|1164|Advisory|Low| Quality|Type "AeWdog_Ip_StatusType" is defined but never used.|/ARTD-CIW266-1/sources/1fec4a627c/output/eclipse/plugins/Wdg_TS_T40D2M19I0R0/include/AeWdog_Ip_Types.h|120|Unclassified|New| 22653941|MISRA C-2012 Rule 2.5|1164|Advisory|Low| Quality|Macro "AEWDOG_IP_DATA_WIDTH_16" is defined but never used.|/ARTD-CIW266-1/sources/1fec4a627c/output/eclipse/plugins/Wdg_TS_T40D2M19I0R0/src/AeWdog_Ip.c|109|Unclassified|New| 22653939|MISRA C-2012 Directive 4.5|1076|Advisory|Low| Quality|Declaration with identifier "AEWDOG_IP_WD_PERIOD_TIME_6_2_512", which is ambiguous.|/ARTD-CIW266-1/sources/1fec4a627c/output/eclipse/plugins/Wdg_TS_T40D2M19I0R0/include/AeWdog_Ip_Types.h|172|Unclassified|New| 22653935|MISRA C-2012 Rule 2.3|1164|Advisory|Low| Quality|Unnamed type is defined but never used.|/ARTD-CIW266-1/sources/1fec4a627c/output/eclipse/plugins/Wdg_TS_T40D2M19I0R0/include/AeWdog_Ip_Types.h|114|Unclassified|New| 22653935|MISRA C-2012 Rule 2.3|1164|Advisory|Low| Quality|Unnamed type is defined but never used.|/ARTD-CIW266-1/sources/1fec4a627c/output/eclipse/plugins/Wdg_TS_T40D2M19I0R0/include/AeWdog_Ip_Types.h|105|Unclassified|New| 22653933|MISRA C-2012 Rule 2.5|1164|Advisory|Low| Quality|Macro "AEWDOG_IP_TYPES_MODULE_ID" is defined but never used.|/ARTD-CIW266-1/sources/1fec4a627c/output/eclipse/plugins/Wdg_TS_T40D2M19I0R0/include/AeWdog_Ip_Types.h|54|Unclassified|New|

<table>
<tr><th>ID</th><th>Subtype</th><th>Headline and Description</th></tr>
<tr><td></td><td></td><td>22653927|MISRA C-2012 Rule 2.3|1164|Advisory|Low| Quality|Type "AeWdog_Ip_ConfigType" is defined but never used.|/ARTD-CIW266-1/sources/1fec4a627c/output/eclipse/plugins/Wdg_TS_T40D2M19I0R0/include/AeWdog_Ip_Types.h|204|Unclassified|New|
10960883|MISRA C-2012 Rule 8.5|694|Required|Low| Quality|Symbol "Wdg_Config_0_VS_0" is declared more than once.|/ARTD-CIW266-1/sources/1fec4a627c/output/eclipse/plugins/Wdg_TS_T40D2M19I0R0/include/Wdg_43_Instance0.h|135|Intentional|Dismissed|When detected because multiple configurations are used: The violation is detected as multiple configurations are needed for exposing as much code as possible.
22653926|MISRA C-2012 Rule 2.5|1164|Advisory|Low| Quality|Macro "AEWDOG_IP_MODULE_ID" is defined but never used.|/ARTD-CIW266-1/sources/1fec4a627c/output/eclipse/plugins/Wdg_TS_T40D2M19I0R0/include/AeWdog_Ip.h|55|Unclassified|New|
22653925|MISRA C-2012 Rule 2.3|1164|Advisory|Low| Quality|Type "AeWdog_Ip_LockType" is defined but never used.|/ARTD-CIW266-1/sources/1fec4a627c/output/eclipse/plugins/Wdg_TS_T40D2M19I0R0/include/AeWdog_Ip_Types.h|108|Unclassified|New|

*Reference:

Preconditions:
[...]

Test Case ID (internal TC that caught the defect) optional:
[...]

Observed behavior:
[...]

Expected behavior:
[...]

All violations fix and follow RTD Quality Criteria

Proposed solution optional:
Fix all MISRA, HIS, VSMD violations</td></tr>
<tr><td>ARTD-29347</td><td>Bug</td><td>[Port] Validation error due to missing RecommendedConfiguration

Detailed description (how to reproduce it):
There is reported the following error while running the bswmd file validations in third party Autosar configuration tool:

Constr_4046-In a BswImplementation <BswImplementation_0> the reference RecommendedConfigurations containing EcucModuleConfigurationValues <>, the reference implementConfigVariant is empty or NULL.[Infos] <BswImplementation_0> : </Port_TS_T40D34M20I0R0/Implementations/BswImplementation_0>,<> : </Port/Port>, ERROR,S32K344_Integration_example,paramdef/bswmd_static/Port_Bswmd_original.arxml

The error is reported because there is the following reference to recommended configuration:

    <RECOMMENDED-CONFIGURATION-REF DEST="ECUC-MODULE-CONFIGURATION-VALUES">/Port/Port</RECOMMENDED-CONFIGURATION-REF>

But there is no recommended configuration available in RTD package in Autosar format (only in Tresos proprietary one in Port_TS_T40D34M20I0R0\config_ext\PortRecConfiguration_JtagPins.xdm).

Preconditions:
Using third party configuration editor (not Tresos)

Test Case ID (internal TC that caught the defect) optional:
N/A

Observed behavior:
Error reported during validation of bswmd file

Expected behavior:
No error reported during validation of bswmd file

Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)

Proposed solution optional:
Add the recommended configuration in Autosar format.
For now we have resolved the issue by deleting the reference from Port bswmd file, i.e. removing this line from Port_Bswmd.arxml file:

    <RECOMMENDED-CONFIGURATION-REF DEST="ECUC-MODULE-CONFIGURATION-VALUES">/Port/Port</RECOMMENDED-CONFIGURATION-REF>

So it looks as follows:
!port_recommended_config_reference.png!</td></tr>
<tr><td>ARTD-30102</td><td>Bug</td><td>[ADC] Remove (c) symbol in Adc_Ipw_CfgDefines.h and Adc_Ip_PBcfg.h

Detailed description (how to reproduce it):
(c) symbol is not being removed in Adc_Ipw_CfgDefines.h and Adc_Ip_PBcfg.h

Preconditions:
N/A

Test Case ID (internal TC that caught the defect) optional:
N/A</td></tr>
</table>

ID	Subtype	Headline and Description
		Observed behavior: (c) symbol is not being removed in Adc_Ipw_CfgDefines.h and Adc_Ip_PBcfg.h Expected behavior: Remove all of (c) symbol Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Related to ticket ARTD-28205 Proposed solution optional: N/A
ARTD-31085	Bug	Port Configuration Generated Wrong Value Detailed description (how to reproduce it): Configured the PAD120 as LCU0_LCU0_OUT1 in EBTresos, the generated configuration value for SSS bit fields is wrong. It should be 10. However, it is 6 in the generation configuration file. !image-2022-08-04-11-20-01-380.png!!image-2022-08-04-11-22-27-458.png! !image-2022-08-04-11-21-34-001.png!!image-2022-08-04-11-20-47-962.png! Preconditions: n/a Test Case ID (internal TC that caught the defect) optional: n/a Observed behavior: n/a Expected behavior: n/a Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]

ARTD-34371	Bug	[RTE] Protected areas fail to resume all interrupts
		
		Detailed description (how to reproduce it):
		[
		
		XPSR.I = 0 (global interrupts enabled)
		
		Enter_50()
		msr_ETH_EXCLUSIVE_AREA_50 = 0
		
		Interrupt !HERE! see in code excerpt below
		
		Enter_51()
		msr_ETH_EXCLUSIVE_AREA_51 = 0
		OsIf_SuspendAllInterrupts > XPSR.I = 1
		
		Enter_50()
		msr_ETH_EXCLUSIVE_AREA_50 = 1 (overwrites value above !)
		
		Exit_50
		XPSR.I not changed XPSR.I = 1
		
		Exit_51
		XPSR.I = 0
		
		continues Enter_50()
		OsIf_SuspendAllInterrupts > XPSR.I = 1 but in this state the msr_ETH_EXCLUSIVE_AREA_50 has wrong value 1
		
		Exit_50
		Sees that the interrupt was not disabled by Enter_50 so no reason to restore it and kept unintentionally disabled
		
		void SchM_Enter_Eth_43_PFE_ETH_EXCLUSIVE_AREA_50(void)
		{
		    uint32 u32CoreId = (uint32)OsIf_GetCoreID(); if(0UL == reentry_guard_ETH_EXCLUSIVE_AREA_50[u32CoreId])
		    {
		        #if (defined MCAL_ENABLE_USER_MODE_SUPPORT)
		        msr_ETH_EXCLUSIVE_AREA_50[u32CoreId] = OsIf_Trusted_Call_Return(Eth_43_PFE_schm_read_msr);
		        #else
		        msr_ETH_EXCLUSIVE_AREA_50[u32CoreId] = Eth_43_PFE_schm_read_msr(); /*read MSR (to store interrupts state)*/
		        #endif / MCAL_ENABLE_USER_MODE_SUPPORT /
		        if (ISR_ON(msr_ETH_EXCLUSIVE_AREA_50[u32CoreId])) /*if MSR[EE] = 0, skip calling Suspend/Resume AllInterrupts*/
		        {
		        /
		        !HERE! INTERRUPT HERE CAUSES GLOBAL INTERRUPT (XPSR.I bit) TO BE DISABLED FOREVER
		        /
		            OsIf_SuspendAllInterrupts();

ID	Subtype	Headline and Description

		#ifdef ARM_DS5_C_S32XX
		ASM_KEYWORD(" nop ");/ Compiler fix forces the CSPID instruction to be generated with 02, Ospace are selected*/
		#endif
		}
		}
		reentry_guard_ETH_EXCLUSIVE_AREA_50[u32CoreId]++;
		}
		]
		
		Preconditions:
		[Main thread and interrupt]
		
		Test Case ID (internal TC that caught the defect) optional:
		[]
		
		Observed behavior:
		[global interrupt disabled on ARM v7M XPSR.I > 1 after exiting from protected areas]
		
		Expected behavior:
		[global interrupt enabled on ARM v7M XPSR.I > 0 after exiting from protected areas]
		
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)
		
		Proposed solution optional:
		[
		
		void SchM_Enter_Eth_43_PFE_ETH_EXCLUSIVE_AREA_50(void)
		{
		uint32 u32CoreId = (uint32)OsIf_GetCoreID();
		/ new local variable to preserve XPSR /
		uint32 msr;
		
		if(0UL == reentry_guard_ETH_EXCLUSIVE_AREA_49[u32CoreId])
		{
		#if (defined MCAL_ENABLE_USER_MODE_SUPPORT)
		msr = OsIf_Trusted_Call_Return(Eth_43_PFE_schm_read_msr);
		#else
		msr = Eth_43_PFE_schm_read_msr(); /*read MSR (to store interrupts state)*/
		#endif / MCAL_ENABLE_USER_MODE_SUPPORT /
		if (ISR_ON(msr)) /*if MSR[EE] = 0, skip calling Suspend/Resume AllInterrupts*/
		{
		OsIf_SuspendAllInterrupts();
		#ifdef ARM_DS5_C_S32XX
		ASM_KEYWORD(" nop ");/ Compiler fix forces the CSPID instruction to be generated with 02, Ospace are selected*/
		#endif
		}
		/ preserved global XSPR should be updated after the global interrupt is disabled /
		msr_ETH_EXCLUSIVE_AREA_50[u32CoreId] = msr;
		}
		reentry_guard_ETH_EXCLUSIVE_AREA_50[u32CoreId]++;
		}
		
		]

ARTD-34782	Bug	[UART] Incorrect Dev Assert condition for Oversampling Ratio Detailed description (how to reproduce it): Incorrect Oversampling Ratio Dev Assert condition in: LPUART_UART_IP_DEV_ASSERT(ClockFrequency >= (ExpectedBaud 5U)); Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: !image-2022-08-22-13-55-50-565.png|thumbnail! [...] Expected behavior: because OSR reg value can go to value of 3 (which means the formula value is OSR+1 == 4) LPUART_UART_IP_DEV_ASSERT(ClockFrequency >= (ExpectedBaud 4U)); Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-34830	Bug	[MCU] ModeID is changed index after import EPC Detailed description (how to reproduce it): Step 1: Create 18 mode in ModeSetingConf in EB (picture 2) Step 2: Generate code, import Mcu.epc into s32ct project Step 3: Check ModeSetingConf in s32ct (picture 1)

ID	Subtype	Headline and Description
		Index for every Mode ID had been change. It will cause confuse for user when they modify configuation. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Mcu_TC_FCT_0001 Observed behavior: N/A Expected behavior: mode ID in EB and Ct have the same index Proposed solution optional: N/A
ARTD-34864	Bug	[LIN] LinWakeupNotification is disabled when two or more Lin channel configured Detailed description (how to reproduce it): Configuration Lin component with 2 channels. Observed behavior: LintWakeupNotification is disabled !image-2022-08-23-09-25-00-672.png! !image-2022-08-23-09-13-58-745.png! Expected behavior: LinWakeupNotification must always be enabled Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-37240	New Feature	[S32K1XX][RM] Add support Dma Mux Ip for S32K1XX Platform NewWorkDescription: Dma mux is move from MCL driver to RM driver from ZSE release. This Ip should be support on all platform Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Update mak file. Update generated file Nothing to be done on this as updates are done on :ARTD-42843
ARTD-37573	Bug	[eth][netc] Timestamp is not working properly when interrupts are enabled Detailed description (how to reproduce it): Use timestamp with interrupts, transmitted frames will have problems at confirmation phase. Preconditions: None. Test Case ID (internal TC that caught the defect) optional: EthSwt_netc_TC_303 Observed behavior: Problems at TX confirmation of management frames. Expected behavior: TX of management frames to work properly with interrupts Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: TX requests for management (timestamping) frames are to be properly registered on PSI0 FIFO0, while keeping track of their requests on the initial FIFOs. If a message is to be sent from PSI0, low-level resources should be reserved on both the initial FIFO and on FIFO0. If the message is a normal frame, the resources of FIFO0 are freed, otherwise, the ones of the initial FIFO are freed, which can be done upon calling the IPW-level Transmit function. The transmit function can also remove requests from the initial FIFO and create them properly on FIFO0's queue when enqueueing management frames. When this is done, management frames are effectively moved from their initial FIFO to FIFO0 on all levels. The only information that needs to be maintained is their buffer IDs as the high-level application has no knowledge of this move. Ensuring the unicity of buffer IDs becomes mandatory, and it can be done by maintaining and using offsets. In order to be

ID	Subtype	Headline and Description
		accessible to the high-level application, packets on FIFO0 have to keep track the buffer ID of their counterparts on their initial FIFOs. Test EthSwt_Netc_TS_302 (TC_303) uses polling. To verify the functionality when using interrupts, a new testing case has to be introduced.

ARTD-37845	New Feature	[BASE] Add support for FREERTOS
		
		NewWorkDescription:
		Currently, in osif does not separate Baremetal and freertos for ResumeAllInterrupts/SuspendAllInterrupts
		
		This could be potential issue because cpsidi and disable all interrupt, hence SW timer of OS will be stopped.
		
		{code:c}
		        / Baremetal or FreeRTOS case /
		        #if (MCAL_PLATFORM_ARM == MCAL_ARM_AARCH64)
		            #define ResumeAllInterrupts() ASM_KEYWORD(" msr DAIFClr,#0x3")
		            #define SuspendAllInterrupts() ASM_KEYWORD(" msr DAIFSet,#0x3")
		        #elif (MCAL_PLATFORM_ARM == MCAL_ARM_AARCH32) (MCAL_PLATFORM_ARM == MCAL_ARM_RARCH)
		            #define ResumeAllInterrupts() Sys_EL1ResumeInterrupts()
		            #define SuspendAllInterrupts() Sys_EL1SuspendInterrupts()
		        #else
		            #define ResumeAllInterrupts() ASM_KEYWORD(" cpsie i")
		            #define SuspendAllInterrupts() ASM_KEYWORD(" cpsid i")
		        #endif / MCAL_PLATFORM_ARM == MCAL_ARM_AARCH64 /
		    #else
		        #if (MCAL_PLATFORM_ARM == MCAL_ARM_AARCH32) (MCAL_PLATFORM_ARM == MCAL_ARM_RARCH)
		            #define ResumeAllInterrupts() OsIf_Trusted_Call(Sys_EL1ResumeInterrupts)
		            #define SuspendAllInterrupts() OsIf_Trusted_Call(Sys_EL1SuspendInterrupts)
		        #else
		            #define ResumeAllInterrupts() Sys_ResumeInterrupts()
		            #define SuspendAllInterrupts() Sys_SuspendInterrupts()
		        #endif
		
		This thread might help:
		
		https://www.freertos.org/FreeRTOS_Support_Forum_Archive/June_2016/freertos_portDISABLE_INTERRUPTS_and_portENABLE_INTERRUPTS_c7121eeaj.html
		
		Requirement source:
		
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		
		Proposed solution optional:
		
		Please find a suitable solution for this functional interface, please aware that RTD can call disable/enable interrupt in driver_init. Which mean os or os task might be not ready to be called. I'm not sure that FreeRtos can support such use-case. If not, we might need to re-analyze init sequence or find another solution.

ARTD-40324	Bug	[CAN] wrong generated code / s32ct / multicore enabled + precompile mode Detailed description (how to reproduce it): Create s32z project Add CAN component Enable multicore Set config time as VARIANT-PRE-COMPILE => wrong generated code in Can_43_FLEXCAN_VS_0_PBcfg.c Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: CAN_TS_MC_155 / s32ct Observed behavior: !image-2022-09-28-15-14-34-131.png|thumbnail! [...] Expected behavior: !image-2022-09-28-15-14-35-636.png|thumbnail! [...] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-40544	Bug	[S32ZSE][DIO] Invalid character "//" in Dio.xdm file Detailed description (how to reproduce it): <a:tst expr="text:uniq(//DioChannelGroupIdentification, .)" false="DioChannelGroupIdentification is repeated for two or more channel groups. Please enter different symbolic name." /> invalid character "//"

ID	Subtype	Headline and Description
		http://ionian.ea.freescale.net/1/project/download/ ZTpcbG9jYWxfMDJcb3V0cHV0XGFydGlmYWN0c1xjdXN0b21fcGx1Z2luY2hlY2tcMjAyMjA5MjYyMjA5NTA5NjAwMDBcZG93bmxvYWRzXGVvbF9wbH Proposed solution optional: Update file Dio.xdm Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: invalid character "//" Expected behavior: No invalid character "//" Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update Dio.xdm file
ARTD-41152	Bug	[I2C] LPI2C0_Master_Slave_IRQHandler function prototype is defined wrong in Lpi2c_Ip_Irq.h file Detailed description (how to reproduce it): {}In the Lpi2c_Ip_Irq.h file, t{}}}{{{}he function prototype is defined as:{}}} #if (LPI2C_INSTANCE_COUNT > 0u)/ LPI2C0 master and slave handler named in startup code. / void LPI2C0_Master_Slave_IRQHandler(void); But LPI2C0_Master_Slave_IRQHandler should be defined as ISR(LPI2C0_Master_Slave_IRQHandler)}} Preconditions: ISR(LPI2C0_Master_Slave_IRQHandler)}} Test Case ID (internal TC that caught the defect) optional: None Observed behavior: void LPI2C0_Master_Slave_IRQHandler(void); Expected behavior: ISR(LPI2C0_Master_Slave_IRQHandler)}} Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: ISR(LPI2C0_Master_Slave_IRQHandler)}}
ARTD-41737	New Feature	[IMPLEMENTATION] [ETH] Add support for internal buffers placed in cacheable memory sections NewWorkDescription: The Eth driver shall support allocating the internal buffers in cacheable memory spaces. This will mean adding invalidate and flush operations in the Eth driver to ensure correct sending and receiving data Requirement source: TCP/IP, stacks, zephyr teams (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: [...]
ARTD-41323	New Feature	[wdg] CPR_RTD_00028 is replaced by CPR_RTD_00664 NewWorkDescription: Requirement CPR_RTD_00028 is rejected and replaced by CPR_RTD_00664: "ISR shall check whether its respective interrupt status flag and interrupt enable flag are set; i.e. whether the interrupt is not spurious. If at least one of the flags is not set, then the interrupt is spurious and the ISR shall not treat the event (return immediately). Rationale: A spurious interrupt is considered when there is a glitch in the hardware mechanisms that trigger the CPU exception. Drivers must ensure only relevant events are treated and spurious ones are discarded." Check and update (if needed) the driver implementation to be according to CPR_RTD_00664. Update FMEA and include also Adrian Herea (Safety Manager) to the review. Requirement source: cPRT {*}Proposed solution{*}: The rationale behind the update is to not consider as spurious the events that may otherwise be treated in a different context (e.g. polling mode) and for which the interrupt enable bit is not set. For interrupt lines shared by multiple events, it should be possible for some of the events to be treated by the ISR, while other to be left uncleared so that they can be treated in the thread (as long as they have the interrupt enabled bit cleared, meaning the interrupt is disabled). The update required here would be to only clear the flags for the events that are enabled and leave the others untouched, as long as they are managed by the driver in a different context. The consequence of that would be to allow multiple events that have

ID	Subtype	Headline and Description
		separate interrupt enable bits but share a single interrupt line to function both in "interrupt" and "polling" modes simultaneously. Hence, the clearing of the flag should only happen if the condition "(interrupt_enable and and interrupt_flag)" is true, otherwise it is assumed that the flag is handled outside the ISR. The rationale applies only for drivers that handle hardware resources that share interrupt lines, but the correct implementation of the ISR logic should be checked for all drivers.
ARTD-41475	New Feature	CLONE - [eth] [S32CT]Update code generation to work in JS strict mode NewWorkDescription: Note: Only Peripherals Tool's components are affected{color}*. There has been an update to the System module: System scripts now run in strict mode and contain include guards to protect against multiple inclusion System utilities / functions have been made read-only Code templates are now evaluated in strict mode so that developers can be aware of overwriting existing variables or using undeclared variables*. Requirement source: Consequence of https://jira.sw.nxp.com/browse/ARTD-23864 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: What you have to do: ## Open S32DS's logging file (swtools.log) or launch the program with console logging activated (e.g. run ".\s32ds.exe consoleLog in CMD/PowerShell) ## Create a new project and instantiate your high-level / low-level component. ## Fix the encountered errors related to your component. Most of them will be along the lines of "ReferenceError: <variable_name> is not defined" or "TypeError: <variable_name> is not a writable property of [object global]"_. Both can be resolved by simply defining the offending variable using "var". ## Remove inclusions of "refsSolver.js" in your code templates or code generator scripts. If there's an API you need from that file which is not provided in codegeneratorUtils.js, please let me know.
ARTD-41495	Bug	[SPI] Update driver code according to requirements Detailed description (how to reproduce it): Update driver code according to finding requirements in attached file Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Req 41.3 INTREQ_SPI_RTD_4.4_S32XX_4.0.0_V03 Observed behavior: Update driver code according to finding requirements in attached file Expected behavior: Update driver code according to finding requirements in attached file Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-41528	Bug	[Crc] Crc_Ip_ProtocolType is not same with structure name in CDD_CRC_016 Detailed description (how to reproduce it): Crc_Ip_ProtocolType is used in driver code, not Crc_ProtocolType !image-2022-10-11-17-44-13-139.png|thumbnail! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: NA Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Correct name this structure
ARTD-42764	Bug	[CAN] re-check all "shared static variables" in local scope -> multicore issue Detailed description (how to reproduce it): The develop branch need to be updated ! Some variable in local scope should not be shared (when using static attribute), otherwise it can be simultaneously used in write-access operation of multi cores:

ID	Subtype	Headline and Description
		for example: static Can_HwType CanIf_Mailbox; static PduInfoType CanIf_PduInfo; in Can_43_FLEXCAN_Ipw_ProcessRxMesgBuffer NOTE: please check all static variables in local scope to fix this issue completely! (similar ticket was known in the past, https://jira.sw.nxp.com/browse/ARTD-24436) Preconditions: Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: !image-2022-10-19-18-32-25-573.png|thumbnail! [...] Expected behavior: please check all static variables in local scope. If they are not shared => static attribute should not be used Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-44614	New Feature	[adc] CPR_RTD_00028 is replaced by CPR_RTD_00664 NewWorkDescription: Requirement CPR_RTD_00028 is rejected and replaced by CPR_RTD_00664: "ISR shall check whether its respective interrupt status flag and interrupt enable flag are set; i.e. whether the interrupt is not spurious. If at least one of the flags is not set, then the interrupt is spurious and the ISR shall not treat the event (return immediately). Rationale: A spurious interrupt is considered when there is a glitch in the hardware mechanisms that trigger the CPU exception. Drivers must ensure only relevant events are treated and spurious ones are discarded." Check and update (if needed) the driver implementation to be according to CPR_RTD_00664. Update FMEA and include also Adrian Herea (Safety Manager) to the review. Requirement source: cPRT {*}Proposed solution{*}: The rationale behind the update is to not consider as spurious the events that may otherwise be treated in a different context (e.g. polling mode) and for which the interrupt enable bit is not set. For interrupt lines shared by multiple events, it should be possible for some of the events to be treated by the ISR, while other to be left uncleared so that they can be treated in the thread (as long as they have the interrupt enabled bit cleared, meaning the interrupt is disabled). The update required here would be to only clear the flags for the events that are enabled and leave the others untouched, as long as they are managed by the driver in a different context. The consequence of that would be to allow multiple events that have separate interrupt enable bits but share a single interrupt line to function both in "interrupt" and "polling" modes simultaneously. Hence, the clearing of the flag should only happen if the condition "(interrupt_enable and and interrupt_flag)" is true, otherwise it is assumed that the flag is handled outside the ISR. The rationale applies only for drivers that handle hardware resources that share interrupt lines, but the correct implementation of the ISR logic should be checked for all drivers.
ARTD-44684	New Feature	[wdg] Implement Fault watchdog feature NewWorkDescription: Implement Fault watchdog for observing interrupt responses. Requirement source: S32M24x Reference Manual, Rev. 1, 08/2022, [LINK|https:// nxp1.sharepoint.com/sites/Zebra/Shared%20Documents/Forms/AllItems.aspx? originalPath=aHR0cHM6Ly9ueHAxLnNoYXJlcG9pbnQuY29tLzpmOi9zL1plYnJhL0V1ZWNqSGJPX3Z4S2lrZDFFbTJnQWI0QlctTnpZTkdCNW9sQ3A2 %5FcnRpbWU9YXBPTHkxUXAyRWcand id=%2Fsites%2FZebra%2FShared%20Documents%2FDocuments%2FPlatforms %2FS32M2xx%2FRMand viewid=49fc7681%2D6f70%2D4be1%2Dbc92%2Dc033431dbf2d] Proposed solution optional: Add Fault Watchdog to AeWdog Instance, support configuration to enable and initialize timeout duration: development activities: add checkbox: enable fault watchdog => generate a macro in code to guard relative code. add textbox for timeout duration => generate variable and add to config structure => add new code to write this value into register FAULT_WD_CFG (perform when calling Wdg_43_Instance2_Init) testing activities: create new test case with fault event to trigger an interrupt. run test, debug and check result.
ARTD-44736	Bug	[UART] Can't gen ECPD file for S32M24x

ID	Subtype	Headline and Description
		Detailed description (how to reproduce it): When edit the codegenerator.js file, S32DS can't gen the ECPD file for Uart module !image-2022-11-11-17-48-10-578.png!!image-2022-11-11-17-47-47-622.png! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Uart_TS_ECPD_01.mak Ip_Lpuart_TS_ECPD_01.mak Ip_Flexio_TS_ECPD_01.mak Observed behavior: Gen fail Expected behavior: Have ECPD file Proposed solution optional: N/A
ARTD-44882	Bug	[S32M24x][i2s] I2s_AbortTransmit set all txRemainingWords and rxRemainingWords to 0 Detailed description (how to reproduce it): This issue found on FLEXIO and this is test scenario: !image-2022-11-14-14-21-29-422.png|width=822,height=312! Tester calls Abort function when byteRemainingCount is 54 (T_I2S_BUFFER_SIZE 10). Affter calling Abort function, tester calls getStatus function and driver is in ABORT_STATUS and expecting that remainingByte shoube <= 54. Actually, getStatus returns remainingByte = 0. This is incorrect. I checked data transmitted to MAF, it's extract 10 bytes. This means remainingByte must be 54. !image-2022-11-14-14-31-15-656.png|width=441,height=213! {*}[16/Dec/2022]{*}After verified on the label PVT_I2S_ARTD_45255 (provided by dev to fix bug), It still has issue as below: !image-2022-12-16-15-54-36-807.png|width=1122,height=565! Abort function is call when remainning words = 53, and status changed to ABORT, but when call GetStatus function and then in MasterGetStatus which shown in above figure. because IsChannelIde = TRUE, RemainWord still = 0 because line 1344 can not be in. This leads BytesRemaining = 0. This means all words were transmitted when call GetStatus function. Preconditions: Calling Abort in middle of transmission. Using FLEXIO. Test Case ID (internal TC that caught the defect) optional: I2s_TC_FCT_1001 Observed behavior: NA Expected behavior: NA Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Abort function is calling Flexio_I2s_Ip_MasterEndTransfer and Flexio_I2s_Ip_MasterEndTransfer sets all value to 0. This causes remaining byte is 0 after calling Abort function: !image-2022-11-14-14-34-36-812.png|width=494,height=185!
ARTD-45025	New Feature	[LINTRCV] Implement wakeup functionality as per Autosar R21-11 NewWorkDescription: There are 3 Wakeup methods described in AUTOSAR_SWS_ECUStateManager. The sequence diagrams must be analyzed and have a conclusion on how to implement on the LinTrcv driver. Analyze also the impact on other modules. If any implementation is required, create tickets. Create dev test and test functionality. Revise the LinTrcv_SetWakeupMode which does not implement SWS_LinTrcv_00135 SWS_LinTrcv_00136 Requirement source:

ID	Subtype	Headline and Description
		ASR SWS Proposed solution optional: [SWS_LinTrcv_00135] ⌈Enabled: If the function LinTrcv_SetWakeupMode is called with TrcvWakeupMode == LINTRCV_WUMODE_ENABLE and if the LinTrcv module has a stored wakeup event pending for the addressed bus, the LinTrcv module shall execute the notification within the API call or immediately after (depending on the implementation).⌋() [SWS_LinTrcv_00136] ⌈Disabled: If the function LinTrcv_SetWakeupMode is called with TrcvWakeupMode == LINTRCV_WUMODE_DISABLE, then the notifications for wakeup events are disabled on the addressed network. It is required by the transceiver device and the underlying communication driver to detect the wakeup events and store it internally in order to raise the event when the wakeup notification is enabled again.⌋() [SWS_LinTrcv_00137] ⌈Clear: If the function LinTrcv_SetWakeupMode is called with TrcvWakeupMode == LINTRCV_WUMODE_CLEAR, then a stored wakeup event is cleared on the addressed network
ARTD-45030	New Feature	[LINTRCV] Add infix feature to LinTrcv driver NewWorkDescription: Add infix feature to LinTrcv driver Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Add infix feature to LinTrcv driver
ARTD-45222	Bug	[WDG] Fix Cwe violation HFA Detailed description (how to reproduce it): Fix CWE violation as in attachment Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Some violations have ticket ID comment Expected behavior: Violations are fixed/commented Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-45341	Bug	[ADC] IPL configurator restricts using same phisical channel in multiple control channels Detailed description (how to reproduce it): A sequence of ADC measurements taken on one pin (S32DS 3.4, RTD version S32K1_2022_02) For single shunt application we need to take four samples of the same current from the same pin and one sample of voltage from another pin in the sequence: CurrentSample1, CurrentSample2, Voltage Sample, CurrentSample3, CurrentSample4. After the last sample(CurrentSample4) we enable End of conversion interrupt and in the interrupt routine we process the measurements, etc… The current measurements need to be taken during one PWM period(FTM3) and need to be triggered completely independent of the SW(by means of PDB) and synchronized with PWM. How we achieved this by means of SDK(in design studio for ARM 2.2) was we mapped the same pin/ADC sense to different ADC channels like so: !image-2022-11-18-18-36-48-974.png! !image-2022-11-18-18-36-55-610.png! In adConv1_ChnConfig0 there is mapped ADC pin EXT6 (current measurement) In adConv1_ChnConfig1 there is mapped the same ADC pin EXT6, but with interrupt on end of conversion In adConv1_ChnConfig2 there is mapped ADC pin EXT7 (voltage measurement) However, I cannot achieve this by RTD/ graphical tool for S32K144(or S32M2). There is a limitation that the same ADC external input pin cannot be assigned to more than one ADC channel. !image-2022-11-18-18-37-17-089.png! The graphical tool then issues an error message !image-2022-11-18-18-37-28-929.png! In my opinion, the problem is the tool looks at ADC Physical channel name(HW pin in fact) whether it is unique within the config. However it would be better to check Name for duplicity.

ID	Subtype	Headline and Description
		Otherwise we cannot use the config tool for an application of the sort I describe above. Preconditions: N.A. Test Case ID (internal TC that caught the defect) optional: N.A. Observed behavior: IPL configurator restricts using same phisical channel in multiple control channels Expected behavior: Same physical channel is allowed to be used for multiple control channels (corresponding to SCx register). Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: In IPL configurator code: use "Name" instead of "AdcChannelName" in this constraint: <constraint cond_expr="$this.findDuplicates(item > item.getSetting(`AdcChannelName`).value()).length() == 0" description="Channel Name must be unique in the channel configurations array." level="error"/>
ARTD-45755	New Feature	[PORT] Recommended configuration can not be integrated by EB automatically Detailed description (how to reproduce it): Elektrobit is reporting an issue when integrating the Port Recommended configuration. Current PortRecConfiguration_JtagPins.xdm for S32K3 2.0.0 is: <d:lst type="TOP-LEVEL-PACKAGES"> <d:ctr name="Port" type="AR-PACKAGE"> <d:lst type="ELEMENTS"> <d:chc name="Port" type="AR-ELEMENT" value="MODULE-CONFIGURATION"> <d:ctr type="MODULE-CONFIGURATION"> <a:a name="DEF" value="ASPath:/TS_T40D34M20I0R0/Port"/> <d:var name="IMPLEMENTATION_CONFIG_VARIANT" type="ENUMERATION" value="VariantPostBuild"> <a:a name="IMPORTER_INFO" value="@DEF"/> </d:var> The fix would be: <d:lst type="TOP-LEVEL-PACKAGES"> <d:ctr name="TS_T40D34M20I0R0" type="AR-PACKAGE"> <d:lst type="ELEMENTS"> <d:chc name="PortRecConfigurationStandard" type="AR-ELEMENT" value="MODULE-CONFIGURATION"> <d:ctr type="MODULE-CONFIGURATION"> <a:a name="DEF" value="ASPath:/TS_T40D34M20I0R0/Port"/> <d:var name="IMPLEMENTATION_CONFIG_VARIANT" type="ENUMERATION" value="VariantPostBuild"> <a:a name="IMPORTER_INFO" value="@DEF"/> </d:var> Preconditions: This issue was reported for S32K3 2.0.0 Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Port can not be fully integrated by EB. Expected behavior: Port should be fully integrated by EB. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Verify and apply EB suggested changed.
ARTD-45917	Bug	[CAN] The bswmd for CAN is generated with infix Can_43_FLEXCAN_Bswmd.arxml while the signing list file is looking for Can_Bswmd.arxml Detailed description (how to reproduce it): [The bwmd for CAN is generated with infix Can_43_FLEXCAN_Bswmd.arxml while the signing list file is looking for Can_Bswmd.arxml] Preconditions: [The bwmd is enabled in release build plan] Test Case ID (internal TC that caught the defect) optional: [ ] Observed behavior: [The bwmd for CAN is generated Can_Bswmd.arxml]

ID	Subtype	Headline and Description
		Expected behavior: [The bwmd for CAN is generated with infix Can_43_FLEXCAN_Bswmd.arxml] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-46120	Bug	[S32M24x CD02] [Gdu] On S32M244LC, PMC block GDU module by raise flag errors on PMC Monitor register Detailed description (how to reproduce it): On S32M244LC, PMC blocks GDU module by raising flag errors on the Monitor register Preconditions: After enabling GDU, there is no GDU fault report by AE Test Case ID (internal TC that caught the defect) optional: Gdu_TS_001 Observed behavior: After enabling GDU, there is GDU fault report by AE Expected behavior: After enabling GDU, there is no GDU fault report by AE Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-46148	Bug	[WDG] The HLD example generate code warning and wrong description on EB Detailed description (how to reproduce it): 1. Open Wdg_example_S32M242/Wdg_example_S32M244 example form one plugin 2. open in cygwin the path to the example root 3. Write make generate into the console click enter 4. Write make build into the console click enter Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: The HLD example generate code warning Wdg_example_S32M244 WARNING 22-11-28,17:06:51 (1648) Created required node "/AUTOSAR/TOP-LEVEL-PACKAGES/Ae/ELEMENTS/Ae/AeHvmHviConfig/AeHvmHviGeneralConfig/AeHvmUnderControlIsrCallback" in container "/Ae/Ae/AeHvmHviConfig/AeHvmHviGeneralConfig" WARNING 22-11-28,17:06:51 (1648) Created required node "/AUTOSAR/TOP-LEVEL-PACKAGES/Wdg_43_Instance2/ELEMENTS/Wdg/WdgSettingsConfig/WdgSettingsConfiguration/WdgWatchdogToken" in container "/Wdg_43_Instance2/Wdg/WdgSettingsConfig/WdgSettingsConfiguration" Wdg_example_S32M242 WARNING 22-11-28,17:09:42 (1069) The node "/AUTOSAR/TOP-LEVEL-PACKAGES/Spi/ELEMENTS/Spi/SpiGeneral/SpiPhyUnit/SpiPhyUnit_1/SpiPhyRxDmaChannel" with value "ASPath:/Mcl/Mcl/MclConfig/DMA_LPSPI1_RX" does not refer to nodes. WARNING 22-11-28,17:09:42 (1069) The node "/AUTOSAR/TOP-LEVEL-PACKAGES/Spi/ELEMENTS/Spi/SpiGeneral/SpiPhyUnit/SpiPhyUnit_1/SpiPhyTxDmaChannel" with value "ASPath:/Mcl/Mcl/MclConfig/DMA_LPSPI1_TX" does not refer to nodes. Unattended wizard "GenerateAllVariants (Execute multiple tasks)(GenerateAllVariants)" exited with warnings. Wrong description on EB !image-2022-11-28-17-40-02-616.png! Expected behavior: The example project generate no warnings, errors and update readme file Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-46151	Bug	[dpga] Examples fail for S32M24x RTD R21-11 1.9.0 CD02 Detailed description (how to reproduce it): There is not DPGA_UM.pdf* Step 1: Open S32DS and get in to Dpga_example_S32M242. Step 2: Open S32 Configuration tools/ open Peripherals. Step 3: Open Documentation of Dpga. There is hard fault after I go to argument Mcu_InitClock on M242.

ID	Subtype	Headline and Description
		Preconditions: Wrong description in S32M244 and S32M242 Example on EB and CT. Besides, There is not information about expect output voltage which to know purpose of this example. !image-2022-11-28-18-07-07-318.png! !image-2022-11-28-18-07-14-295.png! There isn’t DPGA_UM.pdf when I enter documentation on CT. Make sure you typed the name correctly. !image-2022-11-28-18-07-21-846.png!!image-2022-11-28-18-07-28-646.png! This example jump to hard fault after I go to argument Mcu_InitClock(McuClockSettingConfig_0) for M242. !image-2022-11-28-18-07-36-098.png! !image-2022-11-28-18-07-42-774.png! Test Case ID (internal TC that caught the defect) optional Dpga_example_S32M242 Observed behavior: Expected behavior: Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update the example info and config for M242
ARTD-46179	Bug	[WDG] The description of some nodes is not detailed Detailed description (how to reproduce it): Step 1: Clean generate TS Wdg_TS_311 Step 2: Open EB and check description of node Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: The description of some nodes is not detailed !image-2022-11-29-09-39-57-465.png|width=748,height=373! Expected behavior: Update description for new nodes of AeWdog instance Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-46312	Bug	[SPI][M24x] Half duplex async transmit in CS continuous mode can not de-assert CS signal to end the transmission Detailed description (how to reproduce it): If at least 1 Slave is available then Half duplex Master async transmit in CS continuous mode can not de-assert CS signal to end the transmission. Preconditions: At least 1 Slave is available. Test Case ID (internal TC that caught the defect) optional: Spi_HalfDuplexTransfer example Observed behavior: Half duplex Master async transmit in CS continuous mode can not de-assert CS signal to end the transmission. Expected behavior: Half duplex Master async transmit in CS continuous mode can transfer successfully. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-46355	Bug	[ICU] [S32K1XX RTD] Ftm_Icu_Ip_SignalMeasurement warning for boundaries array Detailed description (how to reproduce it): Customer compile source code with option [-Warray-bounds] and receive warnings as below: Description Resource Path Location Type

ID	Subtype	Headline and Description
		array subscript is above array bounds [-Warray-bounds] Ftm_Icu_Ip_Irq.c /MCSPTE1AK144_BLDC_6Step/MCAL/ Icu_TS_T40D2M10I1R0/src line 276 C/C Problem array subscript is above array bounds [-Warray-bounds] Ftm_Icu_Ip_Irq.c /MCSPTE1AK144_BLDC_6Step/MCAL/ Icu_TS_T40D2M10I1R0/src line 277 C/C Problem array subscript is above array bounds [-Warray-bounds] Ftm_Icu_Ip_Irq.c /MCSPTE1AK144_BLDC_6Step/MCAL/ Icu_TS_T40D2M10I1R0/src line 279 C/C Problem array subscript is above array bounds [-Warray-bounds] Ftm_Icu_Ip_Irq.c /MCSPTE1AK144_BLDC_6Step/MCAL/ Icu_TS_T40D2M10I1R0/src line 281 C/C Problem array subscript is above array bounds [-Warray-bounds] Ftm_Icu_Ip_Irq.c /MCSPTE1AK144_BLDC_6Step/MCAL/ Icu_TS_T40D2M10I1R0/src line 469 C/C Problem array subscript is above array bounds [-Warray-bounds] Ftm_Icu_Ip_Irq.c /MCSPTE1AK144_BLDC_6Step/MCAL/ Icu_TS_T40D2M10I1R0/src line 473 C/C Problem array subscript is above array bounds [-Warray-bounds] Ftm_Icu_Ip_Irq.c /MCSPTE1AK144_BLDC_6Step/MCAL/ Icu_TS_T40D2M10I1R0/src line 489 C/C Problem array subscript is above array bounds [-Warray-bounds] Ftm_Icu_Ip_Irq.c /MCSPTE1AK144_BLDC_6Step/MCAL/ Icu_TS_T40D2M10I1R0/src line 493 C/C Problem array subscript is above array bounds [-Warray-bounds] Ftm_Icu_Ip_Irq.c /MCSPTE1AK144_BLDC_6Step/MCAL/ Icu_TS_T40D2M10I1R0/src line 517 C/C Problem array subscript is above array bounds [-Warray-bounds] Ftm_Icu_Ip_Irq.c /MCSPTE1AK144_BLDC_6Step/MCAL/ Icu_TS_T40D2M10I1R0/src line 706 C/C Problem array subscript is above array bounds [-Warray-bounds] Ftm_Icu_Ip_Irq.c /MCSPTE1AK144_BLDC_6Step/MCAL/ Icu_TS_T40D2M10I1R0/src line 708 C/C Problem array subscript is above array bounds [-Warray-bounds] Ftm_Icu_Ip_Irq.c /MCSPTE1AK144_BLDC_6Step/MCAL/ Icu_TS_T40D2M10I1R0/src line 713 C/C Problem Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Warning error with array boundaries of parameter "HwChannel" when hwchannel = 0; The customer analysis is attached in attachment (from lowest position up to top). Expected behavior: There's no warning report Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: We understand that in function Ftm_Icu_Ip_SignalMeasurement, in High time and Low time mode, hwchannel cannot be 0. But it still a potential risk and Customer request to analyze for an update to prevent the risk that it can be zero. Dev team please help to analyze this request and update source code if needed[^Follow-up of calls starting from warning position.c]
ARTD-46517	Bug	Dpga_Ip config from Peripherals tool(non-Autosar driver) does not generate files Detailed description (how to reproduce it): Once Dpga_Ip added to Drivers in peripherals tool and blanking time triggers configured, the error appears !DPGA_Ip-1.bmp! Preconditions: Dpga_Ip used in Drivers in Peripherals tool of S32DS(version described Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: respective code for Dpga_Ip not generated Expected behavior: To generate the respective Dpga_Ip code from Peripherals tool Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Change DpgaDevErrorDetect to Dpga{*}Ip{*}DevErrorDetect in Dpga_Ip_CfgDefines.h !DPGA_Ip2.bmp!
ARTD-46526	New Feature	[DPGA] Dpga_Ip does not allow to set trips and trip levels Detailed description (how to reproduce it): GUI for Dpga_Ip does not give the possibility to set high and low limits and the respective filters !Dpga_Ip3.bmp! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior:

ID	Subtype	Headline and Description
		Settings unavailable Expected behavior: Limits and filters to be adjustable Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-46641	Bug	[LIN] function Lin_SetClockMode should be supported infix in K3XX Detailed description (how to reproduce it): function Lin_SetClockMode should be support infix Preconditions: !image-2022-12-07-14-26-02-064.png! Test Case ID (internal TC that caught the defect) optional: Lin_TC_FCT_0101, Lin_TC_FCT_0100 Observed behavior: N/A Expected behavior: function Lin_SetClockMode should change Lin_43_LPUART_FLEXIO_SetClockMode to support infix Proposed solution optional: [....]
ARTD-46742	Bug	[SPI][S32K3XX] For Flexio, IPW config variable FirstChannel = TRUE, so Flexio IP DMA cannot operate due to checking this variable Detailed description (how to reproduce it): For Flexio, IPW config variable FirstChannel = TRUE, so Flexio IP DMA cannot operate due to checking this variable. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Lpspi_Flexio_Ip_Transfer Example Observed behavior: For Flexio, IPW config variable FirstChannel = TRUE, so Flexio IP DMA cannot operate due to checking this variable. Expected behavior: Flexio IP DMA operate normally. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-46823	Bug	[Platform]SW32ZE_RTD_R21-11_0.9.0_P01: Issues with VERSION INFORMATION of Platform Detailed description (how to reproduce it): According to AUTOSAR_SWS_BSWGeneral.pdf, the version information is defined like <MIP>_VENDOR_ID (see attached Published_Information.png) with MIP being <Ma>[_<vi>_<ai>] (see MIP_Define.png). The problem is that the Module abbreviation "Platform" is reserved for Platform Types (see Module_List.png). The defines like PLATFORM_VENDOR_ID PLATFORM_AR_RELEASE_MAJOR_VERSION are reserved for the PlatformTypes.h file (generate file). But in the MCAL, the CDD Platform uses these macros to define its version information. This can lead to redefinitions. !image-2022-12-09-14-42-03-929.png! !image-2022-12-09-14-46-12-880.png! !image-2022-12-09-14-46-45-077.png! Preconditions: #define PLATFORM_VENDOR_ID 43 #define PLATFORM_AR_RELEASE_MAJOR_VERSION 4 #define PLATFORM_AR_RELEASE_MINOR_VERSION 7 #define PLATFORM_AR_RELEASE_REVISION_VERSION 0 #define PLATFORM_SW_MAJOR_VERSION 0 #define PLATFORM_SW_MINOR_VERSION 9 #define PLATFORM_SW_PATCH_VERSION 0 Test Case ID (internal TC that caught the defect) optional: None

ID	Subtype	Headline and Description
		Observed behavior: #define PLATFORM_TYPES_VENDOR_ID 43 #define PLATFORM_TYPES_AR_RELEASE_MAJOR_VERSION 4 #define PLATFORM_TYPES_AR_RELEASE_MINOR_VERSION 7 #define PLATFORM_TYPES_AR_RELEASE_REVISION_VERSION 0 #define PLATFORM_TYPES_SW_MAJOR_VERSION 0 #define PLATFORM_TYPES_SW_MINOR_VERSION 9 #define PLATFORM_TYPES_SW_PATCH_VERSION 0 Expected behavior: as Preconditions Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: as Preconditions
ARTD-46985	New Feature	[SPI][S32K1xx_M24x] Support frame size from 1 to 64bit followed Autosar 4.7 for HLD Detailed description (how to reproduce it): Update Resource Spi.SpiFrameSizeMax and Min followed Autosar 4.7 for HLD. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Lpspi_Flexio_Ip_Transfer example Observed behavior: Update Resource Spi.SpiFrameSizeMax and Min followed Autosar 4.7 for HLD. Expected behavior: Update Resource Spi.SpiFrameSizeMax and Min followed Autosar 4.7 for HLD. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-46992	Bug	[Uart] Uart header file shall follows the naming convention as CDD_Uart.h Detailed description (how to reproduce it): For CDD drivers, the naming convention should be: CDD_[Module].h, according to the following spec: !image-2022-12-12-13-33-25-264.png|width=752,height=356! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: The header file in Uart is Uart.h, which doesn't follow AUTOSAR spec. Expected behavior: The header file in Uart should be CDD_Uart.h Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: CE's comment: All CDD modules follow the naming convention, except uart. We should have an unique naming convention.
ARTD-47212	Bug	[ETH][S32K1] Enet_Ip_ProvideRxBuff() disables receive interrupt Detailed description (how to reproduce it): Read received frame by Enet_Ip_ReadFrame() and release its Rx buffer by calling Enet_Ip_ProvideRxBuff() in the Rx interrupt callback function. Enet_Ip_ProvideRxBuff() uses incorrect mask to clear INT bit in the enhanced Rx buffer descriptor. Preconditions: Rx interrupt enabled in Enet_Ip configuration (EthCtrlEnableRxInterrupt = true) Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: After reading received frame by Enet_Ip_ReadFrame() and releasing its Rx buffer by calling Enet_Ip_ProvideRxBuff() in the Rx interrupt callback function, only the first N (where N = EthCtrlConfigIngressFifoBufTotal) frames successfully generate ENET Rx interrupt. Subsequent frames do not generate Rx interrupt. Expected behavior: All received frames shall generate Rx interrupt.

ID	Subtype	Headline and Description
		Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: In Enet_Ip_ProvideRxBuff(), line #1213, modify Bd->Enh1 and = ENET_{*}TX{*}_ENH1_INT_MASK; to Bd->Enh1 and = ENET_{*}RX{*}_ENH1_INT_MASK;
ARTD-52394	Bug	[S32K1][pwm] PWM Center Aligned mode doesn't support Paired Channel enable Detailed description (how to reproduce it): When the counter mode of PWN is configured to Center Aligned mode, the click box of Paired Channel enable is unavailable. See following screenshots: !image-2023-01-06-11-18-30-143.png! !image-2023-01-06-11-18-34-178.png! However, paired channel outputs do NOT rely on counter mode. The FTM module supports paired center aligned PWM outputs. Preconditions: Set counter mode to center aligned mode Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: The click box of Paired Channel enable is unavailable Expected behavior: Paired Channels should be configurable in center aligned timer mode Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-52508	Bug	[UART] The number of bytes remaining and the status of the channel do not match Detailed description (how to reproduce it): This issue was raised for: IAD-1504 ticket from iMx team Issue: Use the Uart_GetStatus function to read the transmission status and the remaining bytes of the channel: !screenshot-1.png|thumbnail! Check the return status T_UartStatus is UART_STATUS_NO_ERROR !screenshot-2.png|thumbnail! But the number of remaining bytes is not 0 !screenshot-3.png|thumbnail! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: N/A Expected behavior: Lpuart_Uart_Ip_GetReceiveStatus and Lpuart_Uart_Ip_GetTransmitStatus function need to use exclusive area to protect status checking and update the remaining bytes Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Lpuart_Uart_Ip_GetReceiveStatus and Lpuart_Uart_Ip_GetTransmitStatus function need to use exclusive area
ARTD-52965	New Feature	[S32CT] [GPT] Update CT generation of code to work in JS strict mode NewWorkDescription: Note: Only Peripherals Tool's components are affected{color}*. There has been an update to the System module: System scripts now run in strict mode and contain include guards to protect against multiple inclusion System utilities / functions have been made read-only Code templates are now evaluated in strict mode so that developers can be aware of overwriting existing variables or using undeclared variables*.

ID	Subtype	Headline and Description
		Requirement source: Consequence of https://jira.sw.nxp.com/browse/ARTD-23864 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: What you have to do: ## Open S32DS's logging file (swtools.log) or launch the program with console logging activated (e.g. run ".\s32ds.exe consoleLog in CMD/PowerShell) ## Create a new project and instantiate your high-level / low-level component. ## Fix the encountered errors related to your component. Most of them will be along the lines of "ReferenceError: <variable_name> is not defined" or "TypeError: <variable_name> is not a writable property of [object global]"_. Both can be resolved by simply defining the offending variable using "var". ## Remove inclusions of "refsSolver.js" in your code templates or code generator scripts. If there's an API you need from that file which is not provided in codegeneratorUtils.js, please let me know.
ARTD-53042	New Feature	[Platform] Move MPU IP to Platform NewWorkDescription: According to the SOW, it was decided to move the Mpu Ip (Mpu_M7, Mpu_M33, Smpu) from RM to Platform https://nxp1.sharepoint.com/:w:/r/sites/Zebra/_layouts/15/Doc.aspx?sourcedoc=%7B7CCDA70E-A524-4353-B765-555A665E295B%7Dand file=S32%20RTD%20for%20S32ZSE%200.8.0%20SOW.docxand action=defaultand mobileredirect=trueand cid=8bdb9bbc-3ec2-498a-8a2b-70a6c7b42b23 Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Move Mpu_Ip (Mpu_M7, Mpu_M33, Smpu) code, configuration and requirements from Rm to Platform
ARTD-53067	New Feature	[base] Add LinTrcv_43_AE_MemMap file to support infix according CPR_RTD_00195.lintrcv NewWorkDescription: Add LinTrcv_43_AE_MemMap file to support infix according CPR_RTD_00195.lintrcv Requirement source: [...] CPRT CPR_RTD_00195.lintrcv Proposed solution optional: Add LinTrcv_43_AE_MemMap file to support infix according CPR_RTD_00195.lintrcv

ARTD-53547	New Feature	[SPI][SJA][K1]Update code the SPI driver SW32K3_RTD_4_4_1_0_0 to perform transmit no gap
		
		NewWorkDescription:
		
		The customer expects when Spi transmits 37 bytes conservative no gaps occur.
		
		But with the present Spi driver, this function didn't support the Framesize > 255 or the Framesize < 4
		To SPI transmit no gap with a frame size greater than 255 or frame size is an odd number smaller than 4, Spi driver needs to update as below:
		# File Lpspi_Ip.h
		Change the
		
		Lpspi_Ip_StatusType Lpspi_Ip_UpdateFrameSize(const Lpspi_Ip_ExternalDeviceType ExternalDevice, uint8 FrameSize);
		
		to
		
		Lpspi_Ip_StatusType Lpspi_Ip_UpdateFrameSize(const Lpspi_Ip_ExternalDeviceType ExternalDevice, uint16 FrameSize);
		
		# File Lpspi_Ip_Types.h
		Change:
		
		typedef struct {
		    uint8 FrameSize;
		    /**< Frame size configured /
		    boolean Lsb;
		    /**< Transfer LSB first or MSB first /
		    uint32 DefaultData;
		    /**< Default data to send when TxBuffer is NULL_PTR /
		#if (STD_ON == LPSPI_IP_HALF_DUPLEX_MODE_SUPPORT)
		    Lpspi_Ip_HalfDuplexType TransferType;
		    /**< TransferType /
		#endif
		} Lpspi_Ip_DeviceParamsType;
		
		        To:
		
		typedef struct
		{
		    uint16 FrameSize;
		    /**< Frame size configured /
		    boolean Lsb;
		    /**< Transfer LSB first or MSB first /
		    uint32 DefaultData;

ID	Subtype	Headline and Description

		/**< Default data to send when TxBuffer is NULL_PTR /
		#if (STD_ON == LPSPI_IP_HALF_DUPLEX_MODE_SUPPORT)
		Lpspi_Ip_HalfDuplexType TransferType;
		/**< TransferType /
		#endif
		} Lpspi_Ip_DeviceParamsType;
		3. File Lpspi_Ip.c
		3.1: Change
		
		Lpspi_Ip_StatusType Lpspi_Ip_UpdateFrameSize(const Lpspi_Ip_ExternalDeviceType ExternalDevice, uint8 FrameSize)
		to .
		
		Lpspi_Ip_StatusType Lpspi_Ip_UpdateFrameSize(const Lpspi_Ip_ExternalDeviceType ExternalDevice, uint16 FrameSize)
		
		3.2: Change a code part in the
		
		static void Lpspi_TransmitTxInit(uint8 Instance, uint8 TxBuffer, uint16 NumberOfFrames)
		function as below:
		
		else if (State->ExternalDevice->DeviceParams->FrameSize < 17u)
		{
		if ((NumberOfFrames%2u) == 0)
		
		{ State->ExpectedFifoWrites = NumberOfFrames/2u; }
		else
		
		{ State->ExpectedFifoWrites = NumberOfFrames/2u 1u; }
		}
		else
		{
		if ((NumberOfFrames%4u) == 0)
		
		{ State->ExpectedFifoWrites = NumberOfFrames/4u; }
		else
		
		{ State->ExpectedFifoWrites = NumberOfFrames/4u 1u; }
		}
		
		3.3: Change a code part in the
		
		static void Lpspi_TransmitRxInit(uint8 Instance, uint8 TxBuffer, uint16 NumberOfFrames)
		function as below:
		
		else if (State->ExternalDevice->DeviceParams->FrameSize< 17u)
		{
		if ((NumberOfFrames%2u) == 0)
		
		{ State->ExpectedFifoReads = NumberOfFrames/2u; }
		else
		
		{ State->ExpectedFifoReads = NumberOfFrames/2u 1u; }
		}
		else
		{
		if ((NumberOfFrames%4u) == 0)
		
		{ State->ExpectedFifoReads = NumberOfFrames/4u; }
		else
		
		{ State->ExpectedFifoReads = NumberOfFrames/4u 1u; }
		}
		
		Requirement source:
		N/A
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		
		Proposed solution optional:
		As NewWorkDescription

ARTD-54773	Bug	[Mcu] DIVSLOW couldn't be set to be divided by 1 in Clock tree Detailed description (how to reproduce it): DIVSLOW couldn't be set to be divided by 1 in Clock tree. According to the RefMan, DIVSLOW can be 0b0000, which is divided by 1 !divslow-divider.png! !image-2023-02-09-14-40-21-109.png! Preconditions: Configure FLASH clock in S32DS Clock tree configuration Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior:

ID	Subtype	Headline and Description
		DIVSLOW couldn't be set to be divided by 1 in Clock tree Expected behavior: DIVSLOW could be set to be divided by 1 in Clock tree Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: We can modify the generation code to get DIVSLOW divided by 1. However, after modifying the generation code, there is a case that both DIVSLOW and DIVCORE can't be set to /1, because the MCU driver configures the divider before the selector, and by default SYS_CLK source is FIRC (48MHz), which leads to the case that FLS_CLK is 48 MHz, exceed the maximum value. Please consider that case during your test.
ARTD-55421	Bug	[adc] [GENERAL] PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Detailed description (how to reproduce it): In readme.txt file, the following sentence: PLUGINS_DIR The path to the Tresos plugins directory !john_floros_0-1653582301493.png|width=696,height=261! In fact, PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: PLUGINS_DIR points to EB Tresos plugin folder in readme.txt, which is incorrect. Expected behavior: PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt. PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: PLUGINS_DIR should point to RTD plugin folder : PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins
ARTD-55435	Bug	[crypto] [GENERAL] PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Detailed description (how to reproduce it): In readme.txt file, the following sentence: PLUGINS_DIR The path to the Tresos plugins directory !john_floros_0-1653582301493.png|width=696,height=261! In fact, PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: PLUGINS_DIR points to EB Tresos plugin folder in readme.txt, which is incorrect. Expected behavior: PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt. PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: PLUGINS_DIR should point to RTD plugin folder : PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins
ARTD-55441	Bug	[dio] [GENERAL] PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Detailed description (how to reproduce it): In readme.txt file, the following sentence: PLUGINS_DIR The path to the Tresos plugins directory

ID	Subtype	Headline and Description
		!john_floros_0-1653582301493.png|width=696,height=261! In fact, PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: PLUGINS_DIR points to EB Tresos plugin folder in readme.txt, which is incorrect. Expected behavior: PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt. PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: PLUGINS_DIR should point to RTD plugin folder : PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins
ARTD-55443	Bug	[dpga] [GENERAL] PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Detailed description (how to reproduce it): In readme.txt file, the following sentence: PLUGINS_DIR The path to the Tresos plugins directory !john_floros_0-1653582301493.png|width=696,height=261! In fact, PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: PLUGINS_DIR points to EB Tresos plugin folder in readme.txt, which is incorrect. Expected behavior: PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt. PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: PLUGINS_DIR should point to RTD plugin folder : PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins
ARTD-55457	Bug	[gdu] [GENERAL] PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Detailed description (how to reproduce it): In readme.txt file, the following sentence: PLUGINS_DIR The path to the Tresos plugins directory !john_floros_0-1653582301493.png|width=696,height=261! In fact, PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: PLUGINS_DIR points to EB Tresos plugin folder in readme.txt, which is incorrect. Expected behavior: PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt. PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins

ID	Subtype	Headline and Description
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: PLUGINS_DIR should point to RTD plugin folder : PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins
ARTD-55461	Bug	[i2c] [GENERAL] PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Detailed description (how to reproduce it): In readme.txt file, the following sentence: PLUGINS_DIR The path to the Tresos plugins directory !john_floros_0-1653582301493.png|width=696,height=261! In fact, PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: PLUGINS_DIR points to EB Tresos plugin folder in readme.txt, which is incorrect. Expected behavior: PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt. PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: PLUGINS_DIR should point to RTD plugin folder : PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins
ARTD-55467	Bug	[lin] [GENERAL] PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Detailed description (how to reproduce it): In readme.txt file, the following sentence: PLUGINS_DIR The path to the Tresos plugins directory !john_floros_0-1653582301493.png|width=696,height=261! In fact, PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: PLUGINS_DIR points to EB Tresos plugin folder in readme.txt, which is incorrect. Expected behavior: PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt. PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: PLUGINS_DIR should point to RTD plugin folder : PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins
ARTD-55491	Bug	[platform] [GENERAL] PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Detailed description (how to reproduce it): In readme.txt file, the following sentence: PLUGINS_DIR The path to the Tresos plugins directory !john_floros_0-1653582301493.png|width=696,height=261! In fact, PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt

ID	Subtype	Headline and Description
		Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: PLUGINS_DIR points to EB Tresos plugin folder in readme.txt, which is incorrect. Expected behavior: PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt. PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: PLUGINS_DIR should point to RTD plugin folder : PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins
ARTD-55495	Bug	[port] [GENERAL] PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Detailed description (how to reproduce it): In readme.txt file, the following sentence: PLUGINS_DIR The path to the Tresos plugins directory !john_floros_0-1653582301493.png|width=696,height=261! In fact, PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: PLUGINS_DIR points to EB Tresos plugin folder in readme.txt, which is incorrect. Expected behavior: PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt. PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: PLUGINS_DIR should point to RTD plugin folder : PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins
ARTD-55840	Bug	[ETH]: The RMII output buffer CLK didn't enable when configure in internal loopback mode Detailed description (how to reproduce it): The RMII output buffer CLK didn't enable when configure in internal loopback mode. There is no place to operate the SIM(IP_SIM->MISCTRL0 = SIM_MISCTRL0_RMII_CLK_OBE_MASK;) register in the driver code. Preconditions: When ETH is configured in internal loopback mode. Test Case ID (internal TC that caught the defect) optional: None Observed behavior: Can't work normally in internal loopback mode. Expected behavior: Work normally in internal loopback mode. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Add the process for the SIM register in the initial stage.
ARTD-56989	Bug	[LinIf] Fix build failed by incorrect condition Detailed description (how to reproduce it): The #if condition is incorrect. Need to fix the condition as below:

ID	Subtype	Headline and Description
		#if define() #endif Preconditions: Using Linif Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Incorrect condition Expected behavior: Correct condition Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-58508	Bug	[Gdu] Synchronization enable and HD high voltage detect not propagated to the generated files Detailed description (how to reproduce it): Settings of Synchronization enable and HD high voltage detect are not propagated to the generated files. !GDU1.PNG! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: The settings in the picture above are not propagated to the generated structure. The highlighted parameters in the picture below remain at zero value regardless of the settings in the graphical tool. !GDU2.PNG! Expected behavior: To get propagated to the generated structure Proposed solution optional: N/A
ARTD-58791	Bug	[CAN] Update the driver according with SWS_Can_00441 and SWS_Can_00442 Detailed description (how to reproduce it): The CAN driver MainFunctions names shall be according with SWS_Can_00441 and [SWS_Can_00442] correct. Now we have a deviation mentioned in manual and this will lead to BSWMD error [SWS_Can_00441] ⌈ If more than one main function period is configured by CanMainFunctionRWPeriods (see ECUC_Can_00437), the name of the Can_MainFunction_Write() functions shall be # Can_MainFunction_Write_<CanMainFunctionRWPeriods.ShortName>() for each CanMainFunctionRWPeriods that is referenced by at least one TRANSMIT CanHardwareObject (see ECUC_Can_00438). [SWS_Can_00442] ⌈ If more than one main function period is configured by CanMainFunctionRWPeriods (see ECUC_Can_00437), the name of the Can_MainFunction_Read() functions shall be # Can_MainFunction_Read_<CanMainFunctionRWPeriods.ShortName>() for each CanMainFunctionRWPeriods that is referenced by at least one RECEIVE CanHardwareObject (see ECUC_Can_00438) Don't forgot if an update is taken then to update the documentation too by remove the deviation and check if the driver is still complaint with the VSMD report. Preconditions: Use Polling mode with mainfunctions read\write Test Case ID (internal TC that caught the defect) optional: n/A Observed behavior: We kept ASR 4.3 older requirement. Expected behavior: Comply with standard requirements SWS_Can_00441 and SWS_Can_00442 Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]

ID	Subtype	Headline and Description
ARTD-58867	Bug	[Base]SW32ZE_RTD_R21-11_0.9.0_P01: Issues with VERSION INFORMATION in BASE Detailed description (how to reproduce it): According to AUTOSAR_SWS_BSWGeneral.pdf, the version information is defined like <MIP>_VENDOR_ID (see attached Published_Information.png) with MIP being <Ma>[_<vi>_<ai>] (see MIP_Define.png). The problem is that the Module abbreviation "Platform" is reserved for Platform Types (see Module_List.png). The defines like PLATFORM_VENDOR_ID PLATFORM_AR_RELEASE_MAJOR_VERSION are reserved for the PlatformTypes.h file (generate file). But in the MCAL, the CDD Platform uses these macros to define its version information. This can lead to redefinitions. !image-2022-12-09-14-42-03-929.png! !image-2022-12-09-14-46-12-880.png! !image-2022-12-09-14-46-45-077.png! Preconditions: #define PLATFORM_VENDOR_ID 43 #define PLATFORM_AR_RELEASE_MAJOR_VERSION 4 #define PLATFORM_AR_RELEASE_MINOR_VERSION 7 #define PLATFORM_AR_RELEASE_REVISION_VERSION 0 #define PLATFORM_SW_MAJOR_VERSION 0 #define PLATFORM_SW_MINOR_VERSION 9 #define PLATFORM_SW_PATCH_VERSION 0 Test Case ID (internal TC that caught the defect) optional: None Observed behavior: #define PLATFORM_TYPES_VENDOR_ID 43 #define PLATFORM_TYPES_AR_RELEASE_MAJOR_VERSION 4 #define PLATFORM_TYPES_AR_RELEASE_MINOR_VERSION 7 #define PLATFORM_TYPES_AR_RELEASE_REVISION_VERSION 0 #define PLATFORM_TYPES_SW_MAJOR_VERSION 0 #define PLATFORM_TYPES_SW_MINOR_VERSION 9 #define PLATFORM_TYPES_SW_PATCH_VERSION 0 Expected behavior: as Preconditions Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: as Preconditions
ARTD-58981	Bug	[UART] The number of bytes remaining and the status of the channel do not match belong Flexio Ip and Linflexd Ip Detailed description (how to reproduce it): This issue was raised for: IAD-1504 ticket from iMx team Issue: Use the Uart_GetStatus function to read the transmission status and the remaining bytes of the channel: !screenshot-1.png|thumbnail! Check the return status T_UartStatus is UART_STATUS_NO_ERROR !screenshot-2.png|thumbnail! But the number of remaining bytes is not 0 !screenshot-3.png|thumbnail! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: N/A Expected behavior: Flexio_Uart_Ip_GetStatus function needs to use an exclusive area to protect status checking and update the remaining bytes Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Flexio_Uart_Ip_GetStatus functions need to use an exclusive area
ARTD-58991	Bug	[CRYPTO] The IP files miss Autosar version check Detailed description (how to reproduce it): The IP files now missing the check for Autotar version. More detail at https://bitbucket.sw.nxp.com/projects/ARTD/repos/crypto/pull-requests/746/overview?commentId=1633774 Preconditions: Crypto is cloned

ID	Subtype	Headline and Description
		Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: The IP files now missing the check for Autotar version Expected behavior: The IP files have the check for Autotar version. Proposed solution optional: Add Autosar version check for IP files
ARTD-59354	Bug	[LINTRCV] Cannot generate an *.ecpd file when creating a test ecvd Detailed description (how to reproduce it): .ecpd file not generated !image-2023-03-07-09-17-25-159.png|thumbnail! Preconditions: Test Case ID (internal TC that caught the defect) optional: Observed behavior: Expected behavior: Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: !image-2023-03-07-09-16-49-204.png|thumbnail!
ARTD-59992	Bug	[DIO] Code generated is inconsistent between EB tresos and S32DS when test the MultiCore Detailed description (how to reproduce it): Create config for dio in EB tresos Create config for dio in S32DS Enable Multicore, add channel group Generate code Preconditions: Enable Multicore, add channel group Test Case ID (internal TC that caught the defect) optional: Dio_TS_203 Observed behavior: Data generated in Dio_Cfg.h and Dio_Cfg.c : Code generated is inconsistent between EB tresos and S32DS Dio_Cfg.h !image-2023-03-03-17-23-06-511.png|thumbnail! Dio_Cfg.c !image-2023-03-03-17-22-39-265.png|thumbnail! Expected behavior: code generate between EB tresos and S32DS is consistent Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Added Dio_ConfigName variable such that all DioConfig variables in C code contain the name of DioConfig field in S32DS, the same as in Tresos generated code.
ARTD-60042	New Feature	[port] Enable ECPD generation for IP components (CPR_RTD_00544) Note: This feature must be implemented by all the modules for which this ticket was cloned (including CDDs). No clone shall be closed with "Not Applicable" or "No Action Taken". How to implement: # Add "options_expr" tags (containing all the necessary options) for the fields of the IP component(s). Please refer to the slides covering "EPD Generation" in Reference #2 below. # To ensure that the ECPD is generated only for your component, modify line 512 in "mcu_data/components/PlatformSDK_S32K3_2021_08/system/codegenerator.js" as follows: {code:javascript} if (configComponent.isOptionSet("ecucDefinitionCollection") and and (configComponent.getId() === "Gmac")) { where you can replace "Gmac" with the ID of your IP component. # Check that there are no errors reported in the Problems view and that the ECPD has been generated in Code Preview. How to validate: # Manually create an EBT plugin using the previously generated ECPD. To achieve this: ## Unzip [^Ip_TS_T40D34M10I0R0.zip] in the "output/eclipse/plugins" folder of your K3 work tree / repository. ## Open "Ip_TS_T40D34M10I0R0/plugin.xml" and replace all occurrences of "Gmac" with the ID of your IP component.

ID	Subtype	Headline and Description
		# Create an EBT project (File > New > Configuration Project > Project Name: TstIp > Next > ECU ID: someId; Target: CORTEXM/ S32K3XX > Finish) and add the previously created module (Right click on TstIp in Project Explorer > Module Configurations... > Double click on your module > OK) CHECKPOINT 1: The module should have been successfully imported into the project. # Go to S32DS and create a valid project using your IP component (or use an already existing one). {*}{color:#FF0000}Please assign a value other than the default value to each setting to avoid default values in this configuration{color}{*}. Rename your IP component instance and discard the "_1" prefix (e.g. "Gmac_1" > "Gmac"): !screenshot-1.png! # Generate the [LT/PB]Cfg.h/.c files of your IP component and save them somewhere to avoid overwriting them in a later step. We'll call this set of files {_}OriginalGenerateFiles{_}. # Generate the ECVD file for your IP component instance (Global Settings > EcvdGenerationMethod = INDIVIDUAL > EcvdOutputPath = <EcvdOutputDir> > Click on the "Generate Configuration" button) # Import the previously generated ECVD file in your EBT project (Right click on TstIp in Project Explorer > Im and Exporters... > Green Button > Next > File Name: <Browse and select the ECVD file> > Next > Enable path mapping > Finish > Run Importer) CHECKPOINT 2: The ECVD file should have been successfully applied to your module. # Save (CTRL S) and generate the project (Right click on TstIp in Project Explorer > Generate Project) to trigger the EPC generation. CHECKPOINT 3: The EPC file should have been successfully generated (TstIp/output/output/<Ip>.epc) # Import the previously generated EPC file in your S32DS project (File > Import > Import ECU Configuration > Path: <EpcOutputDir> > Check "Merge into current configuration" > Check "Check ECUC-MODULE-DEF" value > CHECKPOINT 4: The EPC file should have been successfully imported into your S32DS project. If you get the error "The required modules are not available for the current configuration" it means that the ECUC-MODULE-DEF check has failed and you're not properly using M4_XDM_AR_PKG_NAME in your IP component. # Generate again the [LT/PB]Cfg.h/.c files of your IP component and save them somewhere without overwriting the previous set of files. We'll call this new set of files {_}ImportedGenerateFiles{_}. # Compare the OriginalGenerateFiles set with the ImportedGenerateFiles set. CHECKPOINT 5: The comparison should be binary identical. There should be no single difference between the two set of files. Requirements covered by this ticket: CPR_RTD_00544 EA_RTD_00107 EA_RTD_00108 References: # [^Ethernet.template] # [https://nxp1.sharepoint.com/:p:/s/Zebra/EaMbpShSxPVAq7amkzZx3NMB1LWDlRvDlnrzP1m0TZ9elQ?e=erB0sH] (Relevant chapters: "9. EPD Importer Replacing the lost GNU M4 macros", "10. EPD Generation", "11. EPC Importer", "12. EPC Generation") Legend: EPD ECU Parameter Definition (Schema) Used by EBT ECPD ECU Configuration Parameter Definition (Schema) Used by S32CT EPC ECU Parameter Configuration (Configuration File) Used by EBT ECVD ECU Configuration Value Description (Configuration File) Used by S32CT For S32K3, for mapping file, because un-matching about peripherals between EB and Pins tool, so peripheral in mapping files was modified manually with mixing between upper and lowercase: eMIOS, JTAG_TRACEnoETM...
ARTD-60173	New Feature	[adc] Add support for multiple configurations in S32CT NewWorkDescription: Previously, we limited the number of functional groups (multiple configurations) to 1. Now we want to add support for multiple configurations (in the form of post-build variants and classic functional groups) so that we can also replicate in S32CT the module configurations from EB Tresos in which there are two or more post-build variants. Requirement source: SRS_BSW_00405 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Read the newly introduced section ("13. Multiple Configurations") in the following presentation: [NXP_RTD_AUTOSAR_S32CT_V2|https://nxp1.sharepoint.com/:p:/r/sites/Zebra/Shared%20Documents/Configuration%20(code,%20DS,%20EB,%20etc)/ NXP_RTD_AUTOSAR_S32CT_V2.pptx?d=w28a51ba3c45240f5abb6a6933671dcd3and csf=1and web=1and e=r7YbFB] Make sure you have added the SYNC_VALUE and SYNC_SIZE options to each component setting that requires it (as suggested in the presentation, you can use [epdimp2ct|https://nxp1.sharepoint.com/:f:/s/Zebra/ EprWldG7fTNPgdoXtWTgmIsBn1Xo56J_qJQGC37m92ax9A?e=ivi58c] to automate this task as long as you have a matching EPD for your component) Apply the changes shown in the source code references. The main points are: {noformat} [!IF "var:defined('postBuildVariant')"!]{noformat} translates to {noformat} if (useFgInfix()){noformat} The file names and configuration structures should not contain the variant infix when there's no selected post-build variant (actually, most of the changes you'll see in the references are done just to support this use-case) Sanity Check: Check that your driver can be successfully built in all three scenarios: ## ComponentGenerationMethod = EcucPostBuildVariants and VARIANT_NO = 0 ## ComponentGenerationMethod = EcucPostBuildVariants and VARIANT_NO > 1 (VARIANT_NO == 1 is the base case from which you started)

ID	Subtype	Headline and Description
		## ComponentGenerationMethod = FunctionalGroups (with at least 2 FGs configured in the project) Extensive Testing: By default, EB Tresos cannot generate variant aware EPC files without throwing a null pointer exception. To get around this limitation, the dreisoft.tresos.autosar2 plugin must be patched. To do this, download one of the two attached archives, depending on your EBT version, and extract it in <ebt_root_dir>/plugins. # Go to your driver's plugin, open {_}plugin.xml{_}, search for {_}id="EPCGenerator"{_}, and set the value of parameter "allVariants" to {_}"true"{_}. This will enable variant aware EPC file generation for your module. # Create a configuration with at least two post-build variants. The project will need EcuC for post-build variants to be enabled. The configuration should ideally use different values for nodes with postBuildVariantValue=TRUE and different list elements for nodes with postBuildVariantMultiplicity=TRUE. This will ensure that errors in variant handling can be easily caught later on while doing the diffs. # Generate the EPC files by right clicking on your project's configuration and select "Generate Project". If the 1st step has been carried out correctly, there will be a "PostBuildSelectableVariants.arxml" file in <project_root_dir>/output/output and your module's EPC file, lying inside the same directory, will contain lots of "VARIATION-POINT" elements. # Open S32DS and select File > Import > S32 Configuration Tools > Import ECU Configuration. Browse to the location where your EPC files are located within the EBT project and select at least your module's EPC file, EcuC's EPC file and PostBuildSelectableVariants.arxml. Check the "Import the configuration as a new one" radio button and select the processor / package / sdk version accordingly. # Go to the "Global settings" panel and make sure that ComponentGenerationMethod=EcucPostBuildVariants in the System component. # Select "Update Code" to generate your S32DS project files. Perform a comparison/diff of the generated files between S32DS and EBT. These files should be pretty much identical (sans whitespace or formatting) # Go to the "Global settings" panel again to generate the ECVD files (EcvdGenerationMethod=INDIVIDUAL, browse a location for EcvdOutputPath and click on the Generate Configuration button). Compare/diff your module's EPC file with your module's ECVD file. These files should be pretty much identical (sans whitespace or formatting) # The above steps tested the EBT->S32DS compatibility. To also test the S32DS->EBT compatibility, do the above steps in reverse. That is, start from an S32DS project with post-build variants, generate the ECVD files, import them into EBT, compare again the generated files and the configuration files.
ARTD-60229	Bug	[DPGA] the S32DS include both IP and HLD configuration Detailed description (how to reproduce it): Step 1: Open S32DS and create a new application project. Step 2: Open S32 Configuration tools/ open Peripherals. Step 3: add Dpga_Ip Issue: the S32DS include both Dpga and Dpga_Ip Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: see the detail description Expected behavior: the DS only include one layer Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-60242	Bug	[LIN] Lin.h includes LinIf.h Detailed description (how to reproduce it): Lin in RTD 1.0 (S32K1xx): lin.h includes linif.h which causes circular inclusion error when using Vector stack. This was not included in MCAL 4.2 and 4.3 Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: lin.h includes linif.h which causes circular inclusion error when using Vector stack. Expected behavior: Error should not raise Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Analyze to remove include LinIf.h in Lin.h
ARTD-61228	New Feature	[CRC]S32K1_RTD_1_0_1_D2202, DMA Transfer List overwritten by Crc Hardware Calculation We intend to use the DMA in conjunction with the CRC unit to verifiy application (around 512 KB in size).

ID	Subtype	Headline and Description
		The S32K1_RTD_1_0_1_D2202 provides the Crc_Ip_HwCalculate function to start crc calculation in background. The TCD record can be configured in S32DS studio explicitly as DMA_IP_TRANSFER_SIZE_4_BYTE and the CRC Protocol is 32Bit_CUSTOM. However when Crc_Ip_HwCalculate function is called, it overwrites the Transfer Size 1 byte by default values of s_atDmaTransferList structure. This structure can neither be changed at runtime no configured by S32DS. Is this an intended limitation to the NXP driver? How can I fix this behavior without changing the Crc_Ip_Hardware.c source file? !image-2023-03-27-10-11-07-463.png! !image-2023-03-27-10-12-06-589.png!
ARTD-61269	New Feature	[Pwm][S32K1XX] Support FTM reload interrupt Implement the reload interrupt feature for FTM.
ARTD-61584	Bug	[I2C]Multiple definition cause a compilation error Detailed description (how to reproduce it): When I2cErrorCallback was enabled. I2C_MASTER_EVENT_ERROR_FIFO appeared at 2 places: In Lpi2c_Ip_MasterEventType enum in the file Lpi2c_Ip_Callbacks.h Macro generated in the file CDD_I2c_Cfg.h. This caused a compilation error. Preconditions: I2cErrorCallback was enabled Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Has a compilation error. Expected behavior: Has no compilation error. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Rename one of them if all of them need to declared and used.
ARTD-61632	Bug	[S32K3 3.0.0][S32K1 2.0.0] i2s: Node "FlexioI2sCallback" can not generate ecvd in IP layer FLEXIO_I2s Detailed description (how to reproduce it): The configuration can not generate node FlexioI2sCallback in ecvd for IP layer Flexio_i2s Preconditions: For all derivatives, The configuration can not generate node FlexioI2sCallback in ecvd for IP layer Flexio_i2s Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: node FlexioI2sCallback in ecvd for IP layer Flexio_i2s can not generate. See more in picture Expected behavior: The configuration need similar between s32ds and ebt Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-62231	New Feature	[LINTRCV] Analyze the implementing with LinTrcv_GetBusWuReason function NewWorkDescription: There are multiple requirements which describe the implementing with LinTrcv_GetBusWuReason function. Analyze them and have a final solution how to support this function. Requirement source: ASR SWS (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: This ticket was implemented in ARTD-45025. Step to implementing: For M24x and M27x only support two reasons of wake up: Type||Meaning LINTRCV_WU_INTERNALLY The transceiver has detected, that the network has been woken up by the ECU via a request to NORMAL mode. LINTRCV_WU_BY_BUS The transceiver has detected, that the network has

ID	Subtype	Headline and Description
		caused the wake up of the ECU. Call AeLinPhy_LinTrcv_Ip_CheckWakeupFlag function to verify wakeup flag is set or not , if set, set status to LINTRCV_WU_BY_BUS. If no, call AeLinPhy_LinTrcv_Ip_GetMode function to get current mode is NORMAL mode or not, if yes, set status to LINTRCV_WU_INTERNALLY. otherwise , Lintrcv can't detect any reason, set status to LINTRCV_WU_NOT_SUPPORTED. And if driver can't get communication to the transceiver => LINTRCV_WU_ERROR will be return
ARTD-62453	Bug	[I2S] Channel 1 DMA configure wrongly in DS Detailed description (how to reproduce it): Issue 1: When DMA enable with MUX line, 2 channel DMA have to configure. But channel 1 didn't get value from configuration correctly. Issue 2: Functional Group get name to put the file name wrongly with uppercase added in generated file. For example: Functional Group: BOARD_InitPeripherals Generated file: Sai_Ip_BOARD_InitPeripherals_PBcfg.h File included in another file: Sai_Ip_BOARD_INITPERIPHERALS_PBcfg.h. Please check this for all generated files. Preconditions: DMA enable Mux LINE used. Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Issue 1: Channel 1 is 0 if channel 2 is disabled. Issue 2: File included in another file is different name with generated file. Expected behavior: Issue 1: Channel 1 get value from configuration correctly. Issue 2: File included in another file is have similar name with generated file. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Issue 1: aDmaChannel get value for channel 1 from channel 2. See image. Issue 2: Remove .toUpperCase() from variantName for #include "file<variantName>".
ARTD-62459	Bug	[I2C] Macros NUM_INSTANCE_USED do not follow coding conventions Detailed description (how to reproduce it): Open example "I2c_HLD_FLEXIO_Transfer" on S32DS. Open Peripheral and click "Update Code". Check file "Lpi2c_Ip_Cfg.h". !image-2023-04-11-09-49-18-983.png! Preconditions: Functional group properties need to setup with name "BOARD_InitPeripherals". !image-2023-04-11-09-58-22-082.png! Test Case ID (internal TC that caught the defect) optional: Example Observed behavior: After generate, the variable name is not capitalized. Expected behavior: The variable name is capitalized. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-62550	Bug	[Spi] Fail generating code with DMA fast transfer on DS Detailed description (how to reproduce it): Fail generating code with DMA fast transfer on DS DS Tool do not raise errors when configuring wrong with require 2, 3 for Spi/SpiDriver/SpiSequence/SpiEnableDmaFastTransfer node: "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. All parameters in External Device linked to each Job in this Sequence must be the same except SpiCsIdentifier, SpiCsContinous.

ID	Subtype	Headline and Description
		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." Preconditions: Creating DS project with DMA fast transfer Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Log : [DATA] Error resolving: ((system::getParent($this, 2).getSetting(`SpiEnableDmaFastTransfer`).getValue() == false) (system::deref($this).getSetting(`SpiChannelList`).map((x > system::deref(x.getSetting(`SpiChannelAssignment`)))).filter((x > (x.getSetting(`SpiDataWidth`).getValue() > 9))) (((x.getSetting(`SpiEbMaxLength`).getValue() < 32767) and and (x.getSetting(`SpiChannelType`).getValue() == `EB`)) ((x.getSetting(`SpiIbNBuffers`).getValue() < 32767) and and (x.getSetting(`SpiChannelType`).getValue() == `IB`))))), Cannot convert value ([Setting: [Type: struct, Id: Spi.SpiDriver.SpiChannel.7], Setting: [Type: struct, Id: Spi.SpiDriver.SpiChannel.12]]) with type: OBJECT to boolean. Expected behavior: Generating successfully project DMA Fast Transfer Project Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-62978	Bug	[pwm] [GENERAL] PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Detailed description (how to reproduce it): In readme.txt file, the following sentence: PLUGINS_DIR The path to the Tresos plugins directory !john_floros_0-1653582301493.png|width=205,height=77,thumbnail! In fact, PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: PLUGINS_DIR points to EB Tresos plugin folder in readme.txt, which is incorrect. Expected behavior: PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt. PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: PLUGINS_DIR should point to RTD plugin folder : PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins
ARTD-66265	Bug	[ETH] Schema file is not using VendorApiInfix In Eth_43_GMAC.xdm default value for VendorApiInfix is empty, also this parameter is configured as OPTIONAL true : <a:da name="DEFAULT" value=""/> <a:a name="OPTIONAL" value="true" Since Eth driver is using VendorApiInfix the lines should be replaced with : <a:da name="DEFAULT" value="GMAC"/>
ARTD-66345	Bug	[PORT]UM file has a note need to detail explanation Detailed description (how to reproduce it): In the RTD port driver user manual, for the Parameter PortPinModeChangeable, the document says :_"{color:#FF0000}The function for changing the pin modes is not supported by the safety implementation{color}" What does it means exactly? The RTD S32K1 driver should add a detailed explanation for this situation. If don't have a situation occurs as the highlighted content, this content should be removed from UM file. Preconditions: Add detailed explanation Test Case ID (internal TC that caught the defect) optional: None

ID	Subtype	Headline and Description
		Observed behavior: as Detailed description Expected behavior: as Preconditions Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: as Preconditions
ARTD-67383	Bug	[I2C] Missing 'I2c' prefix in Resource files Detailed description (how to reproduce it): Missing 'I2c' prefix for 2 variable in resource file : I2cUnifiedInterrupts:TRUE and I2cMulticoreSupport:TRUE Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Missing 'I2c' prefix for 2 variable in resource file : I2cUnifiedInterrupts:TRUE and I2cMulticoreSupport:TRUE Expected behavior: All variables in Resource files will be added 'I2c' prefix Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Add for I2cUnifiedInterrupts:TRUE and I2cMulticoreSupport:TRUE
ARTD-67533	New Feature	[CXPI] Create preliminary file structure for Cxpi driver NewWorkDescription: Create preliminary file structure for Cxpi in order to have a successful build of the module. Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: add template files (C headers and sources, templates for generated files, XDM schema file for M24X platform and Makefiles). add support for Cxpi module in Zth and include this module in RTD 4.7 S32K1XX S32M24x 2.0.0 release manifest.
ARTD-69383	New Feature	[platform] [S32CT] Update code generation to work in JS strict mode NewWorkDescription: Note: Only Peripherals Tool's components are affected{color}*. There has been an update to the System module: System scripts now run in strict mode and contain include guards to protect against multiple inclusion System utilities / functions have been made read-only Code templates are now evaluated in strict mode so that developers can be aware of overwriting existing variables or using undeclared variables*. Requirement source: Consequence of https://jira.sw.nxp.com/browse/ARTD-23864 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: What you have to do: ## Open S32DS's logging file (swtools.log) or launch the program with console logging activated (e.g. run ".\s32ds.exe consoleLog in CMD/PowerShell) ## Create a new project and instantiate your high-level / low-level component. ## Fix the encountered errors related to your component. Most of them will be along the lines of "ReferenceError: <variable_name> is not defined" or "TypeError: <variable_name> is not a writable property of [object global]"_. Both can be resolved by simply defining the offending variable using "var". ## Remove inclusions of "refsSolver.js" in your code templates or code generator scripts. If there's an API you need from that file which is not provided in codegeneratorUtils.js, please let me know.
ARTD-70394	New Feature	[RTE] Add support for Cxpi exclusive areas in Rte NewWorkDescription: Add exclusive areas support in Rte for the new Cxpi component. Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Run script to generate SchM files, update specific Makefile in Rte.
ARTD-70866	New Feature	[icu] Add support for multiple configurations in S32CT

ID	Subtype	Headline and Description
		NewWorkDescription: Previously, we limited the number of functional groups (multiple configurations) to 1. Now we want to add support for multiple configurations (in the form of post-build variants and classic functional groups) so that we can also replicate in S32CT the module configurations from EB Tresos in which there are two or more post-build variants. Requirement source: SRS_BSW_00405 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Read the newly introduced section ("13. Multiple Configurations") in the following presentation: [NXP_RTD_AUTOSAR_S32CT_V2|https://nxp1.sharepoint.com/:p:/r/sites/Zebra/Shared%20Documents/Configuration%20(code,%20DS,%20EB,%20etc)/NXP_RTD_AUTOSAR_S32CT_V2.pptx?d=w28a51ba3c45240f5abb6a6933671dcd3and csf=1and web=1and e=r7YbFB] Make sure you have added the SYNC_VALUE and SYNC_SIZE options to each component setting that requires it (as suggested in the presentation, you can use [epdimp2ct|https://nxp1.sharepoint.com/:f:/s/Zebra/EprWldG7fTNPgdoXtWTgmIsBn1Xo56J_qJQGC37m92ax9A?e=ivi58c] to automate this task as long as you have a matching EPD for your component) Apply the changes shown in the source code references. The main points are: {noformat} [!IF "var:defined('postBuildVariant')"!]{noformat} translates to {noformat} if (useFgInfix()){noformat} The file names and configuration structures should not contain the variant infix when there's no selected post-build variant (actually, most of the changes you'll see in the references are done just to support this use-case) Sanity Check: Check that your driver can be successfully built in all three scenarios: ## ComponentGenerationMethod = EcucPostBuildVariants and VARIANT_NO = 0 ## ComponentGenerationMethod = EcucPostBuildVariants and VARIANT_NO > 1 (VARIANT_NO == 1 is the base case from which you started) ## ComponentGenerationMethod = FunctionalGroups (with at least 2 FGs configured in the project) Extensive Testing: By default, EB Tresos cannot generate variant aware EPC files without throwing a null pointer exception. To get around this limitation, the dreisoft.tresos.autosar2 plugin must be patched. To do this, download one of the two attached archives, depending on your EBT version, and extract it in <ebt_root_dir>/plugins. # Go to your driver's plugin, open {_}plugin.xml{_}, search for {_}id="EPCGenerator"{_}, and set the value of parameter "allVariants" to {_}"true"{_}. This will enable variant aware EPC file generation for your module. # Create a configuration with at least two post-build variants. The project will need EcuC for post-build variants to be enabled. The configuration should ideally use different values for nodes with postBuildVariantValue=TRUE and different list elements for nodes with postBuildVariantMultiplicity=TRUE. This will ensure that errors in variant handling can be easily caught later on while doing the diffs. # Generate the EPC files by right clicking on your project's configuration and select "Generate Project". If the 1st step has been carried out correctly, there will be a "PostBuildSelectableVariants.arxml" file in <project_root_dir>/output/output and your module's EPC file, lying inside the same directory, will contain lots of "VARIATION-POINT" elements. # Open S32DS and select File > Import > S32 Configuration Tools > Import ECU Configuration. Browse to the location where your EPC files are located within the EBT project and select at least your module's EPC file, EcuC's EPC file and PostBuildSelectableVariants.arxml. Check the "Import the configuration as a new one" radio button and select the processor / package / sdk version accordingly. # Go to the "Global settings" panel and make sure that ComponentGenerationMethod=EcucPostBuildVariants in the System component. # Select "Update Code" to generate your S32DS project files. Perform a comparison/diff of the generated files between S32DS and EBT. These files should be pretty much identical (sans whitespace or formatting) # Go to the "Global settings" panel again to generate the ECVD files (EcvdGenerationMethod=INDIVIDUAL, browse a location for EcvdOutputPath and click on the Generate Configuration button). Compare/diff your module's EPC file with your module's ECVD file. These files should be pretty much identical (sans whitespace or formatting) # The above steps tested the EBT->S32DS compatibility. To also test the S32DS->EBT compatibility, do the above steps in reverse. That is, start from an S32DS project with post-build variants, generate the ECVD files, import them into EBT, compare again the generated files and the configuration files.
ARTD-70879	Bug	[S32K1_M24X RTM 2.0.0] GPT: Unexpected behavior when config Predeftimer 24 and 32 bits Detailed description (how to reproduce it): [When tester choose option "GPT_PREDEF_TIMER_1US_16BIT_ENABLED" and "GPT_PREDEF_TIMER_1US_16_24_32BIT_ENABLED" of node "GptPredefTimer1usEnablingGrade*", EB tresos permit config "GptPredefTimer_1us_24Bit", "GptPredefTimer_1us_32Bit" and "GptPredefTimer_100us_32Bit" in spite of restrict of FTM hardware that has only 16 bit. Thus, tester cannot choose any hardware channel suitable and generated code is not have hardware module type field like first attachment picture.] Preconditions: [None] Test Case ID (internal TC that caught the defect) optional: [None] Observed behavior: [Review driver] Expected behavior: [User can not choose and config Predertimer 24 and 32 bits] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)

ID	Subtype	Headline and Description
		Proposed solution optional: [Whenever choose that two nodes, please raise the error because hardware won't support Predertimer 24 and 32 bits feature and user can not config Predeftimer 24 and 32 bits node or hidden that two nodes.]
ARTD-71064	Bug	[GPT] Code funtioning wrongly while using Gpt_GetTimeElapsed in multicore Detailed description (how to reproduce it): Configure different timers which are used from different cores. While using the Gpt_GetTimeElapsed API this doesn't report the correct value at some point. There's a static variable declared inside the API (static Gpt_HwChannelInfoType returnHwChannelInfo) which means that the value is shared among the cores when this shouldn't to avoid data to be overwritten by each others core. This issue can be happended with Gpt_GetTimeRemaining, Gpt_StopTimer, Gpt_SetMode because of the change in AMNG-980 Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: While using the Gpt_GetTimeElapsed API in Multicore, this doesn't report the correct value at some point. Expected behavior: Gpt_GetTimeElapsed report the correct value Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Delete static keyword. Then the variable variables will be put into stack, and because each core will have their own stack in multicore environment, so this variable will not be concurrently accessed. By my opinion, the root sause is: There are two global variables '{_}Gpt_aChannelInfo{_}' and '{_}Gpt_aStopTime{_}' that store for each channel the '{_}Runtime Internal Context{_}' and the '{_}Timer Value when the Channel is Stopped{_}'; these varibles do not take into account the multicore context {_}*Proposed Solution*{_}: Extending of these variables as arrays of number of partitions #if (GPT_MULTICORE_ENABLED == STD_ON) / brief Global array variable used to store runtime internal context of each logic channel. / static Gpt_ChannelInfoType Gpt_aChannelInfoo[GPT_MAX_PARTITIONS][GPT_HW_CHANNEL_NUM]; / brief Global array variable used to store for each channel the time value when it is stopped / static volatile Gpt_ValueType Gpt_aStopTime[GPT_MAX_PARTITIONS][GPT_HW_CHANNEL_NUM]; #else / brief Global array variable used to store runtime internal context of each logic channel. / static Gpt_ChannelInfoType Gpt_aChannelInfo[1U][GPT_HW_CHANNEL_NUM]; / brief Global array variable used to store for each channel the time value when it is stopped / static volatile Gpt_ValueType Gpt_aStopTime[1U][GPT_HW_CHANNEL_NUM]; #endif Also we should remove the static keywword because '{_}returnHwChannelInfo{_}' is just a local variable used as parameter for '{_}Gpt_Ipw_GetTimeElapsed({_})' function and we don't need to keep its value upon every call of function.
ARTD-71213	Bug	[ocu] Channel Value (CV) register and Modulo (MOD) not reset after calling Deinit Detailed description (how to reproduce it): Do not reset Channel Value register after calling Deinit() !image-2023-05-24-17-24-37-383.png|width=1033,height=482! Do not reset Modulo (MOD) register after calling Deinit() !image-2023-05-24-17-26-44-745.png|width=1181,height=479! Because when resetting the CSC = 0UL register, the channel mode will change to Dual Edge Capture modes. In Input Capture , Capture Test, and Dual Edge Capture modes, any write to a CnV register is ignored, detail in attach image. !image-2023-05-24-17-30-55-477.png|width=829,height=350! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Ip_Ftm_TS_001 Observed behavior: No reset CnV register and MOD register Expected behavior: reset CnV register and MOD register

ID	Subtype	Headline and Description
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: You can change the line code in the file Ftm_Ocu_Ip.c, detailed in the attached image. !image-2023-05-24-17-32-51-830.png|width=1033,height=477!
ARTD-71634	New Feature	[CXPI][ITG] Create preliminary file structure for Cxpi test NewWorkDescription: Create preliminary file structure for Cxpi in order to have a successful build of the module. Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: add template files (C headers and sources, templates for generated files, XDM schema file for M24X platform and Makefiles). add support for Cxpi module in Zth and include this module in RTD 4.7 S32K1XX S32M24x 2.0.0 release manifest.

ARTD-71695	Bug	[MEM_EXFLS][QSPI] Add support to escape from being stuck in BUSY state
		
		The customer's code will be stuck in the BUSY state after several read/write operations, until timeout, and loopback again and then still BUSY and timeout.
		do
		{
		/ Add Fault Injection point for FR_ILLINE flag /
		MCAL_FAULT_INJECTION_POINT(FLS_FIP_FR_ERROR_ABORTSUSPEND);
		
		status = Qspi_Ip_ControllerGetStatus(controllerInstance);
		Fls_Qspi_u32ElapsedTicks = OsIf_GetElapsed(and Fls_Qspi_u32CurrentTicks, (OsIf_CounterType)QSPI_IP_TIMEOUT_TYPE);
		if ((STATUS_QSPI_IP_BUSY == status) and and (Fls_Qspi_u32ElapsedTicks >= Fls_Qspi_u32TimeoutTicks))
		{
		(void) Det_ReportRuntimeError((uint16)FLS_MODULE_ID, FLS_INSTANCE_ID, FLS_MAINFUNCTION_ID, FLS_E_TIMEOUT);
		status = STATUS_QSPI_IP_TIMEOUT;
		}
		}
		while (STATUS_QSPI_IP_BUSY == status);
		
		Customer's general boot flow:
		1. BootROM(QSPI AHB): using QSPI reconfig parameter 133MHz DDR mode
		2. bootloader(QSPI AHB)-> reuse BootROM settings
		3. M7 AutoSRA(QSPI AHB)-> will call Fls_Init function, after do some write and read, using AHB read via DMA to load uboot code
		4. U-boot
		
		Questions:
		1. The API to reset QSPI after the timeout invoked by monitoring the ‘BUSY’. Now, we’re trying with Fls_IPW_InitControllers. Please see the attached mail to get the detailed modification. This seems can fix their currently BUSY issue. Is it reasonable ?
		
		2. If the AHB reading is invoked during the Fls operation, is it possible to get the QSPI stuck at ‘BUSY’ state ?
		3. The API to provide the status of flash/qspi which could tell that the AHB reading by the other software module is safe. Currently, below APIs are being used.
		/ Start DMA transfer /
		if(1==Fls_GetStatus() and and (2==Fls_GetJob()))
		{
		/*AHB read code*/
		}
		4. The recommended setting of the timeout to monitor the ‘BUSY’.
		Please see the attached mail to get the details. Do we have some suggestions about those timeout values?

ARTD-72526	Bug	[S32K1_M24X RTM 2.0.0] GPT: Can not config GptPredefTimer in S32CT Detailed description (how to reproduce it): [When I config GPT PredefTimer_1us_16Bit at derivative K148, I see that I can not config any node of "PredefTimerConfiguration" in this below: !image-2023-06-07-11-25-57-079.png! Beside that, when I open this config file in tresos, I have found another issue related GptPredefTimer. When I change value of node GptPredefTimerFunctionalityApi from false to true !image-2023-06-07-11-42-19-941.png! GptPredefTimer1usEnablingGrade node permits me choose option GPT_PREDEF_TIMER_1US_16_24BIT_ENABLED and GPT_PREDEF_TIMER_1US_16_24_32BIT_ENABLED although FTM counter just support 16 bit counter. !image-2023-06-07-11-45-48-210.png! At that time, GptPredefTimer_1us_32Bit and GptPredefTimer_1us_24Bit can be enable. !image-2023-06-07-11-43-12-212.png!

ID	Subtype	Headline and Description
		Test Case ID (internal TC that caught the defect) optional: [Gpt_TS_M09] Observed behavior: [..] Expected behavior: [S32CT: It could be config as tresos Tresos: Should not show 2 option: GPT_PREDEF_TIMER_1US_16_24BIT_ENABLED and GPT_PREDEF_TIMER_1US_16_24_32BIT_ENABLED if not support FTM have more 16bit-counter. GptPredefTimer_1us_32Bit and GptPredefTimer_1us_24Bit can not be enable.] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [None]
ARTD-72761	Bug	[S32K1_M24X RTM 2.0.0] GPT: GptPredefTimer do not support 24 and 32 bits Detailed description (how to reproduce it): [When I change value of node GptPredefTimerFunctionalityApi from false to true by tresos !image-2023-06-12-10-27-04-255.png! GptPredefTimer1usEnablingGrade node permits me choose option GPT_PREDEF_TIMER_1US_16_24BIT_ENABLED and GPT_PREDEF_TIMER_1US_16_24_32BIT_ENABLED although FTM counter support only 16 bit counter. !image-2023-06-12-10-29-40-046.png! At that time, GptPredefTimer_1us_32Bit and GptPredefTimer_1us_24Bit can be enable. !image-2023-06-12-10-29-19-759.png! Although, tresos notice that we have not any channel meet Predef timer 24 and 32 bits feature and in CT did correct thing that we could not config any Predef 24 and 32 bits node. !image-2023-06-12-10-17-34-089.png! !image-2023-06-12-10-18-07-567.png! When I try to config 24 and 32 bits counter by tresos, the result log have no instance at build phase !image-2023-06-12-10-19-29-759.png!] Preconditions: [..] Test Case ID (internal TC that caught the defect) optional: [Gpt_TS_M09] Observed behavior: [..] Expected behavior: [Both Tresos and CT should surround the two option in K1XX and M24X, user might choose GPT_PREDEF_TIMER_1US_16BIT_ENABLED and GPT_PREDEF_TIMER_1US_DISABLED instead they must aware that FTM support only 16 bit counter.] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [..]
ARTD-72825	Bug	[ICU]Difference between files generated by EB and S32CT Detailed description (how to reproduce it): There are some differences between the generated files generated on the EB and S32DS interfaces, files generated from eb and s32ds must be the same The difference is described in detail in the attached file there is an incorrect channel index error generated by s32ct in the file Icu_Ipw_PBcfg.c Preconditions: Use eb and s32ct same configuration on interface Test Case ID (internal TC that caught the defect) optional: Icu_TC_WBT_0025,Icu_TC_WBT_009,0Icu_TC_WBT_0024(Icu_TS_037) Observed behavior: some files generated are not the same when configured the same on the EB and S32DS interface Expected behavior: variable values, macro need to be same when config same on EB and S32DS Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)

ID	Subtype	Headline and Description
		Proposed solution optional: need to check the condition so that the variables ,marco are the same when configuring the same on the EB and S32DS interfaces
ARTD-72839	Bug	[QDEC] Code generated by Tresos and S32CT are different Detailed description (how to reproduce it): There are some differences between the gen code on tresos and s32ct. See more detail in attachment files Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Code generate is different Expected behavior: Code generate between EB and CT is same Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA

ARTD-73099	Bug	[ICU]Fix some bugs related to CMP
		
		Detailed description (how to reproduce it):
		there are some bugs related to Comparator (CMP)
		Isssue 1:
		the value of FILTER_CNT cannot be set to the value 3,4,5,6,7 in the driver, although in some modes of CMP these values can be used according to RM such as:
		CMP_SAMPLED_FILTRED_INT_CLK,CMP_SAMPLED_FILTRED_EXT_CLK,CMP_WINDOWED_FILTRED(C0[FILTER_CNT]>1)
		the value of FILTER_CNT cannot be set to the value 1 in the driver in CMP_WINDOWED_RESAMPLED mode:
		case CMP_IP_FUNCTIONALMODE_WINDOWED_RESAMPLED:
		Cmp_C0 = CMP_C0_EN(1U);
		Cmp_C0 = CMP_C0_WE(1U);
		Cmp_C0 = CMP_C0_SE(0U);
		Cmp_C0 = CMP_C0_FPR(userConfig->Comparator.FilterSamplePeriod);
		break;
		the value of SE cannot be set to the value 1 in the driver in CMP_IP_FUNCTIONALMODE_SAMPLED_FILTERED_EXT_CLK mode:
		Cmp_C0 = CMP_C0_EN(1U);
		Cmp_C0 = CMP_C0_WE(0U);
		Cmp_C0 = CMP_C0_SE(0U);
		Cmp_C0 = CMP_C0_FILTER_CNT(2U);
		Cmp_C0 = CMP_C0_FPR(userConfig->Comparator.FilterSamplePeriod);
		RoundRobinEnChannelMask value incorrectly written to ACOn register(this is the field to write to the pre-set state of channel n):
		Cmp_C2 = CMP_C2_ACOn(userConfig->Trigger.RoundRobinEnChannelMask);
		element redundancy:
		boolean EnableComparatorInvert; in struct type Cmp_Ip_ComparatorConfigType because CMP does not support this bit
		FilterSamplePeriod is not set in the modes
		CMP_IP_FUNCTIONALMODE_SAMPLED_NONFILTERED_EXT_CLK,CMP_IP_FUNCTIONALMODE_SAMPLED_FILTERED_INT_CLK
		
		Isssue 2:
		In CMP trigger mode round robin, the driver also does not support writing to the CHNn bit of the C1 register, which makes the CHnF interrupt flag of the C2 register in robin mode also not set to 1.
		the interrupt handler function handles incorrectly because the interrupt flag of
		robin mode CHnF of register C2 is not related to interrupt flags CFR,CFF of register C0:
		if (flag_Cmp and and (flag_Cmp flag_RR))
		{
		/ clear w1c bits /
		base->C0 = CMP_C0_CFF_MASK CMP_C0_CFR_MASK;
		base->C2 = CMP_C2_CH0F_MASK CMP_C2_CH1F_MASK CMP_C2_CH2F_MASK CMP_C2_CH3F_MASK
		CMP_C2_CH4F_MASK CMP_C2_CH5F_MASK CMP_C2_CH6F_MASK CMP_C2_CH7F_MASK;
		if (Cmp_Ip_axState[instance].NotificationEnabled)
		{
		if (NULL_PTR != Cmp_Ip_axState[instance].ComparatorNotification)
		{
		if (0U != flag_Cmp)
		{
		Cmp_Ip_axState[instance].ComparatorNotification(Cmp_Ip_axState[instance].CallbackParam1, FALSE);
		}
		}
		if (NULL_PTR != Cmp_Ip_axState[instance].TriggerNotification)
		{
		if (0U != flag_RR)
		{
		Cmp_Ip_axState[instance].TriggerNotification(Cmp_Ip_axState[instance].CallbackParam1, FALSE);
		}
		}
		}
		}

ID	Subtype	Headline and Description
		Issue 3:The function Cmp_Ip_Deinit does not reset the values of the bits of the C0,C1,C2 registers Preconditions: configure on the interface and use some functions of CMP Test Case ID (internal TC that caught the defect) optional: Icu_TS_116 Observed behavior: There are some bugs as described above Expected behavior: fix the above errors Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: in the modes CMP_SAMPLED_FILTRED_INT_CLK,CMP_SAMPLED_FILTRED_EXT_CLK,CMP_WINDOWED_FILTRED to capture FILTER_CNT values in the range 2 to 7, need to change the statement: Cmp_C0 = CMP_C0_FILTER_CNT(2U) to Cmp_C0 = CMP_C0_FILTER_CNT(userConfig->Comparator.FilterSampleCount) add statements so that FilterSamplePeriod is set in the modes CMP_IP_FUNCTIONALMODE_SAMPLED_NONFILTERED_EXT_CLK,CMP_IP_FUNCTIONALMODE_SAMPLED_FILTERED_INT_CLK The RoundRobinEnChannelMask value generated on the interface needs to be changed to write to the value of the CHNn bit of the C1 register. function Cmp_Ip_Deinit need reset value: bit IER,IEF,CFR,CFF,FPR,SE,WE,PMODE,INVT,COS,OPE,EN,FILTER_CNT,OFFSET,HYSTCTR,DMAEN of C0 register bit INPSEL,INNSEL,CHNn,DACEN,VRSEL,PSEL,MSEL,VOSEL of C1 register bit RRE,RRIE,FXMP,FXMXCH,CHnF,NSAM,INITMOD,ACOn of C2 register
ARTD-73171	Bug	[QDEC][S32DS] Build fail when config variant = 0 Detailed description (how to reproduce it): Code generate on S32DS when config variant = 0 is not correct, it still include VS_0 !image-2023-06-14-17-29-33-141.png! In file Qdec_Cfg.h !image-2023-06-16-09-13-53-157.png! Preconditions: Config ECUC with Post build variant is 0 Config ComponentGenerationMethod is EcucPostBuildVariants Test Case ID (internal TC that caught the defect) optional: Qdec_TS_002 Observed behavior: build fail Expected behavior: build success Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-73364	Bug	[ICU]Missing resource for S32M24X Detailed description (how to reproduce it): some resources are missing in some IPvalts: FTM,CMP and PORT_CI FTM:mising channel FTM_2_CH_6,FTM_2_CH_7 PORT_CI: PORT_0:0->17 PORT_1:0->17 PORT_2:0->17 PORT_3:0->17 PORT_4:0->16 redundant resources for CMP CMP use pins only:CMP0_IN2,CMP0_IN4,CMP0_IN5 Preconditions: check resource for S32M24X Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Missing resource for S32M24X Expected behavior: add full correct resource for S32M24x Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)

ID	Subtype	Headline and Description
		Proposed solution optional: add resources for FTM,CMP and PORT_CI for S32M24x
ARTD-73377	Bug	[Pwm] Fixing fail test at generate and build - S32K1XX Detailed description (how to reproduce it): When config channel ID fail, except logs to be error : !image-2023-06-16-12-12-09-943.png|width=461,height=473! but cfg_sets = s32k116_qfn32 : !image-2023-06-16-12-14-39-536.png|width=388,height=462! void Pwm_Ipw_FlexioNotification function undefined but appear extern void Pwm_Ipw_FlexioNotification ==> Build fail ! image-2023-06-16-13-09-41-623.png! When generate test , there are some differences between eb tresos and s32ct !image-2023-06-27-09-19-00-371.png|width=824,height=295! Preconditions: Generate on Eb tresos and CT and compare generated files Test Case ID (internal TC that caught the defect) optional: Pwm_TS_COT_003 CFG_SETS=s32k116_qfn32 Pwm_TS_100_Flexio deravative : s32k148 Pwm_TS_COT_008 CFG_SETS= 3 Observed behavior: Pwm_TS_COT_003 : when config fail channel ID , unexpected log out error Pwm_TS_100_Flexio : Pwm_Ipw_FlexioNotification function undefined Expected behavior: Pwm_TS_COT_003 : when config fail channel ID , error expect : Value \"CH_\" of node \"/AUTOSAR/TOP-LEVEL-PACKAGES/Pwm/ELEMENTS/Pwm/PwmChannelConfigSet/PwmFtm/PwmFtm_0/PwmFtmCh/PwmFtmCh_0/PwmFtmChId\" not in range \"[CH_0, CH_1, CH_2, CH_3, CH_4, CH_5, CH_6, CH_7]" Pwm_TS_100_Flexio : Define Pwm_Ipw_FlexioNotification function Pwm_TS_COT_008 : Similarity between generated files Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-73538	Bug	[ADC][PDB_IP] The hardfault error when initializing Adc_DeInit => Pdb_Adc_Ip_Init Detailed description (how to reproduce it): The hardfault error when use Adc_DeInit() function for S32M24x. Here Pdb_Adc_Ip_Init() not configuration for CH2C1 register of PDB0. Address of CH2C1 register in Reference Manual is 4003_6060h. But when run then this is an unknown register address area. leads to a jump into Hardfault(). Not configures for registers have address bigger than 4003_6060h. Detail see attach file. Preconditions: Adc_DeInit => Adc_Ipw_DeInit => Pdb_Adc_Ip_DeInit => Pdb_Adc_Ip_Init => Pdb_Adc_HwAcc_ConfigAdcPretriggers In the Pdb_Adc_HwAcc_ConfigAdcPretriggers function, it configures the CH2C1 value at the register address 4003_6060h (PDB0: 4003_6000h) Test Case ID (internal TC that caught the defect) optional: Adc_TC_WIR_0001.c Observed behavior: The hardfault error when use Adc_DeInit() function for S32M24x. Expected behavior: The Adc_DeInit() configures for PDB_IP module is success Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-73573	Bug	[ICU]Input filter FILTER value of FTM channels is not set to 0 after DeInit Detailed description (how to reproduce it): When using multiple channels, the input filter value is not reset to 0 after calling the DeInit function:

ID	Subtype	Headline and Description

		/ Clear filter for each chanel that support this feature. /
		if (channel < (uint8)FTM_FILTER_MAX_NO_CH)
		{
		ftmIcuBase[instance]->FILTER = ((uint32)FTM_FILTER_CH0FVAL_MASK << ((uint32)4U (uint32)channel));
		}
		The value of input filter of hwChannel channel is also set incorrectly in period time mode:
		if (hwChannel < CHAN4_IDX)
		{
		Ftm_Icu_Ip_SetChnInputCaptureFilter(ftmBase, (uint8)hwChannel, (*userConfig->pChannelsConfig)[index].filterValue);
		#if (STD_ON == FTM_ICU_SIGNAL_MEASUREMENT_API)
		if ((FTM_ICU_PERIOD_TIME != Ftm_Icu_Ip_ChState[instance][hwChannel].measurement)and and \\
		(FTM_ICU_MODE_SIGNAL_MEASUREMENT == (*userConfig->pChannelsConfig)[index].chMode))
		{
		Ftm_Icu_Ip_SetChnInputCaptureFilter(ftmBase, (uint8)(hwChannel 1U), (*userConfig->pChannelsConfig)[index].filterValue);
		}
		#endif
		Preconditions:
		Use filters in some cases
		
		Test Case ID (internal TC that caught the defect) optional:
		Icu_TS_103
		
		Observed behavior:
		Filter values of channels are set incorrectly in some cases
		
		Expected behavior:
		Filter values of channels are set correctly in some cases
		
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)
		
		Proposed solution optional:
		replace statements:
		
		ftmIcuBase[instance]->FILTER = ((uint32)FTM_FILTER_CH0FVAL_MASK << ((uint32)4U (uint32)channel))
		by
		ftmIcuBase[instance]->FILTER and = ((uint32)FTM_FILTER_CH0FVAL_MASK << ((uint32)4U (uint32)channel));
		
		must write value chMode before that set filter value

ARTD-73744	Bug	[I2C] Analyzing and Re-structure the inclusion between layers according to Autosar Standard Detailed description (how to reproduce it): In current implementation in i2c driver, IPW.c file is calling CDD.h file to get some typedef struct on Types.h file. This doesn't follow to inclusion of RTD project. !MicrosoftTeams-image (1).png|thumbnail!. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: IPW layer include CDD layer Expected behavior: Should be CDD include IPW include IPV Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Analyze and re-struct all layers in repo
ARTD-73776	Bug	[Ocu] Get wrong channel when calling Ocu_SetPinState and Ocu_EnableNotification Detailed description (how to reproduce it): Configure FTM_1_CH_4 but calling Ocu_SetPinState calls instance 1 channel 255. detail in attach image: !image-2023-06-21-16-54-56-067.png|width=749,height=329! Get value when debugging. !image-2023-06-21-16-53-55-824.png|width=684,height=434! API Ocu_EnableNotification: Configure FTM_1_CH_0 but calling Ocu_EnableNotification calls instance 1 channel 255. detail in attach image: !image-2023-06-27-16-56-06-257.png|width=1104,height=385! I think the cause is getting the value of channelIdx. !image-2023-06-22-08-52-20-338.png|width=1319,height=630! The configuration is attached below. Preconditions:

ID	Subtype	Headline and Description
		N/A Test Case ID (internal TC that caught the defect) optional: Ocu_TS_001 Observed behavior: Get wrong channel when calling Ocu_SetPinState Get wrong channel when calling Ocu_EnableNotification Expected behavior: Get correct channel when calling Ocu_SetPinState Get wrong channel when calling Ocu_EnableNotification Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-73799	New Feature	[Mem_Eep] [S32K1XX] Ftfc_Eep_Ip_Write propose improvement to prevent potential bug NewWorkDescription: Customer used Ftfc_Eep_Ip_Write() to write data from RAM to FlexRam 1.pu8SrcAddress input parameter is 0x2000015f (unaligned with 4byte) 2.PageSize is 4 byte 3.FTFC_EEP_IP_ALIGNED_RAM_ACCESS option is STD_OFF When debug until function Ftfc_Eep_Ip_WriteIntoFlexram(): FTFC_EEP_IP_ALIGNED_RAM_ACCESS is off so RTD's EEPROM driver does not care for unaligned addresses of pu8SrcAddress and causes a HardFault on write: !image-2023-02-28-16-33-14-384.png! HardFault occurs when the LDR instruction accesses an address that is not aligned with 4 bytes of R1 (0x2000015F) As the S32K118 ARM Coretex-M0 Reference Manual states that access to unaligned addresses causes a HardFault !image-2023-02-28-16-34-23-197.png! Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Ftfc_Eep_Ip_Write should return Fail if parameter is invalid for pu8SrcAddress, or we should document this point about parameter input for this function must be aligned with 4byte (or 2byte) address if PageSize is 4Byte (or 2byte) in the User Manual For further information, please check the propose workaround from customer in attachment[^Request check the workaround.pdf]
ARTD-73884	Bug	[ocu] To investigate and update driver code according to errata ERR010856: Detailed description (how to reproduce it): a new errata for FTM Ip says that we need to follow the steps from the attached document Preconditions: to use FTM Ip Test Case ID (internal TC that caught the defect) optional: Ocu_TS_001 Observed behavior: Safe state is not removed from channel outputs after fault condition ends if SWOCTRL is being used to control the pin Expected behavior: If an FTM channel output is being controlled using the software output control register (FTM_SWOCTRL) and fault detection is also enabled for the channel, then when a fault is detected the output is forced to its safe value. However, when the fault condition has been cleared, the channel output will stay in the safe state instead of reverting to the value programmed by the FTM_SWOCTRL register. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: follow the steps from errata pdf
ARTD-73888	Bug	[ICU]Interrupt flag clear instruction is incorrect in case many channel occurs interrupt at the same time Detailed description (how to reproduce it): In the function interrupt of Lpit harware, there is mistake in processing interrupt in case many channel occurs interrupt at the same time. command line causing interrupt of the previous channel to clear the interrupt flag of the following channel: / Clear pending interrupt serviced. / lpitBase[instance]->MSR = (uint32)(LPIT_MSR_TIF_START_MASK << (uint32)channel); !image-2023-06-22-17-53-20-904.png|thumbnail!

ID	Subtype	Headline and Description
		Preconditions: use interrupts from two or more channels simultaneously Test Case ID (internal TC that caught the defect) optional: Icu_TC_WBT_0104(Icu_TS_083) Observed behavior: The instruction in the interrupt function causes the interrupt of the previous channel to clear the interrupt flag of the next channel Expected behavior: The interrupt of the previous channel does not clear the interrupt flag of the following channel Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: add a ChannelMask mask variable to not clear other bits

ARTD-74194	Bug	[FEE][ASRR21.11] Fee_Cancel() can not call NvM_JobErrorNotification
		
		Detailed description (how to reproduce it):
		
		called Fee_Cancel(), the Fee_Cancel() will call to Fls_Cancel(), then Fls_Cancel() call the call back jobErrorNotificationPtr Fee_JobErrorNotification(). in Fee_JobErrorNotification() it won't call back to FEE_NVM_JOB_ERROR_NOTIFICATION to notice upper layer the cancel is occurred.
		
		if (MEMIF_JOB_CANCELED == Fee_eJobResult)
		{
		/ Fls job has been canceled. Do nothing in this callback.
		The NvM_JobErrorNotification() callback will be called from the Fee_Cancel()function which called the Fls_Cancel() function /
		}
		
		in Fee_Cancel(), only the if (MEMIF_JOB_PENDING == Fee_eJobResult) is true, then it run into it. however, when called Fls_cancel(), the Fls_eJobResult will change to MEMIF_JOB_CANCELED, the (MEMIF_JOB_PENDING == Fee_eJobResult) won't be TRUE.
		
		So, that is one dead loop when customer call Fee_Cancel() and expect the NvM_JobErrorNotification() is called back.
		
		Preconditions:
		N/A
		
		Test Case ID (internal TC that caught the defect) optional:
		N/A
		
		Observed behavior:
		NvM_JobErrorNotification() can not be called by Fee_Cancel(). It not follow Fee_Cancel process of AUTOSAR special.
		
		Expected behavior:
		NvM_JobErrorNotification() should be called by Fee_JobEndNotification()
		
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)
		
		Proposed solution optional:
		N/A

ARTD-74560	Bug	[I2C] Create the CDD_I2C_CfgDefines.h file in the HLD layer. Detailed description (how to reproduce it): All files in HLD layer is doing incorrect include files (.c and .h). Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: The files in the HLD layer are not correctly include. Expected behavior: All include files at tiers need to be followed according to the rules. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Include files (.c and .h) that need to follow the rules: [https://nxp1.sharepoint.com/:p:/r/sites/Zebra/Shared%20Documents/General/Zebra %20Training/06_NXP_RealTImeDrivers_Configuration.pptx?d=w8385793d9bd44136815bd8c3fb7bc5bband csf=1and web=1] !_thumb_1014453.png|thumbnail!
ARTD-74618	Bug	[DPGA] DPGA is disabled in function "Dpga_Ip_ConfigureVoltageMonitor" Detailed description (how to reproduce it): Enable DpgaEnableVoltageMonitoring to raise interrupt

ID	Subtype	Headline and Description
		Preconditions: Plugins(DPGA_024) Test Case ID (internal TC that caught the defect) optional: Dpga_TC_FCT_0006 Observed behavior: In function "Dpga_Ip_ConfigureVoltageMonitor" CtrlReg is overwritten to another value so DPGA is also disable. !image-2023-06-28-16-43-50-785.png! Expected behavior: N/A Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Can enable DPGA at the end of Dpga_Ip_Init()
ARTD-74621	Bug	[SPI][S32K1] With Flexio SpiShiftClockIdleLevel = HIGH, for SpiDataShiftEdge = LEADING, must to generate SHIFTCTLn using negedge of clock Detailed description (how to reproduce it): Spi Flexio has 4 combinations of (SpiShiftClockIdleLevel, SpiDataShiftEdge): (LOW, LEADING) is generated correctly as (TIMCFGn initial clock state is logic 0, SHIFTCTLn using posedge of clock) (LOW, TRAILING) is generated correctly as (TIMCFGn initial clock state is logic 0, SHIFTCTLn using negedge of clock) But (HIGH, LEADING) must be generated as (TIMCFGn initial clock state is logic 1, SHIFTCTLn using negedge of clock) And (HIGH, TRAILING) must be generated as (TIMCFGn initial clock state is logic 1, SHIFTCTLn using posedge of clock) Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Lpspi_Flexio_Ip_Transfer Example Observed behavior: Spi Flexio generated code for 2 cases are wrong. Expected behavior: Spi Flexio generated code are correct. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-74805	New Feature	[Mem_Eep] Update IP layer requirements to start with Ftfc_Mem_Eep_ in the ReqID and in the text NewWorkDescription: Update IP layer requirements to start with Ftfc_Mem_Eep in the ReqID and in the text. To avoid clashes between Mem_Eep and Mem_InFls reqs and functions Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Update IP layer requirements to start with Ftfc_Mem_Eep in the ReqID and in the text
ARTD-74808	New Feature	[Mem_InFls] Update IP layer requirements to start with Ftfc_Mem_InFls_ in the ReqID and in the text NewWorkDescription: Update IP layer name of Ftfc driver from Ftfc_Fls_Ip to Ftfc_Mem_InFls_Ip To avoid clashes between Mem_Eep and Mem_InFls reqs and functions Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Update Ftfc IP driver
ARTD-74839	Bug	[S32K1_M24X RTM 2.0.0] Mem_Eep: Build fail in IAR Detailed description (how to reproduce it): test build fail when user iar compiler Preconditions: compiler is iar Test Case ID (internal TC that caught the defect) optional: All test Observed behavior: !image-2023-06-30-10-32-44-201.png|thumbnail!

ID	Subtype	Headline and Description
		Expected behavior: build pass with iar compiler Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-74844	Bug	[ICU] Not exits error when tick config other mode in EDGE_COUNT mode Detailed description (how to reproduce it): When choose mode edge counter, tick in other config for other mode but not exits warning or error !image-2023-06-30-11-11-05-867.png! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Icu_TS_037 Observed behavior: Not have error Log Expected behavior: Have error log when cofig Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-74857	Bug	[S32K1_M24X RTM 2.0.0] MemAcc: Build fail when use Mem_EEP config. Detailed description (how to reproduce it): Build fail when user Mem_Eep config Preconditions: user Mem_Eep config Test Case ID (internal TC that caught the defect) optional: All test Observed behavior: !screenshot-1.png|thumbnail! Expected behavior: build pass with Mem_Eep config Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Adding resource list: MEMACC_MEM_HW_EEP
ARTD-74898	Bug	[I2C] I2c_Init service cannot return I2C_E_INVALID_POINTER when driver initialize with invalid configuration pointer Detailed description (how to reproduce it): Enable Det error report Init I2c driver with invalid configuration pointer Check Det error report Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: I2c_TC_FCT_0200 Observed behavior: I2c_Init service cannot return I2C_E_INVALID_POINTER Expected behavior: Driver need to work correctly with I2c requirement: SWS_CDD_I2C_00004 Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-74949	Bug	[DPGA] Generate failed ECPD file on S32DS Detailed description (how to reproduce it): Fix codegenerator.js to generate ECPD file on S32DS Preconditions: plugins(DPGA_024)

ID	Subtype	Headline and Description
		Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: N/A !image-2023-07-03-14-00-51-333.png! Expected behavior: N/A Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-74996	Bug	[S32K1_M24X RTM 2.0.0] PORT: Inconsistent generated code between EB and S32DS in config variant Precompile Detailed description (how to reproduce it): Port_VS_0_PBcfg.h file is inconsistent between EB and CT Preconditions: config VARIANT-PRE-COMPILE Test Case ID (internal TC that caught the defect) optional: Port_TS_014 Observed behavior: Code generated is inconsistent between EB tresos and S32DS Data generated in Port_VS_0_PBcfg.h for eb and Port_VS_0_PBcfg.h for ct: Port_VS_0_PBcfg.h for eb: !image-2023-07-03-15-53-31-390.png! Port_VS_0_PBcfg.h for ct: !image-2023-07-03-15-54-14-995.png! Expected behavior: code generate between EB tresos and S32DS is consistent Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-75003	Bug	[icu] PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Detailed description (how to reproduce it): In readme.txt file, the following sentence: PLUGINS_DIR The path to the Tresos plugins directory !john_floros_0-1653582301493.png|width=696,height=261! In fact, PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: PLUGINS_DIR points to EB Tresos plugin folder in readme.txt, which is incorrect. Expected behavior: PLUGINS_DIR should point to RTD plugin folder instead of EB Tresos plugin folder in readme.txt. PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: PLUGINS_DIR should point to RTD plugin folder : PLUGINS_DIR The path to the RTD plugins directory Ex: PLUGINS_DIR = C:/NXP/SW32_RTD/eclipse/plugins
ARTD-75021	Bug	[S32K1_M24X RTM 2.0.0] GPT: Undefined pointer at LPTMR IP Detailed description (how to reproduce it): [The pointer pHwChannelConfig inside Lptmr_Gpt_Ip_Init() is used but undefined before. Detail at picture below: !image-2023-07-03-18-43-33-864.png!]

ID	Subtype	Headline and Description
		Preconditions: [Build by function at Lptmr IP] Test Case ID (internal TC that caught the defect) optional: [Gpt_TS_CO_003] Observed behavior: [] Expected behavior: [That pointer is defined.] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [.]
ARTD-75063	Bug	[I2C] Build fail with variant no greater than 1 Detailed description (how to reproduce it): Create test project with variant no greater than 1 (2 or 3) Generate code and build project Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: I2c_TC_FCT_0001 Observed behavior: Build fail with bellow error log: !image-2023-07-04-09-50-19-207.png! Expected behavior: Project passing build with all variant no cases Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-75080	Bug	[ADC] Driver build fail on ghs compiler due to last line of file ends without a newline Detailed description (how to reproduce it): Driver build fail on ghs compiler due to last line of file ends without a newline !image-2023-07-04-13-39-05-019.png|width=671,height=248! Preconditions: ADC_437 Test Case ID (internal TC that caught the defect) optional: Adc_TS_Wir_01 Observed behavior: See the description Expected behavior: Build successfully Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-75076	Bug	[MCU] Wrong SOSC s32ct code generation Detailed description (how to reproduce it): In EB, user can configurable: SCG_SOSCCFG[HGO] by McuSOSCHighGainOscillatorSelect SCG_SOSCCFG[EREFS] by McuSOSCExternalReferenceSelect (picture 1 and 3) Issue 1 in S32DS, code generation for EREFS bit is not update when SOSC configurable external reference clock (picture 2) Issue 2: User config SOSC using high internal crystal (picture 4) but Mcu component checked in to HGO and EREFS both. (picture 5). EREFS should be 1 in this mode. This mean Select external reference clock should be unchecked. Preconditions: s32ct,sosc

ID	Subtype	Headline and Description
		Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: can not configurable EREFS Expected behavior: code generation for EREFS is corresponding to SOSC configuration Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-75107	Bug	[Dpga] No error for unsupported derivative Detailed description (how to reproduce it): Install the release, create a new project, add Dpga, set the resources s32k358_mapbga289, generate code for the current project. Preconditions: EBT 29.2.0 SW32K3_RTD4.4_R21-11_3.0.0_D2303.exe Test Case ID (internal TC that caught the defect) optional: Observed behavior: The Dpga is not applicable on S32K derivative. Expected behavior: To have an error for it when Dpga is added in the project. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Check it in the configurator and give us an error.
ARTD-75115	Bug	[S32M24x] [Gdu] [S32K1_M24x 2.0.0] Generate false config struct for INTEN field Detailed description (how to reproduce it): When enable low side interrupt node in tresos, the generated config struct won't match the config in EB. Preconditions: GduDesaturationLs0IE, GduDesaturationLs1IE, GduDesaturationLs2IE are enabled. Test Case ID (internal TC that caught the defect) optional: Observed behavior: Generated struct have the wrong values. The struct point to the wrong node and have the wrong mask for INTEN bit field. Expected behavior: The INTEN register value should match the configuration in EB. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update source code of the generate config struct part to have it take the value of the correct node and use the correct mask in the struct.
ARTD-75120	Bug	[S32K1_M24X RTM 2.0.0] PORT: (S32K144W) Generate error when importing epc file from EBT into S32DS in ECPD test Detailed description (how to reproduce it): Get error at 2 two pins PTE4 PTE6 in pin tools when import epc file from EBT to S32DS S31K144W LQFP64 Preconditions: 1. Create a new mex configuration in S32DS 2. Generate ecvd and ecpd file 3. Create a project on EBT and import ecvd file, create epc file 4. Import epc from EBT into S32DS Test Case ID (internal TC that caught the defect) optional: PORT_TS_COT_019 Observed behavior: errors at PTE4 PTE6 on pin tools after importing epc from EBT into S32DS !image-2023-07-04-16-56-44-963.png|width=936,height=255! error when import ecvd file from DS to EB

ID	Subtype	Headline and Description
		!image-2023-07-04-17-19-22-715.png|width=534,height=381! !image-2023-07-04-17-19-59-308.png|width=533,height=440! S32K148_LQFP176 {*}{{*}}error on EB when import ecvd file from DS to EB (mscr =153) !image-2023-07-04-17-43-58-555.png|width=837,height=460! {_}*S32K144_LQFP100*{_}{*}{{*}}{_}error when import epc from eb into s32ds, no data upload{_}{*}{{*}}{*}{{*}}{_}*! image-2023-07-05-14-08-23-892.png|width=725,height=308!*{_}{*}{{*}} S32K146_LQFP144 {_}errors at pin Jtag on pin tools after importing epc from EBT into S32DS{_}{_}!image-2023-07-05-14-11-52-000.png| width=1116,height=527!{_} {_}*Expected behavior:*{_}No error appears Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-75132	Bug	[MCU] TRACEDIV and TRACEFRAC can not set to 0 Detailed description (how to reproduce it): Step 1: Configurable TRACE clock, set trace divider = 1 and trace fraction multiply =1 (picture 1) Step 2: Generate code and debug into Clock_Ip_SetSimTraceDivMul_TrustedCall() function. Issue 1: TRACEDIV and TRACEFRAC can not set to 0 due to Clock_Ip_SetSimTraceDivMul_TrustedCall() not has action to clear bit (picture 2). IP_SIM->CLKDIV4 should be read out and clear TRACEDIV, TRACEFRAC bit field before write into it. TRACEDIVEN should be disabled before set TRACEDIV and TRACEFRAC (picture3). Please check all peripheral which has fraction divider (LPTMR). Issue 2: Trace_clk not has undercontrol button in s32ct. Preconditions: trace_clk Test Case ID (internal TC that caught the defect) optional: Mcu_TC_FCT_0668 Observed behavior: Wrong trace divider setting Expected behavior: tracediv and tracefrac can be set to 0 Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Add step read, clear TRACEDIV, TRACEFRAC bit field before write divider into SIM->CLKDIV4 in Clock_Ip_SetSimTraceDivMul_TrustedCall() function
ARTD-75138	Bug	[S32K1_M24X RTM 2.0.0] GPT: Type is not allowed before parsing the parmeter when call a function Detailed description (how to reproduce it): [Type name of "uint8" is not allowed here when parsing instance. Detail in picture below: !image-2023-07-05-08-16-39-971.png!] Preconditions: [.] Test Case ID (internal TC that caught the defect) optional: [Gpt_TS_CO_003, Ip_Gpt_SRtc_TS_COT_002, Gpt_TS_M03, Gpt_TS_M10] Observed behavior: [.] Expected behavior: [Remove uint8 here.] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [..]
ARTD-75144	Bug	[S32K1_M24x 2.0.0] [Gdu] Gdu have compiler warnings Detailed description (how to reproduce it): Gdu has compiler warnings Preconditions: Gdu has no compiler warnings Test Case ID (internal TC that caught the defect) optional: All compile only test case

ID	Subtype	Headline and Description
		Observed behavior: Gdu has compiler warnings Expected behavior: Gdu has no compiler warnings Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: n/A
ARTD-75183	Bug	[PLATFORM]: Build failed on IAR Compiler Detailed description (how to reproduce it): On Compiler IAR Error[Lc036]: no block or place matches the pattern "ro code section .acmem_43_INFLS_code_rom in Ftfc_Fls_Ip_Ac_c.o symbols: [Ftfc_Fls_Ip_AccessCode]" Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Fee_TC_FCT_0101 Observed behavior: N/A Expected behavior: Test pass Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Add section acmem_43_INTFLS_code_rom to linker file. It's implemented in commit 1547ab47 of ARTD-67963
ARTD-75175	Bug	[PWM] Warnings in compiler warning report Detailed description (how to reproduce it): There are some warnings in the Compiler Warnings report. For more details, please check in Attachment file below. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: warnings in the Compiler Warnings report Expected behavior: No warnings in the Compiler Warnings report Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-75221	Bug	[I2C] Cannot enable user mode for I2c module Detailed description (how to reproduce it): According the CPR_RTD_00352.i2c requirement, I2c driver shall be able to run in user mode. But cannot be configure node I2cEnableUserModeSupport on Tresos !image-2023-07-05-14-00-05-939.png! !image-2023-07-05-13-58-00-763.png! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: I2c_TS_001 Observed behavior: See the description Expected behavior: I2cEnableUserModeSupport node can be configured by user

ID	Subtype	Headline and Description
		Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-75242	Bug	[Gpt] Fix some issue about compiler warnings Detailed description (how to reproduce it): For detail, please see excel file below. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: NA Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-75245	Bug	[S32K1_M24X RTM 2.0.0] GPT: Undefined macro FTM_PROT_MEM_U32 Detailed description (how to reproduce it): [Macro FTM_PROT_MEM_U32 in FTM_Gpt_Ip.c is undefined in this below: !image-2023-07-05-15-33-55-553.png! and here: !image-2023-07-05-15-40-28-506.png!] Preconditions: [.] Test Case ID (internal TC that caught the defect) optional: [Gpt_TS_CO_003, Ip_Gpt_Ftm_TS_COT_002] Observed behavior: [.] Expected behavior: [The macro is defined.] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [..]
ARTD-75282	Bug	[I2C] The function Lpi2c_Ip_MasterIRQHandler and Lpi2c_Ip_SlaveIRQHandler not follow requirement CPR_RTD_00664.i2c, CPR_RTD_00011.i2c Detailed description (how to reproduce it): Not yet implement requirement CPR_RTD_00664.i2c, CPR_RTD_00011.i2c for Lpi2c_Ip_MasterIRQHandler and Lpi2c_Ip_SlaveIRQHandler !image-2023-07-05-17-12-32-475.png|width=508,height=456! !image-2023-07-05-17-13-58-883.png|width=504,height=593! Preconditions: Not defined LPI2C_IP_COMMON_IRQ_MASTER_AND_SLAVE. Using separate interrupt handler for master and slave Test Case ID (internal TC that caught the defect) optional: I2c_TS_103 Observed behavior: See the description Expected behavior: The handle interrupt function follow requirement Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)

ID	Subtype	Headline and Description
		Proposed solution optional:
ARTD-75287	Bug	[I2C] The condition to check the spurious interrupt need to be strictly Detailed description (how to reproduce it): The condition to check the spurious interrupt need to be strictly. It is necessary to check that interrupt enable bit and interrupt flag bit need to be set together. !image-2023-07-11-23-53-13-819.png|width=449,height=218! Preconditions: Defined LPI2C_IP_COMMON_IRQ_MASTER_AND_SLAVE Test Case ID (internal TC that caught the defect) optional: I2c_TS_006 Observed behavior: See the description Expected behavior: Driver implement correctly Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Maybe correct the condition as bellow if ((MasterIsrEnable and MasterIsrStatus) (SlaveIsrEnable and SlaveIsrStatus))
ARTD-75316	Bug	[ICU] LPTMR must not support ICU_BOTH_EDGES Detailed description (how to reproduce it): Driver only sussport rising and falling edge, not sussport both of edge Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: [Ip_Lptmr_TS_COT_002|http://panama.ea.freescale.net/0/project/fileexplorer/ ZTpcbG9jYWxfMDZcb3V0cHV0XGFydGlmYWN0c1xhcl9jb19mb3JfY3RfaWN1X2lhclwyMDIzMDcwNTA1MjI1MDE0MjAwMFx0YXJnZXRcYnVpbGRcS Observed behavior: !image-2023-07-06-10-34-44-427.png! !image-2023-07-06-10-37-05-087.png! Expected behavior: Not exits config for both of edge or have error message Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]

ARTD-75319	Bug	[ICU]Function Port_Ci_Icu_Ip_GetInputState does not exist Interrupt enable flag check condition
		
		Detailed description (how to reproduce it):
		Port_Ci_Icu_Ip_GetInputState function lacks interrupt enable flag check condition:
		The test conditions in the Port_Ci_Icu_Ip_GetInputState function are missing a check for
		Interrupt enable flag (IRQC)
		The DOZE_EN bit only works in the STOP1 mode of the MCU, so it needs to support the button on the interface to turn this feature on and off.
		
		lack of wake up feature for ipvalt LPIT:
		void Icu_Ipw_SetSleepMode(const Icu_Ipw_ChannelConfigType ChannelConfig)
		{
		uint8 channel;
		uint8 module = ChannelConfig->instanceNo;
		Icu_Ipw_ModuleType ipType = ChannelConfig->channelIp;
		
		/ Select IP type case. /
		switch(ipType)
		{
		case ICU_FTM_MODULE:
		{
		channel = (ChannelConfig->pFtmHwChannelConfig)->hwChannel;
		Ftm_Icu_Ip_SetSleepMode(module, channel);
		break;
		}
		case ICU_LPTMR_MODULE:
		{
		Lptmr_Icu_Ip_SetSleepMode(module);
		break;
		}
		case ICU_PORT_CI_MODULE:

ID	Subtype	Headline and Description

		{
		channel = (ChannelConfig->pPortCiHwChannelConfig)->pinId;
		Port_Ci_Icu_Ip_SetSleepMode(module, channel);
		break;
		}
		default:
		{
		/ Do nothing. /
		}
		break;
		}
		}
		
		Preconditions:
		use the function Port_Ci_Icu_Ip_GetInputState
		
		Test Case ID (internal TC that caught the defect) optional:
		Ip_Port_TC_FCT_6000(Ip_Port_TS_230)
		
		Observed behavior:
		Port_Ci_Icu_Ip_GetInputState function lacks interrupt enable flag check condition and add button to use on interface for bit DOZE
		
		Expected behavior:
		The Port_Ci_Icu_Ip_GetInputState function conditional checks the interrupt enable flag and button to use on the interface for the DOZE bit
		
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)
		
		Proposed solution optional:
		add a test condition for the IRQC bit in the Port_Ci_Icu_Ip_GetInputState function and
		button to use on the interface for the DOZE bit
		added wakeup feature for LPIT

ARTD-75336	Bug	[S32K1_M24X RTM 2.0.0] "DISABLE mode" configuration on pintool and eb tresos is inconsistent Detailed description (how to reproduce it): On a single port, configure more than one pin as "Disable Mode" (Pin tool and EB tresos) Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Error on S32DS and Warning on EB tresos s32ds: !image-2023-07-06-13-58-03-836.png! EB tresos: !image-2023-07-06-14-00-35-876.png! !image-2023-07-06-14-00-57-140.png! Expected behavior: configuration on EB and S32CT is consistent Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-75399	New Feature	[GDU] Add user mode support NewWorkDescription: Add a configuration parameter for user mode support Requirement source: CPR_RTD_00352.gdu Proposed solution optional: [....]
ARTD-75484	Bug	[MemAcc] The start address of subAddressArea does not match with size of sector batch Detailed description (how to reproduce it): when i set a value which does not match with size of sector batch at MemAccLogicalStartAddress node and saw that no any error as show in attched picture . Size of sector batch is 4096 but i set 2048 saw that no any error Preconditions: N/A

ID	Subtype	Headline and Description
		Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: when i set a value which does not match with size of sector batch at MemAccLogicalStartAddress node and saw that no any error as show in attched pciture Expected behavior: it must have error when user set a value does not match with size of sector batch Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: update check validation or automaton set value when user add more subaddressArea
ARTD-75621	Bug	[S32K1_M24X RTM 2.0.0] GPT: Enable User Mode when use SRTC lead to hardfault Detailed description (how to reproduce it): [Tester config GPT module with SRTC and enable user mode, so it lead to hardfault.] Preconditions: [Enable node GptEnableUserModeSupport when use SRTC Ip] Test Case ID (internal TC that caught the defect) optional: [GPT_TS_001] Observed behavior: [.] Expected behavior: [When user config SRTC and enable User mode at the same time, EB will log a error that SRTC cannot config with GPT enable User Mode] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: 1. Add code snippet below to '{_}SRtc_Ip.h{_}' file #if (STD_ON == SRTC_IP_ENABLE_USER_MODE_SUPPORT) #include "Reg_eSys.h" #endif 2. replace in the code defined(MCAL_SRTC_REG_PROT_AVAILABLE) by (defined(MCAL_SRTC_REG_PROT_AVAILABLE) and and (STD_ON == MCAL_SRTC_REG_PROT_AVAILABLE)) [..]
ARTD-75640	Bug	[Port] [S32K1] Epd test failure at generate Detailed description (how to reproduce it) Generate epd test with port.xdm (attached below) Preconditions: epd file gen from tag : PVT_PORT_S32K1XX_S32M24x_RTM_2.0.0_008 Test Case ID (internal TC that caught the defect) optional: Port_TS_COT_013 Observed behavior: Error: not in range "[GPIO]" !image-2023-07-10-16-59-07-206.png|width=836,height=633! 2. pcr index out out range when using epd file to config (142W,116...) XDM !image-2023-07-11-10-02-58-535.png|width=639,height=654! EPD !image-2023-07-11-10-03-10-365.png|width=933,height=443! Expected behavior: Generate successfully

ID	Subtype	Headline and Description
		Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-75753	Bug	[S32K1_M24X RTM 2.0.0] GPT: Can not get number IP hardware module Detailed description (how to reproduce it): [The fixed VSMD error caused an issue that cannot get number hardware module. !image-2023-07-11-08-51-19-241.png! In this picture above, EB tresos cannot get number hardware instance of FTM module and cause generation error when use more than one channel !image-2023-07-11-08-55-54-217.png! In the first picture, EB tresos just get only one element instead get number hw instance in resource !image-2023-07-11-09-01-10-755.png!] Preconditions: [Config with more channel or instance with FTM module.] Test Case ID (internal TC that caught the defect) optional: [GPT_TS_002] Observed behavior: [.] Expected behavior: [Gpt] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [The issue could be fixed like this : <a:da name="MAX" type="XPath" expr="num:i({*}count{*}({*}ecu:list{*}('Gpt.Num_Ftm_Hw_Modules')))"/> to <a:da name="MAX" type="XPath" expr="num:i({*}ecu:get{*}('Gpt.Num_Ftm_Hw_Modules'))"/> or to this: <a:da name="MAX" type="XPath" expr="num:i(count(ecu:list('{*}Gpt.GptChannelConfigSet.GptFtm.GptFtmModule.List{*}')))"/>]
ARTD-75848	Bug	[S32K1XX][MEM_EEP] Remove DEM reference for MEM_EEP at IP layer Detailed description (how to reproduce it): Remove DEM reference for MEM_EEP at IP layer Preconditions: Implement test with IP layer, enable production error reporting Test Case ID (internal TC that caught the defect) optional: Observed behavior: Build fail Expected behavior: Build pass Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:
ARTD-75854	New Feature	[S32K1XX][ICU] (ITG) Create new TC to cover FMEA requirement 00285 NewWorkDescription: Create new TC to cover this requirement: CPR_RTD_00285: Before recording the data read from a register, the driver shall filter the value read from the data register clearing all bit values that do not belong in the actual data bitfield. Configured data alignment in the register shall also be considered. Requirement source: CPR_RTD_00285 Proposed solution optional: Implement in another tickit ARTD-78681

ID	Subtype	Headline and Description
		Creat test case Icu_TC_FLT_001 for function Icu_GetInputLevel, Icu_GetEdgeNumbers, Lptmr_Icu_Ip_GetEdgeNumbers, Ftm_Icu_Ip_GetInputLevel
ARTD-75863	New Feature	[S32K1XX][OCU] (ITG) Create new TC to cover FMEA requirement 00285 NewWorkDescription: Create new TC to cover this requirement: CPR_RTD_00285: Before recording the data read from a register, the driver shall filter the value read from the data register clearing all bit values that do not belong in the actual data bitfield. Configured data alignment in the register shall also be considered. Requirement source: [..] Proposed solution optional: [..]
ARTD-75874	Bug	[Mem] Unspecified value when get InstanceID before initialization module Detailed description (how to reproduce it): If the module is UNINIT and the driver get InstanceID(or get any value from e.x: Mem_43_INFLS_pConfigPtr), the hard fault will be raise. !image-2023-07-11-18-14-55-531.png! Preconditions: Test Case ID (internal TC that caught the defect) optional: Observed behavior: Expected behavior: Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-75877	Bug	[S32K1_M24X RTM 2.0.0][I2S] Build fail due to missing define header file for M243 M241 Detailed description (how to reproduce it): Build fail due to missing define header file for M243 M241 Preconditions: Using Derivative S32M241 S32M243 Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Build fail due to missing define header file for M243 M241 Expected behavior: Build passed Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: !image-2023-07-12-13-43-44-045.png|thumbnail! !image-2023-07-12-13-43-53-190.png|thumbnail!
ARTD-75888	Bug	[S32K1_M24X RTM 2.0.0][Port] Fix compiler warnings Detailed description (how to reproduce it): Compiler warning report have waring on GCC sheet (attached file below) link station: [http://somov.ea.freescale.net/2/project/custom_compilerwarning/details|http://sulu.ea.freescale.net/2/project/custom_compilerwarning/details] Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Port_TS_COT_001 Port_TS_COT_005 Observed behavior: 167|unused parameter 'ConfigPtr' [-Wunused-parameter]| 167 \\| const Port_ConfigType ConfigPtr Expected behavior: No warning in compiler warning report

ID	Subtype	Headline and Description
		Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-75883	Bug	[S32K1_M24X RTM 2.0.0] MemAcc: MemAcc_Erase return not ok when start erase address correct but not align with sector size of current subaddressarea Detailed description (how to reproduce it): MemAcc: MemAcc_Erase return not ok when start erase address correct but not align with sector size of current subaddressarea Preconditions: adressarea contains multiple HW type Test Case ID (internal TC that caught the defect) optional: Memacc_TS_020 Observed behavior: MemAcc return E_NOT_OK Expected behavior: MemAcc return E_OK Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-75943	Bug	[ICU]Fix some warning and error for Example Detailed description (how to reproduce it): Wrong channel FTM in readme of example s32k118(EBtresos) !image-2023-07-12-16-10-24-641.png! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Icu_example_S32K148_EB Observed behavior: Wrong readme Expected behavior: Update to right readme Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-76035	Bug	[PWM] No fault interrupt for HLD Detailed description (how to reproduce it): When interrupt fault is enabled and fault is used, there is no fault handling function. Preconditions: Interrupt fault is enabled and fault is used. Test Case ID (internal TC that caught the defect) optional: Pwm_TC_FCT_0215 Observed behavior: there is no fault handling function. Expected behavior: there is fault handling function. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-77944	Bug	[S32K1_M24X RTM 2.0.0] OCOTP: Fix bug about DEV_ASSERT_OCOTP Detailed description (how to reproduce it): !screenshot-1.png|thumbnail! Preconditions: Generating TS pass Test Case ID (internal TC that caught the defect) optional: OCOTP_AE_TC_FLT_0001

ID	Subtype	Headline and Description
		Observed behavior: Build fail Expected behavior: Build pass Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update define DEV_ASSERT_OCOTP in generate file
ARTD-78581	Bug	[ICU]MARCO FTM_ICU_n_OVF_ISR_USED not generated on S32K11X platform Detailed description (how to reproduce it): when configuring using feature IcuOverflowNotificationApi marco FTM_ICU_n_OVF_ISR_USED is not generated Preconditions: using the IcuOverflowNotificationApi feature on the S32K11X Test Case ID (internal TC that caught the defect) optional: Icu_TC_WBT_8088(Icu_TS_036) Observed behavior: MARCO FTM_ICU_n_OVF_ISR_USED not generated on S32K11X platform Expected behavior: MARCO FTM_ICU_n_OVF_ISR_USED generated when using the feature Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: add the code in the file Ftm_Icu_Ip_Defines.h to generate this macro
ARTD-78621	Bug	I2C_EVENT_ERROR_FIFO_MASTER macro defined but not used Detailed description (how to reproduce it): I2C_MASTER_EVENT_ERROR_FIFO was renamed to I2C_EVENT_ERROR_FIFO_MASTER in ARTD-61584 but is not used !image-2023-07-18-17-13-10-877.png! Preconditions: Test Case ID (internal TC that caught the defect) optional: TS_0999 Observed behavior: Error in misra report: [ARTD CI-I2c 854: Artifacts (nxp.com)|https://bamboo3.sw.nxp.com/browse/ARTD-CII2C-854/artifact] Expected behavior: No error Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update devtest to make use of I2C_EVENT_ERROR_FIFO_MASTER
ARTD-78675	Bug	[Icu] Fix compiler warning Detailed description (how to reproduce it): Have compiler warning of Ip Driver when build test example: !image-2023-07-19-15-16-16-017.png! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Example Port_Ci_Icu_Ip_BlinkLed_S32K118 Observed behavior: Have warning Expected behavior: Not exits warning Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-78681	New Feature	[ICU] [ITG] Update data test code covarage and profilling NewWorkDescription:

ID	Subtype	Headline and Description
		Update data test code covarage and profiling Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Update data test code covarage and profiling,fix test failed
ARTD-78700	Bug	[PORT] MISRA Rule 8.13 shall not be commented out Detailed description (how to reproduce it): There's one MISRA Rule 8.13: "{_}A pointer should point to a const-qualified type whenever possible{_}" This rule normally can be commented out if the function is one of ASR function (with ASR requirements) In this PORT case, the function Port_Ci_Port_Ip_ConfigureInterleave is non-ASR function. Preconditions: MISRA Rule 8.13 Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: When reviewing the MISRA report, found 1 rule is being commented out but not the case from deviation list. Expected behavior: MISRA Rule 8.13 should not be commented out for the function Port_Ci_Port_Ip_ConfigureInterleave Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update driver to modify the input parameters of the function. !image-2023-07-19-18-38-32-143.png!
ARTD-78707	Bug	[S32K1_M24X RTM 2.0.0] [DEV] CAN: Missing check code field corruption when Errata ERR050443 happened Detailed description (how to reproduce it): Driver allways check INACTIVE case when Errata ERR050443 happened, but not allways the code field is INACTIVE, specifically in this case (pic bellow). !image-2023-07-19-19-43-42-349.png! Preconditions: Need 3 frame to trans and lock the Rx buffer Test Case ID (internal TC that caught the defect) optional: Ip_FlexCAN_TC_FCT_3026 Observed behavior: After driver fix the code field, it still wrong, so can't be receive correct frame anymore Expected behavior: After driver fix the code field, the code field need to be correct Proposed solution optional: Update driver to check all the case of FLEXCAN_RX
ARTD-78704	Bug	[I2C] Macros following generate are uncapitalized. Detailed description (how to reproduce it): Because the macros in the generate file are not capitalized, the build process will fail. !image-2023-07-19-10-38-13-616.png! Preconditions: Create test for I2C or use Dev_test TS_I2C_0999. Rename VS_0 to boarrdddd. !image-2023-07-19-10-41-55-949.png! Update again for EcuC !image-2023-07-19-10-43-08-491.png! Re-generate and build. Test Case ID (internal TC that caught the defect) optional: TS_I2C_0999 Observed behavior: The macros in the generate file are not capitalized. Expected behavior: The macros in the generate file are capitalized. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]
ARTD-78857	Bug	[S32K1_M24X RTM 2.0.0] OCOTP: Fix bug about build fail ccov Detailed description (how to reproduce it):

ID	Subtype	Headline and Description
		Preconditions: Generating TS pass Test Case ID (internal TC that caught the defect) optional: OCOTP_AE_TC_FLT_0001 Observed behavior: Build fail Expected behavior: Build pass Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution
ARTD-78861	Bug	[Gpt] Issue about code generate when using node DozeEnable with S32CT Detailed description (how to reproduce it): When using derivative S32K14x and enabling the DozeEnable node with S32CT, then it's not enabled, issue in file LPit_Gpt_Ip_Cfg_Defines.h line 212, see the picture below: !image-2023-07-21-14-14-20-780.png! so when using this node with S32K14x it won't be right Preconditions: NA Test Case ID (internal TC that caught the defect) optional: Gpt_TC_FCT_0206.c Observed behavior: NA Expected behavior: NA Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-79425	Bug	[S32K1_M24X RTM 2.0.0][EXAMPLE] DIO-PORT: Can't open config file on S32K118 Detailed description (how to reproduce it): Open S32DS, import Gpio_Dio_Ip_Example_S32K118, Port_Ci_Port_Ip_Example_S32K118 open mex file by S32 Configuration Tool Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Gpio_Dio_Ip_Example_S32K118 Port_Ci_Port_Ip_Example_S32K118 Observed behavior: !image-2023-07-25-10-30-30-263.png! Expected behavior: No warning/error Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-80248	Bug	[Mem_InFls] Example for K144 need Enable cache Detailed description (how to reproduce it): Example For K144 Not run. Because Node caches not enable Preconditions: !image-2023-07-31-13-57-02-180.png! Test Case ID (internal TC that caught the defect) optional: Observed behavior:

ID	Subtype	Headline and Description
		Expected behavior: !image-2023-07-31-13-57-22-611.png! Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [....]

4.4 Change List for 1.0.1

ID	Subtype	Headline and Description
ARTD-4422	New Feature	[FEE] Add support for bad Sector Management and Sector Retirement (part 1)<*> NewWorkDescription: Add support for bad Sector Management and Sector Retirement (CPR_RTD_00505.fee). This involves removing Sectors from configuration when they become unusable. Rationale: some flash memory specs have low endurance/write cycles availability, which needs to be improved by the ability to retire bad sectors or locations affected by permanent ECC, at the expense of allocated flash size the endurance being a ppm specification." Requirement source: cPRD (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Add support to use un-consecutive logical sector addresses.
ARTD-14705	Bug	[BASE] "hasExclusiveOwnership" always throws error when the calling component is disabled ,,"Preconditions: S32DS 3.4 U1 B210415 S32CC_RTD_4_4_RTM_2_0_0_DS_updatesite_2104 (210413) Test Case ID (internal TC that caught the defect) optional: 1. Create project for S32R45 enable RTD 2. Open Peripherals tool 3. Add GMAC component 4. Disable GMAC component Observed behavior: The errors are still showed when disable component. The configurations in EthGeneral tab is still light. Expected behavior: The error should be disappeared and the configuration should be gray when disable component.
ARTD-10467	New Feature	[icu] Improve define generation for ISR used to be cleaner/clearer<*> Please investigate if improvements for more clarity clearness can be done on generation code that is used for ISR used defines . [https://bitbucket.sw.nxp.com/projects/ARTD/repos/icu/pull-requests/207/overview?commentId=827200]
ARTD-10767	New Feature	[adc] Analyze MISRA violations and fix everything not in the deviations list<*> Analyze the MISRA violations in the driver and fix everything that is not in the Deviation list. Note 1: Read the comments for the allowed MISRA errors, because some are only allowed in certain situation, not everywhere. For example Rule 12.3 is only allowed when it is repoted because of Osif_Trusted_Call Note 2: If a certain MISRA error cannot be fixed or the fix has a high impact on maintainability or performance please discuss it with your component owner Feel free to create clones for this ticket in order to cover all platforms when time allows. Clean-up the MISRA comments from PC Lint from your code. More specifically: 1. Remove comment block from the start of the file that defines the MISRA violations and global MISRA deviations / page misra_violations MISRA-C:2012 violations 2. Remove comments from above the C code line with the violation. Example: / violates ref Mcl_Dma_h_REF_1 MISRA 2012 Required Directive 4.10, Precautions shall be taken in order to prevent the contents of a header file being included more than once. / More information on Coverity and MISRA can be found here: [https://confluence.sw.nxp.com/pages/viewpage.action?pageId=133734230]
ARTD-11539	Bug	[Platform] Wrong folder name 'generate_PC'<*> [https://bitbucket.sw.nxp.com/projects/ASPA/repos/real_time_drivers/browse/Generic/Change%20and%20Configuration%20Management/RTD_AASWS_Configuration%20Management%20Plan.docx] From the file in the link above (Chapter 4.2), I understand that the naming convention for the folders containing generated header file and source file templates should be named 'generate' instead of 'generate_PC'. !image-2021-06-11-13-43-55-112.png!

ID	Subtype	Headline and Description
ARTD-12331	New Feature	[BUILD_ENV] Update Ecuc.xdm in project template to match newly added container<*> NewWorkDescription: EcucPduCollection is de-populated after the tresosproject.pl script is run at generation. This causes drivers dependend on that container (i.e Fr) to fail at generation. Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Add EcucPduCollection in the new template.
ARTD-13195	Bug	[MCU] FIRC, SOSC, SPLL are not disabled in VLPR, VLPS mode. ,,"Detailed description (how to reproduce it): FIRC, SOSC, SPLL are not disabled in VLPR, VLPS mode. Preconditions: POWER_MODE_CHANGE_NOTIFICATION == STD_ON Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: FIRC, SOSC, SPLL are not disabled in VLPR, VLPS mode. Expected behavior: FIRC, SOSC, SPLL are disabled in VLPR, VLPS mode. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: remove case BEFORE_POWER_MODE_CHANGED !image-2021-07-06-16-53-48-200.png!
ARTD-13812	Bug	[MCU] FIRC is not disabled follow configuration<*> Detailed description (how to reproduce it): System clock under MCU control = false FIRC Enable = false System clock is not select FIRC Call Mcu_Init Call Mcu_Initclok FIRC is not disabled follow configuration Preconditions: NA Test Case ID (internal TC that caught the defect) optional: Mcu_TC_FCT_0501 Observed behavior: FIRC is not disabled follow configuration Expected behavior: FIRC is disabled follow configuration Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Can't enter condition "(config->regulator) ? 0UL : 1UL" !image-2021-07-08-11-21-05-905.png!
ARTD-13850	New Feature	[PINS] SIM Channel interleave support for K1<*> NewWorkDescription: Investigate and propose implementation for supporting additional routing settings for the signals between PINS and IPs. Concrete case at this point is to support : Channel interleave support for K1. Controlling by CHIPCTL register of SIM module to allow signal to be routed to both ADC instances. Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Investigate if is possible to add selection for pins to contain routing to 1 ADC input or 2 ADC inputs.
ARTD-14119	Bug	[OS] OsApplicationCoreRef and OsAppEcucPartitionRef shall be implemented as parameters<*> Detailed description (how to reproduce it): In SW32K3_RTD_4.4_0.9.0_D2103 MCAL, many drivers use OsAppEcucPartitionRef and OsApplicationCoreRef as reference, for instance in Eth driver: [!LOOP "as:modconf('Os')[1]/OsApplication/*/OsAppEcucPartitionRef/*"!] [!IF "node:value(.) = $EcucPartitionRef"!] [!IF "node:refvalid(../../OsApplicationCoreRef/*[1])"!] [!VAR "CoreId" = "num:i(node:value(node:ref(../../OsApplicationCoreRef/*[1])/EcucCoreId))"!] In this case the OsAppEcucPartitionRef and OsApplicationCoreRef are used as an list. According with Autosar specification these two containers is EcucReferenceDef, which can be only one parameter, and they shall not be a list, which contains one or more parameters (see attachment). Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A

ID	Subtype	Headline and Description
		Observed behavior: In customer's OS, the OsAppEcucPartitionRef and OsApplicationCoreRef are implemented as parameter, while Eth driver is used as lists. As a result, the configuration and generation are failed. Expected behavior: OsApplicationCoreRef and OsAppEcucPartitionRef shall be implemented as parameters, instead of lists Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-14220	Bug	[ADC] Adc_SetChannel function prototype does not match with requirement because of const qualifiers<*> Detailed description (how to reproduce it): Some function prototypes do not match with requirement Preconditions: Review new interface requirements Test Case ID (internal TC that caught the defect) optional: Review requirement Observed behavior: CPR_RTD_00329.adc: The optional API prototype shall be: void Adc_SetChannel(Adc_GroupType Group, Adc_GroupDefType Channel, #if (ADC_DELAY_AVAILABLE == STD_ON) uint16 Delays, uint32 ChannelUpdateMask, #endif / (ADC_DELAY_AVAILABLE == STD_ON) / Adc_ChannelIndexType NumberOfChannel); In driver: void Adc_SetChannel(const Adc_GroupType Group, const Adc_GroupDefType Channel, const uint16 Delays, const uint32 ChannelUpdateMask, const Adc_ChannelIndexType NumberOfChannel) SWS_Adc_00082: Service name: IoHwAb_AdcNotification – Syntax: void IoHwAb_AdcNotification(void) – In driver: Not found comment in ticket ARTD-14077 Expected behavior: Matching between requirement and driver Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-14221	Bug	[S32K1XX][PORT] Some functions were not mapped between ReqExport and Port driver .h<*> Detailed description (how to reproduce it): check the interface of all functions in "ReqExport.txt" file and "*.h" files of driver code: [http://savu.ea.freescale.net/0/project/download/ZTpcbG9jYWxfMDFcb3V0cHV0XGFydGlmYWN0c1xjdXN0b21fY2hlY2tpbnRlcmZhY2VcMjAyMTA3MjEwNTIzNTc0MjUwMDBcZG93bmxvYWRzXFBPUlRfcmVzdWx0LnR4dA==http://savu.ea.freescale.net/0/project/download/ZTpcbG9jYWxfMDFcb3V0cHV0XGFydGlmYWN0c1xjdXN0b21fY2hlY2tpbnRlcmZhY2VcMjAyMTA3MjEwNTIzNTc0MjUwMDBcZG93bmxvYWRzXFBPUlRfcmVzdWx0LnR4dA==http://savu.ea.freescale.net/0/project/download/ZTpcbG9jYWxfMDFcb3V0cHV0XGFydGlmYWN0c1xjdXN0b21fY2hlY2tpbnRlcmZhY2VcMjAyMTA3MjEwNTIzNTc0MjUwMDBcZG93bmxvYWRzXERJT19yZXN1bHQudHh0|http://savu.ea.freescale.net/0/project/download/ZTpcbG9jYWxfMDFcb3V0cHV0XGFydGlmYWN0c1xjdXN0b21fY2hlY2tpbnRlcmZhY2VcMjAyMTA3MjEwNTIzNTc0MjUwMDBcZG93bmxvYWRzXERJT19yZXN1bHQudHh0] or attach file Preconditions: Plugin_2172021 Reqs export BASELINE VERSION: 17.7 INTREQ_DIO_RTD_4.4_S32K1_1.0.0}} Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: In ReqExport: void Port_Init(const Port_ConfigType ConfigPtr) void Port_GetVersionInfo(*Std_VersionInfoType versioninfo) void Port_Ci_Port_Ip_DisableDigitalFilter(*GPIO_Type base, uint32 Pin) void Port_Ci_Port_Ip_EnableDigitalFilter(*GPIO_Type base, uint32 Pin) PortStatusType Port_Ci_Port_Ip_Init(uint32 pinCount, const Port_Ci_Port_Ip_PinSettingsConfig config[]) In Port.h and Port_Ci_Port_Ip.h: void Port_Init(const Port_ConfigType* ConfigPtr) void Port_GetVersionInfo(*Std_VersionInfoType* versioninfo) void Port_Ci_Port_Ip_DisableDigitalFilter(*PORT_Type const base, uint32 pin) void Port_Ci_Port_Ip_EnableDigitalFilter(*PORT_Type const base, uint32 pin) Port_Ci_Port_Ip_PortStatusType Port_Ci_Port_Ip_Init(uint32 pinCount, const Port_Ci_Port_Ip_PinSettingsConfig config[]) Expected behavior: All function similarities between "ReqExport.txt" file and driver code Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-14314	New Feature	[BASE] SIM Channel interleave support for K1 in IPL<*> NewWorkDescription: Investigate and propose implementation for supporting additional routing settings for the signals between PINS and IPs.

ID	Subtype	Headline and Description
		Concrete case at this point is to support : Channel interleave support for K1. Controlling by CHIPCTL register of SIM module to allow signal to be routed to both ADC instances. Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Investigate if is possible to add selection for pins to contain routing to 1 ADC input or 2 ADC inputs.
ARTD-14331	Bug	[MCU] Correct clock name of QSPI<*> Detailed description (how to reproduce it): In clock tool S32CT K148 Observed behavior: Some clock name of QSPI not same with SIM clock out in Mcu componet Expected behavior: Correct clock name of QSPI in CLKOUT0_CLK QSPI0_MODULE_CLK---->QSPI_CLK QSPI0_2xSFIF_CLK >QSPI_2xSFIF_CLK QSPI0_SFIF_CL ----->QSPI_SFIF_CLK QSPI0_SFIF_CLK_HYP_PREMUX_CLK----->QSPI_SFIF_CLK_HYP_PREMUX_CLK
ARTD-14413	Bug	[SAI] Unfixed MISRA violations of rule 8.13<*> Detailed description (how to reproduce it): Unfixed MISRA violations for rule 8.13, outside list of accepted project level deviations (MISRA violations for Rules are only accepted in cases specified by column D with disclaimer from column E for document with project level deviations): aData is assigned to pData in the state structure which is used for both TX and RX, so cannot be const This ticket is raised to analyze and confirm if code can be refactored to avoid these violations. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-14481	Bug	[MCU] The function Mcu_Ipw_CmuClearClockFailIrqFlags isn't called in HLD<*> Detailed description (how to reproduce it): The function Mcu_Ipw_CmuClearClockFailIrqFlags didn't be called in HLD !image-2021-07-29-17-18-43-915.png! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: The function Mcu_Ipw_CmuClearClockFailIrqFlags didn't be called in HLD Expected behavior: Remove the function Mcu_Ipw_CmuClearClockFailIrqFlags in driver code Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]

ARTD-14485	Bug	[ADC] Configurator generates multiple times the notification prototype, when same notification is used for multiple groups
		
		,,"Detailed description (how to reproduce it):
		Tresos configurator generates multiple times the notification prototype, when same notification is used for multiple groups
		To check if also S32CT has same issue
		Preconditions:
		Have a configuration with multiple groups that use the same notification.
		Test Case ID (internal TC that caught the defect) optional:
		TS_Shared_000
		Observed behavior:
		Each notifications declared in the EB config is generated more than once:
		/
		brief ADC Notification functions.
		details ADC Nofification functions defined inside the Plugin.
		/
		extern void Notification_0(void);
		extern void Notification_1(void);
		extern void T_ADC_Notification(void);
		extern void T_ADC_Notification(void);
		extern void Notification_0(void);
		extern void Notification_0(void);
		extern void T_ADC_Notification(void);
		extern void T_ADC_Notification(void);

ID	Subtype	Headline and Description

		extern void Notification_4(void);
		extern void Notification_5(void);
		extern void T_ADC_Notification(void);
		extern void Notification_1(void);
		extern void Notification_0(void);
		extern void Notification_0(void);
		extern void Notification_0(void);
		extern void Notification_0(void);
		extern void Notification_0(void);
		extern void Notification_0(void);
		extern void Notification_0(void);
		extern void Notification_0(void);
		Expected behavior:
		Each notification must be declared only once.
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)
		Proposed solution optional:
		[...]

ARTD-14487	Bug	[ADC] IPW notification prototype is defined both in driver code as well as generated code<*> Detailed description (how to reproduce it): IPW notification prototype is defined both in Adc_Ipw_Irq.c file as well as Adc_Ip_PBCfg.h generated file. Preconditions: Use a HLD config that is using interrupts. Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: IPW notification prototype is defined both in driver code as well as generated code. Expected behavior: IPW notification should be defined only once. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Remove IPW notification prototype from Adc_Ipw_Irq.c.
ARTD-14495	Bug	[CAN] ecvd/Ip_FlexCAN fails to export data for multiple controllers<*> Detailed description (how to reproduce it): A*> add more than one controllers to s32k148/Ip_FlexCAN (IPL layer) projects with default configuration (fix all errors if have) export ecvd import ecvd => error as below: !image-2021-07-30-10-51-10-710.png! B*> !image-2021-07-30-10-56-02-840.png! C>* **_ please re-check if all required nodes are exported (e.g. pe_clock_frequency) !image-2021-07-30-11-05-31-694.png! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-14559	New Feature	[EEP] Protect access to the FTFx registers shared between Fls, Eep and Crypto ,,"NewWorkDescription: Accesses to shared FTFx registers need to be guarded by exclusive areas. Analyze sync/async scenarios. Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Write / Erase / Cmd operations: The exlusive area needs to start when CCIF is read, and it needs to end after the FSTAT status flags have been checked.
ARTD-14569	Bug	[WDG] File version check issues: missing check for Ewm_Ip.c and Osif.h and wrong define name used for Wdg_Ipw_PBcfg.c<*> Detailed description (how to reproduce it): check RTD_WDG_CheckDefsReport report. It has warning about Multiple Definition See report to detail : [^RTD_WDG_CheckDefsReport.xlsx] Preconditions: [...]

ID	Subtype	Headline and Description
		Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: Wdg_43_Instance0_Ipw_PBcfg.c and Wdg_43_Instance0_PBcfg.c duplicated definition It has warning about Multiple Definition missing check version for Ewm_Ip.c and Osif.h Expected behavior: Fix warning Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Fix warning in Wdg_43_Instance0_Ipw_PBcfg.c Add checking version for Ewm_Ip.c and Osif.h
ARTD-14605	New Feature	[S32K1xx][Port] PFE field should raise errors to notice the users to configurate the correct Ports<*> NewWorkDescription: The PFE field should raise some errors to prevent the users from generating incorrect configuration source. Details: When users enable the PFE field by using PCR5 or PCR99 then change it to other PCR, even though the configuration of PFE was unable to tick, it remains TRUE so the source generated will be TRUE despite its PCR. Examples: !image-2021-08-02-17-38-13-510.png|width=1134,height=390! Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: [...]
ARTD-14607	Bug	[CRYPTO] Remove clear the error code on Csec_Ip_CancelCommand<*> Detailed description (how to reproduce it): Function have error code CSEC_IP_ERC_NO_RESPONSE but it have clear error on Csec_Ip_CancelCommand Preconditions: set timeout for test is minimum (Csec_Ip_pState->u32Timeout = 0) Test Case ID (internal TC that caught the defect) optional: Crypto_TC_FCT_0075 Observed behavior: IP function no return error Expected behavior: IP function will return error CSEC_IP_ERC_NO_RESPONSE Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-14615	Bug	[PWM] Stuck in interrupt function when re-enable notification for Flexio<*> Detailed description (how to reproduce it): When re-enable notification for channel flexio is 0 and 1, the stuck occurs in the interrupt function: Initilize 2 channels flexio 0 and 1 Loop for enable notification 2 channels. timerIrqMask variable will be set to 3(0x11) when re-enable notification for channel 0 timerIrqMask = timerIrqMask & ((uint8)0x01U << userCfg->timerId) in Flexio_Pwm_Ip_UpdateInterruptMode function will set timerIrqMask to 2(0x10)-> value 0 for channel 0. But interrupt still occur, it causes the program to get stuck in the interrupt function Preconditions: Build with GCC compiler Test Case ID (internal TC that caught the defect) optional: Pwm_TC_FCT_0213 Flexio Observed behavior: get stuck in the interrupt function Expected behavior: No stuck in the interrupt function Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-14680	Bug	[LIN][BASE] - Update name of Ds component for Lin<*> Detailed description (how to reproduce it): The actual component name for Lin is Lin_43. It should be updated to Lin. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: HLD component should be named Lin. Not Lin_43. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)

ID	Subtype	Headline and Description
		Proposed solution optional: In base driver we should update for S32K1 platform from Lin_43 to Lin in the modules.h file. For Lin driver the makefile, the sdk_manifest the generation files should be updated in order to change the name for Lin component in Lin
ARTD-14726	New Feature	[EcuC] Add requiresIndex to EcucPartition<*> Detailed description (how to reproduce it): ADC get the list Partition from EcucPartitionCollection/EcucPartition array. When importing epc output from Ecuc of EB tresos to S32DS S32DS import the partition order following "Name" field but the order in EB following "index" column. this makes ADC has different in generated code bw Eb and CT !image-2021-08-05-14-46-19-370.png! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: ADC_TS_001 DevTest Observed behavior: [...] Expected behavior: generated code is indentical Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Use <REQUIRES-INDEX> attribute as the post in team [https://teams.microsoft.com/dl/launcher/launcher.html?url=%2F_%23%2Fl%2Fmessage%2F19%3Ae451c595d3ba410c8f9c1f5efc19a8a3%40thread.tacv2%2F1618569101137%3FtenantId%3D686ea1d3-bc2b-4c6f-a92c-d99c5c301635%26groupId%3Df9decd68-7b74-4101-a4d2-ad54e86f727f%26parentMessageId%3D1618569101137%26teamName%3DZebra%26channelName%3DConfiguration%2520(code%252C%2520DS%252C%2520EB%252C%2520etc)%26createdTime%3D1618569101137&type=message&deeplinkId=7af9cc4b-4bf4-4f14-8e9c-c1a7f75cf125&directDl=true&msLaunch=true&enableMobilePage=true&suppressPrompt=true]
ARTD-14757	Bug	[SPI] Different in SpiFlexioEnable node location between EB and CT<*> Detailed description (how to reproduce it): In EB, the SpiFlexioEnable is located in SpiAutosarExt container but in CT it is SpiGeneral. Must be synchronous between them Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: SpiFlexioEnable is in different container in EB and HLD CT Expected behavior: in same container Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Synchronous them
ARTD-14846	New Feature	[EEP] Follow up ISO26262 Safety Assessors findings<*> NewWorkDescription: 1. The requirement CPR_RTD_00211 is an output of EEP_FMEA analysis, therefore it is a safety related requirement, the verification criteria should not be limited to review only ! it should be proven by test as well This get from pull request [Pull Request #181: Feature/ARTD-12130 s32k1 rtm eep generate quality documents dev for the module NXP Bitbucket|https://bitbucket.sw.nxp.com/projects/ARTD/repos/eep/pull-requests/181/overview] Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: NA
ARTD-14854	New Feature	[GPT] Follow up ISO26262 Safety Assessors findings<*> NewWorkDescription: Srtc_Ip_ProcessInterrupt Cyclomatic complexity is too high =19. Please provide counter measure to reduce it This get from pull request [Pull Request #287: Feature/ARTD-11887 s32k1 rtm gpt generate quality documents dev for the module NXP Bitbucket|https://bitbucket.sw.nxp.com/projects/ARTD/repos/gpt/pull-requests/287/overview] Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: NA
ARTD-14862	Bug	[SPI] Add condition related to must enable pin configuration for Flexio on CT<*> Detailed description (how to reproduce it): Now, CT tool does not raise any an error if not configure for spi pins of flexio Preconditions: NA Test Case ID (internal TC that caught the defect) optional:

ID	Subtype	Headline and Description
		NA Observed behavior: No error is raised to notification for user Expected behavior: An error if pins is not configured Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Add an constraint on CT template file
ARTD-14978	New Feature	[ADC] Remove per Group indexation of pResultsBufferPtr<*> NewWorkDescription: Remove per Group indexation of pResultsBufferPtr, because it is part of Adc_GroupConfigurationType, so can already be accessed per group: Adc_pCfgPtr[u32CoreId]->pGroups[GroupIndex].pResultsBufferPtr[Group]
ARTD-15062	Bug	[LIN] Features Lin Frame Timeout Disable only applied with case node is slave.<*> Detailed description (how to reproduce it): About features Lin Frame Timeout Disable, Now driver only applied with case node is slave, lacking with case node is master for LPUART Driver don't need to checking timer out when master node transmits header frame for FLEXIO Preconditions: !image-2021-08-11-17-17-53-703.png! Test Case ID (internal TC that caught the defect) optional: Lin_TC_FCT_0019 Observed behavior: NA Expected behavior: NA Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-15148	Bug	[BASE] Some wizard data files are missing the copyright header<*> Detailed description (how to reproduce it): At least Clock_Ip_Cfg_Defines.h, Clock_Ip_Cfg_Defines.c, Clock_Ip_PBcfg.c, Clock_Ip_PBcfg.h, Clock_Ip_Cfg.c, Clock_Ip_Cfg.h, Siul2_Port_Ip_Cfg.c, Siul2_Port_Ip_Cfg.h are missing the comment header block including the copyright Preconditions: None Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Missing copyright Expected behavior: Comment header block should be there Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Review the files in Base and add the comment header block
ARTD-15151	Bug	[Uart] Update code follow new changes of requirement<*> Detailed description (how to reproduce it): Update code follow req changes on ticket: https://jira.sw.nxp.com/browse/AAI-907 Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-15226	Bug	[Spi] No error raise when SpiMaxDmaFastTransfert not enable and Dma Fast transfer support<*> Detailed description (how to reproduce it): To support Dma Fast transfer, SpiPhyUnit is used by External Device in Job must has SpiMaxDmaFastTransfer enabled. Preconditions: SpiEnableDmaFastTransfer is True SpiEnableDmaFastTransferSupport is True SpiGlobalDmaEnable is True Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: NA

ID	Subtype	Headline and Description
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Raise an error
ARTD-15311	Bug	[WDG] Findings need to fix follow to Code Checklist Review<*> Detailed description (how to reproduce it): Something didn't follow to coding rule (checklist in ticket: https://jira.sw.nxp.com/browse/ARTD-9123) Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Need to fix
ARTD-15312	Bug	[SAI] Fix code review and UML review check list items failed<*> Detailed description (how to reproduce it): Fix code review and UML review check list items failed Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-15318	Bug	[BASE] Templating engine does not correctly report Java exceptions<*> Detailed description (how to reproduce it): 1. Make the setup from UCT-4375 2. The error message is undefined As we analyzed it, this is from a bug in templatingengine.js file. !image-2021-08-16-17-38-59-868.png! error.stack doesn't exist. error variable is from class *IllegalFormatConversionException, so you can use the next variant: scriptApi.logError(error);* ** (the result will be like in next print) !image-2021-08-16-17-38-23-705.png!
ARTD-15466	New Feature	[BUILD_ENV] Create subchapter in IM about User mode configuration in AutosarOS<*> NewWorkDescription: Create subchapter in IM about User mode configuration in AutosarOS. This will help user understand our User Mode approach and know what needs to be done in AutosarOS. Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Create subchapter in IM about User mode configuration in AutosarOS.
ARTD-15484	New Feature	[SPI] Synchronous between Spi.component and Spi.xdm<*> NewWorkDescription: Some node in xdm and component is different type that is prescribed in NXP_RTD_AUTOSAR_S32CT.pptx as SpiChannelHalfDuplexSupport Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Remove node of component is not available in xdmd Change to array with node having optional attribute
ARTD-15535	New Feature	[SPI] Implement new requirement: CPR_RTD_00543.spi<*> NewWorkDescription: Implement new requirements: CPR_RTD_00543.spi and CPR_RTD_00544.spi Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)

ID	Subtype	Headline and Description
		Proposed solution optional: Follow guiding: https://nxp1.sharepoint.com/:p:/r/sites/Zebra/_layouts/15/Doc.aspx?sourcedoc=%7B28A51BA3-C452-40F5-ABB6-A6933671DCD3%7D&file=NXP_RTD_AUTOSAR_S32CT_V2.pptx&action=edit&mobileredirect=true
ARTD-15669	Bug	[CAN] re-implement SWS_Can_00384<*> Detailed description (how to reproduce it): SWS_Can_00384|Each time the CAN controller state machine is triggered with the state transition value CAN_CS_STARTED, the function Can_SetControllerMode shall re-initialize the CAN controller with the same controller configuration set previously used by functions Can_SetBaudrate or Can_Init. but as implemented in the code. driver just switch from stopped state (disable mode) to started state(normal mode) => a re-init (reset all then init again) is necessary to reset buffers, fifo, bus-off,.. Test Case ID (internal TC that caught the defect) optional:__*Preconditions: NA Observed behavior: driver didn't re-init controller when started state required Expected behavior: driver does re-init controller when started state required Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: call FlexCan_Ip_Init at Ipw_SetcontrollertoStart and re-configure baudrate by Can_au16ControllerBaudRateSel and Can_Ipw_eClockMode which are saved at runtime
ARTD-15773	New Feature	[BASE] Add support for REQUIRES-INDEX in generateEcpd.js<*> NewWorkDescription: Add support for REQUIRES-INDEX in generateEcpd.js Requirement source: AUTOSAR_TPS_ECUConfiguration (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Add support for REQUIRES-INDEX in generateEcpd.js
ARTD-16267	Bug	[WDG] Wdog_Ip reset value of CS register is incorrect<*> Detailed description (how to reproduce it): Reset value of CS register isnt matching with S32k1xx_RM_Rev13 In Wdog_Ip_FeatureDefines.h: #define WDOG_IP_FEATURE_CS_RESET_VALUE_U16 (0x2520U) In RM: It is 0x2980 !image-2021-09-01-10-37-22-282.png! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: Reset value of CS register isnt matching with S32k1xx_RM_Rev13 Expected behavior: WDOG_IP_FEATURE_CS_RESET_VALUE_U16 should be 0x2980 Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update Wdog_Ip_FeatureDefines.h
ARTD-16336	Bug	[base] [EBT] OsApplicationCoreRef and OsAppEcucPartitionRef shall be accessed as parameters<*> Note: This issue is only affecting EBT configuration and source files! Detailed description (how to reproduce it): Our OS stub configuration defined OsAppEcucPartitionRef and OsApplicationCoreRef as lists, but the AUTOSAR specification defines them as optional parameters. We fixed this in the OS module, but now all the other modules that depend on OS to implement multicore features are affected. Preconditions: Create a multicore configuration. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as lists. As a result, the configuration and generation are failing. Expected behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as parameters. As a result, the configuration and generation are successful. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: # Replace all occurrences of {code}OsApplicationCoreRef/*[1] with {code}OsApplicationCoreRef # Replace all occurrences of {code}OsAppEcucPartitionRef/*[1] with {code}OsAppEcucPartitionRef Warning: Do not forget to adjust relative paths if needed (e.g. in SELECT-expressions).
ARTD-16337	Bug	[can] [EBT] OsApplicationCoreRef and OsAppEcucPartitionRef shall be accessed as parameters<*> Note: This issue is only affecting EBT configuration and source files! Detailed description (how to reproduce it):

ID	Subtype	Headline and Description
		Our OS stub configuration defined OsAppEcucPartitionRef and OsApplicationCoreRef as lists, but the AUTOSAR specification defines them as optional parameters. We fixed this in the OS module, but now all the other modules that depend on OS to implement multicore features are affected. Preconditions: Create a multicore configuration. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as lists. As a result, the configuration and generation are failing. Expected behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as parameters. As a result, the configuration and generation are successful. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: # Replace all occurrences of {code}OsApplicationCoreRef/*[1] with {code}OsApplicationCoreRef # Replace all occurrences of {code}OsAppEcucPartitionRef/*[1] with {code}OsAppEcucPartitionRef Warning: Do not forget to adjust relative paths if needed (e.g. in SELECT-expressions).
ARTD-16340	Bug	[crypto] [EBT] OsApplicationCoreRef and OsAppEcucPartitionRef shall be accessed as parameters<*> Note: This issue is only affecting EBT configuration and source files! Detailed description (how to reproduce it): Our OS stub configuration defined OsAppEcucPartitionRef and OsApplicationCoreRef as lists, but the AUTOSAR specification defines them as optional parameters. We fixed this in the OS module, but now all the other modules that depend on OS to implement multicore features are affected. Preconditions: Create a multicore configuration. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as lists. As a result, the configuration and generation are failing. Expected behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as parameters. As a result, the configuration and generation are successful. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: # Replace all occurrences of {code}OsApplicationCoreRef/*[1] with {code}OsApplicationCoreRef # Replace all occurrences of {code}OsAppEcucPartitionRef/*[1] with {code}OsAppEcucPartitionRef Warning: Do not forget to adjust relative paths if needed (e.g. in SELECT-expressions).
ARTD-16342	Bug	[eth] [EBT] OsApplicationCoreRef and OsAppEcucPartitionRef shall be accessed as parameters<*> Note: This issue is only affecting EBT configuration and source files! Detailed description (how to reproduce it): Our OS stub configuration defined OsAppEcucPartitionRef and OsApplicationCoreRef as lists, but the AUTOSAR specification defines them as optional parameters. We fixed this in the OS module, but now all the other modules that depend on OS to implement multicore features are affected. Preconditions: Create a multicore configuration. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as lists. As a result, the configuration and generation are failing. Expected behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as parameters. As a result, the configuration and generation are successful. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: # Replace all occurrences of {code}OsApplicationCoreRef/*[1] with {code}OsApplicationCoreRef # Replace all occurrences of {code}OsAppEcucPartitionRef/*[1] with {code}OsAppEcucPartitionRef Warning: Do not forget to adjust relative paths if needed (e.g. in SELECT-expressions).
ARTD-16345	Bug	[gpt] [EBT] OsApplicationCoreRef and OsAppEcucPartitionRef shall be accessed as parameters<*> Note: This issue is only affecting EBT configuration and source files! Detailed description (how to reproduce it): Our OS stub configuration defined OsAppEcucPartitionRef and OsApplicationCoreRef as lists, but the AUTOSAR specification defines them as optional parameters. We fixed this in the OS module, but now all the other modules that depend on OS to implement multicore features are affected. Preconditions: Create a multicore configuration. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as lists. As a result, the configuration and generation are failing. Expected behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as parameters. As a result, the configuration and generation are successful. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: # Replace all occurrences of {code}OsApplicationCoreRef/*[1] with {code}OsApplicationCoreRef

ID	Subtype	Headline and Description
		# Replace all occurrences of {code}OsAppEcucPartitionRef/*[1] with {code}OsAppEcucPartitionRef Warning: Do not forget to adjust relative paths if needed (e.g. in SELECT-expressions).
ARTD-16299	New Feature	[BASE] Update Can_GeneralTypes.h with SWS_CanTrcv_00164 and SWS_CanTrcv_00165<*> Update Can_GeneralTypes.h with SWS_CanTrcv_00164 and SWS_CanTrcv_00165
ARTD-16346	Bug	[i2c] [EBT] OsApplicationCoreRef and OsAppEcucPartitionRef shall be accessed as parameters<*> Note: This issue is only affecting EBT configuration and source files! Detailed description (how to reproduce it): Our OS stub configuration defined OsAppEcucPartitionRef and OsApplicationCoreRef as lists, but the AUTOSAR specification defines them as optional parameters. We fixed this in the OS module, but now all the other modules that depend on OS to implement multicore features are affected. Preconditions: Create a multicore configuration. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as lists. As a result, the configuration and generation are failing. Expected behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as parameters. As a result, the configuration and generation are successful. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: # Replace all occurrences of {code}OsApplicationCoreRef/*[1] with {code}OsApplicationCoreRef # Replace all occurrences of {code}OsAppEcucPartitionRef/*[1] with {code}OsAppEcucPartitionRef Warning: Do not forget to adjust relative paths if needed (e.g. in SELECT-expressions).
ARTD-16355	Bug	[port] [EBT] OsApplicationCoreRef and OsAppEcucPartitionRef shall be accessed as parameters<*> Note: This issue is only affecting EBT configuration and source files! Detailed description (how to reproduce it): Our OS stub configuration defined OsAppEcucPartitionRef and OsApplicationCoreRef as lists, but the AUTOSAR specification defines them as optional parameters. We fixed this in the OS module, but now all the other modules that depend on OS to implement multicore features are affected. Preconditions: Create a multicore configuration. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as lists. As a result, the configuration and generation are failing. Expected behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as parameters. As a result, the configuration and generation are successful. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: # Replace all occurrences of {code}OsApplicationCoreRef/*[1] with {code}OsApplicationCoreRef # Replace all occurrences of {code}OsAppEcucPartitionRef/*[1] with {code}OsAppEcucPartitionRef Warning: Do not forget to adjust relative paths if needed (e.g. in SELECT-expressions).
ARTD-16356	Bug	[pwm] [EBT] OsApplicationCoreRef and OsAppEcucPartitionRef shall be accessed as parameters<*> Note: This issue is only affecting EBT configuration and source files! Detailed description (how to reproduce it): Our OS stub configuration defined OsAppEcucPartitionRef and OsApplicationCoreRef as lists, but the AUTOSAR specification defines them as optional parameters. We fixed this in the OS module, but now all the other modules that depend on OS to implement multicore features are affected. Preconditions: Create a multicore configuration. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as lists. As a result, the configuration and generation are failing. Expected behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as parameters. As a result, the configuration and generation are successful. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: # Replace all occurrences of {code}OsApplicationCoreRef/*[1] with {code}OsApplicationCoreRef # Replace all occurrences of {code}OsAppEcucPartitionRef/*[1] with {code}OsAppEcucPartitionRef Warning: Do not forget to adjust relative paths if needed (e.g. in SELECT-expressions).
ARTD-16357	Bug	[qd] [EBT] OsApplicationCoreRef and OsAppEcucPartitionRef shall be accessed as parameters<*> Note: This issue is only affecting EBT configuration and source files! Detailed description (how to reproduce it): Our OS stub configuration defined OsAppEcucPartitionRef and OsApplicationCoreRef as lists, but the AUTOSAR specification defines them as optional parameters. We fixed this in the OS module, but now all the other modules that depend on OS to implement multicore features are affected. Preconditions:

ID	Subtype	Headline and Description
		Create a multicore configuration. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as lists. As a result, the configuration and generation are failing. Expected behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as parameters. As a result, the configuration and generation are successful. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: # Replace all occurrences of {code}OsApplicationCoreRef/*[1] with {code}OsApplicationCoreRef # Replace all occurrences of {code}OsAppEcucPartitionRef/*[1] with {code}OsAppEcucPartitionRef Warning: Do not forget to adjust relative paths if needed (e.g. in SELECT-expressions).
ARTD-16361	Bug	[spi] [EBT] OsApplicationCoreRef and OsAppEcucPartitionRef shall be accessed as parameters<*> Note: This issue is only affecting EBT configuration and source files! Detailed description (how to reproduce it): Our OS stub configuration defined OsAppEcucPartitionRef and OsApplicationCoreRef as lists, but the AUTOSAR specification defines them as optional parameters. We fixed this in the OS module, but now all the other modules that depend on OS to implement multicore features are affected. Preconditions: Create a multicore configuration. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as lists. As a result, the configuration and generation are failing. Expected behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as parameters. As a result, the configuration and generation are successful. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: # Replace all occurrences of {code}OsApplicationCoreRef/*[1] with {code}OsApplicationCoreRef # Replace all occurrences of {code}OsAppEcucPartitionRef/*[1] with {code}OsAppEcucPartitionRef Warning: Do not forget to adjust relative paths if needed (e.g. in SELECT-expressions).
ARTD-16363	Bug	[uart] [EBT] OsApplicationCoreRef and OsAppEcucPartitionRef shall be accessed as parameters<*> Note: This issue is only affecting EBT configuration and source files! Detailed description (how to reproduce it): Our OS stub configuration defined OsAppEcucPartitionRef and OsApplicationCoreRef as lists, but the AUTOSAR specification defines them as optional parameters. We fixed this in the OS module, but now all the other modules that depend on OS to implement multicore features are affected. Preconditions: Create a multicore configuration. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as lists. As a result, the configuration and generation are failing. Expected behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as parameters. As a result, the configuration and generation are successful. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: # Replace all occurrences of {code}OsApplicationCoreRef/*[1] with {code}OsApplicationCoreRef # Replace all occurrences of {code}OsAppEcucPartitionRef/*[1] with {code}OsAppEcucPartitionRef Warning: Do not forget to adjust relative paths if needed (e.g. in SELECT-expressions).
ARTD-16387	Bug	[crypto] [SWS_BSW_00037] Main functions of un-initialized modules shall return immediately<*> !SWS_BSW_00037.png|width=620,height=100! Detailed description (how to reproduce it): In Mdl_MainFunction, according to SWS_BSW_00037, if the module is not initialized and the main function is executed then no error shall be reported and the main function shall return immediately. Preconditions: Mdl_MainFunction() is called before Mdl_Init() or Mdl_Init() is called without success. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: In case the driver is un-initialized, Mdl_MainFunction() raises a DET error or doesn't return immediately. Expected behavior: In case the driver is un-initialized, Mdl_MainFunction() doesn't raise a DET error and returns immediately. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: In the main function(s), check if the module has been initialized and return immediately if not. Don't rely on the values of un-initialized variables (besides the global driver state).
ARTD-16388	Bug	[eep] [SWS_BSW_00037] Main functions of un-initialized modules shall return immediately<*> !SWS_BSW_00037.png|width=620,height=100! Detailed description (how to reproduce it):

ID	Subtype	Headline and Description
		In Mdl_MainFunction, according to SWS_BSW_00037, if the module is not initialized and the main function is executed then no error shall be reported and the main function shall return immediately. Preconditions: Mdl_MainFunction() is called before Mdl_Init() or Mdl_Init() is called without success. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: In case the driver is un-initialized, Mdl_MainFunction() raises a DET error or doesn't return immediately. Expected behavior: In case the driver is un-initialized, Mdl_MainFunction() doesn't raise a DET error and returns immediately. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: In the main function(s), check if the module has been initialized and return immediately if not. Don't rely on the values of un-initialized variables (besides the global driver state).
ARTD-16393	Bug	[fls] [SWS_BSW_00037] Main functions of un-initialized modules shall return immediately<*> !SWS_BSW_00037.png|width=620,height=100! Detailed description (how to reproduce it): In Mdl_MainFunction, according to SWS_BSW_00037, if the module is not initialized and the main function is executed then no error shall be reported and the main function shall return immediately. Preconditions: Mdl_MainFunction() is called before Mdl_Init() or Mdl_Init() is called without success. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: In case the driver is un-initialized, Mdl_MainFunction() raises a DET error or doesn't return immediately. Expected behavior: In case the driver is un-initialized, Mdl_MainFunction() doesn't raise a DET error and returns immediately. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: In the main function(s), check if the module has been initialized and return immediately if not. Don't rely on the values of un-initialized variables (besides the global driver state).
ARTD-16375	Bug	[CAN] Ip_Flexcan fails to return out-of-range status / enabled fd / s32k312<*> Detailed description (how to reproduce it): Initialize Flexcan_0/s32k312/IPL with invalid maxmb value: payload 16, maxmb = 43 Verificaition point: FlexCAN_Ip_Init return FLEXCAN_STATUS_BUFF_OUT_OF_RANGE => fail at this step Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Ip_FlexCAN_TC_FCT_2080 Observed behavior: behavior does not match requirement Expected behavior: driver should return out-of-range status Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-16399	Bug	[ADC] Update driver according to some requirements with Verification Criteria is "Review" implemented differently ,,"After complete to Review all the requirements with Verification Criteria is ""Review"", I found some requirement implemented different to the expectation due to the description did not match, missing condition to check dependency node, attribute was set to ReadOnly... Check the excel for further information and fixed if needed
ARTD-16411	Bug	[platform] [EBT] OsApplicationCoreRef and OsAppEcucPartitionRef shall be accessed as parameters<*> Note: This issue is only affecting EBT configuration and source files! Detailed description (how to reproduce it): Our OS stub configuration defined OsAppEcucPartitionRef and OsApplicationCoreRef as lists, but the AUTOSAR specification defines them as optional parameters. We fixed this in the OS module, but now all the other modules that depend on OS to implement multicore features are affected. Preconditions: Create a multicore configuration. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as lists. As a result, the configuration and generation are failing. Expected behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as parameters. As a result, the configuration and generation are successful. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: # Replace all occurrences of {code}OsApplicationCoreRef/*[1] with {code}OsApplicationCoreRef # Replace all occurrences of {code}OsAppEcucPartitionRef/*[1] with {code}OsAppEcucPartitionRef

ID	Subtype	Headline and Description
		Warning: Do not forget to adjust relative paths if needed (e.g. in SELECT-expressions).

ARTD-16449	New Feature	[base] All trusted functions should NOT defined as static or inline functions and should listed out in IM<*>
		
		NewWorkDescription:
		There are some points that need to be checked in each modules:
		1. All trusted functions should NOT defined as static or inline functions*, so OS Application can be able to call them from outside RTD drivers.
		Check in driver code all IP functions that needs to be called as trusted functions using macros OsIf_Trusted_Call[_Return][1-6params].
		Ensure these functions are defined without "static" or "inline" keywords, so OS Application can use "extern" keyword to declare and call them outside RTD drivers.
		For example:
		a. OsIf_Trusted_Call1param(*Adc_Sar_ClrUserAccessAllowed*, pBase);
		Adc_Sar_ClrUserAccessAllowed() function should be defined without "static" or "inline" as:
		void Adc_Sar_ClrUserAccessAllowed(const ADC_Type const pBase)
		{
		CLR_USER_ACCESS_ALLOWED((uint32)pBase, SAR_ADC_PROT_MEM_U32);
		}
		b. OsIf_Trusted_Call_Return1param(*C40_Ip_SetLockProtect*, Fls_VirtualSector);
		C40_Ip_SetLockProtect() function should be defined without "static" or "inline" as:
		C40_Ip_StatusType C40_Ip_SetLockProtect(C40_Ip_VirtualSectorsType Fls_VirtualSector)
		{
		}
		2. Create a new separate header file <IpName>_Ip_TrustedFunctions.h for each IPV has trusted functions, the AutosarOS just needs to include this new header file when user mode is used.
		In <IpName>_Ip_TrustedFunctions.h:
		Declare all trusted functions with "extern" keyword like:
		extern void C40_Ip_SetLockProtect(C40_Ip_VirtualSectorsType Fls_VirtualSector);
		Include all header files contain the definition of all types which are used by parameters or return type of those trusted functions.
		3. All these trusted functions need to be configured in OS, so they must be clearly listed in the driver IM (chapter 5.7 User Mode Support)
		List these functions follow guiding mentioned in slide 17 Coding RTD_RunningInUserMode.pptx
		All functions should be listed out in the table like:
		Function syntax||Description||Available via
		void C40_Ip_SetLockProtect(
		    C40_Ip_VirtualSectorsType Fls_VirtualSector
		)|Set lock bit for flash sectors|C40_Ip_TrustedFunctions.h
		4. Subchapter "*User Mode configuration in AutosarOS*" needs to be added in IM:
		!image-2021-08-19-09-26-20-823.png|thumbnail!
		In order to add this subchapter, there are 2 files added in build_env: user_mode_support.dox, user_mode_config_in_autosar_os.dox (added on the ticket ARTD-15466).
		In module repository, each module needs to rename from user_mode_support*.dox to user_mode_config_in_module*.dox and also its title from "*User Mode support*" to "*User Mode configuration in the module*".
		Please refer the silde [Coding RTD_RunningInUserMode.pptx|https://nxp1.sharepoint.com/:p:/r/sites/Zebra/Shared%20Documents/General/Zebra%20Training/Coding%20-%20RTD_RunningInUserMode.pptx?d=w21c9578fff794ff387db07a106880274&csf=1&web=1&e=lpbnNC] to understand more about User mode implementation in RTD.
		For Testing side*: Ensure that we have test scenarios mentioned from slides 22->25 Coding RTD_RunningInUserMode.pptx.
		Follow slide 23 to ensure we have one test suite to check all trusted functions able to call from outside RTD drivers.
		Refer to the implementation example on FLS:* ARTD-15610
		Requirement source:
		RTD implementation
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		Proposed solution optional:
		Follow Coding RTD_RunningInUserMode.pptx

ARTD-16458	New Feature	[lin] All trusted functions should NOT defined as static or inline functions and should listed out in IM<*>
		
		NewWorkDescription:
		There are some points that need to be checked in each modules:
		1. All trusted functions should NOT defined as static or inline functions*, so OS Application can be able to call them from outside RTD drivers.
		Check in driver code all IP functions that needs to be called as trusted functions using macros OsIf_Trusted_Call[_Return][1-6params].
		Ensure these functions are defined without "static" or "inline" keywords, so OS Application can use "extern" keyword to declare and call them outside RTD drivers.
		For example:
		a. OsIf_Trusted_Call1param(*Adc_Sar_ClrUserAccessAllowed*, pBase);
		Adc_Sar_ClrUserAccessAllowed() function should be defined without "static" or "inline" as:
		void Adc_Sar_ClrUserAccessAllowed(const ADC_Type const pBase)
		{
		CLR_USER_ACCESS_ALLOWED((uint32)pBase, SAR_ADC_PROT_MEM_U32);
		}
		b. OsIf_Trusted_Call_Return1param(*C40_Ip_SetLockProtect*, Fls_VirtualSector);
		C40_Ip_SetLockProtect() function should be defined without "static" or "inline" as:
		C40_Ip_StatusType C40_Ip_SetLockProtect(C40_Ip_VirtualSectorsType Fls_VirtualSector)
		{
		}
		2. Create a new separate header file <IpName>_Ip_TrustedFunctions.h for each IPV has trusted functions, the AutosarOS just needs to include this new header file when user mode is used.
		In <IpName>_Ip_TrustedFunctions.h:
		Declare all trusted functions with "extern" keyword like:
		extern void C40_Ip_SetLockProtect(C40_Ip_VirtualSectorsType Fls_VirtualSector);
		Include all header files contain the definition of all types which are used by parameters or return type of those trusted functions.
		3. All these trusted functions need to be configured in OS, so they must be clearly listed in the driver IM (chapter 5.7 User Mode Support)

ID	Subtype	Headline and Description
		List these functions follow guiding mentioned in slide 17 Coding RTD_RunningInUserMode.pptx All functions should be listed out in the table like: Function syntax||Description||Available via void C40_Ip_SetLockProtect(C40_Ip_VirtualSectorsType Fls_VirtualSector)|Set lock bit for flash sectors|C40_Ip_TrustedFunctions.h 4. Subchapter "*User Mode configuration in AutosarOS*" needs to be added in IM: !image-2021-08-19-09-26-20-823.png|thumbnail! In order to add this subchapter, there are 2 files added in build_env: user_mode_support.dox, user_mode_config_in_autosar_os.dox (added on the ticket ARTD-15466). In module repository, each module needs to rename from user_mode_support*.dox to user_mode_config_in_module*.dox and also its title from "*User Mode support*" to "*User Mode configuration in the module*". Please refer the silde [Coding RTD_RunningInUserMode.pptx|https://nxp1.sharepoint.com/:p:/r/sites/Zebra/Shared%20Documents/General/Zebra%20Training/Coding%20-%20RTD_RunningInUserMode.pptx?d=w21c9578fff794ff387db07a106880274&csf=1&web=1&e=lpbnNC] to understand more about User mode implementation in RTD. For Testing side*: Ensure that we have test scenarios mentioned from slides 22->25 Coding RTD_RunningInUserMode.pptx. Follow slide 23 to ensure we have one test suite to check all trusted functions able to call from outside RTD drivers. Refer to the implementation example on FLS:* ARTD-15610 Requirement source: RTD implementation (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Follow Coding RTD_RunningInUserMode.pptx

ARTD-16466	New Feature	
		[adc] All trusted functions should NOT defined as static or inline functions and should listed out in IM<*>
		
		NewWorkDescription:
		There are some points that need to be checked in each modules:
		1. All trusted functions should NOT defined as static or inline functions*, so OS Application can be able to call them from outside RTD drivers.
		Check in driver code all IP functions that needs to be called as trusted functions using macros OsIf_Trusted_Call[_Return][1-6params].
		Ensure these functions are defined without "static" or "inline" keywords, so OS Application can use "extern" keyword to declare and call them outside RTD drivers.
		For example:
		a. OsIf_Trusted_Call1param(*Adc_Sar_ClrUserAccessAllowed*, pBase);
		Adc_Sar_ClrUserAccessAllowed() function should be defined without "static" or "inline" as:
		void Adc_Sar_ClrUserAccessAllowed(const ADC_Type const pBase)
		{
		CLR_USER_ACCESS_ALLOWED((uint32)pBase, SAR_ADC_PROT_MEM_U32);
		}
		b. OsIf_Trusted_Call_Return1param(*C40_Ip_SetLockProtect*, Fls_VirtualSector);
		C40_Ip_SetLockProtect() function should be defined without "static" or "inline" as:
		C40_Ip_StatusType C40_Ip_SetLockProtect(C40_Ip_VirtualSectorsType Fls_VirtualSector)
		{
		}
		2. Create a new separate header file <IpName>_Ip_TrustedFunctions.h for each IPV has trusted functions, the AutosarOS just needs to include this new header file when user mode is used.
		In <IpName>_Ip_TrustedFunctions.h:
		Declare all trusted functions with "extern" keyword like:
		extern void C40_Ip_SetLockProtect(C40_Ip_VirtualSectorsType Fls_VirtualSector);
		Include all header files contain the definition of all types which are used by parameters or return type of those trusted functions.
		3. All these trusted functions need to be configured in OS, so they must be clearly listed in the driver IM (chapter 5.7 User Mode Support)
		List these functions follow guiding mentioned in slide 17 Coding RTD_RunningInUserMode.pptx
		All functions should be listed out in the table like:
		Function syntax||Description||Available via
		void C40_Ip_SetLockProtect(
		   C40_Ip_VirtualSectorsType Fls_VirtualSector
		)|Set lock bit for flash sectors|C40_Ip_TrustedFunctions.h
		4. Subchapter "*User Mode configuration in AutosarOS*" needs to be added in IM:
		!image-2021-08-19-09-26-20-823.png|thumbnail!
		In order to add this subchapter, there are 2 files added in build_env: user_mode_support.dox, user_mode_config_in_autosar_os.dox (added on the ticket ARTD-15466).
		In module repository, each module needs to rename from user_mode_support*.dox to user_mode_config_in_module*.dox and also its title from "*User Mode support*" to "*User Mode configuration in the module*".
		Please refer the silde [Coding RTD_RunningInUserMode.pptx|https://nxp1.sharepoint.com/:p:/r/sites/Zebra/Shared%20Documents/General/Zebra%20Training/Coding%20-%20RTD_RunningInUserMode.pptx?d=w21c9578fff794ff387db07a106880274&csf=1&web=1&e=lpbnNC] to understand more about User mode implementation in RTD.
		For Testing side*: Ensure that we have test scenarios mentioned from slides 22->25 Coding RTD_RunningInUserMode.pptx.
		Follow slide 23 to ensure we have one test suite to check all trusted functions able to call from outside RTD drivers.
		Refer to the implementation example on FLS:* ARTD-15610
		Requirement source:
		RTD implementation
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		Proposed solution optional:
		Follow Coding RTD_RunningInUserMode.pptx

ARTD-16476	New Feature	 [icu] All trusted functions should NOT defined as static or inline functions and should listed out in IM<*> NewWorkDescription: There are some points that need to be checked in each modules: 1. All trusted functions should NOT defined as static or inline functions*, so OS Application can be able to call them from outside RTD drivers. Check in driver code all IP functions that needs to be called as trusted functions using macros OsIf_Trusted_Call[_Return][1-6params].

ID	Subtype	Headline and Description

		Ensure these functions are defined without "static" or "inline" keywords, so OS Application can use "extern" keyword to declare and call them outside RTD drivers.
		For example:
		a. OsIf_Trusted_Call1param(*Adc_Sar_ClrUserAccessAllowed*, pBase);
		Adc_Sar_ClrUserAccessAllowed() function should be defined without "static" or "inline" as:
		void Adc_Sar_ClrUserAccessAllowed(const ADC_Type const pBase)
		{
		CLR_USER_ACCESS_ALLOWED((uint32)pBase, SAR_ADC_PROT_MEM_U32);
		}
		b. OsIf_Trusted_Call_Return1param(*C40_Ip_SetLockProtect*, Fls_VirtualSector);
		C40_Ip_SetLockProtect() function should be defined without "static" or "inline" as:
		C40_Ip_StatusType C40_Ip_SetLockProtect(C40_Ip_VirtualSectorsType Fls_VirtualSector)
		{
		}
		2. Create a new separate header file <IpName>_Ip_TrustedFunctions.h for each IPV has trusted functions, the AutosarOS just needs to include this new header file when user mode is used.
		In <IpName>_Ip_TrustedFunctions.h:
		Declare all trusted functions with "extern" keyword like:
		extern void C40_Ip_SetLockProtect(C40_Ip_VirtualSectorsType Fls_VirtualSector);
		Include all header files contain the definition of all types which are used by parameters or return type of those trusted functions.
		3. All these trusted functions need to be configured in OS, so they must be clearly listed in the driver IM (chapter 5.7 User Mode Support)
		List these functions follow guiding mentioned in slide 17 Coding RTD_RunningInUserMode.pptx
		All functions should be listed out in the table like:
		Function syntax||Description||Available via
		void C40_Ip_SetLockProtect(
		    C40_Ip_VirtualSectorsType Fls_VirtualSector
		)|Set lock bit for flash sectors|C40_Ip_TrustedFunctions.h
		4. Subchapter "*User Mode configuration in AutosarOS*" needs to be added in IM:
		!image-2021-08-19-09-26-20-823.png|thumbnail!
		In order to add this subchapter, there are 2 files added in build_env: user_mode_support.dox, user_mode_config_in_autosar_os.dox (added on the ticket ARTD-15466).
		In module repository, each module needs to rename from user_mode_support*.dox to user_mode_config_in_module*.dox and also its title from "*User Mode support*" to "*User Mode configuration in the module*".
		Please refer the silde [Coding RTD_RunningInUserMode.pptx|https://nxp1.sharepoint.com/:p:/r/sites/Zebra/Shared%20Documents/General/Zebra%20Training/Coding%20-%20RTD_RunningInUserMode.pptx?d=w21c9578fff794ff387db07a106880274&csf=1&web=1&e=lpbnNC] to understand more about User mode implementation in RTD.
		For Testing side*: Ensure that we have test scenarios mentioned from slides 22->25 Coding RTD_RunningInUserMode.pptx.
		Follow slide 23 to ensure we have one test suite to check all trusted functions able to call from outside RTD drivers.
		Refer to the implementation example on FLS:* ARTD-15610
		Requirement source:
		RTD implementation
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		Proposed solution optional:
		Follow Coding RTD_RunningInUserMode.pptx

ARTD-16477	New Feature	[ocu] All trusted functions should NOT defined as static or inline functions and should listed out in IM<*>
		
		NewWorkDescription:
		There are some points that need to be checked in each modules:
		1. All trusted functions should NOT defined as static or inline functions*, so OS Application can be able to call them from outside RTD drivers.
		Check in driver code all IP functions that needs to be called as trusted functions using macros OsIf_Trusted_Call[_Return][1-6params].
		Ensure these functions are defined without "static" or "inline" keywords, so OS Application can use "extern" keyword to declare and call them outside RTD drivers.
		For example:
		a. OsIf_Trusted_Call1param(*Adc_Sar_ClrUserAccessAllowed*, pBase);
		Adc_Sar_ClrUserAccessAllowed() function should be defined without "static" or "inline" as:
		void Adc_Sar_ClrUserAccessAllowed(const ADC_Type const pBase)
		{
		CLR_USER_ACCESS_ALLOWED((uint32)pBase, SAR_ADC_PROT_MEM_U32);
		}
		b. OsIf_Trusted_Call_Return1param(*C40_Ip_SetLockProtect*, Fls_VirtualSector);
		C40_Ip_SetLockProtect() function should be defined without "static" or "inline" as:
		C40_Ip_StatusType C40_Ip_SetLockProtect(C40_Ip_VirtualSectorsType Fls_VirtualSector)
		{
		}
		2. Create a new separate header file <IpName>_Ip_TrustedFunctions.h for each IPV has trusted functions, the AutosarOS just needs to include this new header file when user mode is used.
		In <IpName>_Ip_TrustedFunctions.h:
		Declare all trusted functions with "extern" keyword like:
		extern void C40_Ip_SetLockProtect(C40_Ip_VirtualSectorsType Fls_VirtualSector);
		Include all header files contain the definition of all types which are used by parameters or return type of those trusted functions.
		3. All these trusted functions need to be configured in OS, so they must be clearly listed in the driver IM (chapter 5.7 User Mode Support)
		List these functions follow guiding mentioned in slide 17 Coding RTD_RunningInUserMode.pptx
		All functions should be listed out in the table like:
		Function syntax||Description||Available via
		void C40_Ip_SetLockProtect(
		    C40_Ip_VirtualSectorsType Fls_VirtualSector
		)|Set lock bit for flash sectors|C40_Ip_TrustedFunctions.h
		4. Subchapter "*User Mode configuration in AutosarOS*" needs to be added in IM:
		!image-2021-08-19-09-26-20-823.png|thumbnail!
		In order to add this subchapter, there are 2 files added in build_env: user_mode_support.dox, user_mode_config_in_autosar_os.dox (added on the ticket ARTD-15466).

ID	Subtype	Headline and Description
		In module repository, each module needs to rename from user_mode_support*.dox to user_mode_config_in_module*.dox and also its title from "*User Mode support*" to "*User Mode configuration in the module*". Please refer the silde [Coding RTD_RunningInUserMode.pptx|https://nxp1.sharepoint.com/:p:/r/sites/Zebra/Shared%20Documents/General/Zebra%20Training/Coding%20-%20RTD_RunningInUserMode.pptx?d=w21c9578fff794ff387db07a106880274&csf=1&web=1&e=lpbnNC] to understand more about User mode implementation in RTD. For Testing side*: Ensure that we have test scenarios mentioned from slides 22->25 Coding RTD_RunningInUserMode.pptx. Follow slide 23 to ensure we have one test suite to check all trusted functions able to call from outside RTD drivers. Refer to the implementation example on FLS:* ARTD-15610 Requirement source: RTD implementation (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Follow Coding RTD_RunningInUserMode.pptx

ARTD-16479	New Feature	[qd] All trusted functions should NOT defined as static or inline functions and should listed out in IM<*>
		
		NewWorkDescription:
		There are some points that need to be checked in each modules:
		1. All trusted functions should NOT defined as static or inline functions*, so OS Application can be able to call them from outside RTD drivers.
		Check in driver code all IP functions that needs to be called as trusted functions using macros OsIf_Trusted_Call[_Return][1-6params].
		Ensure these functions are defined without "static" or "inline" keywords, so OS Application can use "extern" keyword to declare and call them outside RTD drivers.
		For example:
		a. OsIf_Trusted_Call1param(*Adc_Sar_ClrUserAccessAllowed*, pBase);
		Adc_Sar_ClrUserAccessAllowed() function should be defined without "static" or "inline" as:
		void Adc_Sar_ClrUserAccessAllowed(const ADC_Type const pBase)
		{
		CLR_USER_ACCESS_ALLOWED((uint32)pBase, SAR_ADC_PROT_MEM_U32);
		}
		b. OsIf_Trusted_Call_Return1param(*C40_Ip_SetLockProtect*, Fls_VirtualSector);
		C40_Ip_SetLockProtect() function should be defined without "static" or "inline" as:
		C40_Ip_StatusType C40_Ip_SetLockProtect(C40_Ip_VirtualSectorsType Fls_VirtualSector)
		{
		}
		2. Create a new separate header file <IpName>_Ip_TrustedFunctions.h for each IPV has trusted functions, the AutosarOS just needs to include this new header file when user mode is used.
		In <IpName>_Ip_TrustedFunctions.h:
		Declare all trusted functions with "extern" keyword like:
		extern void C40_Ip_SetLockProtect(C40_Ip_VirtualSectorsType Fls_VirtualSector);
		Include all header files contain the definition of all types which are used by parameters or return type of those trusted functions.
		3. All these trusted functions need to be configured in OS, so they must be clearly listed in the driver IM (chapter 5.7 User Mode Support)
		List these functions follow guiding mentioned in slide 17 Coding RTD_RunningInUserMode.pptx
		All functions should be listed out in the table like:
		Function syntax||Description||Available via
		void C40_Ip_SetLockProtect(
		    C40_Ip_VirtualSectorsType Fls_VirtualSector
		)|Set lock bit for flash sectors|C40_Ip_TrustedFunctions.h
		4. Subchapter "*User Mode configuration in AutosarOS*" needs to be added in IM:
		!image-2021-08-19-09-26-20-823.png|thumbnail!
		In order to add this subchapter, there are 2 files added in build_env: user_mode_support.dox, user_mode_config_in_autosar_os.dox (added on the ticket ARTD-15466).
		In module repository, each module needs to rename from user_mode_support*.dox to user_mode_config_in_module*.dox and also its title from "*User Mode support*" to "*User Mode configuration in the module*".
		Please refer the silde [Coding RTD_RunningInUserMode.pptx|https://nxp1.sharepoint.com/:p:/r/sites/Zebra/Shared%20Documents/General/Zebra%20Training/Coding%20-%20RTD_RunningInUserMode.pptx?d=w21c9578fff794ff387db07a106880274&csf=1&web=1&e=lpbnNC] to understand more about User mode implementation in RTD.
		For Testing side*: Ensure that we have test scenarios mentioned from slides 22->25 Coding RTD_RunningInUserMode.pptx.
		Follow slide 23 to ensure we have one test suite to check all trusted functions able to call from outside RTD drivers.
		Refer to the implementation example on FLS:* ARTD-15610
		Requirement source:
		RTD implementation
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		Proposed solution optional:
		Follow Coding RTD_RunningInUserMode.pptx

ARTD-16491	Bug	[Spi] On DS can generate LPSPI_TCR_CONT(1) even though SpiCsSelection is disabled<*> Detailed description (how to reproduce it): On DS can generate LPSPI_TCR_CONT(1) even though SpiCsSelection is disabled Preconditions: SpiCsContinous is True SpiCsSelection is disable Code generate LPSPI_TCR_CONT(1) Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: NA Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA

ID	Subtype	Headline and Description
ARTD-16516	Bug	[FLS][S32K3XX] Fls_CheckLoadAc condition-check is missing in case Multiple Sectors configured<*> Detailed description (how to reproduce it): During check ccov test hardfault , it showed: Due to LDRA build characteristic: It required to make an custom linker to build acc_code_rom to map to a sector at BLOCK_3, this make sure to test could run with access_code_load to Ram feature without Read-While_Write due to auxiliary files was generated (and mapped to code sectors of Block1) If configures Multiple sector (02 sectors in test case) in boundary location(from BLOCK2_SECTOR_383 the ending sector of Block2): driver shall check the next sector (BLOCK3_SECTOR_384) is satisfied CheckLoadAc() condition and implement load access-code to RAM section, but now, it doesn't. Preconditions: LDRA build run test. Test Case ID (internal TC that caught the defect) optional: Fls_TC_FCT_1000 Observed behavior: hardfault occur due to Read-While-Write condition. Expected behavior: CheckLoadAC shall be ensure such in case multiple sectors configured , especially in Boundary configured use-case (sector from Block1 to Block2, Block2 to Block3, ...) as long as sector to test stays on the same block of AccessCode. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-16525	Bug	[ADC] Need to add constraint when AdcConvTimeOnce is enable and container AdcNormalConvTimings is disable<*> Detailed description (how to reproduce it): Need to add constraint when AdcConvTimeOnce is enable and container AdcNormalConvTimings is disable Preconditions: AdcConvTimeOnce is enable Tab Hw unit container AdcNormalConvTimings is disable More detail in attachment Test Case ID (internal TC that caught the defect) optional: Adc_TC_FCT_0001-Adc_TS_009 Observed behavior: AdcConvTimeOnce is enable container AdcNormalConvTimings is disable => Real status: generate pass Expected behavior: Generate fail beacause in this case users can not configure sample time value and what the values in sample time register after calling Adc_Init Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-16572	New Feature	[S32CT][ECPD] Create a "literals" option for enumerations represented by dynamic_enums ,,"NewWorkDescription: Create a ""literals"" option for enumerations represented by dynamic_enums. If it exists, then the values for the ECUC-ENUMERATION-LITERAL-DEF nodes will be taken from this option. Otherwise, the values will be taken from the ""items"" expression of the dynamic_enum. This option represents a list of ""fallback"" values. Update the following presentation: https://nxp1.sharepoint.com/:p:/r/sites/Zebra/Shared%20Documents/Configuration%20(code,%20DS,%20EB,%20etc)/NXP_RTD_AUTOSAR_S32CT_V2.pptx?d=w28a51ba3c45240f5abb6a6933671dcd3&csf=1&web=1&e=s8y7Hc Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Update generateEcpd.js to implement the described behaviour.
ARTD-16644	Bug	[icu] [S32K1XX] [EBT] OsApplicationCoreRef and OsAppEcucPartitionRef shall be accessed as parameters<*> Note: This issue is only affecting EBT configuration and source files! Detailed description (how to reproduce it): Our OS stub configuration defined OsAppEcucPartitionRef and OsApplicationCoreRef as lists, but the AUTOSAR specification defines them as optional parameters. We fixed this in the OS module, but now all the other modules that depend on OS to implement multicore features are affected. Preconditions: Create a multicore configuration. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as lists. As a result, the configuration and generation are failing. Expected behavior: OsAppEcucPartitionRef and OsApplicationCoreRef are accessed as parameters. As a result, the configuration and generation are successful. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: # Replace all occurrences of {code}OsApplicationCoreRef/*[1] with {code}OsApplicationCoreRef # Replace all occurrences of {code}OsAppEcucPartitionRef/*[1] with {code}OsAppEcucPartitionRef Warning: Do not forget to adjust relative paths if needed (e.g. in SELECT-expressions).

ID	Subtype	Headline and Description
ARTD-16724	Bug	[ADC] Not return warning error when DMA channel be shared among configurations<*> Detailed description (how to reproduce it): When using the same DMA channel for configurations, it does not give an error warning. Configure ADC module select DMA channel (node: AdcDmaChannelId): dmaLogicChannel_Type_0 Configure CTU FIFO select DMA channel (Node: BctuFifoDmaChannelId): dmaLogicChannel_Type_0 Preconditions: Use EB Tresos. Test Case ID (internal TC that caught the defect) optional: Adc_TC_FCT_0701 of Adc_TS_012_cfgdma Observed behavior: When using the same DMA channel for configurations, it does not give an error warning Expected behavior: When using the same DMA channel for configurations, it does give an error warning Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-16728	Bug	[ADC] There is some information that does not match EB and S32CT when comparing code generation<*> Detailed description (how to reproduce it): Step 1: clean generate Adc_TS_COT_001 for EB and Adc_TS_COT_101_CT for CT Step 2: Compare include folder and src folder of S32CT with EB Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: There is some information that does not match EB and S32CT when comparing code generation. Detail in share point link: [https://nxp1.sharepoint.com/sites/Zebra/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2Fsites%2FZebra%2FShared%20Documents%2FGroup%204%2FEBvsS32CT%2FADC] File: "Compare Config S32CT and EB Tresos_EPC_S32K3xx.xlsx" Expected behavior: EB and S32CT need to generate the same file. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-16805	Bug	[LIN][LPUART] Report LIN_ERR_HEADER to LinIf when a frame error occurs in the header part of the frame<*> Detailed description (how to reproduce it): Driver is missing LIN_ERR_HEADER report to LinIf when a frame error occurs in the header part of the frame Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: NA Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-16814	Bug	[UART] Remove duplicate callback name if user configured with same name<*> Detailed description (how to reproduce it): If user config 1 callback name for 2 or more channel of Uart, duplicate name will happen !image-2021-09-14-23-12-45-227.png|thumbnail! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: Remove duplicate callback name if user configured with same name Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-16845	Bug	[LIN] Failure at generating on S32DS for K1XX<*> Detailed description (how to reproduce it): Open S32DS and error log(see attached file) !image-2021-09-15-15-15-00-517.png|thumbnail! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional:

ID	Subtype	Headline and Description
		N/A Observed behavior: Can not generate code on S32DS due to LIN module Expected behavior: Can generate code on S32DS Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-16875	Bug	[ADC] Hard fault error when calling Adc_SetClockMode(ADC_ALTERNATE) with AdcEnableDualClockMode is enable and AdcAlternateConvTimings is disable<*> Detailed description (how to reproduce it): Hard fault error when calling Adc_SetClockMode(ADC_ALTERNATE) with AdcEnableDualClockMode is enable and AdcAlternateConvTimings is disable Preconditions: AdcEnableDualClockMode is enable Tab hw unit container AdcAlternateConvTimings is disable Case 1: AdcConvTimeOnce is enable (Adc_TS_008) Case 2: AdcConvTimeOnce is disable (Adc_TS_009) More detail in attachment Test Case ID (internal TC that caught the defect) optional: Adc_TC_FCT_0002 Adc_TS_008 Adc_TC_FCT_0001 Adc_TS_009 Observed behavior: Calling Adc_SetClockMode(ADC_ALTERNATE) > Real status: Hard fault error in all 2 cases Expected behavior: no hard fault error Case 1: AdcConvTimeOnce is enable > calling Adc_SetClockMode(ADC_ALTERNATE) return e_not_ok or add constraint between AdcEnableDualClockMode and container AdcAlternateConvTimings (Adc_ts_008) Case 2: AdcConvTimeOnce is disable > calling Adc_SetClockMode(ADC_ALTERNATE) return ok because configure in group tab (Adc_ts_009) Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-16880	Bug	[S32K3 RTM] FLS: Missing fls user mode check when defined MCAL_ENABLE_USER_MODE_SUPPORT<*> Detailed description (how to reproduce it): 1. enable fls user mode and not define MCAL_ENABLE_USER_MODE_SUPPORT 2. disable fls user mode and define MCAL_ENABLE_USER_MODE_SUPPORT No error generated in both case. Observed behavior: No error generated in both above case. Expected behavior: An error will generated in both above case. Proposed solution optional: [...]
ARTD-16926	Bug	[RTD_4.4_S32K3x_1.0.0] Warning "Invalid project path: Missing project folder or file: \...\include for Source path" displayed after detach RTD for NPW project ,," Install S32DS 3.4 SP2.EAR1 release and S32K3XX_RTD_4_4_RTM_1_0_0_DS_updatesite_2109_RC4.zip Test Case: # Create NPW for S32K3XX enable RTD 1.0.0 # Select project > SDKs > Detach RTD 1.0.0 # Check Problem view Observed behavior: There is a warning message displayed: ""_*Invalid project path: Missing project folder or file: \\{project_name}\include for Source path*_"" after detach RTD 1.0.0 Expected behavior: No warning message displayed
ARTD-17012	New Feature	[FLS] Update Fls_DataAddressType to fix Misra 11.6<*> NewWorkDescription: Fix Misra 11.6 "A cast should not be performed between pointer to void and an arithmetic type." Requirement source: Misra 2012 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Update Fls_DataAddressType to uint32 to avoid conversions from pointer to void to uint32.
ARTD-17116	Bug	[FLS] FlsCallCycle parameter is 0 and cannot be configured<*> Detailed description (how to reproduce it): FlsCallCycle parameter is 0 and cannot be configured. This parameter is used by Fls driver in bswmd file (eclipse\plugins \Fls_TS_T40D11M20I0R0\generate_swcd\swcd\Fls_Bswmd.arxml) as follows: <PERIOD>[!IF "node:exists(FlsConfigSet/FlsCallCycle)"!][!"FlsConfigSet/FlsCallCycle"!][!ELSE!]0[!ENDIF!]</PERIOD>

ID	Subtype	Headline and Description
		As a result the generated value for Fls main function PERIOD in bswmd file is always 0 and this results in error in Autosar RTE as 0 is not a valid period of main function. Preconditions: FLS driver used together with Autosar RTE Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Not able to generate Autosar RTE because the Fls main function period in bswmd file is 0 Expected behavior: To be able to generate Autosar RTE (period is not 0 in bswmd file) Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Make the FlsCallCycle parameter configurable by these changes in Fls.xdm file: original FlsCallCycle definition: <v:var name="FlsCallCycle" type="FLOAT_LABEL"> <a:a name="LABEL" value="Fls Call Cycle"/> <a:a name="DESC"> .... Cycle time of calls of the flash driver main function Note: Not supported by the driver. .... </a:a> .... <a:da name="DEFAULT" value="0.0"/> .... <a:da name="READONLY" value="true"/> </v:var> Fixed definition (modified type of parameter from FLOAT_LABEL to FLOAT, modified DESC field to remove note that not supported by driver, modified default value from 0 to 0.01 and removed READONLY attribute): <v:var name="FlsCallCycle" type="FLOAT"> <a:a name="LABEL" value="Fls Call Cycle"/> <a:a name="DESC"> .... Cycle time of calls of the flash driver main function .... </a:a> .... <a:da name="DEFAULT" value="0.01"/> .... </v:var> Find attached the updated Fls.xdm file.
ARTD-17128	Bug	[CAN] inital segments generated by auto calculation baudrate in FlexCAN_Ip_PbCfg.c is not same with the ones in HLD<*> Detailed description (how to reproduce it): with input paramaters: Can_CLK = 24Mhz and !image-2021-09-24-18-32-47-140.png! segments generated in IPL (FlexCAN_Ip_PbCfg.c) !image-2021-09-24-18-34-19-636.png! segments generated in HLD (Can_PbCfg.c) !image-2021-09-24-18-34-54-919.png! => NOT identical. so affter Can_Init called, segments generated by IP used => not works need to check for S32CT code also. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: segments generated by HLD and IP are not identical Expected behavior: segments generated by HLD and IP are identical Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-17307	Bug	[S32DS] There are many Semantic errors message raised by S32DS but they are already defined in the project<*> Detailed description (how to reproduce it): There are many Semantic errors message raised by S32DS but they are already defined in the project and the project still build successful. This kind of error will be only appeared in almost of files after open them. They are not raised if the source files are not opened. This problem is appeared on all RTD releases, can check this problem on Examples project. !image-2021-09-27-14-22-49-376.png! Preconditions: The problem can be also reproduced on S32K3: Install the packages: S32 Design Studio 3.4 Service pack 2 with S32K3xx development package build 210923 [LINK|https://nxp1.sharepoint.com/sites/freeshareprivate/S32DSproject/Lists/Builds/DispForm.aspx?ID=1744] RTD S32K3XX RTM100 update side Open example: Can_example_S32K344 Generate code by press "Update code" button on GUI of peripheral tool Build project Open some source files in "RTD" and "board" folders in the project on S32DS GUI

ID	Subtype	Headline and Description
		Test Case ID (internal TC that caught the defect) optional: Examples project Observed behavior: Semantic errors appears but they are already defined in other files and the project still build successfully Expected behavior: There is no Semantic errors Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-17198	Bug	[SPI] SPI_EXCLUSIVE_AREA_02 is missing in bswmd file<*> Detailed description (how to reproduce it): SPI_EXCLUSIVE_AREA_02 is not assigned to any function in bswmd file. This results in compiler error as this exclusive area is not generated by Autosar RTE. Preconditions: Spi driver used with Autosar RTE Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Exclusive areas 02 is not generated by Autosar RTE Expected behavior: Exclusive areas 02 is generated by Autosar RTE Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update the exclusive_areas_to_be_defined_in_bsw_scheduler.dox to map the exclusive areas to a function which is generated into bswmd file.
ARTD-17279	Bug	[BASE] No need to declare trusted functions in OsIf_Timer_Systick_TrustedFunctions.h when AutosarOS used<*> Detailed description (how to reproduce it): When AutosarOS used, the functions OsIf_GetCounter() and OsIf_GetElapsed() will call corresponding to GetCounterValue() and GetElapsedValue() in Os.h if enable OsIfUseSystemTimer. So, the functions OsIf_Timer_System_Internal_Init(), OsIf_Timer_System_Internal_GetCounter() and OsIf_Timer_System_Internal_GetElapsed() will only called when using OS_FREERTOS or OS_BAREMETAL. AutosarOS will not called these functions. So OsIf_Timer_Systick_TrustedFunctions.h file can be removed. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: In OsIf_Timer_System.c: the functions OsIf_Timer_System_Internal_Init(), OsIf_Timer_System_Internal_GetCounter() and OsIf_Timer_System_Internal_GetElapsed() are not called when AutosarOS is used(USING_OS_AUTOSAROS defined). Expected behavior: NA Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Remove OsIf_Timer_Systick_TrustedFunctions.h and update the subchapter User mode in IM
ARTD-17352	Bug	[LIN] Driver and requirement are inconsistent<*> Detailed description (how to reproduce it): Driver and requirement are incompatible as below: No||Function name||Remark||_*Solution 1|void Flexio_Lin_Ip_GotoIdleState(const uint8 Channel);|Different name "Reqexport.txt" : "Flexio_Lin_Ip_GoToIdleState" " .h " file : "Flexio_Lin_Ip_GotoIdleState"|Change from Flexio_Lin_Ip_GotoIdleState to Flexio_Lin_Ip_GoToIdleState 2|void Lin_GetVersionInfo (Std_VersionInfoType VersionInfo);|Different name "Reqexport.txt" : "versioninfo" " .h " file : "VersionInfo"|The parameter in Lin_GetVersionInfo function need to be updated from VersionInfo to versioninfo. 3|void Lpuart_Lin_Ip_GotoIdleState(const uint8 Instance);|Different name "Reqexport.txt" : "Lpuart_Lin_Ip_GoToIdleState" " .h " file : "Lpuart_Lin_Ip_GotoIdleState"|Change from Lpuart_Lin_Ip_GotoIdleState to Lpuart_Lin_Ip_GoToIdleState 4|void Lpuart_Lin_Ip_TimerExpiredService(uint8 Instance);|Lacking const "Reqexport.txt" : "(const uint8 Instance);" " .h " file : "(uint8 Instance);"|The Instance parameter type in Lpuart_Lin_Ip_TimerExpiredService function should be changed to const. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Driver and requirement are inconsistent Expected behavior: Driver and requirement are consistent Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA

ID	Subtype	Headline and Description
ARTD-17515	New Feature	[RESOURCE] Requirement CPR_RTD_00220.resource has changed to non-traceable<*> NewWorkDescription: CPR_RTD_00220.resource has changed to non-traceable, so "@implement ResourceSubderivative_Object" in Resource.xdm should be removed. Requirement source: cPRT (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: NA
ARTD-17575	New Feature	[BUILD_ENV] Update compiler options<*> NewWorkDescription: There are some points conflict and needs to update after review and compare compiler option between SOW with S32DS in the ticket _**_ ARTD-16608 # For GHS: !image-2021-10-06-18-58-23-983.png! Option “-nostartfiles” is an linker option. It also mentioned in GHS document: !image-2021-10-06-18-44-13-702.png! So, this option should be moved from Compiler section to Linker section in excel file [LINK|https://nxp1.sharepoint.com/:x:/r/sites/Zebra/Shared%20Documents/Tools%20(build,%20misra,%20vsmd,%20etc)/Compilers/RTD_Compiler_Options.xlsx?d=wc54fdf4ed62b4036b28bba16838efb73&csf=1&web=1&e=qdKltl] 2. For IAR: !image-2021-10-06-18-57-39-415.png! The define "-DEU_DISABLE_ANSILIB_CALLS" is only used in EUnit.c for testing and not used in our driver code. So it should be removed from our compiler option and move to EUnit. Option “-c” is described in excel file but it is not exist on IAR compiler. So, this option should be removed. !image-2021-10-06-18-56-53-569.png! For more detail please check in email attached. Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: NA

ARTD-17736	Bug	[FLS] Qspi_Ip_Read function checks error callout for NULL_PTR instead of ECC callout<*>
		
		Detailed description (how to reproduce it):
		Source code : Qspi_Ip.c
		Function : Qspi_Ip_Read
		There is a process to call eccCheckCallout on line 1168, but line 1166 checks whether errorCheckCallout is NULL_PTR. We believe that it is necessary to check eccCheckCallout.
		/ Call user callout, if available, to check ecc status /
		if ((STATUS_QSPI_IP_SUCCESS == status) && (NULL_PTR != state->configuration->errorCheckCallout))
		{
		status = state->configuration->eccCheckCallout(instance, crtAddress, chunkSize);
		}
		Preconditions:
		N/A
		Test Case ID (internal TC that caught the defect) optional:
		N/A
		Observed behavior:
		errorCheckCallout is checked if it is NULL_PTR
		Expected behavior:
		eccCheckCallout shall be checked if it is NULL_PTR
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)
		Proposed solution optional:
		Replace the check for NULL_PTR as follows (errorCheckCallout modified to eccCheckCallout):
		/ Call user callout, if available, to check ecc status /
		if ((STATUS_QSPI_IP_SUCCESS == status) && (NULL_PTR != state->configuration->eccCheckCallout))
		{
		status = state->configuration->eccCheckCallout(instance, crtAddress, chunkSize);
		}

ARTD-17766	New Feature	[Wdg] Replace Swt_Ip_FeatureDefine.h with resource symbols and generate in CfgDefines.h<*> NewWorkDescription: [Wdg] Replace Swt_Ip_FeatureDefine.h with resource symbols and generate in CfgDefines.h Requirement source: Internal refactoring Proposed solution optional: [Wdg] Replace Swt_Ip_FeatureDefine.h with resource symbols and generate in CfgDefines.h
ARTD-17812	Bug	[Adc] MISRA Rule 8.5 Symbol "Adc_Sar_x_Isr" is declared more than once. ,,"Detailed description (how to reproduce it): Symbol ""Adc_Sar_x_Isr"" is declared more than once. Preconditions: NA Test Case ID (internal TC that caught the defect) optional:

ID	Subtype	Headline and Description
		NA Observed behavior: Symbol ""Adc_Sar_x_Isr"" is declared more than once. Expected behavior: Violation is fixed Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:
ARTD-17838	New Feature	[LIN] - Ecum_Externals.h must be optionally included in driver<*> _*NewWorkDescription: Ecum_Externals.h shall be optionally included in driver due to 8.6.2 Optional Interfaces chapter in ASR LIN standard. Requirement source: ASR SWS Proposed solution optional: Add a macro to guard the types and inclusion of this header.
ARTD-17933	Bug	[I2c] Cannot migrate EB tresos configuration to CT configuration because note slave address<*> Detailed description (how to reproduce it): [...] I converted example from EB to CT. I import i2c.epc file to CT. Open peripheral tool in S32DS. Have error CDD_i2c component. I saw that the node error belong to Ip layer. It should not add in HLD CDD_i2c component. !image-2021-10-18-16-59-18-790.png! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...] Remove node slave address on CT configuration
ARTD-17951	Bug	[ADC] Unused ISR code when interrupts are disabled in HLD configurator AdcInterrupt tab<*> Detailed description (how to reproduce it): At IPL for all IPs, ISRs are not removed at precompiled, even if interrupts are not being configured. Preconditions: None. Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: ISR code is always compiled. Expected behavior: ISR code not present when corresponding interrupts are not configured. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-17988	New Feature	[BASE] Use linkerFile tag instead of resourceFile tag in itm file for IAR profile<*> NewWorkDescription: IAR profile setting in itm file should update according to the advice from S32DS team mentioned in the ticket S32DS-24032. For option 'incPaths' tag, 'configId' attribute should be removed. For linker file configuration, linkerFile tag should be used instead of resourceFile tag. So, for Debug_RAM profile, no need to duplicate the options 'incPaths' and 'icfFile'. Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: NA
ARTD-18055	Bug	[MCU] Wrong diagram in clock configuration tool of S32DS V3.4<*> Detailed description (how to reproduce it): The S32K1's PCC has a clock gate function for peripherals. However, there is no clock gate at the output of the clock selector in the PCC. For example, in the case of FTM, PCC_FLEXTMRn[CGC] can be used to set the gating of the FTM system clock. I can also select a clock with PCC_FLEXTMRn[PCS], but this clock cannot be gated. However, in the Clocks Diagram on ConfigTools, the clock gate is attached to the output of the clock selector in PCC. Please refer to the image below: !image-2021-10-21-15-11-51-604.png!

ID	Subtype	Headline and Description
		Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Clocks Diagram on ConfigTools is not correct Expected behavior: Clocks Diagram on ConfigTools should be correct Note:* in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-18759	Bug	[CAN] Timestamp name does not match R41's RM<*> Detailed description (how to reproduce it): Create S32R41 project for CAN testing: Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: !image-2021-10-26-15-39-45-435.png! Expected behavior: Timestamp name should be re-checked for PFE !image-2021-10-26-15-39-19-914.png! Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-18833	New Feature	[BASE] Split ASM_KEYWORD into volatile and non-volatile<*> NewWorkDescription: Split ASM_KEYWORD into volatile and non-volatile in order to avoid compiler warnings related to volatile asm statements used out of functions. Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Split ASM_KEYWORD into volatile and non-volatile.
ARTD-18949	Bug	[SPI] Function parameter of Spi_SetupEB was not following the requirement and autosar specs<*> Detailed description (how to reproduce it): The function declaring of Spi_SetupEB was not following the requirement and autosar specs Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Std_ReturnType Spi_SetupEB (Spi_ChannelType Channel, Spi_DataBufferType SrcDataBufferPtr, Spi_DataBufferType DesDataBufferPtr, Spi_NumberOfDataType Length) Expected behavior: Std_ReturnType Spi_SetupEB (Spi_ChannelType Channel, const Spi_DataBufferType SrcDataBufferPtr, Spi_DataBufferType DesDataBufferPtr, Spi_NumberOfDataType Length) Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Change Spi_SetupEB to follow requirement and autosar specs
ARTD-18978	Bug	[ADC] BCTU and CTU Ip includes Det.h<*> Detailed description (how to reproduce it): BCTU and CTU Ip includes Det.h in Bctu_Ip_PBcfg.c and Ctu_Ip_PBcfg.c generated from EBT or S32CT Adc_Ipw_Irq.c included Det.h to checking spurious interrupt. But there is no Det need to be reported here because of Rationale: The DEM or DET callback is long in its full implementation, delaying the ISR. Currently, Adc report all DET error at HLD layer Preconditions: N.A. Observed behavior: BCTU and CTU Ip includes Det.h in Bctu_Ip_PBcfg.c and Ctu_Ip_PBcfg.c generated from EBT or S32CT Expected behavior: BCTU and CTU Ip must not have dependency on Det.h

ID	Subtype	Headline and Description
		Consider remove Det.h also in Adc_Ipw_Irq.c Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N.A.
ARTD-19020	New Feature	[ETH][ENET] Improve zero-copy operating mode<*> NewWorkDescription: Improve zero-copy operating mode Requirement source: Internal driver requirements (GMAC_IP_001) (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Add compiler-time switches EthCtrlAllocateTxDataBuffers and EthCtrlAllocateRxDataBuffers to avoid extraneous memory consumption when the application provides its own buffers. Add examples of usage in the documentation.
ARTD-19035	Bug	[EEP] Eep_Cancel function gets trapped with async write job in HS200 and HS400 mode<*> Detailed description (how to reproduce it): Eep_Cancel get trapped when canceling an in-progress Async Write job Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Eep_TC_FCT_0006 from Eep_TS_101.mak (cfg 12: SD_IP_DATABUSWIDTH_8BIT_DDR HS400 mode) Observed behavior: Program get trapped in the while loop of *uSDHC_Ip_Abort when Eep_Cancel is called. INT_STATUS[TC] and INT_STATUS[TC] never set Data line is always busy The pending bit of ISR(uSDHC0_Isr) in NVIC is set The issue happens when run through the *SchM_Enter_Eep_EEP_EXCLUSIVE_AREA_06 in *Eep_Cancel (line 1153) Currently it calls the function SuspendAllInterrupts to set PRIMASK = 1: raise execution priority to 0 #define SuspendAllInterrupts() ASM_KEYWORD(" cpsid i") This action might interfere to the ISR(uSDHC0_Isr) Expected behavior: Eep_Cancel shall work smoothly and cancel any current process. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Investigate the design purpose of EA in *Eep_Cancel Update code
ARTD-19081	Bug	[ETH] The start of index for each fifo read incorrectly<*> Detailed description (how to reproduce it): follow code driver: !image-2021-11-05-19-22-32-604.png|thumbnail! The start of index of each fifo equal the start of index of previous fifo number of buffers configured for fifo which was calculating. This is incorrectly. It should be he start of index number of buffers configured of previous fifo. Preconditions: Number of fifo > 1 Number of buffers configured in each fifo is different Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: The start of fifo's index is incorrectly. Expected behavior: The start of fifo's index need to calculate correctly. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Code should be changed to !image-2021-11-05-19-29-41-691.png|thumbnail!
ARTD-19078	Bug	[ADC] Missing exclusive areas in bswmd file<*> Detailed description (how to reproduce it): The folowing excusive areas is not getting generated in rte.c: ADC_EXCLUSIVE_AREA_46 ADC_EXCLUSIVE_AREA_42 ADC_EXCLUSIVE_AREA_43 ADC_EXCLUSIVE_AREA_44 They are not generated because these exclusive areas are missing or not assigned to a function in bswmd file. Preconditions: Adc driver used with Autosar RTE Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Exclusive areas not generated by RTE Expected behavior: Exclusive areas are generated by RTE Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:

ID	Subtype	Headline and Description
		Add exclusive areas into bswmd file. Internally update the exclusive_areas_to_be_defined_in_bsw_scheduler.dox file to map the exclusive areas to functions and then also add the IP functions into NonASR_ServiceID.xml file.
ARTD-19258	New Feature	[BUILD_ENV] Add a generic chapter to document the user mode flow in all IMs<*> NewWorkDescription: The current documentation of RTD does not clearly state that the integrator needs to define wrapper functions, called from our Osif implementation, in which to do extra OS specific operations (like marshaling of the parameters) and call the RTD APIs. Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Add a new generic IM chapter in build_env to appear in all modules IMs Describe the attached user mode sequence, add also a visual diagram. Describe the flow and highlight which functions need to be defined by the integrator. Example: • <RTD_module>_Init_Privileged function defined in RTD code and must be declared in OS configuration, for OS call and indexing. • TRUSTED_<RTD_module>_Init_Privileged function must be defined and declared in Integration/User code. • Call_<RTD_module>_Init_Privileged_TRUSTED function must be defined and declared in Integration/User code*, visible in Os.h for RTD to call it; doing the marshaling of the parameters in OS specific manner. • Trusted functions should not be declared as static or inline functions, they need to be declared with external linkage, so the OS can call them. • All trusted driver functions shall be declared in a separate header file <IPName>_Ip_TrustedFunctions.h; the OS, applications can include this header. • The <IPName>_Ip_TrustedFunctions.h will include the definition of all types used in the exported trusted functions
ARTD-19261	New Feature	[BASE][STUBS][S32CT] Add support for postBuildVariants<*> NewWorkDescription: Add support for postBuildVariants Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Update the generic "system" component to provide mechanisms that can enable the creation and usage of postBuildVariants
ARTD-19267	Bug	[LIN] LinNodeType has been generated incorrectly with multi channels setup<*> Detailed description (how to reproduce it): Configuration lin driver with multi channels: Channel 0: Lin node type is Slave. Channel 1: Lin node type is Master. ...... Observed behavior: All LinNodeTypes have been generated to LIN_SLAVE_NODE Expected behavior: LinNodeType of channel 0 is LIN_SLAVE_NODE. LinNodeType of channel 1 is LIN_MASTER_NODE. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-19276	New Feature	CLONE - [spi] Enable ECPD generation for IP components (CPR_RTD_00544)<*> Note: This feature must be implemented by all the modules for which this ticket was cloned (including CDDs). No clone shall be closed with "Not Applicable" or "No Action Taken". How to implement: # Add "options_expr" tags (containing all the necessary options) for the fields of the IP component(s). Please refer to the slides covering "EPD Generation" in Reference #2 below. # To ensure that the ECPD is generated only for your component, modify line 512 in "mcu_data/components/PlatformSDK_S32K3_2021_08/system/codegenerator.js" as follows: {code:javascript} if (configComponent.isOptionSet("ecucDefinitionCollection") && (configComponent.getId() === "Gmac")) { where you can replace "Gmac" with the ID of your IP component. # Check that there are no errors reported in the Problems view and that the ECPD has been generated in Code Preview. How to validate: # Manually create an EBT plugin using the previously generated ECPD. To achieve this: ## Unzip https://jira.sw.nxp.com/secure/attachment/828304/Ip_TS_T40D34M10I0R0.zip in the "output/eclipse/plugins" folder of your K3 work tree / repository. ## Open "Ip_TS_T40D34M10I0R0/plugin.xml" and replace all occurrences of "Gmac" with the ID of your IP component. # Create an EBT project (File > New > Configuration Project > Project Name: TstIp > Next > ECU ID: someId; Target: CORTEXM/S32K3XX > Finish) and add the previously created module (Right click on TstIp in Project Explorer > Module Configurations... > Double click on your module > OK) CHECKPOINT 1: The module should have been successfully imported into the project. # Go to S32DS and create a valid project using your IP component (or use an already existing one). Please assign a value other than the default value to each setting to avoid default values in this configuration{color}*. Rename your IP component instance and discard the "_1" prefix (e.g. "Gmac_1" > "Gmac"): {panel}!screenshot-1.png! # Generate the [LT/PB]Cfg.h/.c files of your IP component and save them somewhere to avoid overwriting them in a later step. We'll call this set of files OriginalGenerateFiles_. # Generate the ECVD file for your IP component instance (Global Settings > EcvdGenerationMethod = INDIVIDUAL > EcvdOutputPath = <EcvdOutputDir> > Click on the "Generate Configuration" button)

ID	Subtype	Headline and Description
		# Import the previously generated ECVD file in your EBT project (Right click on TstIp in Project Explorer > Im and Exporters... > Green Button > Next > File Name: <Browse and select the ECVD file> > Next > Enable path mapping > Finish > Run Importer) CHECKPOINT 2: The ECVD file should have been successfully applied to your module. # Save (CTRL S) and generate the project (Right click on TstIp in Project Explorer > Generate Project) to trigger the EPC generation. CHECKPOINT 3: The EPC file should have been successfully generated (TstIp/output/output/<Ip>.epc) # Import the previously generated EPC file in your S32DS project (File > Import > Import ECU Configuration > Path: <EpcOutputDir> > Check "Merge into current configuration" > Check "Check ECUC-MODULE-DEF" value > CHECKPOINT 4: The EPC file should have been successfully imported into your S32DS project. If you get the error "The required modules are not available for the current configuration" it means that the ECUC-MODULE-DEF check has failed and you're not properly using M4_XDM_AR_PKG_NAME in your IP component. # Generate again the [LT/PB]Cfg.h/.c files of your IP component and save them somewhere without overwriting the previous set of files. We'll call this new set of files ImportedGenerateFiles_. # Compare the OriginalGenerateFiles set with the ImportedGenerateFiles set. CHECKPOINT 5: The comparison should be binary identical. There should be no single difference between the two set of files. Requirements covered by this ticket: CPR_RTD_00544 EA_RTD_00107 EA_RTD_00108 References: # https://jira.sw.nxp.com/secure/attachment/828307/Ethernet.template # https://nxp1.sharepoint.com/:p:/s/Zebra/EaMbpShSxPVAq7amkzZx3NMB1LWDlRvDlnrzP1m0TZ9elQ?e=erB0sH (Relevant chapters: "9. EPD Importer Replacing the lost GNU M4 macros", "10. EPD Generation", "11. EPC Importer", "12. EPC Generation") Legend: EPD ECU Parameter Definition (Schema) Used by EBT ECPD ECU Configuration Parameter Definition (Schema) Used by S32CT EPC ECU Parameter Configuration (Configuration File) Used by EBT ECVD ECU Configuration Value Description (Configuration File) Used by S32CT
ARTD-19320	Bug	[ETH] Dependency on DEM is still expected when ETH_DEM_EVENT_DETECT = STD_OFF<*> Detailed description (how to reproduce it): Not all inclusions of Dem.h are guarded by macro ETH_DEM_EVENT_DETECT Preconditions: Set EthDisableDemEventDetect = TRUE in configuration Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Build fails if DEM isn't included in the build even though the configuration tells the driver to disable DEM. Expected behavior: Build is passing even if DEM isn't included in the build when configuration tells the driver to disable DEM. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Guard the #include "Dem.h" directives with ETH_DEM_EVENT_DETECT in files Eth.c and Eth_Ipw_Irq.c (their corresponding file version checks)
ARTD-19329	New Feature	CLONE - [lin] Enable ECPD generation for IP components (CPR_RTD_00544)<*> Note: This feature must be implemented by all the modules for which this ticket was cloned (including CDDs). No clone shall be closed with "Not Applicable" or "No Action Taken". How to implement: # Add "options_expr" tags (containing all the necessary options) for the fields of the IP component(s). Please refer to the slides covering "EPD Generation" in Reference #2 below. # To ensure that the ECPD is generated only for your component, modify line 512 in "mcu_data/components/PlatformSDK_S32K3_2021_08/system/codegenerator.js" as follows: {code:javascript} if (configComponent.isOptionSet("ecucDefinitionCollection") && (configComponent.getId() === "Gmac")) { where you can replace "Gmac" with the ID of your IP component. # Check that there are no errors reported in the Problems view and that the ECPD has been generated in Code Preview. How to validate: # Manually create an EBT plugin using the previously generated ECPD. To achieve this: ## Unzip https://jira.sw.nxp.com/secure/attachment/828304/Ip_TS_T40D34M10I0R0.zip in the "output/eclipse/plugins" folder of your K3 work tree / repository. ## Open "Ip_TS_T40D34M10I0R0/plugin.xml" and replace all occurrences of "Gmac" with the ID of your IP component. # Create an EBT project (File > New > Configuration Project > Project Name: TstIp > Next > ECU ID: someId; Target: CORTEXM/S32K3XX > Finish) and add the previously created module (Right click on TstIp in Project Explorer > Module Configurations... > Double click on your module > OK) CHECKPOINT 1: The module should have been successfully imported into the project. # Go to S32DS and create a valid project using your IP component (or use an already existing one). Please assign a value other than the default value to each setting to avoid default values in this configuration{color}*. Rename your IP component instance and discard the "_1" prefix (e.g. "Gmac_1" > "Gmac"): {panel}!screenshot-1.png! # Generate the [LT/PB]Cfg.h/.c files of your IP component and save them somewhere to avoid overwriting them in a later step. We'll call this set of files OriginalGenerateFiles_. # Generate the ECVD file for your IP component instance (Global Settings > EcvdGenerationMethod = INDIVIDUAL > EcvdOutputPath = <EcvdOutputDir> > Click on the "Generate Configuration" button) # Import the previously generated ECVD file in your EBT project (Right click on TstIp in Project Explorer > Im and Exporters... > Green Button > Next > File Name: <Browse and select the ECVD file> > Next > Enable path mapping > Finish > Run Importer) CHECKPOINT 2: The ECVD file should have been successfully applied to your module. # Save (CTRL S) and generate the project (Right click on TstIp in Project Explorer > Generate Project) to trigger the EPC generation. CHECKPOINT 3: The EPC file should have been successfully generated (TstIp/output/output/<Ip>.epc) # Import the previously generated EPC file in your S32DS project (File > Import > Import ECU Configuration > Path: <EpcOutputDir> > Check "Merge into current configuration" > Check "Check ECUC-MODULE-DEF" value >

ID	Subtype	Headline and Description
		CHECKPOINT 4: The EPC file should have been successfully imported into your S32DS project. If you get the error "The required modules are not available for the current configuration" it means that the ECUC-MODULE-DEF check has failed and you're not properly using M4_XDM_AR_PKG_NAME in your IP component. # Generate again the [LT/PB]Cfg.h/.c files of your IP component and save them somewhere without overwriting the previous set of files. We'll call this new set of files ImportedGenerateFiles_. # Compare the OriginalGenerateFiles set with the ImportedGenerateFiles set. CHECKPOINT 5: The comparison should be binary identical. There should be no single difference between the two set of files. Requirements covered by this ticket: CPR_RTD_00544 EA_RTD_00107 EA_RTD_00108 References: # https://jira.sw.nxp.com/secure/attachment/828307/Ethernet.template # https://nxp1.sharepoint.com/:p:/s/Zebra/EaMbpShSxPVAq7amkzZx3NMB1LWDlRvDlnrzP1m0TZ9elQ?e=erB0sH (Relevant chapters: "9. EPD Importer Replacing the lost GNU M4 macros", "10. EPD Generation", "11. EPC Importer", "12. EPC Generation") Legend: EPD ECU Parameter Definition (Schema) Used by EBT ECPD ECU Configuration Parameter Definition (Schema) Used by S32CT EPC ECU Parameter Configuration (Configuration File) Used by EBT ECVD ECU Configuration Value Description (Configuration File) Used by S32CT
ARTD-19338	Bug	[Gpt] GPT_CONFIG_VS_0_PB macro is generated when VariantPreCompile is used<*> Detailed description (how to reproduce it): This ticket is cloned because bamboo plan can't run be run for 6 fix versions because of limited space on bamboo servers When I run the test with test code covarage, I get a build failed error like the image described below This error is caused by the GPT_CONFIG_VS_0_PB macro that does not contain data in the Gpt_n_PBcfg.h file: #define GPT_CONFIG_VS_0_PB \\ Preconditions: Configuration in local file CCOV_ENABLE := ON LDRA_DIR := C:/LDRA_Toolsuite ALLOW_MULTIPLE_INSTANCES:=ON Test Case ID (internal TC that caught the defect) optional: Gpt_TC_FCT_0203(Gpt_TS_C07) Observed behavior: Build failed when running test code coverage Expected behavior: Build is successful when running test code coverage Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: in case of precompile variant this macro shouldn't be generated
ARTD-19435	Bug	[LIN] Function Lpuart_Lin_Ip_AutoBaudCapture can't detect wakeup signal<*> Detailed description (how to reproduce it): After Init driver call function Lpuart_Lin_Ip_GoToSleepMode and waiting master node send wakeup signal. But driver can't wakeup when autobaudrate feature enable. Preconditions: In func Lpuart_Lin_Ip_AutoBaudCapture, driver only handle case wakeup signal if func Lpuart_Lin_Ip_AutoBaudCapture called at least 3 times. But in this case after Init, driver go into SLEEP mode and Lpuart_Lin_Ip_AutoBaudCapture never called before. So when received wakeup signal driver only call Lpuart_Lin_Ip_AutoBaudCapture 2 time and ignore wakeup signal. !image-2021-11-15-10-59-35-409.png|width=906,height=387! Test Case ID (internal TC that caught the defect) optional: Ip_Lpuart_Lin_TC_FCT_0021 Observed behavior: Can't wakeup Expected behavior: Can wakeup Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-19486	New Feature	[BASE][S32CT] Add support to filter generated artifacts and components<*> NewWorkDescription: Add support to filter generated artifacts and components: Invoking the S32DS CLI generator with option "-ExportArgs" will let the user filter the generated artifacts (e.g. "-ExportArgs ecpd" means that only ECPDs will be generated; "-ExportArgs c h" means that only .c and .h files will be generated) Invoking the S32DS CLI generator with option "-ExportComponentIds" will let the user filter the generated components (e.g. "-ExportComponentIds Eth_43_GMAC" means that only the generator for Eth_43_GMAC will be invoked) Note: The arguments to both options are case-insensitive (e.g. "eCpD" and "ecpd" are equivalent; "eTh_43_gMac" and "Eth_43_GMAC" are equivalent). The options themselves are optional. If "-ExportArgs" is omitted, no filtering on artifacts is performed. If "-ExportComponentIds" is omitted, no filtering on components is performed. Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Update System files to support filtering generated artifacts and components.
ARTD-19529	Bug	[PORT] Cannot set port pin to input mode if the current mode is HIGHZ.<*> Detailed description (how to reproduce it):

ID	Subtype	Headline and Description

		Cannot set port pin to input mode if the current mode is HIGHZ.
		The issue happen because our code does not clear the PIDR if input mode:
		if (PORT_PIN_OUT == eDirection)
		{
		/ violates ref Port_Port_Ci_C_REF_5 Violates MISRA 2004 Required Rule 11.1, cast from unsigned long to pointer. /
		/ violates ref Port_Port_Ci_C_REF_10 A cast should not be performed between a pointer type and an integral type. /
		REG_BIT_SET32(GPIO_PDDR_ADDR32(GPIO_PORT_U32(Pin)), (uint32)(1UL << GPIO_CHANNEL_U32(Pin)));
		}
		/ Configures Port Pin as Input or High-Z*/
		else if ((PORT_PIN_IN == eDirection) (PORT_PIN_HIGH_Z == eDirection))
		{
		/ violates ref Port_Port_Ci_C_REF_5 Violates MISRA 2004 Required Rule 11.1, cast from unsigned long to pointer. /
		/ violates ref Port_Port_Ci_C_REF_10 A cast should not be performed between a pointer type and an integral type. /
		REG_BIT_CLEAR32(GPIO_PDDR_ADDR32(GPIO_PORT_U32(Pin)), (uint32)(1UL << GPIO_CHANNEL_U32(Pin))); / Check if the pin is HIGH-Z. In this case the driver needs to disable port input in PIDR register of GPIO IP*/
		if(PORT_PIN_HIGH_Z == eDirection)
		{
		/ violates ref Port_Port_Ci_C_REF_5 Violates MISRA 2004 Required Rule 11.1, cast from unsigned long to pointer. /
		/ violates ref Port_Port_Ci_C_REF_10 A cast should not be performed between a pointer type and an integral type. /
		REG_BIT_SET32(GPIO_PIDR_ADDR32(GPIO_PORT_U32(Pin)), (uint32)(1UL << GPIO_CHANNEL_U32(Pin)));
		}
		else
		{
		//Missing clear the PIDR
		}
		}
		else
		{
		/ Do nothing /
		}
		Preconditions:
		N/A
		Test Case ID (internal TC that caught the defect) optional:
		N/A
		Observed behavior:
		The driver cannot set to GPIO_INPUT mode if the current mode is HIGH-Z
		Expected behavior:
		The driver can set to GPIO_INPUT mode if the current mode is HIGH-Z
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)
		Proposed solution optional:
		N/A

ARTD-19575	Bug	[PORT] Port driver have to add some Det_reportError function following requirement<*> Detailed description (how to reproduce it): If Det is enabled, some function shall report specify error and return without any other action. Detail: CPR_RTD_00423.port: The function Port_ResetPinMode shall report PORT_E_MODE_UNCHANGEABLE error and return without any other action, if the parameter PortPinModeChangeable is set to FALSE. CPR_RTD_00426.port*: The function Port_SetAsUnusedPin shall report PORT_E_MODE_UNCHANGEABLE error and return without any other action, if the parameter PortPinModeChangeable is set to FALSE. CPR_RTD_00428.port*: The function Port_SetAsUsedPin shall report PORT_E_MODE_UNCHANGEABLE error and return without any other action, if the parameter PortPinModeChangeable is set to FALSE. SWS_Port_00223: The function Port_SetPinMode shall reportPORT_E_MODE_UNCHANGEABLE error and return without any other action, if the parameter PortPinModeChangeable is set to FALSE. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Port driver should add some Det_reportError function with specify error following the requirement in Detailed Description.
ARTD-19732	Bug	[eep] Update missing exclusive areas in BSWMD<*> Detailed description (how to reproduce it): Do a repo sync on bswmd_creator to sync with the manifest (make sure that you've checked out BLN_BSWMD_CREATOR_04.08.03) Compile the plugin passing parameter GENERATE_BSWMD_FILE=ON to the make command (`p s1 b` in ZTH) If the plugin was compiled successfully and there are no errors from bswmd_creator regarding missing or unmapped exclusive areas, then you can close your clone with "Fixed No Action Taken". Otherwise, proceed with the solutions given in the "Proposed solution" section. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Missing exclusive areas in <Module>_Bswmd.arxml Expected behavior: All exclusive areas are included in <Module>_Bswmd.arxml Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)

ID	Subtype	Headline and Description
		Proposed solution optional: Potential Issue #1 Missing exclusive areas for ISRs Solution #1 The bswmd_creator script will automatically fetch ISRs from the source code as long as they're defined with ISR(...). Every ISR mentioned in the "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" files shall also be wrapped in ISR(...). For example: ADC_EXCLUSIVE_AREA_70 is used in function ISR(Ctu_Error_Isr) to protect the updates for CTU_IFR register Potential Issue #2 Missing exclusive areas for IPL functions which are not called by any HLD function Solution #2 # Have the EA mentioned in "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" only as used by the IPL function. For example: ADC_EXCLUSIVE_AREA_17 is used in function Adc_Sar_Ip_AbortConversion to protect the updates for ADC_MCR register # Add the IPL function in NonASR_ServiceID.xml Note: Do not use other workarounds like artificially mentioning in .dox that the IPL function is called by an HLD function which is included in NonASR_ServiceID.xml (even though, in reality, it's not)
ARTD-19738	Bug	[i2c] Update missing exclusive areas in BSWMD<*> Detailed description (how to reproduce it): Do a repo sync on bswmd_creator to sync with the manifest (make sure that you've checked out BLN_BSWMD_CREATOR_04.08.03) Compile the plugin passing parameter GENERATE_BSWMD_FILE=ON to the make command (`p s1 b` in ZTH) If the plugin was compiled successfully and there are no errors from bswmd_creator regarding missing or unmapped exclusive areas, then you can close your clone with "Fixed No Action Taken". Otherwise, proceed with the solutions given in the "Proposed solution" section. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Missing exclusive areas in <Module>_Bswmd.arxml Expected behavior: All exclusive areas are included in <Module>_Bswmd.arxml Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Potential Issue #1 Missing exclusive areas for ISRs Solution #1 The bswmd_creator script will automatically fetch ISRs from the source code as long as they're defined with ISR(...). Every ISR mentioned in the "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" files shall also be wrapped in ISR(...). For example: ADC_EXCLUSIVE_AREA_70 is used in function ISR(Ctu_Error_Isr) to protect the updates for CTU_IFR register Potential Issue #2 Missing exclusive areas for IPL functions which are not called by any HLD function Solution #2 # Have the EA mentioned in "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" only as used by the IPL function. For example: ADC_EXCLUSIVE_AREA_17 is used in function Adc_Sar_Ip_AbortConversion to protect the updates for ADC_MCR register # Add the IPL function in NonASR_ServiceID.xml Note: Do not use other workarounds like artificially mentioning in .dox that the IPL function is called by an HLD function which is included in NonASR_ServiceID.xml (even though, in reality, it's not)
ARTD-19741	Bug	[lin] Update missing exclusive areas in BSWMD<*> Detailed description (how to reproduce it): Do a repo sync on bswmd_creator to sync with the manifest (make sure that you've checked out BLN_BSWMD_CREATOR_04.08.03) Compile the plugin passing parameter GENERATE_BSWMD_FILE=ON to the make command (`p s1 b` in ZTH) If the plugin was compiled successfully and there are no errors from bswmd_creator regarding missing or unmapped exclusive areas, then you can close your clone with "Fixed No Action Taken". Otherwise, proceed with the solutions given in the "Proposed solution" section. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Missing exclusive areas in <Module>_Bswmd.arxml Expected behavior: All exclusive areas are included in <Module>_Bswmd.arxml Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Potential Issue #1 Missing exclusive areas for ISRs Solution #1 The bswmd_creator script will automatically fetch ISRs from the source code as long as they're defined with ISR(...). Every ISR mentioned in the "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" files shall also be wrapped in ISR(...). For example: ADC_EXCLUSIVE_AREA_70 is used in function ISR(Ctu_Error_Isr) to protect the updates for CTU_IFR register Potential Issue #2 Missing exclusive areas for IPL functions which are not called by any HLD function Solution #2

ID	Subtype	Headline and Description
		# Have the EA mentioned in "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" only as used by the IPL function. For example: ADC_EXCLUSIVE_AREA_17 is used in function Adc_Sar_Ip_AbortConversion to protect the updates for ADC_MCR register # Add the IPL function in NonASR_ServiceID.xml Note: Do not use other workarounds like artificially mentioning in .dox that the IPL function is called by an HLD function which is included in NonASR_ServiceID.xml (even though, in reality, it's not)
ARTD-19750	Bug	[pwm] Update missing exclusive areas in BSWMD<*> Detailed description (how to reproduce it): Do a repo sync on bswmd_creator to sync with the manifest (make sure that you've checked out BLN_BSWMD_CREATOR_04.08.03) Compile the plugin passing parameter GENERATE_BSWMD_FILE=ON to the make command (`p s1 b` in ZTH) If the plugin was compiled successfully and there are no errors from bswmd_creator regarding missing or unmapped exclusive areas, then you can close your clone with "Fixed No Action Taken". Otherwise, proceed with the solutions given in the "Proposed solution" section. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Missing exclusive areas in <Module>_Bswmd.arxml Expected behavior: All exclusive areas are included in <Module>_Bswmd.arxml Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Potential Issue #1 Missing exclusive areas for ISRs Solution #1 The bswmd_creator script will automatically fetch ISRs from the source code as long as they're defined with ISR(...). Every ISR mentioned in the "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" files shall also be wrapped in ISR(...). For example: ADC_EXCLUSIVE_AREA_70 is used in function ISR(Ctu_Error_Isr) to protect the updates for CTU_IFR register Potential Issue #2 Missing exclusive areas for IPL functions which are not called by any HLD function Solution #2 # Have the EA mentioned in "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" only as used by the IPL function. For example: ADC_EXCLUSIVE_AREA_17 is used in function Adc_Sar_Ip_AbortConversion to protect the updates for ADC_MCR register # Add the IPL function in NonASR_ServiceID.xml Note: Do not use other workarounds like artificially mentioning in .dox that the IPL function is called by an HLD function which is included in NonASR_ServiceID.xml (even though, in reality, it's not)
ARTD-19756	Bug	[spi] Update missing exclusive areas in BSWMD<*> Detailed description (how to reproduce it): Do a repo sync on bswmd_creator to sync with the manifest (make sure that you've checked out BLN_BSWMD_CREATOR_04.08.03) Compile the plugin passing parameter GENERATE_BSWMD_FILE=ON to the make command (`p s1 b` in ZTH) If the plugin was compiled successfully and there are no errors from bswmd_creator regarding missing or unmapped exclusive areas, then you can close your clone with "Fixed No Action Taken". Otherwise, proceed with the solutions given in the "Proposed solution" section. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Missing exclusive areas in <Module>_Bswmd.arxml Expected behavior: All exclusive areas are included in <Module>_Bswmd.arxml Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Potential Issue #1 Missing exclusive areas for ISRs Solution #1 The bswmd_creator script will automatically fetch ISRs from the source code as long as they're defined with ISR(...). Every ISR mentioned in the "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" files shall also be wrapped in ISR(...). For example: ADC_EXCLUSIVE_AREA_70 is used in function ISR(Ctu_Error_Isr) to protect the updates for CTU_IFR register Potential Issue #2 Missing exclusive areas for IPL functions which are not called by any HLD function Solution #2 # Have the EA mentioned in "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" only as used by the IPL function. For example: ADC_EXCLUSIVE_AREA_17 is used in function Adc_Sar_Ip_AbortConversion to protect the updates for ADC_MCR register # Add the IPL function in NonASR_ServiceID.xml Note: Do not use other workarounds like artificially mentioning in .dox that the IPL function is called by an HLD function which is included in NonASR_ServiceID.xml (even though, in reality, it's not)
ARTD-19758	Bug	[uart] Update missing exclusive areas in BSWMD<*> Detailed description (how to reproduce it):

ID	Subtype	Headline and Description
		Do a repo sync on bswmd_creator to sync with the manifest (make sure that you've checked out BLN_BSWMD_CREATOR_04.08.03) Compile the plugin passing parameter GENERATE_BSWMD_FILE=ON to the make command (`p s1 b` in ZTH) If the plugin was compiled successfully and there are no errors from bswmd_creator regarding missing or unmapped exclusive areas, then you can close your clone with "Fixed No Action Taken". Otherwise, proceed with the solutions given in the "Proposed solution" section. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Missing exclusive areas in <Module>_Bswmd.arxml Expected behavior: All exclusive areas are included in <Module>_Bswmd.arxml Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Potential Issue #1 Missing exclusive areas for ISRs Solution #1 The bswmd_creator script will automatically fetch ISRs from the source code as long as they're defined with ISR(...). Every ISR mentioned in the "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" files shall also be wrapped in ISR(...). For example: ADC_EXCLUSIVE_AREA_70 is used in function ISR(Ctu_Error_Isr) to protect the updates for CTU_IFR register Potential Issue #2 Missing exclusive areas for IPL functions which are not called by any HLD function Solution #2 # Have the EA mentioned in "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" only as used by the IPL function. For example: ADC_EXCLUSIVE_AREA_17 is used in function Adc_Sar_Ip_AbortConversion to protect the updates for ADC_MCR register # Add the IPL function in NonASR_ServiceID.xml Note: Do not use other workarounds like artificially mentioning in .dox that the IPL function is called by an HLD function which is included in NonASR_ServiceID.xml (even though, in reality, it's not)
ARTD-19912	Bug	[port] Update missing exclusive areas in BSWMD for S32K1<*> Detailed description (how to reproduce it): Do a repo sync on bswmd_creator to sync with the manifest (make sure that you've checked out BLN_BSWMD_CREATOR_04.08.03) Compile the plugin passing parameter GENERATE_BSWMD_FILE=ON to the make command (`p s1 b` in ZTH) If the plugin was compiled successfully and there are no errors from bswmd_creator regarding missing or unmapped exclusive areas, then you can close your clone with "Fixed No Action Taken". Otherwise, proceed with the solutions given in the "Proposed solution" section. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Missing exclusive areas in <Module>_Bswmd.arxml Expected behavior: All exclusive areas are included in <Module>_Bswmd.arxml Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Potential Issue #1 Missing exclusive areas for ISRs Solution #1 The bswmd_creator script will automatically fetch ISRs from the source code as long as they're defined with ISR(...). Every ISR mentioned in the "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" files shall also be wrapped in ISR(...). For example: ADC_EXCLUSIVE_AREA_70 is used in function ISR(Ctu_Error_Isr) to protect the updates for CTU_IFR register Potential Issue #2 Missing exclusive areas for IPL functions which are not called by any HLD function Solution #2 # Have the EA mentioned in "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" only as used by the IPL function. For example: ADC_EXCLUSIVE_AREA_17 is used in function Adc_Sar_Ip_AbortConversion to protect the updates for ADC_MCR register # Add the IPL function in NonASR_ServiceID.xml Note: Do not use other workarounds like artificially mentioning in .dox that the IPL function is called by an HLD function which is included in NonASR_ServiceID.xml (even though, in reality, it's not)
ARTD-20029	Bug	[CAN][HLD-S32CT] This warning 'Issue: CanController_0 should be referred by CanIfCtrlCanCtrlRef' appears even if the controller is referred by CanIf<*> Detailed description (how to reproduce it): Project is created by EBT >import epc files from EBT to S32CT. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: Can_TS__CT_115 Observed behavior: !image-2021-11-30-11-48-36-294.png! Expected behavior: No warning appears. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)

ID	Subtype	Headline and Description
		Proposed solution optional: NA
ARTD-20327	Bug	[CRC] The data in ECVD file different from interface<*> Detailed description (how to reproduce it): N/A Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: When user add CRC64 custom with polynomial value on S32CT is 0x42F0E1EBA9EA3693, but polynomial value in ECVD file is 0x42F0E1EBA9EA37D8 Expected behavior: Update crc to polynomial value in ECVD file is the same interface Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-20322	New Feature	[CRC] Add support for CRC64<*> Detailed description (how to reproduce it): N/A Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: CRC driver code and configuration files do not support CRC64 mode Expected behavior: CRC driver code and configuration files must support CRC64 mode Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-20328	Bug	[CAN][S32CT/IPL] Build failed when tranfer type is set to USING DMA but LegacyFIFO is disabled<*> Detailed description (how to reproduce it): Transfer type is set to using_dma but legacy fifo feature is disabled. !image-2021-12-07-19-04-56-139.png! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: Ip_FlexCAN_TS_TRUST_001 Observed behavior: !image-2021-12-07-19-06-47-800.png! Expected behavior: No error at building Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-20696	New Feature	[S32K1 1.0.1] Driver activities for UART<*> Drivers/IPs list in SOW: [see SOW for details|https://nxp1.sharepoint.com/:w:/r/sites/Zebra/_layouts/15/Doc.aspx?sourcedoc=%7BCAFEE555-7211-425F-A217-11EA75D1364D%7D&file=S32K1%20RTD%20ASR%204.4%201.0.1%20SOW.docx&action=default&mobileredirect=true]
ARTD-22288	New Feature	[ETH]Add infix support in the Eth driver<*> NewWorkDescription: Add infix support in the Eth driver Requirement source: Planned activity (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: For having separate and independent namespaces, drivers with the defined multiplicity greater than 1 need to have their name extended with an infix. Steps: Outside of the driver: Add in Base a new MemMap Eth_43_NETC_MemMap.h generated for the new naming Add in Rte new SchM files Schm_Eth_43_NETC,c and Schm_Eth_43_NETC.h generated for the new naming In the driver: Add in the HLD and IPW M4 tags that will be replaced with the infix The files need to be renamed, the types, the functions, etc. Do not rename the types and defines that are specified in Eth_GeneralTypes. Those need to keep their name, as they will be used as defined by all Eth drivers (if more drivers are present in a project) Update the xdm file to change the package name and use the short_name of the driver instead of MODULE_NAME where needed Update the mak file of the driver to rename all files and to propagate the m4_infix_value in all needed files In the tests:

ID	Subtype	Headline and Description
		Change the xdm configuration for the tests to use the new format Change the mak file of the tests to compile the correct plugin folder Create a wrapper file Eth.h which includes Eth_43_NETC.h and redefines all needed macros, typedefs and functions to point to the newly named entities
ARTD-22323	Bug	[ETH]Funtions ReadMMD/ WriteMMD operated incorrectly<*> Detailed description (how to reproduce it): For clause 45: Need to write address first then read/write value to PHY. Driver is handling as clause 22. Preconditions: EthCtrlEnableMmd need to enable. Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: MMD function operate incorrectly Expected behavior: MMD operate correctly Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Add the line to write address of PHY first.
ARTD-22377	Bug	[WDG] Incorrect values of min/max_expr in S32CT for WdgTimeoutPeriod and WdgWindowPeriod<*> There are some value mismatches between EBT and CT. Please see for example the attached screenshot. The CT values should be momentarily hardcoded to match the EBT ones for S32CC 3.0.0 release. The min_expr and max_expr in CT should be fixed to match the ones in EBT. Verify EPD/EPC support for HLD and IPLD. Please refer to [this post on Teams https://teams.microsoft.com/l/message/19:e451c595d3ba410c8f9c1f5efc19a8a3@thread.tacv2/1606904026692?tenantId=686ea1d3-bc2b-4c6f-a92c-d99c5c301635&groupId=f9decd68-7b74-4101-a4d2-ad54e86f727f&parentMessageId=1606904026692&teamName=Zebra&channelName=Configuration%20(code%2C%20DS%2C%20EB%2C%20etc)&createdTime=1606904026692] for additional details. From the presentation, the main focus should be on the following sections: Mapping XDM to Component EPD Importer EPD Generation EPC Importer EPC Generation
ARTD-22454	Bug	[platform] Analysis the consistent between functionalities requirement and driver implementation<*> Situation: During the last minutes of S32K3 1.0.0, we used an owner tool to check the consistent between requirement and source code implementation and found some drivers that have the inconsistent between the functions' requirements and drivers implementation. Something is really to fix in the below scenarios: Miss or redundant parameters. Platform is declared mark for this situation Wrong function name Functions are not expose to Users (in header files) We already fixed in Platform, LIN, MCL, MCU, LIN, SENT, FLS, ADC, SAI drivers. The rests should be analysis and dig into the inconsistent if it is wrong or not. The following 3 categories were found : The functions are declared in "*.h" files of driver code (exported to user) but not included in "ReqExport.txt" file (no requirement are found in Doors > no traceability > no test) The functions are found in "ReqExport.txt" (requirement exists) but the functions are not declared in "*.h" files of driver code (are not present for user) The function name is found in both "ReqExport.txt" and in "*.h" file but there is at least one mismatch on data type, variable name, ... Proposal: With the remaining findings, please to: This ticket should perform after requirement analyis ticket Export all requirement and take a review to conclude the consistent between requirement and implementation. Please take a review on Crucible, the checklist at attachment. In the case there is the needed to update driver or to update requirement, please create a ticket to implement and link to this ticket For each driver, analysis and resolution should be completed with relevant information Reference: Findings from S32K3 1.0.0 release (attachment), as optional Contact SW Testers to provide the latest results for each drivers (it's mandatory input)
ARTD-22512	Bug	[ADC] Fix findings from code review checklist S32K1<*> Detailed description (how to reproduce it): Fix remaining findings that were postponed on ARTD-15668. #|Source Code file|Issue |Proposed Correction|Status|Comment 1|All EB and CT generate files using AdcEnableDmaTrasferMode and CtuEnableDmaTrasferMode nodes|Rule 3|AdcEnableDmaTransferMode, CtuEnableDmaTransferMode|Postponed|Postponed due to affect test code

ID	Subtype	Headline and Description
		5|All files using source file version information of IPW, some header wrappers in IPL.|Rule 19,25: some defines don't have appropriate prefix equivalent to layer (HLD/IPW/IPL)|Add prefix ADC_IPW, ADC_SAR_IP, BCTU_IP to defines|Postponed|Postponed due to affect test code 6|Adc_Sar_Ip_Types.h|Rule 20,25: RESOLUTION_x of Adc_Sar_Ip_Resolution doesn't have prefix|Add prefix ADC_SAR_IP to defines and update generation code|Postponed|Postponed due to affect test code 7|Adc_Ipw_Types.h|Rule 22: Some typedef missing Ipw prefix|Add prefix Ipw to typedefs|Postponed|Postponed due to affect test code 9|Adc.c, Adc_Ipw.c, Adc_Ip.c, Ctu_Ip.c, Bctu_Ip.c|Rule 28|Correct prefix of global variables|Postponed|Postponed due to affect test code Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: NA Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-22528	New Feature	[PWM] Correct and update the naming convention according to rule 19 and 29 in Code Review Checklist<*> NewWorkDescription: Rule 29 in Code Review Checklist : Are the local variables, function parameters and struct members using only PascalCase naming, without any special prefix (e.g. Hungarian notation or <Msn>, <Ip> etc.)? <VarName> [Rule 29] Example: Channel, State, Index, Config, etc. Rule 19 in Code Review Checklist : All macros used in preprocessor directives shall be in uppercase, with multiple words separated by an underscore following the format: HLD: <MSN>[_<NAME>] IPW: <MSN>_IPW[_<NAME>] IP: <IP>_<MSN>_IP[_<NAME>] for all shared IPs <IP>_IP[_<NAME>] for IPs that are not shared Example: #define CAN_DEV_ERROR_DETECTION STD_ON Requirement source: Rule 19 and 29 in Code review checklist (attached file). (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Update the driver code to use PascalCase naming, without any special prefix.
ARTD-22537	New Feature	[BASE] Remove release note link in rtd.collateral.release_id.xml file<*> NewWorkDescription: The release notes document link (from Flexera) will be available on RTD Updatesite description !image-2021-12-20-14-48-58-473.png! The release note document file inside Updatesite is no longer exist. So, the old link to release note pushed in itm.<PlatformName>.rtd.collateral.release_id.xml should be removed. Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: NA
ARTD-22719	Bug	[UART] Ecuc Path should only be read when Uart enable Multicore support<*> Detailed description (how to reproduce it): Ecuc Path should only be read when Uart enable Multicore support Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: There is no condition to check if (Uart is using multicore or not) !image-2021-12-22-10-23-00-935.png|thumbnail! Expected behavior: Update condition to check uart enable multicore support or not to read Ecuc Path Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-22720	New Feature	[S32K1XX] Add support for S32K1XX RTM 1.0.1<*> NewWorkDescription: Add support for S32K1XX RTM 1.0.1 Requirement source: (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional:
ARTD-22722	New Feature	

ID	Subtype	Headline and Description
		[BASE] Add support for S32K1XX RTM 101<*> NewWorkDescription: Add support for S32K1XX RTM 101 Requirement source: (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional:
ARTD-22724	Bug	[UART] Fix build fail with new tag<*> Detailed description (how to reproduce it): Driver build fail with MCL Preconditions: NA Test Case ID (internal TC that caught the defect) optional: Dev test: Uart_TC_0001 Observed behavior: NA Expected behavior: UART test case build and run pass Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-22849	Bug	[ITG] [Uart] HLD Transmit and Receive functions need to check both receive and transmit status for reporting det error<*> Detailed description (how to reproduce it): This ticket has been cloned from https://jira.sw.nxp.com/browse/ARTD-17018. In the https://jira.sw.nxp.com/browse/ARTD-19289 the checking on both transmission and reception on HLD for busy channel has been removed because it is not the correct approach on the loopback mode. Due to missing information on the ARTD-17018, we cannot figure out if the both operations checkins are necessary. What we need to do in this ticket bug is analyze if the updates are impacted the other functionality. If the impact exists, update the code in the way that driver works correctly both on internal loopback and normal mode. Another analysis required is that: is the last bug impacting all the platforms? If yes, then let's update the bug according If no, then let's update the code accordingly (maybe an update on the generic file is not correct). Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: NA Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-22864	Bug	[S32K1XX RTM 1.0.1][ICU] There are some errors in driver<*> Detailed description (how to reproduce it): The variable "channel" in Dev_Assert is not same as the parameter passed as following underlined: !image-2021-12-24-15-36-06-466.png! !image-2021-12-24-15-36-37-736.png! !image-2021-12-24-15-38-28-266.png! Preconditions: Dev error detect is turned on. Test Case ID (internal TC that caught the defect) optional: Ip_Lpit_TC_FCT_7000 Observed behavior: The name is wrong. Expected behavior: The name channel must be consistent. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-22865	New Feature	[EEP] Add Injection points to cover FMEA<*> NewWorkDescription: Add Injection point and modify driver code in Ftfc_Eep_Ip_WaitCcifTimeout (uint32 TimeoutCounter) function to support check FMEA requirement. Requirement source: [EEP] Add Injection points to cover FMEA Proposed solution optional: Add Injection point and modify driver code as suggest picture in attchment file
ARTD-22866	Bug	[EEP] Det reports incorrect API_ID at Eep_Init<*>

ID	Subtype	Headline and Description
		Detailed description (how to reproduce it): Create a timeout in Eep_Init, Driver return Write Time Out error code when Error Detect on Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: Eep_TC_FCT_0038 Observed behavior: Although Eep_Init timeout, Driver return EEP_WRITE_ID = EEP_E_TIMEOUT. Expected behavior: Expected Eep_Init timeout, Driver return EEP_INIT_ID = EEP_E_TIMEOUT. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Refer attchment picture as below for some sugesstions
ARTD-22877	Bug	[I2c] Fix issue can not link to channel of mcl and dma<*> Detailed description (how to reproduce it): [...] Flexio_Ip !image-2021-12-27-14-50-42-143.png! Lpi2c_Ip !image-2021-12-27-14-51-29-690.png! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]

ARTD-22882	New Feature	[platform] All trusted functions should NOT defined as static or inline functions and should listed out in IM<*>
		
		NewWorkDescription:
		There are some points that need to be checked in each modules:
		1. All trusted functions should NOT defined as static or inline functions*, so OS Application can be able to call them from outside RTD drivers.
		Check in driver code all IP functions that needs to be called as trusted functions using macros OsIf_Trusted_Call[_Return][1-6params].
		Ensure these functions are defined without "static" or "inline" keywords, so OS Application can use "extern" keyword to declare and call them outside RTD drivers.
		For example:
		a. OsIf_Trusted_Call1param(*Adc_Sar_ClrUserAccessAllowed*, pBase);
		Adc_Sar_ClrUserAccessAllowed() function should be defined without "static" or "inline" as:
		void Adc_Sar_ClrUserAccessAllowed(const ADC_Type const pBase)
		{
		CLR_USER_ACCESS_ALLOWED((uint32)pBase, SAR_ADC_PROT_MEM_U32);
		}
		b. OsIf_Trusted_Call_Return1param(*C40_Ip_SetLockProtect*, Fls_VirtualSector);
		C40_Ip_SetLockProtect() function should be defined without "static" or "inline" as:
		C40_Ip_StatusType C40_Ip_SetLockProtect(C40_Ip_VirtualSectorsType Fls_VirtualSector)
		{
		}
		2. Create a new separate header file <IpName>_Ip_TrustedFunctions.h for each IPV has trusted functions, the AutosarOS just needs to include this new header file when user mode is used.
		In <IpName>_Ip_TrustedFunctions.h:
		Declare all trusted functions with "extern" keyword like:
		extern void C40_Ip_SetLockProtect(C40_Ip_VirtualSectorsType Fls_VirtualSector);
		Include all header files contain the definition of all types which are used by parameters or return type of those trusted functions.
		3. All these trusted functions need to be configured in OS, so they must be clearly listed in the driver IM (chapter 5.7 User Mode Support)
		List these functions follow guiding mentioned in slide 17 Coding RTD_RunningInUserMode.pptx
		All functions should be listed out in the table like:
		Function syntax||Description||Available via
		void C40_Ip_SetLockProtect(
		    C40_Ip_VirtualSectorsType Fls_VirtualSector
		)|Set lock bit for flash sectors|C40_Ip_TrustedFunctions.h
		4. Subchapter "*User Mode configuration in AutosarOS*" needs to be added in IM:
		!image-2021-08-19-09-26-20-823.png|thumbnail!
		In order to add this subchapter, there are 2 files added in build_env: user_mode_support.dox, user_mode_config_in_autosar_os.dox (added on the ticket ARTD-15466).
		In module repository, each module needs to rename from user_mode_support*.dox to user_mode_config_in_module*.dox and also its title from "*User Mode support*" to "*User Mode configuration in the module*".
		Please refer the silde [Coding RTD_RunningInUserMode.pptx|https://nxp1.sharepoint.com/:p:/r/sites/Zebra/Shared%20Documents/General/Zebra%20Training/Coding%20-%20RTD_RunningInUserMode.pptx?d=w21c9578fff794ff387db07a106880274&csf=1&web=1&e=lpbnNC] to understand more about User mode implementation in RTD.
		For Testing side*: Ensure that we have test scenarios mentioned from slides 22->25 Coding RTD_RunningInUserMode.pptx.
		Follow slide 23 to ensure we have one test suite to check all trusted functions able to call from outside RTD drivers.
		Refer to the implementation example on FLS:* ARTD-15610

ID	Subtype	Headline and Description
		Requirement source: RTD implementation (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Follow Coding RTD_RunningInUserMode.pptx
ARTD-22884	Bug	[LIN] Missing case LIN_DUAL_CLOCK_MODE with IPV FlexIO for K1<*> Detailed description (how to reproduce it): Missing case LIN_DUAL_CLOCK_MODE with IPV FlexIO Preconditions: Missing case LIN_DUAL_CLOCK_MODE with IPV FlexIO Test Case ID (internal TC that caught the defect) optional: Lin_TC_FCT_0003 Observed behavior: NA Expected behavior: NA Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: add case LIN_DUAL_CLOCK_MODE with IPV FlexIO
ARTD-22890	New Feature	[S32k1 1.0.1] Update test case Ocu_TC_FCT_0031<*> NewWorkDescription: Update test case Ocu_TC_FCT_0031, incorrect error ID Requirement source: Update to correct error ID (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: [...]
ARTD-22900	Bug	[S32K1XX RTM 1.0.1][PORT] The generated patch version in S32DS ip layer is different with driver<*> Detailed description (how to reproduce it): The build error is as following: !image-2021-12-29-10-17-03-323.png! !image-2021-12-29-10-17-21-236.png! !image-2021-12-29-10-17-45-589.png! !image-2021-12-29-14-13-17-284.png! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-22902	Bug	[CAN] Can_SetBaudrate still change internal state in case of failure/timeout<*> Detailed description (how to reproduce it): !image-2021-12-29-10-27-41-462.png! Can_au16BaudrateIDConfig[Controller]* is being changed unconditionally (without checking eRetVal status => this will cause related api (Can_SetControllerMode will work on unexpected previous config state of {_}*Can_au16BaudrateIDConfig*{_}{_}){_} Please also re-check other similar API Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: CAN_TC_FCT_1666 Observed behavior: Expected behavior: status of setting api should be checked to make sure not to save non-working configuration, for example, as below expected handler: !image-2021-12-29-10-26-33-630.png! Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-22907	Bug	[gpt] Update missing exclusive areas in BSWMD<*> Detailed description (how to reproduce it): Do a repo sync on bswmd_creator to sync with the manifest (make sure that you've checked out BLN_BSWMD_CREATOR_04.08.03) Compile the plugin passing parameter GENERATE_BSWMD_FILE=ON to the make command (`p s1 b` in ZTH)

ID	Subtype	Headline and Description
		If the plugin was compiled successfully and there are no errors from bswmd_creator regarding missing or unmapped exclusive areas, then you can close your clone with "Fixed No Action Taken". Otherwise, proceed with the solutions given in the "Proposed solution" section. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Missing exclusive areas in <Module>_Bswmd.arxml Expected behavior: All exclusive areas are included in <Module>_Bswmd.arxml Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Potential Issue #1 Missing exclusive areas for ISRs Solution #1 The bswmd_creator script will automatically fetch ISRs from the source code as long as they're defined with ISR(...). Every ISR mentioned in the "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" files shall also be wrapped in ISR(...). For example: ADC_EXCLUSIVE_AREA_70 is used in function ISR(Ctu_Error_Isr) to protect the updates for CTU_IFR register Potential Issue #2 Missing exclusive areas for IPL functions which are not called by any HLD function Solution #2 # Have the EA mentioned in "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" only as used by the IPL function. For example: ADC_EXCLUSIVE_AREA_17 is used in function Adc_Sar_Ip_AbortConversion to protect the updates for ADC_MCR register # Add the IPL function in NonASR_ServiceID.xml Note: Do not use other workarounds like artificially mentioning in .dox that the IPL function is called by an HLD function which is included in NonASR_ServiceID.xml (even though, in reality, it's not)
ARTD-22911	New Feature	[S32K1 1.0.1] OCU: Update test case, test suite, makefile to correct format ,,"NewWorkDescription: Update test case, test suite, makefile to correct format Templates here: [^tc_template.c] Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: [...]
ARTD-22916	New Feature	[pwm] Enable ECPD generation for IP components (CPR_RTD_00544)<*> Note: This feature must be implemented by all the modules for which this ticket was cloned (including CDDs). No clone shall be closed with "Not Applicable" or "No Action Taken". How to implement: # Add "options_expr" tags (containing all the necessary options) for the fields of the IP component(s). Please refer to the slides covering "EPD Generation" in Reference #2 below. # To ensure that the ECPD is generated only for your component, modify line 512 in "mcu_data/components/PlatformSDK_S32K3_2021_08/system/codegenerator.js" as follows: {code:javascript} if (configComponent.isOptionSet("ecucDefinitionCollection") && (configComponent.getId() === "Gmac")) { where you can replace "Gmac" with the ID of your IP component. # Check that there are no errors reported in the Problems view and that the ECPD has been generated in Code Preview. How to validate: # Manually create an EBT plugin using the previously generated ECPD. To achieve this: ## Unzip https://jira.sw.nxp.com/secure/attachment/828304/Ip_TS_T40D34M10I0R0.zip in the "output/eclipse/plugins" folder of your K3 work tree / repository. ## Open "Ip_TS_T40D34M10I0R0/plugin.xml" and replace all occurrences of "Gmac" with the ID of your IP component. # Create an EBT project (File > New > Configuration Project > Project Name: TstIp > Next > ECU ID: someId; Target: CORTEXM/S32K3XX > Finish) and add the previously created module (Right click on TstIp in Project Explorer > Module Configurations... > Double click on your module > OK) CHECKPOINT 1: The module should have been successfully imported into the project. # Go to S32DS and create a valid project using your IP component (or use an already existing one). Please assign a value other than the default value to each setting to avoid default values in this configuration{color}*. Rename your IP component instance and discard the "_1" prefix (e.g. "Gmac_1" > "Gmac"): {panel}!screenshot-1.png! # Generate the [LT/PB]Cfg.h/.c files of your IP component and save them somewhere to avoid overwriting them in a later step. We'll call this set of files OriginalGenerateFiles_. # Generate the ECVD file for your IP component instance (Global Settings > EcvdGenerationMethod = INDIVIDUAL > EcvdOutputPath = <EcvdOutputDir> > Click on the "Generate Configuration" button) # Import the previously generated ECVD file in your EBT project (Right click on TstIp in Project Explorer > Im and Exporters... > Green Button > Next > File Name: <Browse and select the ECVD file> > Next > Enable path mapping > Finish > Run Importer) CHECKPOINT 2: The ECVD file should have been successfully applied to your module. # Save (CTRL S) and generate the project (Right click on TstIp in Project Explorer > Generate Project) to trigger the EPC generation. CHECKPOINT 3: The EPC file should have been successfully generated (TstIp/output/output/<Ip>.epc) # Import the previously generated EPC file in your S32DS project (File > Import > Import ECU Configuration > Path: <EpcOutputDir> > Check "Merge into current configuration" > Check "Check ECUC-MODULE-DEF" value > CHECKPOINT 4: The EPC file should have been successfully imported into your S32DS project. If you get the error "The required modules are not available for the current configuration" it means that the ECUC-MODULE-DEF check has failed and you're not properly using M4_XDM_AR_PKG_NAME in your IP component.

ID	Subtype	Headline and Description
		# Generate again the [LT/PB]Cfg.h/.c files of your IP component and save them somewhere without overwriting the previous set of files. We'll call this new set of files ImportedGenerateFiles_. # Compare the OriginalGenerateFiles set with the ImportedGenerateFiles set. CHECKPOINT 5: The comparison should be binary identical. There should be no single difference between the two set of files. Requirements covered by this ticket: CPR_RTD_00544 EA_RTD_00107 EA_RTD_00108 References: # https://jira.sw.nxp.com/secure/attachment/828307/Ethernet.template # https://nxp1.sharepoint.com/:p:/s/Zebra/EaMbpShSxPVAq7amkzZx3NMB1LWDlRvDlnrzP1m0TZ9elQ?e=erB0sH (Relevant chapters: "9. EPD Importer Replacing the lost GNU M4 macros", "10. EPD Generation", "11. EPC Importer", "12. EPC Generation") Legend: EPD ECU Parameter Definition (Schema) Used by EBT ECPD ECU Configuration Parameter Definition (Schema) Used by S32CT EPC ECU Parameter Configuration (Configuration File) Used by EBT ECVD ECU Configuration Value Description (Configuration File) Used by S32CT
ARTD-22917	New Feature	[gpt] Enable ECPD generation for IP components (CPR_RTD_00544)<*> Note: This feature must be implemented by all the modules for which this ticket was cloned (including CDDs). No clone shall be closed with "Not Applicable" or "No Action Taken". How to implement: # Add "options_expr" tags (containing all the necessary options) for the fields of the IP component(s). Please refer to the slides covering "EPD Generation" in Reference #2 below. # To ensure that the ECPD is generated only for your component, modify line 512 in "mcu_data/components/PlatformSDK_S32K3_2021_08/system/codegenerator.js" as follows: {code:javascript} if (configComponent.isOptionSet("ecucDefinitionCollection") && (configComponent.getId() === "Gmac")) { where you can replace "Gmac" with the ID of your IP component. # Check that there are no errors reported in the Problems view and that the ECPD has been generated in Code Preview. How to validate: # Manually create an EBT plugin using the previously generated ECPD. To achieve this: ## Unzip https://jira.sw.nxp.com/secure/attachment/828304/Ip_TS_T40D34M10I0R0.zip in the "output/eclipse/plugins" folder of your K3 work tree / repository. ## Open "Ip_TS_T40D34M10I0R0/plugin.xml" and replace all occurrences of "Gmac" with the ID of your IP component. # Create an EBT project (File > New > Configuration Project > Project Name: TstIp > Next > ECU ID: someId; Target: CORTEXM/S32K3XX > Finish) and add the previously created module (Right click on TstIp in Project Explorer > Module Configurations... > Double click on your module > OK) CHECKPOINT 1: The module should have been successfully imported into the project. # Go to S32DS and create a valid project using your IP component (or use an already existing one). Please assign a value other than the default value to each setting to avoid default values in this configuration{color}*. Rename your IP component instance and discard the "_1" prefix (e.g. "Gmac_1" > "Gmac"): {panel}!screenshot-1.png! # Generate the [LT/PB]Cfg.h/.c files of your IP component and save them somewhere to avoid overwriting them in a later step. We'll call this set of files OriginalGenerateFiles_. # Generate the ECVD file for your IP component instance (Global Settings > EcvdGenerationMethod = INDIVIDUAL > EcvdOutputPath = <EcvdOutputDir> > Click on the "Generate Configuration" button) # Import the previously generated ECVD file in your EBT project (Right click on TstIp in Project Explorer > Im and Exporters... > Green Button > Next > File Name: <Browse and select the ECVD file> > Next > Enable path mapping > Finish > Run Importer) CHECKPOINT 2: The ECVD file should have been successfully applied to your module. # Save (CTRL S) and generate the project (Right click on TstIp in Project Explorer > Generate Project) to trigger the EPC generation. CHECKPOINT 3: The EPC file should have been successfully generated (TstIp/output/output/<Ip>.epc) # Import the previously generated EPC file in your S32DS project (File > Import > Import ECU Configuration > Path: <EpcOutputDir> > Check "Merge into current configuration" > Check "Check ECUC-MODULE-DEF" value > CHECKPOINT 4: The EPC file should have been successfully imported into your S32DS project. If you get the error "The required modules are not available for the current configuration" it means that the ECUC-MODULE-DEF check has failed and you're not properly using M4_XDM_AR_PKG_NAME in your IP component. # Generate again the [LT/PB]Cfg.h/.c files of your IP component and save them somewhere without overwriting the previous set of files. We'll call this new set of files ImportedGenerateFiles_. # Compare the OriginalGenerateFiles set with the ImportedGenerateFiles set. CHECKPOINT 5: The comparison should be binary identical. There should be no single difference between the two set of files. Requirements covered by this ticket: CPR_RTD_00544 EA_RTD_00107 EA_RTD_00108 References: # https://jira.sw.nxp.com/secure/attachment/828307/Ethernet.template # https://nxp1.sharepoint.com/:p:/s/Zebra/EaMbpShSxPVAq7amkzZx3NMB1LWDlRvDlnrzP1m0TZ9elQ?e=erB0sH (Relevant chapters: "9. EPD Importer Replacing the lost GNU M4 macros", "10. EPD Generation", "11. EPC Importer", "12. EPC Generation") Legend: EPD ECU Parameter Definition (Schema) Used by EBT ECPD ECU Configuration Parameter Definition (Schema) Used by S32CT EPC ECU Parameter Configuration (Configuration File) Used by EBT ECVD ECU Configuration Value Description (Configuration File) Used by S32CT
ARTD-22918	New Feature	[icu] Enable ECPD generation for IP components (CPR_RTD_00544)<*> Note: This feature must be implemented by all the modules for which this ticket was cloned (including CDDs). No clone shall be closed with "Not Applicable" or "No Action Taken". How to implement: # Add "options_expr" tags (containing all the necessary options) for the fields of the IP component(s). Please refer to the slides covering "EPD Generation" in Reference #2 below.

ID	Subtype	Headline and Description
		# To ensure that the ECPD is generated only for your component, modify line 512 in "mcu_data/components/PlatformSDK_S32K3_2021_08/system/codegenerator.js" as follows: {code:javascript} if (configComponent.isOptionSet("ecucDefinitionCollection") && (configComponent.getId() === "Gmac")) { where you can replace "Gmac" with the ID of your IP component. # Check that there are no errors reported in the Problems view and that the ECPD has been generated in Code Preview. How to validate: # Manually create an EBT plugin using the previously generated ECPD. To achieve this: ## Unzip https://jira.sw.nxp.com/secure/attachment/828304/Ip_TS_T40D34M10I0R0.zip in the "output/eclipse/plugins" folder of your K3 work tree / repository. ## Open "Ip_TS_T40D34M10I0R0/plugin.xml" and replace all occurrences of "Gmac" with the ID of your IP component. # Create an EBT project (File > New > Configuration Project > Project Name: TstIp > Next > ECU ID: someId; Target: CORTEXM/S32K3XX > Finish) and add the previously created module (Right click on TstIp in Project Explorer > Module Configurations... > Double click on your module > OK) CHECKPOINT 1: The module should have been successfully imported into the project. # Go to S32DS and create a valid project using your IP component (or use an already existing one). Please assign a value other than the default value to each setting to avoid default values in this configuration{color}*. Rename your IP component instance and discard the "_1" prefix (e.g. "Gmac_1" > "Gmac"): {panel}!screenshot-1.png! # Generate the [LT/PB]Cfg.h/.c files of your IP component and save them somewhere to avoid overwriting them in a later step. We'll call this set of files OriginalGenerateFiles_. # Generate the ECVD file for your IP component instance (Global Settings > EcvdGenerationMethod = INDIVIDUAL > EcvdOutputPath = <EcvdOutputDir> > Click on the "Generate Configuration" button) # Import the previously generated ECVD file in your EBT project (Right click on TstIp in Project Explorer > Im and Exporters... > Green Button > Next > File Name: <Browse and select the ECVD file> > Next > Enable path mapping > Finish > Run Importer) CHECKPOINT 2: The ECVD file should have been successfully applied to your module. # Save (CTRL S) and generate the project (Right click on TstIp in Project Explorer > Generate Project) to trigger the EPC generation. CHECKPOINT 3: The EPC file should have been successfully generated (TstIp/output/output/<Ip>.epc) # Import the previously generated EPC file in your S32DS project (File > Import > Import ECU Configuration > Path: <EpcOutputDir> > Check "Merge into current configuration" > Check "Check ECUC-MODULE-DEF" value > CHECKPOINT 4: The EPC file should have been successfully imported into your S32DS project. If you get the error "The required modules are not available for the current configuration" it means that the ECUC-MODULE-DEF check has failed and you're not properly using M4_XDM_AR_PKG_NAME in your IP component. # Generate again the [LT/PB]Cfg.h/.c files of your IP component and save them somewhere without overwriting the previous set of files. We'll call this new set of files ImportedGenerateFiles_. # Compare the OriginalGenerateFiles set with the ImportedGenerateFiles set. CHECKPOINT 5: The comparison should be binary identical. There should be no single difference between the two set of files. Requirements covered by this ticket: CPR_RTD_00544 EA_RTD_00107 EA_RTD_00108 References: # https://jira.sw.nxp.com/secure/attachment/828307/Ethernet.template # https://nxp1.sharepoint.com/:p:/s/Zebra/EaMbpShSxPVAq7amkzZx3NMB1LWDlRvDlnrzP1m0TZ9elQ?e=erB0sH (Relevant chapters: "9. EPD Importer Replacing the lost GNU M4 macros", "10. EPD Generation", "11. EPC Importer", "12. EPC Generation") Legend: EPD ECU Parameter Definition (Schema) Used by EBT ECPD ECU Configuration Parameter Definition (Schema) Used by S32CT EPC ECU Parameter Configuration (Configuration File) Used by EBT ECVD ECU Configuration Value Description (Configuration File) Used by S32CT
ARTD-22919	New Feature	[S32K1 1.0.1] OCU: Enable ECPD generation for IP components (CPR_RTD_00544)<*> Note: This feature must be implemented by all the modules for which this ticket was cloned (including CDDs). No clone shall be closed with "Not Applicable" or "No Action Taken". How to implement: # Add "options_expr" tags (containing all the necessary options) for the fields of the IP component(s). Please refer to the slides covering "EPD Generation" in Reference #2 below. # To ensure that the ECPD is generated only for your component, modify line 512 in "mcu_data/components/PlatformSDK_S32K3_2021_08/system/codegenerator.js" as follows: {code:javascript} if (configComponent.isOptionSet("ecucDefinitionCollection") && (configComponent.getId() === "Gmac")) { where you can replace "Gmac" with the ID of your IP component. # Check that there are no errors reported in the Problems view and that the ECPD has been generated in Code Preview. How to validate: # Manually create an EBT plugin using the previously generated ECPD. To achieve this: ## Unzip https://jira.sw.nxp.com/secure/attachment/828304/Ip_TS_T40D34M10I0R0.zip in the "output/eclipse/plugins" folder of your K3 work tree / repository. ## Open "Ip_TS_T40D34M10I0R0/plugin.xml" and replace all occurrences of "Gmac" with the ID of your IP component. # Create an EBT project (File > New > Configuration Project > Project Name: TstIp > Next > ECU ID: someId; Target: CORTEXM/S32K3XX > Finish) and add the previously created module (Right click on TstIp in Project Explorer > Module Configurations... > Double click on your module > OK) CHECKPOINT 1: The module should have been successfully imported into the project. # Go to S32DS and create a valid project using your IP component (or use an already existing one). Please assign a value other than the default value to each setting to avoid default values in this configuration{color}*. Rename your IP component instance and discard the "_1" prefix (e.g. "Gmac_1" > "Gmac"): {panel}!screenshot-1.png! # Generate the [LT/PB]Cfg.h/.c files of your IP component and save them somewhere to avoid overwriting them in a later step. We'll call this set of files OriginalGenerateFiles_. # Generate the ECVD file for your IP component instance (Global Settings > EcvdGenerationMethod = INDIVIDUAL > EcvdOutputPath = <EcvdOutputDir> > Click on the "Generate Configuration" button) # Import the previously generated ECVD file in your EBT project (Right click on TstIp in Project Explorer > Im and Exporters... > Green Button > Next > File Name: <Browse and select the ECVD file> > Next > Enable path mapping > Finish > Run Importer) CHECKPOINT 2: The ECVD file should have been successfully applied to your module.

ID	Subtype	Headline and Description
		# Save (CTRL S) and generate the project (Right click on TstIp in Project Explorer > Generate Project) to trigger the EPC generation. CHECKPOINT 3: The EPC file should have been successfully generated (TstIp/output/output/<Ip>.epc) # Import the previously generated EPC file in your S32DS project (File > Import > Import ECU Configuration > Path: <EpcOutputDir> > Check "Merge into current configuration" > Check "Check ECUC-MODULE-DEF" value > CHECKPOINT 4: The EPC file should have been successfully imported into your S32DS project. If you get the error "The required modules are not available for the current configuration" it means that the ECUC-MODULE-DEF check has failed and you're not properly using M4_XDM_AR_PKG_NAME in your IP component. # Generate again the [LT/PB]Cfg.h/.c files of your IP component and save them somewhere without overwriting the previous set of files. We'll call this new set of files ImportedGenerateFiles_. # Compare the OriginalGenerateFiles set with the ImportedGenerateFiles set. CHECKPOINT 5: The comparison should be binary identical. There should be no single difference between the two set of files. Requirements covered by this ticket: CPR_RTD_00544 EA_RTD_00107 EA_RTD_00108 References: # https://jira.sw.nxp.com/secure/attachment/828307/Ethernet.template # https://nxp1.sharepoint.com/:p:/s/Zebra/EaMbpShSxPVAq7amkzZx3NMB1LWDlRvDlnrzP1m0TZ9elQ?e=erB0sH (Relevant chapters: "9. EPD Importer Replacing the lost GNU M4 macros", "10. EPD Generation", "11. EPC Importer", "12. EPC Generation") Legend: EPD ECU Parameter Definition (Schema) Used by EBT ECPD ECU Configuration Parameter Definition (Schema) Used by S32CT EPC ECU Parameter Configuration (Configuration File) Used by EBT ECVD ECU Configuration Value Description (Configuration File) Used by S32CT
ARTD-22920	New Feature	[qd] Enable ECPD generation for IP components (CPR_RTD_00544)<*> Note: This feature must be implemented by all the modules for which this ticket was cloned (including CDDs). No clone shall be closed with "Not Applicable" or "No Action Taken". How to implement: # Add "options_expr" tags (containing all the necessary options) for the fields of the IP component(s). Please refer to the slides covering "EPD Generation" in Reference #2 below. # To ensure that the ECPD is generated only for your component, modify line 512 in "mcu_data/components/PlatformSDK_S32K3_2021_08/system/codegenerator.js" as follows: {code:javascript} if (configComponent.isOptionSet("ecucDefinitionCollection") && (configComponent.getId() === "Gmac")) { where you can replace "Gmac" with the ID of your IP component. # Check that there are no errors reported in the Problems view and that the ECPD has been generated in Code Preview. How to validate: # Manually create an EBT plugin using the previously generated ECPD. To achieve this: ## Unzip https://jira.sw.nxp.com/secure/attachment/828304/Ip_TS_T40D34M10I0R0.zip in the "output/eclipse/plugins" folder of your K3 work tree / repository. ## Open "Ip_TS_T40D34M10I0R0/plugin.xml" and replace all occurrences of "Gmac" with the ID of your IP component. # Create an EBT project (File > New > Configuration Project > Project Name: TstIp > Next > ECU ID: someId; Target: CORTEXM/S32K3XX > Finish) and add the previously created module (Right click on TstIp in Project Explorer > Module Configurations... > Double click on your module > OK) CHECKPOINT 1: The module should have been successfully imported into the project. # Go to S32DS and create a valid project using your IP component (or use an already existing one). Please assign a value other than the default value to each setting to avoid default values in this configuration{color}*. Rename your IP component instance and discard the "_1" prefix (e.g. "Gmac_1" > "Gmac"): {panel}!screenshot-1.png! # Generate the [LT/PB]Cfg.h/.c files of your IP component and save them somewhere to avoid overwriting them in a later step. We'll call this set of files OriginalGenerateFiles_. # Generate the ECVD file for your IP component instance (Global Settings > EcvdGenerationMethod = INDIVIDUAL > EcvdOutputPath = <EcvdOutputDir> > Click on the "Generate Configuration" button) # Import the previously generated ECVD file in your EBT project (Right click on TstIp in Project Explorer > Im and Exporters... > Green Button > Next > File Name: <Browse and select the ECVD file> > Next > Enable path mapping > Finish > Run Importer) CHECKPOINT 2: The ECVD file should have been successfully applied to your module. # Save (CTRL S) and generate the project (Right click on TstIp in Project Explorer > Generate Project) to trigger the EPC generation. CHECKPOINT 3: The EPC file should have been successfully generated (TstIp/output/output/<Ip>.epc) # Import the previously generated EPC file in your S32DS project (File > Import > Import ECU Configuration > Path: <EpcOutputDir> > Check "Merge into current configuration" > Check "Check ECUC-MODULE-DEF" value > CHECKPOINT 4: The EPC file should have been successfully imported into your S32DS project. If you get the error "The required modules are not available for the current configuration" it means that the ECUC-MODULE-DEF check has failed and you're not properly using M4_XDM_AR_PKG_NAME in your IP component. # Generate again the [LT/PB]Cfg.h/.c files of your IP component and save them somewhere without overwriting the previous set of files. We'll call this new set of files ImportedGenerateFiles_. # Compare the OriginalGenerateFiles set with the ImportedGenerateFiles set. CHECKPOINT 5: The comparison should be binary identical. There should be no single difference between the two set of files. Requirements covered by this ticket: CPR_RTD_00544 EA_RTD_00107 EA_RTD_00108 References: # https://jira.sw.nxp.com/secure/attachment/828307/Ethernet.template # https://nxp1.sharepoint.com/:p:/s/Zebra/EaMbpShSxPVAq7amkzZx3NMB1LWDlRvDlnrzP1m0TZ9elQ?e=erB0sH (Relevant chapters: "9. EPD Importer Replacing the lost GNU M4 macros", "10. EPD Generation", "11. EPC Importer", "12. EPC Generation") Legend: EPD ECU Parameter Definition (Schema) Used by EBT ECPD ECU Configuration Parameter Definition (Schema) Used by S32CT EPC ECU Parameter Configuration (Configuration File) Used by EBT ECVD ECU Configuration Value Description (Configuration File) Used by S32CT
ARTD-22933	Bug	

ID	Subtype	Headline and Description
		[CAN] CAN_BUSOFF_POLLING_SUPPORT is generated differently between S32CT and EB<*> Detailed description (how to reproduce it): Create s32ct / s32k148 project: config CanController_0/ bus-off processing type as POLL config CanController_1/ bus-off processing type as INTERRUPT Make the same settings on EB, then comparing two generated code Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: !image-2021-12-31-15-28-51-444.png! Expected behavior: Two generators should generate similar code Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-22935	Bug	[S32K1XX][Crypto] Build fail CCOV due to missing define<*> Detailed description (how to reproduce it): Build fail CCOV test due to missing define CRYPTO_SPT_ENC_AUTH_KEY_EXPORT in driver and function Crypto_AuthCipherModeIsCcmOrGcm not used in CSEC on tag CRYPTO_104 Observed behavior: Missing define CRYPTO_SPT_ENC_AUTH_KEY_EXPORT in driver and function Crypto_AuthCipherModeIsCcmOrGcm not fenced to only use to HSE Expected behavior: Define CRYPTO_SPT_ENC_AUTH_KEY_EXPORT and function Crypto_AuthCipherModeIsCcmOrGcm need to be fenced with M4 tag to avoid compiler warnings. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-22971	Bug	[S32K1XX RTM 1.0.0] [ICU] There are some error with ecpd file in S32DS<*> Detailed description (how to reproduce it): The name in line 14 of the ecpd file for FTM Ip is wrong so the plugin from DS is error to convert to EB. It must be "Ftm_Icu".{*}{*} Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: Ip_Ftm_TC_COT_0500 (Ip_Ftm_TS_COT_0500). Observed behavior: ecpd for Ftm is error and ecpd for other HW is not generated. Expected behavior: [...] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-22966	Bug	[CAN] Missing constraint for numbers of CanIcomRxMessage<*> Detailed description (how to reproduce it): Create S32K148 project Enable CanIcom Add CanIcomConfig Add two CanIcomRxMessage_0, CanIcomRxMessage_1 => verification point: gui shall not allow more than 1 CanIcomRxMessag per CanIcomConfig Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: !image-2022-01-04-10-53-42-522.png! Expected behavior: S32CT should apply contraint as similar as EB Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-23002	Bug	[S32K1XX] [Crypto] Wrong field short name in ECVD file<*> Detailed description (how to reproduce it): When config IP layer on S32DS, field short name in file ECVD being filled is "crypto" ,it should be " Csec" Observed behavior: Field short name in ECVD file being filled is "crypto" when config IP layer Csec Expected behavior: Correct field short name is "Csec" when gen ECVD file

ID	Subtype	Headline and Description
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-23015	New Feature	CLONE - [i2c] Enable ECPD generation for IP components (CPR_RTD_00544)<*> Note: This feature must be implemented by all the modules for which this ticket was cloned (including CDDs). No clone shall be closed with "Not Applicable" or "No Action Taken". How to implement: # Add "options_expr" tags (containing all the necessary options) for the fields of the IP component(s). Please refer to the slides covering "EPD Generation" in Reference #2 below. # To ensure that the ECPD is generated only for your component, modify line 512 in "mcu_data/components/PlatformSDK_S32K3_2021_08/system/codegenerator.js" as follows: {code:javascript} if (configComponent.isOptionSet("ecucDefinitionCollection") && (configComponent.getId() === "Gmac")) { where you can replace "Gmac" with the ID of your IP component. # Check that there are no errors reported in the Problems view and that the ECPD has been generated in Code Preview. How to validate: # Manually create an EBT plugin using the previously generated ECPD. To achieve this: ## Unzip https://jira.sw.nxp.com/secure/attachment/828304/Ip_TS_T40D34M10I0R0.zip in the "output/eclipse/plugins" folder of your K3 work tree / repository. ## Open "Ip_TS_T40D34M10I0R0/plugin.xml" and replace all occurrences of "Gmac" with the ID of your IP component. # Create an EBT project (File > New > Configuration Project > Project Name: TstIp > Next > ECU ID: someId; Target: CORTEXM/S32K3XX > Finish) and add the previously created module (Right click on TstIp in Project Explorer > Module Configurations... > Double click on your module > OK) CHECKPOINT 1: The module should have been successfully imported into the project. # Go to S32DS and create a valid project using your IP component (or use an already existing one). Please assign a value other than the default value to each setting to avoid default values in this configuration{color}*. Rename your IP component instance and discard the "_1" prefix (e.g. "Gmac_1" > "Gmac"): {panel}!screenshot-1.png! # Generate the [LT/PB]Cfg.h/.c files of your IP component and save them somewhere to avoid overwriting them in a later step. We'll call this set of files OriginalGenerateFiles_. # Generate the ECVD file for your IP component instance (Global Settings > EcvdGenerationMethod = INDIVIDUAL > EcvdOutputPath = <EcvdOutputDir> > Click on the "Generate Configuration" button) # Import the previously generated ECVD file in your EBT project (Right click on TstIp in Project Explorer > Im and Exporters... > Green Button > Next > File Name: <Browse and select the ECVD file> > Next > Enable path mapping > Finish > Run Importer) CHECKPOINT 2: The ECVD file should have been successfully applied to your module. # Save (CTRL S) and generate the project (Right click on TstIp in Project Explorer > Generate Project) to trigger the EPC generation. CHECKPOINT 3: The EPC file should have been successfully generated (TstIp/output/output/<Ip>.epc) # Import the previously generated EPC file in your S32DS project (File > Import > Import ECU Configuration > Path: <EpcOutputDir> > Check "Merge into current configuration" > Check "Check ECUC-MODULE-DEF" value > CHECKPOINT 4: The EPC file should have been successfully imported into your S32DS project. If you get the error "The required modules are not available for the current configuration" it means that the ECUC-MODULE-DEF check has failed and you're not properly using M4_XDM_AR_PKG_NAME in your IP component. # Generate again the [LT/PB]Cfg.h/.c files of your IP component and save them somewhere without overwriting the previous set of files. We'll call this new set of files ImportedGenerateFiles_. # Compare the OriginalGenerateFiles set with the ImportedGenerateFiles set. CHECKPOINT 5: The comparison should be binary identical. There should be no single difference between the two set of files. Requirements covered by this ticket: CPR_RTD_00544 EA_RTD_00107 EA_RTD_00108 References: # https://jira.sw.nxp.com/secure/attachment/828307/Ethernet.template # https://nxp1.sharepoint.com/:p:/s/Zebra/EaMbpShSxPVAq7amkzZx3NMB1LWDlRvDlnrzP1m0TZ9elQ?e=erB0sH (Relevant chapters: "9. EPD Importer Replacing the lost GNU M4 macros", "10. EPD Generation", "11. EPC Importer", "12. EPC Generation") Legend: EPD ECU Parameter Definition (Schema) Used by EBT ECPD ECU Configuration Parameter Definition (Schema) Used by S32CT EPC ECU Parameter Configuration (Configuration File) Used by EBT ECVD ECU Configuration Value Description (Configuration File) Used by S32CT
ARTD-23018	Bug	[icu] Update missing exclusive areas in BSWMD<*> Detailed description (how to reproduce it): Do a repo sync on bswmd_creator to sync with the manifest (make sure that you've checked out BLN_BSWMD_CREATOR_04.08.03) Compile the plugin passing parameter GENERATE_BSWMD_FILE=ON to the make command (`p s1 b` in ZTH) If the plugin was compiled successfully and there are no errors from bswmd_creator regarding missing or unmapped exclusive areas, then you can close your clone with "Fixed No Action Taken". Otherwise, proceed with the solutions given in the "Proposed solution" section. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Missing exclusive areas in <Module>_Bswmd.arxml Expected behavior: All exclusive areas are included in <Module>_Bswmd.arxml Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Potential Issue #1

ID	Subtype	Headline and Description
		Missing exclusive areas for ISRs Solution #1 The bswmd_creator script will automatically fetch ISRs from the source code as long as they're defined with ISR(...). Every ISR mentioned in the "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" files shall also be wrapped in ISR(...). For example: ADC_EXCLUSIVE_AREA_70 is used in function ISR(Ctu_Error_Isr) to protect the updates for CTU_IFR register Potential Issue #2 Missing exclusive areas for IPL functions which are not called by any HLD function Solution #2 # Have the EA mentioned in "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" only as used by the IPL function. For example: ADC_EXCLUSIVE_AREA_17 is used in function Adc_Sar_Ip_AbortConversion to protect the updates for ADC_MCR register # Add the IPL function in NonASR_ServiceID.xml Note: Do not use other workarounds like artificially mentioning in .dox that the IPL function is called by an HLD function which is included in NonASR_ServiceID.xml (even though, in reality, it's not)
ARTD-23029	Bug	[PWM] Unable to generate phase shift pulses if the phase shift plus the duty cycle is beyond 100%<*> Detailed description (how to reproduce it): In {*}FTM Modified Combine Mode{*}, it's allowed to have the phase shift plus the duty cycle beyond 100%, i.e. C(n)V > C(n+1)V. See below picture: !Cn-Cn+1.png|width=425,height=124! However, in RTD driver, with below condition in Ftm_Pwm_Ip_UpdatePwmChannel(), !image-2022-01-07-10-05-43-635.png|width=473,height=156! The case that the phase shift plus the duty cycle beyond 100% returns an errors, because the secondEdge would be bigger than the {_}ftmPeriod{_}. By changing the code not to return an error when the overflow of the secondEdge_ occurs, it was able to generate a PWM pulse with the phase shift plus the duty cycle beyond 100% Preconditions: FTM is selected as hw peripheral, Modified Combine Mode is used Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Unable to generate a PWM pulse with the phase shift plus the duty cycle beyond 100% Expected behavior: Able to generate a PWM pulse with the phase shift plus the duty cycle beyond 100% Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-23034	Bug	[S32K1XX][DIO] Can not generate ecpd file from S32DS<*> Detailed description (how to reproduce it): 1. Install S32DS 3.4 and devpackage for S32K1XX 2. Layout and generate ecpd file use command: eclipsec.exe noSplash application com.nxp.swtools.framework.application launcher.ini s32ds.ini HeadlessTool Peripherals MCU S32K148 SDKversion PlatformSDK_S32K1_2022_02 ExportSrc ../ S32K1XX_101/output/eclipse/plugins/ecpd ExportArgs ecpd ExportComponentIds Gpio_Dio Preconditions: Dio tag: PVT_DIO_S32K1XX_1.0.1_V01 Test Case ID (internal TC that caught the defect) optional: IP_Dio_TC_COT_0000 Observed behavior: ecpd file was not generated in folder plugins Expected behavior: ecpd file should appear int plugins/ecpd after running command mentioned in detailed description Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-23035	Bug	[CAN] missing information when exporting ecvd for FlexCAN / IP layer / S32K1xx and can't export ecpd<*> Detailed description (how to reproduce it): {*}A{*}> Create s32k148 s32ct project Add FlexCAN Configuration with maximum features: include PN network Update code > original configuration (A) Export ecvd, then re-import ecvd Update code > new configuration (B) => compare A and B, you can observe many diff in .mex files you can re-use attached .mex for reproducing this issue !image-2022-01-07-12-39-57-736.png! B> enable ecpd generation for FlexCAN => code generation fail error !image-2022-01-07-12-46-58-887.png! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Review Observed behavior: ecvd was not exported with all gui data Expected behavior: ecvd should be exported with all gui data

ID	Subtype	Headline and Description
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-23045	Bug	[I2c] Fix different short name node in ecvd file<*> Detailed description (how to reproduce it): [...] Fix different in ecvd file on LPi2c DS !image-2022-01-07-16-18-09-660.png! On HLD DS !image-2022-01-10-10-25-24-930.png!!image-2022-01-10-10-26-05-981.png!!image-2022-01-10-10-29-51-225.png!!image-2022-01-10-10-30-45-638.png! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-23049	Bug	[ADC] Adc Hardware Group and Software Continuous can not transfer by DMA since limit check failed<*> Detailed description (how to reproduce it): Adc Hardware Group and Software Continuous can not transfer by DMA since limit check failed because DMA Request was disabled automatically after first transferring is completed. Software Single Access does not affected because if user want to restart conversion, they have to call Adc_StartGroupConversion API again. This will setup DMA again for the group. Preconditions: Limit check is enable Using Dma transfer Using hw trigger oneshot group or software continuous Test Case ID (internal TC that caught the defect) optional: Adc_TC_FCT_1401 Adc_TS_006 cfg BetweenDMA Observed behavior: Enable Hw trigger Start trigger with out range voltage Disable hw trigger Verify that group state is busy because limit check failed Change voltage to in range voltage Start trigger Expect: group status is stream complete Real status: group status is busy and dma did not transfer data ERQ bit of dma to enable hw request from adc did not set Expected behavior: Group status is stream complete Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-23074	Bug	[Port] Difference from generated source files<*> Detailed description (how to reproduce it): Use tag PVT_PORT_S32K1XX_1.0.1_V03 for the plugins Create 2 configuration with the same setting for both requiring the feature Lock Register is enabled. After generating sources, open Port_VS_0_PBcfg.c in both CT and EB 's generated files. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: 1. au8Port_PartitionList_VS_0 in CTHL was missing the static keyword as below: !image-2022-01-10-15-12-43-679.png! 2. the array Port_aPinConfigDefault_VS_0 in CT was missing the LK bit enable for Lock Register feature even it was checked in setting phase: !image-2022-01-10-15-13-37-668.png! Expected behavior: au8Port_PartitionList_VS_0 in both CT and EBHL should have the same keyword definition (both have static or both be removed) and Port_aPinConfigDefault_VS_0* in CTHL must generate the PCR value with LK bit (if it is configurated)
ARTD-23079	Bug	[UART] Uart HLD and Flexio_Uart IP are missing in S32DS Component window<*> Detailed description (how to reproduce it): Uart HLD and Flexio_Uart IP missing in S32DS Component window Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA

ID	Subtype	Headline and Description
		Observed behavior: Uart HLD and Flexio_Uart IP are missing in S32DS Component window Expected behavior: Uart HLD and Flexio_Uart IP NOT missing in S32DS Component window !image-2022-01-10-16-27-35-895.png|thumbnail! Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-23092	Bug	[SPI] ECPD HLD missing SpiChannelList in SpiJob container<*> Detailed description (how to reproduce it): ECPD HLD missing SpiChannelList in SpiJob container: !image-2022-01-11-09-45-26-337.png|thumbnail! spi ecvd and ecpd file have not SpiChannelList in SpiJob container Preconditions: NA Test Case ID (internal TC that caught the defect) optional: Spi_TS_COT_002 Observed behavior: NA Expected behavior: NA Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-23096	Bug	[S32K1XX 1.0.1 RTM] Fix ECPD generate fail<*> Detailed description (how to reproduce it): Can't generate ECPD file on S32DS Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: Generate ECPD successfully Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-23100	Bug	[PWM] Fail at build when running CCOV enable<*> Detailed description (how to reproduce it): MCAL_FTM_REG_PROT_AVAILABLE is undefined in Ftm_Pwm_Ip_Cfg.h file. when running with CCOV_ENABLE. it cause fail at build Preconditions: CCOV_ENABLE = ON Test Case ID (internal TC that caught the defect) optional: Pwm_TS_C001 Observed behavior: N/A Expected behavior: N/A Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-23102	Bug	[LIN] Function interface don't matching with function API in driver<*> Detailed description (how to reproduce it): Function interface don't matching with function API in driver Preconditions: !image-2022-01-11-14-58-20-399.png! Test Case ID (internal TC that caught the defect) optional: !image-2022-01-11-14-58-25-115.png! Observed behavior: NA Expected behavior: NA Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-23115	Bug	[S32K1XX][PORT] Hard Fault Handle when config PTB0 as ADC interleave mode<*> Detailed description (how to reproduce it): Use Plugin_1012022, config PTB0 as ADC interleave in IPL, check Port_Ci_Port_Ip_Init

ID	Subtype	Headline and Description
		Preconditions: Port_Ci_Port_Ip_Init can initialize when config ADC interleave Test Case ID (internal TC that caught the defect) optional: IP_Port_Ci_TC_0002/4 Observed behavior: Hard fault when over line 269 !image-2022-01-12-09-34-42-765.png! Expected behavior: Driver can support ADC interleave when call Port_Ci_Port_Ip_Init Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA

ARTD-23116	Bug	[WDG] Init Wdog error after calling Wdog_Ip_DeInit function<*>
		
		Detailed description (how to reproduce it):
		After Wdog_Ip_DeInit, "UPDATE" bit in CS register is clear to 0 (due to reset value 0000_2980h)
		When we call Wdog_Ip_Init to Init wdg again, it will raise an error with the check
		if (!Wdog_Ip_IsUpdateEnabled(Base))
		{
		Status = WDOG_IP_STATUS_ERROR;
		}
		So we can Init again after call Wdog_Ip_Init.
		Preconditions:
		[...]
		Test Case ID (internal TC that caught the defect) optional:
		[...]
		Observed behavior:
		"UPDATE" bit is disable after call Wdog_Ip_DeInit function
		Expected behavior:
		"UPDATE" bit should be set after call Wdog_Ip_DeInit function
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)
		Proposed solution optional:
		[...]

ARTD-23125	Bug	[WDG] Can't generate ecvd file for HLD and IP on S32DS<*> Detailed description (how to reproduce it): Step 1: Create new project into S32DS Step 2: For IP: add Wdog_Ip, Ewm_Ip components For HLD: add Wdg, Wdg_43_Instance1, Mcu components Step 3: Click Peripherals > Global Settings > Select Directory > Generate Configuration Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: Cannot generate ecvd file for HLD and IP after creating configuration !image-2022-01-12-15-01-30-164.png! Expected behavior: Can generate ecvd file Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-23127	Bug	[ADC] Build fail: "unresolved symbols: EndConversionNotification" when configuring transfer types are different between multi VS ,,"Detailed description (how to reproduce it): Build fail: ""unresolved symbols: EndConversionNotification"" when configuring transfer types are different between multi VS Preconditions: VS_0: ADC0: AdcTransferType: ADC_INTERRUPT ADC1: AdcTransferType: ADC_DMA VS_1: ADC0: AdcTransferType: ADC_DMA ADC1: AdcTransferType: ADC_INTERRUPT Test Case ID (internal TC that caught the defect) optional: Adc_TC_FCT_1601 Adc_TS_023 cfg 3VS_PC_DMA Observed behavior: Build fail log: [elxr] (error #412) unresolved symbols: 2 Adc_Ipw_Adc1EndConversionNotification from Adc_Ip_VS_1_PBcfg_c.o Adc_Ipw_Adc0DmaTransferCompleteNotification from Dma_Ip_VS_1_PBcfg_c.o Only ADC_UNIT_1_DMA_TRANSFER_USED and ADC_UNIT_0_END_CONVERSION_NOTIF_USED define in ADC_CfgDefines.h Expected behavior: Build done Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A

ID	Subtype	Headline and Description
ARTD-23128	Bug	[Rte] PRIMASK might not be read during usermode /s32k11x<*> Detailed description (how to reproduce it): On {*}s32k116{*}, i have observed that PRIMASK (when could not be read during {*}usermode{*}) !image-2022-01-12-17-43-58-256.png! Preconditions: s32k11x and usermode Test Case ID (internal TC that caught the defect) optional: CAN_TC_FCT_0563, CAN_TC_FCT_0564 Observed behavior: set PRIMASK = 1 (by invoking sys_disableAllInterrupts) => Can_schm_read_msr return 0 Expected behavior: set PRIMASK = 1 (by invoking sys_disableAllInterrupts) => Can_schm_read_msr return 1 Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-23155	Bug	[CAN] Fail to handle rx data frame, in case of sooner response / poll method ,,"Detailed description (how to reproduce it): A*> Please see attached file B*> FlexCAN_Ip_MainFunctionWrite should check state->mbs[mb_idx].state (similar as FlexCAN_Ip_MainFunctionRead) to ensure safety C*> FlexCAN_Ip_MainFunctionWrite does not invoke callback after tx complete (but FlexCAN_Ip_MainFunctionRead invoke callback after rx complete) => this seem to be inconsistent. For example: users installed callback and being using tx, rx interrupt with both ones notified. Then they change manner (at runtime) from tx,rx interrupt => tx,rx poll, now rx callback still remain but tx callback will not. => This inconsistency should be documented Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Ip_FlexCAN_TC_FCT_1027 Observed behavior: sooner response frame can not be handled in POLL mode. Expected behavior: apply similar implemetation for FlexCAN_Ip_MainFunctionWrite Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-23159	Bug	[ADC] Compiler warning for adc driver<*> Detailed description (how to reproduce it): Compiler warning for adc driver Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: All tests on station Observed behavior: Compiler warning report Expected behavior: No warning Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-23161	Bug	[SPI] Spi_SyncTransmit with length is 0 not Timeout erorr<*> Detailed description (how to reproduce it): Sync Transmit 1 job / 1 channel with 0 lengh EB. Lpspi_Ip_SyncTransmit: State->RxIndex = State->ExpectedFifoReads =0 not into while loop so not check timeout and return LPSPI_IP_IDLE !image-2022-01-14-16-12-26-000.png|thumbnail! Preconditions: Sync Transmit 1 job / 1 channel with 0 lengh EB: Spi_SetupEB(SpiConf_SpiChannel_007, Spi_TestDataTx[0], Spi_TestDataRx[0], 0) !image-2022-01-14-16-17-47-639.png|thumbnail! Test Case ID (internal TC that caught the defect) optional: Spi_TC_PER_1000 Observed behavior: Lpspi_Ip_SyncTransmit return LPSPI_IP_IDLE Spi_SyncTransmit return E_OK Expected behavior: Lpspi_Ip_SyncTransmit return LPSPI_IP_TIMEOUT Spi_SyncTransmit return E_NOT_OK Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA

ID	Subtype	Headline and Description
ARTD-23199	Bug	[BASE][S32CT] ECVD files containing INDEX elements don't pass the XSD validation<*> Detailed description (how to reproduce it): Generate the ECVD file of a component containing a setting that defines a "requiresIndex" option. Validate the generated ECVD file against the XSD. Preconditions: ECVD file containing an INDEX element Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: ECVD files containing INDEX elements don't pass the validation check of the schema definition (AUTOSAR_00046.xsd) Expected behavior: ECVD files containing INDEX elements pass the validation check of the schema definition (AUTOSAR_00046.xsd) Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: INDEX elements should not be generated in the middle of another group's sequence. In particular, for ECUC-CONTAINER-VALUE, the INDEX element shall be placed between SHORT-NAME and DEFINITION-REF elements.
ARTD-23248	Bug	[gpt] Update missing exclusive areas in BSWMD<*> Detailed description (how to reproduce it): Do a repo sync on bswmd_creator to sync with the manifest (make sure that you've checked out BLN_BSWMD_CREATOR_04.08.06) Do a repo sync on perl_common_utils to sync with the manifest (make sure that you've checked out BLN_PERL_COMMON_UTILS_01.01.03) Compile the plugin passing parameter GENERATE_BSWMD_FILE=ON to the make command (`p s1 b` in ZTH) If the plugin was compiled successfully and there are no errors from bswmd_creator regarding missing or unmapped exclusive areas, then you can close your clone with "Fixed No Action Taken". Otherwise, proceed with the solutions given in the "Proposed solution" section. After generating the plugin, there are some errors as image below: !screenshot-1.png|thumbnail! !screenshot-2.png|thumbnail! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Missing exclusive areas in <Module>_Bswmd.arxml Expected behavior: All exclusive areas are included in <Module>_Bswmd.arxml Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Potential Issue #1 Missing exclusive areas for ISRs Solution #1 The bswmd_creator script will automatically fetch ISRs from the source code as long as they're defined with ISR(...). Every ISR mentioned in the "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" files shall also be wrapped in ISR(...). For example: ADC_EXCLUSIVE_AREA_70 is used in function ISR(Ctu_Error_Isr) to protect the updates for CTU_IFR register Potential Issue #2 Missing exclusive areas for IPL functions which are not called by any HLD function Solution #2 # Have the EA mentioned in "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" only as used by the IPL function. For example: ADC_EXCLUSIVE_AREA_17 is used in function Adc_Sar_Ip_AbortConversion to protect the updates for ADC_MCR register # Add the IPL function in NonASR_ServiceID.xml Note: Do not use other workarounds like artificially mentioning in .dox that the IPL function is called by an HLD function which is included in NonASR_ServiceID.xml (even though, in reality, it's not)
ARTD-23252	Bug	[S32K14XW][EEP] Fix compiler warning in function Ftfc_Eep_Ip_Write<*> Detailed description (how to reproduce it): For S32K14XW, when building a test suite by gcc compiler, appeared 2 warning. Preconditions: Make generate completely Observed behavior: Appear warning when building test suite with S32K14XW Expected behavior: No warning when building test suite with S32K14XW Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-23263	Bug	[S32K1XX_1.0.1][ICU] Fix version checking AUTOSAR of inter module inclusion<*> Detailed description (how to reproduce it): Checking Icu_Ipw_Types.h Autosar major version is not checked for Cmp_Ip_Types.h Autosar minor version is not checked for Cmp_Ip_Types.h Autosar revision version is not checked for Cmp_Ip_Types.h Vendor ID is not checked for Cmp_Ip_Types.h

ID	Subtype	Headline and Description
		Checking Ftm_Icu_Ip_Irq.c Autosar major version is not checked for Mcal.h Autosar minor version is not checked for Mcal.h Ftm_Icu_Ip_Irq.c and Mcal.h need at least MINOR and MAJOR checking for Autosar version checking of inter module inclusion Checking Lpit_Icu_Ip_Irq.c Autosar major version is not checked for Mcal.h Autosar minor version is not checked for Mcal.h Lpit_Icu_Ip_Irq.c and Mcal.h need at least MINOR and MAJOR checking for Autosar version checking of inter module inclusion Checking Lptmr_Icu_Ip_Irq.c Autosar major version is not checked for Mcal.h Autosar minor version is not checked for Mcal.h Lptmr_Icu_Ip_Irq.c and Mcal.h missing DISABLE_MCAL_INTERMODULE_ASR_CHECK for autosar version checking of intermodule inclusion Checking Port_Ci_Icu_Ip_Irq.c Autosar major version is not checked for Mcal.h Autosar minor version is not checked for Mcal.h Port_Ci_Icu_Ip_Irq.c and Mcal.h need at least MINOR and MAJOR checking for Autosar version checking of inter module inclusion Preconditions: use version checking between files Test Case ID (internal TC that caught the defect) optional: Icu_TC_WIR_003(Icu_TS_001) Observed behavior: There are still some version checking errors between files as described Expected behavior: No more error checking version between files as described Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Verify all files & fix version checking between each file that mentioned at attachment file
ARTD-23261	Bug	[ADC] Import ECVD with errors from S32ConfigurationTool to Tresos configurator<*> Detailed description (how to reproduce it): [ADC] Import ECVD with errors from S32CT to EBT Steps: # Create new project wizard on S32DS and add component Adc # Generate ECVD for Adc component # Create new project wizard on EBT and add component Adc # Import ECVD and enable auto mapping # Run importer # VERIFICATION_POINT: Run importer without error, no error on EBT project => FAIL !image-2022-01-18-14-50-14-333.png! Workaround: replace AdcUserCfg to Adc in ecvd file !image-2022-01-18-14-54-12-843.png! Preconditions: Test Case ID (internal TC that caught the defect) optional: Adc_TS_COT_003 Observed behavior: Expected behavior: Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-23274	Bug	[S32K1XX][Crypto] Fix compiler warning<*> Detailed description (how to reproduce it): Some compiler warning founded by tool when run test with tag CRYPTO_106 : warning #1797-D: external identifier shall have exactly one definition@8 uint8 Crypto_au8NvramBlob1[CRYPTO_SIZEOF_NVRAM_BLOB_1] = CRYPTO_NVRAM_BLOB_1_INITIALIZER; unused parameter 'u32PartitionId' [-Wunused-parameter] const uint32 u32PartitionId, Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: See detail at attachment file Expected behavior: No compiler warning is founded from driver Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-23275	Bug	[UART] - Build fail on CT IP driver when adding a callback parameter<*> Detailed description (how to reproduce it): Import any IP uart driver CT component in S32DS. Configure a callback parameter. Generate the configuration files. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: There is a build fail on this configuration

ID	Subtype	Headline and Description
		Expected behavior: Analyze the Uart Callback Parameter usage. Define a way to define/declare the callback in parameter. Fix build error. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-23278	Bug	[s32k1xx 1.0.1][fee]ferror gen config is different between EB and DS<*> Detailed description (how to reproduce it): the config files is generated are different between EB and DS Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: the config files is generated are different between EB and DS Expected behavior: the config files is generated are samebetween EB and DS Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-23291	Bug	[WDG] Generate fail ECVD for Wdg_43_Instance1 on S32DS<*> Detailed description (how to reproduce it): Step 1: Custom tool [^validate_ip_ecpd.pl] Step 2: Update file [^plugin.xml] Step 3: Update [^Wdg_TS_COT_005.mak] file Step 4: clean generate Wdg_TS_COT_005 CFG_SETS = s32k148_lqfp176 Step 5: Compare all files in original_configuration folder with new_configuration folder Step 6: Import Wdg_43_Instance1.epc at path: " ...\output\S32K1XX_S32K148\wdg \Wdg_TS_COT_005_cfgs32k148_lqfp176\generate_s32ct\output " into S32DS Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: Wdg_43_Instance1.ecpd not matching between EB and CT !image-2022-01-19-16-12-00-689.png|width=1042,height=151! Unable to generate .c and .h files related to Wdg_43_instance1 for EB !image-2022-01-19-16-14-39-466.png|width=972,height=221! Error when importing epc file into S32DS !image-2022-01-19-16-13-37-326.png|width=644,height=370! Expected behavior: Generate ecvd successfully for Wdg_43_Instance1 Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-23290	Bug	[Port] Wrong macro usage in Port_Ci_Port_Ip_SetGlobalPinControl<*> Detailed description (how to reproduce it): Use tag PORT_134 for the plugins. Open Port_Ci_Port_Ip.c and look for the Port_Ci_Port_Ip_SetGlobalPinControl function. Test Case ID (internal TC that caught the defect) optional: IP_Port_Ci_TC_0001.c Observed behavior: The macro PORT_GPCLR_GPWE_SHIFT used in the switch case was duplicated from PORT_GLOBAL_CONTROL_LOWER_HALF_PINS to PORT_GLOBAL_CONTROL_UPPER_HALF_PINS as below: !image-2022-01-19-15-26-26-085.png! According to the Header files, It should be PORT_GPCHR_GPWE_SHIFT. Expected behavior: The macro should be correct corresponding each register definitions
ARTD-23302	Bug	[S32K1XX RTM 1.0.0] [ICU] There are some error with ecpd file in S32DS<*> Detailed description (how to reproduce it): When import ICU ecvd file into EB project, there are 4 fields in channel turned on: IcuSignalEdgeDetection, IcuSignalMeasurement, IcuTimestampMeasurement, IcuWakeup . With project in EB, user have to disable the field not corresponding with the mode of channel manually. So when user import ecvd file to EB, the error appear as the following : !image-2022-01-20-09-06-09-109.png! !image-2022-01-20-09-06-38-290.png! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: Icu_TS_COT_001 Observed behavior: Expected behavior: The components of other mode is disabled automatically.

ID	Subtype	Headline and Description
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-23312	Bug	[S32DS][S32K11X] Build fails with new project on S32DS<*> Detailed description (how to reproduce it): 1. Install S32DS3.4 with devpackage: SW32K1_S32DS_3.4.1_D2106.zip and update_site: S32K1XX_RTD_4_4_RTM_1_0_1_DS_updatesite_2201_signed.zip 2. Integrate GHS 202014 and IAR 8.40.3 into S32DS 2. Create new project following steps: Step1: In S32DS choose File > New > S32DS Application project Step2: Fulfil project name and Choose Family (S32K1xx) > S32K1XX(Test all derivatives: S32K116, S32K118, S32K142, S32K142W, S32K144, S32K144W, S32K146, S32K148) > Next Step3: Select required tool chain plugin from toolchain tab x number of tools chains Step4: Type a project name(e.g NewProject). Step5: Select SDK version Step6: Click Finish Step7: Build project with RAM and FLASH Step8: Click on Update code Step9: Build project with RAM and FLASH Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: Build fails with RAM on GCC 9.2 on S32K116 before and after updating code: Invoking: Standard S32DS C Linkerarm-none-eabi-gcc o "gcc.elf" "@gcc.args" d:/zebra/integration/s32k1xx_101/cycle1/s32k1xx_101_c1/s32ds/build_tools/gcc_v9.2/gcc-9.2-arm32-eabi/bin/../lib/gcc/arm-none-eabi/9.2.0/../../../../arm-none-eabi/bin/real-ld.exe: gcc.elf section `.sram' will not fit in region `int_sram'd:/zebra/integration/s32k1xx_101/cycle1/s32k1xx_101_c1/s32ds/build_tools/gcc_v9.2/gcc-9.2-arm32-eabi/bin/../lib/gcc/arm-none-eabi/9.2.0/../../../../arm-none-eabi/bin/real-ld.exe: region `int_sram' overflowed by 6232 bytescollect2.exe: error: ld returned 1 exit statusmake: [makefile:39: gcc.elf] Error 1"make j4 all" terminated with exit code 2. Build might be incomplete. Warnings when building with FLASH on GCC 9.2 on S32K116 and both of FLASH/RAM on S32K118 before updating code: arm-none-eabi-gcc "@RTD/src/Clock_Ip.args" MMD MP MF"RTD/src/Clock_Ip_Divider.d" MT"RTD/src/Clock_Ip_Divider.o" o "RTD/src/Clock_Ip_Divider.o" "../RTD/src/Clock_Ip_Divider.c" ../RTD/src/Clock_Ip.c:138:16: warning: 'Clock_Ip_bObjectsAreInitialized' defined but not used [-Wunused-variable] 138 static boolean Clock_Ip_bObjectsAreInitialized = FALSE; / Clock objects are initialized. / Build fails with flash on GHS compiler S32K118 after updating code: make all make: No rule to make target 'D:/Zebra/Integration/S32K1XX_101/cycle1/S32K1XX_101_C1/ws.it.c1/ghs118/generate/include/Clock_Ip_BOARD_InitPeripherals_PBcfg.h', needed by 'generate/src/Clock_Ip_Cfg.o'. Stop. "make all" terminated with exit code 2. Build might be incomplete. Warning when building with FLASH/ RAM before and after updating code on IAR for all derivative: IAR ELF Linker V8.40.3.268/W32 for ARM Copyright 2007-2020 IAR Systems AB. Warning[Ls015]: [stack usage analysis] at least one function appears to be uncalled. Example: "BusFault_Handler". A complete list of uncalled functions is in the map file. 576 bytes of readonly code memory 892 bytes of readonly data memory 1'048 bytes of readwrite data memory Expected behavior: Have no warnings, errors when building new project on all compilers Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-23311	Bug	[EEP] Fix warning duplicate when run CheckDef tool<*> Detailed description (how to reproduce it): 1. Setup and run CheckDefs tool. 2. Check RTD_EEP_CheckDefsReport.xlsx file 3. It appeared some WARNING Multiple Definition and Different Value at Ftfc_Eep_Ip.c file. Preconditions: Test Case ID (internal TC that caught the defect) optional: Observed behavior: Expected behavior: Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:
ARTD-23317	Bug	[Uart] Need insert error message when configure Baudrate divider is out of range<*> Detailed description (how to reproduce it): If user configure high reference clock and low baudrate, baudrate divider is more than 255 which is out of range. So driver needs to insert error message to avoid hard fault of hardware !image-2022-01-20-16-03-19-745.png|thumbnail! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A

ID	Subtype	Headline and Description
		Observed behavior: When user configure high reference clock and low baudrate, baudrate divider is more than 255 which is out of range driver didn't insert error message to avoid hard fault of hardware. Expected behavior: Error message appeared when user configured invalid Baudrate divider Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update valid condition expr attribute of Desire Baudrate node in S32CT
ARTD-23328	Bug	[PORT] Disable the configuration of CommonPublishedInformation container in S32DS<*> Detailed description (how to reproduce it): In S32CT, Port driver still allows user to config all the parameters in CommonPublishedInformation container. It will violate the Autosar expectation. Please refer General Specification of Basic Software Modules AUTOSAR CP Release 4.4.0 documentaion chapter 10.3Published Information Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Expected behavior: Port driver disables the configuration of that container in S32DS Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-23329	Bug	[DIO] Disable the configuration of CommonPublishedInformation container in S32DS<*> Detailed description (how to reproduce it): In S32CT, Port driver still allows user to config all the parameters in CommonPublishedInformation container. It will violate the Autosar expectation. Please refer General Specification of Basic Software Modules AUTOSAR CP Release 4.4.0 documentaion chapter 10.3Published Information Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Expected behavior: Port driver disables the configuration of that container in S32DS Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-23332	Bug	[CAN] redundant check pState->mbs[u32MbIdx].isPolling inside FlexCAN_IRQHandlerTxMB<*> Detailed description (how to reproduce it): [...] Preconditions: analysis ccov report (MCDC tab) !image-2022-01-21-14-05-50-639.png|thumbnail! Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: redundant check causes uncovered code in MCDC report !image-2022-01-21-14-12-20-287.png|thumbnail! Expected behavior: analysis redundant code Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-23334	Bug	[S32K1XX 1.0.1] Update example in S32DS<*> Detailed description (how to reproduce it): In S32DS interface, there are 4 fields in channel turned on automatically with old project: IcuSignalEdgeDetection, IcuSignalMeasurement, IcuTimestampMeasurement, IcuWakeup .This is due to the change relating to ecpd in ticket bug ARTD-23302 . So user have to remove the component of the field not corresponding with the mode of channel manually, the error appear as the following : !image-2022-01-21-14-23-57-087.png! Preconditions: [N/A] Test Case ID (internal TC that caught the defect) optional: [N/A] Observed behavior: Just one mode of channel is enabed. Expected behavior: Remove the fields not belong current channel

ID	Subtype	Headline and Description
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Remove the fields not belong current channel
ARTD-23360	Bug	[pwm] Fix all violations (HIS + MISRA + VSMD reports)<*> Detailed description (how to reproduce it): Create HIS MISRA VSMD reports Fix all violations for HIS MISRA VSMD reports *Reference: MISRA Deviation List: [https://bitbucket.sw.nxp.com/projects/ASPA/repos/real_time_drivers/browse/Generic/SW%20Development/RTD%20MISRA%20Deviations.xlsx] [https://confluence.sw.nxp.com/pages/viewpage.action?pageId=133734230] RTD Quality Criteria: [https://bitbucket.sw.nxp.com/projects/ASPA/repos/real_time_drivers/browse/Generic/Quality%20Assurance/RTD%20AASWS%20Gate%20and%20Release%20Criteria.xlsx] VSMD report: 0 warning, 0 error for RTM release MISRA report: 0 unjustified MISRA violations HIS report: !image-2021-08-30-12-26-52-249.png! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: NA All violations fix and follow RTD Quality Criteria Proposed solution optional: Fix all MISRA, HIS, VSMD violations
ARTD-23375	Bug	[LIN] The driver doesn't report error if PID is set incorrectly by user<*> Detailed description (how to reproduce it): A LIN frame was configured with an incorrect PID value, e.g. PID=0x10. However, the driver doesn't report any error in Lin_SendFrame() function, although the frame couldn't be sent at all. Preconditions: A LIN frame was configured with an incorrect PID value. Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: The LIN driver doesn't report any error in Lin_SendFrame() function Expected behavior: The LIN driver shall report error in Lin_SendFrame() function in case PID was incorrect set by user. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: CE's comment: When going down the Lin_SendFrame() path it was seen that the parity was set incorrectly, so Lin_Ipw_CheckFrameInfo() was returning E_NOT_OK. However, TempReturn does not change from initialization in Lin_Ipw_SendFrame(), so eventually Lin_SendFrame() will return E_OK. It was a typical issue of missing “else” condition in the implementation. A possible workaround would be adding the missing “else” condition in the “if” statement. Something like: !image-2022-01-25-14-07-05-171.png!
ARTD-23394	Bug	[S32K1][SPI] Fix violations HIS reports<*> Detailed description (how to reproduce it): Fix all violations for HIS reports Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: NA Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Fix all MISRA, HIS, VSMD violations
ARTD-23399	Bug	[CRC] Example on both CT and EB build with error<*> Detailed description (how to reproduce it): Import Example on S32DS and EB Generate and update code Build example project > some error occurred Observed behavior: Build with error on both EB and CT example Expected behavior:

ID	Subtype	Headline and Description
		All examples should build and run OK

ARTD-23404	Bug	[WDG] Some structs of driver is in invalid state when calling Wdg_SetMode FAST MODE or OFF MODE in Instance 1 EWM<*>
		
		Detailed description (how to reproduce it):
		Step 1: Init wdg with Instance 1(EWM)
		Step 2: Call Set Mode in EWM
		Std_ReturnType Wdg_Ipw_SetMode(Wdg_Ipw_InstanceType Instance, const Wdg_Ipw_ConfigType const IpwConfig)
		{
		Std_ReturnType Ret = (Std_ReturnType) E_OK;
		#if (WDOG_IP_USED == STD_ON)
		Wdog_Ip_StatusType RetIp = WDOG_IP_STATUS_SUCCESS;
		#endif /*(WDOG_IP_USED == STD_ON)*/
		Wdg_Ipw_aeIp[Instance] = IpwConfig->eWdgIp;
		switch (Wdg_Ipw_aeIp[Instance])
		{
		#if (WDOG_IP_USED == STD_ON)
		case WDG_IPW_WDOG_IP:
		RetIp = Wdog_Ip_Config((uint8)Instance, IpwConfig->pWdogConfig);
		if (RetIp != WDOG_IP_STATUS_SUCCESS)
		{
		if (WDOG_IP_STATUS_TIMEOUT == RetIp)
		{
		/ Raise a runtime det for Osif timeout expired*/
		(void)Det_ReportRuntimeError((uint16)WDG_IPW_MODULE_ID, (uint8)Instance, (uint8)WDG_IPW_SETMODE_ID, (uint8)WDG_IPW_E_PARAM_TIMEOUT);
		}
		Ret = (Std_ReturnType) E_NOT_OK;
		}
		break;
		#endif /*(WDOG_IP_USED == STD_ON)*/
		#if (EWM_IP_USED == STD_ON)
		case WDG_IPW_EWM_IP:
		(void)IpwConfig;
		Ret = (Std_ReturnType) E_NOT_OK;
		break;
		#endif /*(EWM_IP_USED == STD_ON)*/
		default:
		Ret = (Std_ReturnType) E_NOT_OK;
		break;
		}
		return Ret;
		}
		If mode = Slow mode: it will go to case case WDG_IPW_EWM_IP => Correct case
		If mode = Fast or Off mode: It will go to case default because the Wdg_ModeSettings[Mode] in EWM only have data in SlowMode, it is NULL_PTR in Fast and OFF mode => so Wdg_Ipw_aeIp[Instance] = IpwConfig->eWdgIp will return random value => the code will go to default case, and sometime lead to hardfault
		!screenshot-1.png|thumbnail!
		!screenshot-2.png|thumbnail!
		Preconditions:
		[...]
		Test Case ID (internal TC that caught the defect) optional:
		[...]
		Observed behavior:
		[...]
		Expected behavior:
		[...]
		Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)
		Proposed solution optional:
		[...]

ARTD-23414	Bug	[dio] [S32CT] Incorrect enable attributes for ConfigTimeSupport fields<*> Detailed description (how to reproduce it): Manually check the ConfigTimeSupport fields of the S32CT component / template Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: POST_BUILD_VARIANT_USED has enable="false" IMPLEMENTATION_CONFIG_VARIANT has enable="false" Expected behavior: POST_BUILD_VARIANT_USED has enable="false" if and only if postBuildVariantSupport="false" in the config set's options_expr. IMPLEMENTATION_CONFIG_VARIANT has enable="true" or no enable attribute (which is equivalent to enable="true"). Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: POST_BUILD_VARIANT_USED has enable="false" if and only if postBuildVariantSupport="false" in the config set's options_expr. IMPLEMENTATION_CONFIG_VARIANT has enable="true" or no enable attribute (which is equivalent to enable="true").
ARTD-23435	Bug	[qd] [S32CT] Incorrect enable attributes for ConfigTimeSupport fields<*> Detailed description (how to reproduce it):

ID	Subtype	Headline and Description
		Manually check the ConfigTimeSupport fields of the S32CT component / template Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: POST_BUILD_VARIANT_USED has enable="false" IMPLEMENTATION_CONFIG_VARIANT has enable="false" Expected behavior: POST_BUILD_VARIANT_USED has enable="false" if and only if postBuildVariantSupport="false" in the config set's options_expr. IMPLEMENTATION_CONFIG_VARIANT has enable="true" or no enable attribute (which is equivalent to enable="true"). Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: POST_BUILD_VARIANT_USED has enable="false" if and only if postBuildVariantSupport="false" in the config set's options_expr. IMPLEMENTATION_CONFIG_VARIANT has enable="true" or no enable attribute (which is equivalent to enable="true").
ARTD-23491	New Feature	[MCU] Investigate test case failure on S32K1XX<*> Investigate test case failure and request station run again
ARTD-23551	Bug	[I2c] Fix warning gennerate and build for example K1XX<*> Detailed description (how to reproduce it): [...] !image-2022-02-07-10-06-46-831.png! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-23554	Bug	[QDEC] Example generate fail<*> Detailed description (how to reproduce it): all ds example generate fail as in attach file: example HLD jump to hard_fault_handle after call Mcu_Init(NULL_PTR) function, please double check ConfigTimeSupport for Mcu Preconditions: NA Test Case ID (internal TC that caught the defect) optional: Ftm_Qdec_Ip_Example, Qdec_Example Observed behavior: generate fail Expected behavior: generate pass Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: {_}should be update ConfigTimeSupport:{_}{_}!image-2022-02-10-09-35-00-145.png!{_}
ARTD-23558	Bug	[WDG] Generate fail when using Bus CLK on S32K11X due to incorrect clock value range<*> Detailed description (how to reproduce it): Step 1: Prepare a test in with config of Instance 0: WdgClockValue = 48000 WdgClockSelection = Bus_Clock Step 2: Generate test and get error: Invalid value for node "/AUTOSAR/TOP-LEVEL-PACKAGES/Wdg/ELEMENTS/Wdg/WdgSettingsConfig/WdgSettingsFast/WdgClockValue": Value out of range: is "48000" but must be ">= 0" and "<= 40000" It seems that the clock value condition is incorrect because Bus clock frequency can be larger than 40Mhz !image-2022-02-07-13-43-52-121.png|thumbnail! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: Generate fail when using bus clock 's frequency larger than 40Mhz Expected behavior: Correct value range to make sure generate test normally Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-23581	Bug	[S32K1XX] [Crypto] Missing field CommonPublishedInformation in ECVD file of HLD<*>

ID	Subtype	Headline and Description
		Detailed description (how to reproduce it): When config HL layer on S32DS , ECVD file missing fied " CommonPublishedInformation" Test Case ID (internal TC that caught the defect) optional: Crypto_TS_COT_007 Observed behavior: When config HL layer on S32DS , ECVD file missing field " CommonPublishedInformation", see detail at attachments site Expected behavior: When config HL layer on S32DS, ECVD file have all field Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-23587	Bug	[ICU][S32K1XX RTM 1.0.1] Fix the configuration for example<*> Detailed description (how to reproduce it): In the example Ftm_Icu_Ip_BlinkLed_S32K118 in S32DS, the OSIF_COUNTER_DUMMY need to be selected as default to avoid the stop in function Clock_Ip_Init: !image-2022-02-08-17-34-25-984.png! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-23611	Bug	[PORT] Verify the MISRA's violations for S32K1<*> Detailed description (how to reproduce it): !image-2022-02-09-15-37-58-140.png! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: There are some MISRA, HIS, VSMD violations appear in the report. Expected behavior: All the violations need to be fixed/commented out Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-23615	Bug	[Port] CTHL Cannot generate ECPD file from S32DS<*> Detailed description (how to reproduce it): Use tag PVT_PORT_S32K1XX_1.0.1_V06 for the plugins Try to generate ecpd file from random configuration in S32DS layout by one of below ways: 1. Open codegenerator.js in eclipse\mcu_data\components\PlatformSDK_S32K1_2022_02\system and change some lines of code in ecpd session 2. Run cmd in eclipse to take the template configuration ecpd Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Both ways cannot generate ecpd files. With S32DS interface if correct the codegenerator.js, many errors occur as the picture below: !image-2022-02-09-16-38-57-607.png! !image-2022-02-09-16-39-25-065.png! With the using cmd, the error is shown as the following captured: !image-2022-02-09-16-40-13-377.png! Expected behavior: Ecpd file could be generated from the S32DS
ARTD-23621	New Feature	[Gpt]Update Profile test data<*> NewWorkDescription: There are some nodes that change properties of SRTC IP like Alarm Configuration so we need to update test data for it Update ecvd file because base update(generateEcvd.js file change) Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Create a new configuration file for ip srtc Update ecvd file
ARTD-23658	Bug	[UART][EXAMPLE] Update S32K118 pin from Q64 to Q48 as SOW<*>

ID	Subtype	Headline and Description
		Detailed description (how to reproduce it): S32K118 Examples are using Q64 pin not match with SOW(Q48 pin) Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: Update S32K118 Example to use Q48 pin Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-23689	Bug	[PORT] Fix compiler warnings for PORT S32K1XX RTM 1.0.1<*> Detailed description (how to reproduce it): There are some compiler warnings existed in the report. [http://ionian.ea.freescale.net/2/project/custom_compilerwarning/details] Preconditions: NA Test Case ID (internal TC that caught the defect) optional: IP_Port_Ci_TS_001 IP_Port_Ci_TS_002 Observed behavior: There are some compiler warning in the report. Expected behavior: There is no compiler warning in the report. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-23696	Bug	[UART][Example] Update example configuration files<*> Detailed description (how to reproduce it): Wrong resource configuration file for S32K142W EB example Port is not configured in S32K144W CT Example Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Wrong resource configuration file for S32K142W EB example Port is not configured in S32K144W CT Example Expected behavior: Resource configuration file for S32K142W EB example should be used as Q64 Config untourch port pin in Port module for S32K144W example Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-23712	Bug	[ICU][S32K1XX RTM 1.0.1] Fix the compiler warning<*> Detailed description (how to reproduce it): There is an warning in derivative S32K11X chip relating to TOF function not be used.See the compiler warning in the attached file. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: N/A Expected behavior: The function TOF is not used for S32K11X derivative should be guarded. Note: in the “Expected behavior” field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:

4.5 Change List for 1.0.0

ID	Subtype	Headline and Description
ARTD-1435	New Feature	New Feature

ID	Subtype	Headline and Description
		[ADC] S32ConfiguratorTool and Tresos remove .members from structs (for C90 compliance), keep generated files bit-exact with Tresos" ,,"Tresos and S32CT file generation remove .members from structs (for C90 compliance) Validate dev_tests after changes Validate that IPL example is still compiling after changes.
ARTD-6298	Bug	[SAI] Wrong transmit data after aborting/timeout transfer<*> Detailed description (how to reproduce it): Wrong transmit data after aborting/timeout transfer. Please refer the MAF log and the logic analyzer Sai_Abort_Sending.logicdata: the first transmission is abort at middle, then try to send again !image-2021-01-14-11-07-28-788.png! Initialize SAI driver as master mode, master sends data with I2S protocol Initialize MAF I2S interface: slave mode, receiver and 8-bit word size SAI driver starts an asynchronous transaction on the SAI bus Wait until the SAI transmission status is not SAI_STATUS_BUSY Abort on going transfer Verification Point: The SAI transmission status is SAI_STATUS_ABORTED MAF waits until the transmitting transaction finishes Verify: the MAF return status is E_MAF_OK MAF test the received frames, compare with the passed data array with data in the MAF RX buffer Verification Point: + the MAF return status is E_MAF_OK + The callback counter for SAI_IP_RUN_ERROR event equals 1 SAI driver starts an asynchronous transaction on the SAI bus Wait until the SAI transmission status is not SAI_STATUS_BUSY Verification Point: The SAI transmission status is SAI_STATUS_COMPLETED MAF test the received frames, compare with the passed data array with data in the MAF RX buffer Verification Point: + the MAF return status is E_MAF_OK => Failed at this verification point + The callback counter for SAI_IP_RUN_ERROR event equals 0 DeInitializes the SAI module Preconditions: [...] Observed behavior: Wrong transmit data after aborting/timeout event Expected behavior: Correct transmit data after aborting/timeout event Proposed solution optional:
ARTD-7085	New Feature	New Feature [ADC] Support External Dma for without Interrupts ,,"Consider to support AdcExtDMAChanEnable when AdcWithoutInterrupts is enabled. Proposed Solution: An alternative solution might be to use in this case the buffer registered by the user with Adc_SetupResultBuffer()"
ARTD-7734	New Feature	New Feature [ADC] Extend HLD examples with DMA usecase ,,"the Adc DMA is a common application for customer, suggest to add a Adc DMA case

ID	Subtype	Headline and Description
		Extend HLD EBT and S32CT examples with DMA usecase"
ARTD-7928	New Feature	New Feature [SAI] Optimize interrupt loops ,,"Optimize interrupt loops: Move switch outside while/for loops to avoid checking condition for each data written in FIFO"
ARTD-7938	New Feature	New Feature [ICU] Improve validation regarding input signals used on LPCMP ,,Analise and add validation based on available input channels used by CMP
ARTD-8230	Bug	[PWM] Fix Typos in FlexIO API requirements<*> {code:c} Requirement FLEXIO_PWM_IP_005_001: Service name: Flexio_Pwm_Ip_UpdatePeriodDuty Syntax: Flexio_Pwm_Ip_StatusType Flexio_Pwm_Ip_UpdatePeriodDuty(uint8 instance, uint8 channel, uint ... OK Requirement FLEXIO_PWM_IP_006_001: Service name: Flexio_Pwm_Ip_GetOutputState Syntax: boolean Flexio_Pwm_Ip_GetOutputState(uint8 instanceId, uint8 channel) Sync/Async: Sync Reent ... should use one naming parameter (instance or instanceId) for all of apis to synchronize between the driver code and the requirement The Flexio source code has all APIs defeined with instanceId. FlexIO requirements should be updated to reflect this.
ARTD-10161	New Feature	New Feature [GPT] Improve Predefine Timer functionality in Design Studio ,,"NewWorkDescription: PredefTimer functionality should automatically calculate the prescaler based on mcu clock reference and selected period value(1us or 100us) Requirement source: none (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: none"
ARTD-10242	Bug	[MCL] The prototype of Mcl_SelectCommonTimebase function is not exact with RTD form in CPR_RTD_00392.mcl<*> Detailed description (how to reproduce it): Check MCL Collected Requirements Check CPR_RTD_00392.mcl requirement > The prototype of Ftm function in the requirement is different with the prototype of driver code Observed behavior: The prototype of Ftm function in the requirement is different with the prototype of driver code Expected behavior:

ID	Subtype	Headline and Description
		Need update CPR_RTD_00392.mcl requirement with RTD form
ARTD-10265	New Feature	New Feature [ADC] SetChannel feature support on K1 ,,"NewWorkDescription: Implement/check SetChannel feature support on K1. After implementing/checking the feature, requirements for it must be marked as 'fulfilled in': CPR_RTD_00328.adc CPR_RTD_00329.adc Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: [...]"
ARTD-10270	New Feature	New Feature [ADC] WithoutInterrupts feature support on K1 ,,"NewWorkDescription: Implement/check WithoutInterrupts feature support on K1. After implementing/checking the feature, requirements for it must be marked as 'fulfilled in'. Requirement source: CPR_RTD_00048.adc Proposed solution optional: [...]"
ARTD-10349	New Feature	New Feature [SPI] Create SPI feature for 4-bit half duplex mode ,,"NewWorkDescription: Implement 4-bit half duplex mode according requirements # The Spi driver shall be able to transfer in half duplex mode supporting 2/4/8-bit parallel transmission or reception on each clock edge. # A vendor specific pre-compile boolean configuration parameter SpiHalfDuplexModeSupport shall enable/disable this functionality. # By default this optional functionality and configuration parameters shall be disabled. ITWG ticket: https://jira.sw.nxp.com/browse/AAI-632 Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: [...]"
ARTD-10386	New Feature	New Feature [ETH]Some functions have HIS_PARAM > 5 ,,"NewWorkDescription: Some functions have HIS_PARAM > 5: Gmac_Ip_MDIOReadMMD Gmac_Ip_MDIOWriteMMD GMAC_WriteManagementFrame Eth_Transmit Requirement source:

ID	Subtype	Headline and Description
		NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: NA"
ARTD-10473	New Feature	New Feature [WDG] Add TestMode feature ,,"For Wdog IPL, add a checkbox to enabled Wdog Test Mode; Add a pair of APIs at IP level Wdog_SetTestMode and Wdog_GetTestMode, guarded by a define generated according to the previous mentioned checkbox; The proposed API would be: IPL: Wdog_Ip_StatusType Wdog_Ip_SetTestMode(const uint8 Instance, Wdog_Ip_TestModeType TestMode) Wdog_Ip_TestModeType Wdog_Ip_GetTestMode(const uint8 Instance) For IPL, the APIs and the Wdog_Ip_TestModeType enum will be guarded by the define WDOG_IP_TEST_MODE_ENABLE, generated according to a Test Mode Enable checkbox; Update the requirements for WDOG IP according to the excel attached in AAI-858 to describe the new APIs."
ARTD-10732	New Feature	New Feature [WDG] Add Det_ReportRuntimeError when the Wdog can not be unlocked ,,"Add Det_ReportRuntimeError when the Wdog unlock sequence times out and the Wdog can not be unlocked. Update/add requirements Proposed solution: Move all error reporting enums from Wdg_Channel.h in a platform specific header file, which will be included by the IPW. In this way, the HLD remains unchanged and can still access error reporting enums, and error reporting code can also be added in the IPW. For S32K1, in case the IPL funcitons return the timeout status, it can be verified in the IPW and a Det_ReportRuntineError can be added."
ARTD-10739	New Feature	New Feature [ADC] Double buffering optimization - optimize DMA streaming ,,"NewWorkDescription: Add dev test to validate double buffering optimization. Investigate and fix any eventual issue CPR_RTD_00384: The ADC driver shall provide an optional configuration parameter for reducing the number of interrupts required for processing the conversions of Adc Groups that consist of one or more channels (depending on HW capabilities) and which are configured as ADC_ACCESS_MODE_STREAMING. This parameter shall be available only when DMA transfer is used. When this feature is enabled, only one interrupt will be raised after the completion of all stream conversions (as configured by AdcStreamingNumSamples parameter). An additional interrupt to be raised after half of the stream is converted shall also be configurable. Requirement source: CPR_RTD_00384 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional:

ID	Subtype	Headline and Description
		[...]"
ARTD-10740	New Feature	New Feature [ADC] DMA Support for Streaming Group without interrupts ,,"NewWorkDescription: Implement DMA support for Streaming Group and without interrupts enabled. CPR_RTD_00048.adc, CPR_RTD_00488.adc Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: [...]"
ARTD-10769	Bug	[MCL] FTM need to support on S32K1XX RTM_100<*> Detailed description (how to reproduce it): Check MCL Collected Requirements with newest base line Observed : Requirement of FTM still exist on S32K1XX Observed behavior: Requirement of FTM still exist on S32K1XX, it is not supported for this release Expected behavior: It must be removed from S32K1XX requirement
ARTD-10775	New Feature	New Feature [ADC] Add support for AdcCalibrationPrescale feature ,,"NewWorkDescription: Add support for AdcCalibrationPrescale and/or update automatically prescale value before calibration Check available implementation in legacy SDK See attached email for details Update requirements Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: N.A."
ARTD-10831	Bug	[S32K1XX] Fix file version checking<*> Detailed description (how to reproduce it): Fix all violation version checking Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: [...] Expected behavior: No file version checking errors. Please ensure all violations fix for your driver, reference link: [http://ceram.ea.freescale.net/0/project/custom_file_verchecking/details] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:

ID	Subtype	Headline and Description
		[...] - NewWorkDescription: Fix all file version checking for RTD drivers Requirement source: Attachment baseline report on 2021.05.24 Proposed solution optional: [...]
ARTD-10837	Bug	[S32K1XX] Fix VSMD error<*> Detailed description (how to reproduce it): Create VSMD report for driver Fix any VSMD error if any Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-10842	Bug	[S32K1XX] Fix and comment static analysis violations (MISRA + HIS + CERT-C)<*> Detailed description (how to reproduce it): Get MISRA, HIS & CERT-C reports from Bamboo CI build Fix violations Guideline following: [https://confluence.sw.nxp.com/pages/viewpage.action?pageId=133734230] Preconditions: NA Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Fix with best efforts for EAR release
ARTD-10891	New Feature	New Feature [S32K1XX] Support enable/disable SIM_MISCTRL0[FTMx_OBE_CTRL] bit in SIM module ,,"NewWorkDescription: OCU. FTM module has a feature that need enable the SIM_MISCTRL0[FTMx_OBE_CTRL] bit in SIM module: !image-2021-05-24-14-45-27-968.png|width=561,height=110! Requirement source: Add feature enable/disable FTMx_OBE_CTRL bit.

ID	Subtype	Headline and Description
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: N/A"
ARTD-10896	Bug	FTM Overflow Calculation is wrong when overflow notification is enabled in RTD ICU driver<*> Detailed description (how to reproduce it): When FTM is working as signal measurement mode, and the overflow notification is enabled, the period and duty calculation are incorrect. Preconditions: 1) FTM0 CH2, PA14, generated a 500Hz PWM signal 2) FTM1 CH0, PB02, working as input capture 3) Using a wire to connect J13-08 of J49 and J3-04 of J56 on VNP-S32G-EVB board. 4) Run \Icu_example_S32G274A_M7\debug\run.cmm 5) Print via UART0 to see the captured result Test Case ID (internal TC that caught the defect) optional: Test code see attachment Observed behavior: !image-2021-05-24-16-40-12-485.png! Correct result is, duty cycle = 20000, period = 40000 Incorrect result is, duty cycle = 40000, period = 60000 Expected behavior: Correct result is, duty cycle = 20000, period = 40000 Proposed solution optional: [...]
ARTD-10962	Bug	[ADC] Adc_ReadRawData read data from control channel index (SC index) instead of physical channel id<*> Detailed description (how to reproduce it): Adc_ReadRawData read data from control channel index (SC index) instead of physical channel id Preconditions: Adc_ReadRawData is enable Adc 0 group with channel SE6_ADCH6, SE14_ADCH14 Test Case ID (internal TC that caught the defect) optional: Adc_TC_1801 Adc_TS_021 Observed behavior: Start group of adc 0 Wait until complete flag of SC1 channel set Adc_ReadRawData with channel array {6, 14} Real status: wrong data in buffer read group {0 , 0} Complete flag check failed because input parameter is physical channel Expected behavior: Read group buffer matching with {0xE35, 0xE7C} Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-10965	Bug	[ADC] Missing exclusive area from Pdb_Adc_Ip_SwTrigger<*> Detailed description (how to reproduce it): Exclusive area 38 that should guard SC register access inside Pdb_Adc_Ip_SwTrigger function of Pdb_Adc IPL module is currently missing.

ID	Subtype	Headline and Description
		Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: The exclusive area enter and exit functions are not added. Adding them causes TS_000 to hang because the user notification function is no longer being called. Expected behavior: Exclusive area present and tests passing. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-10967	New Feature	New Feature [SAI] SAI IP generate defines with instance number ,,"NewWorkDescription: SAI IP generate defines with instance number Update IPL example to replace defines from main.c: #define INST_SAI0 0U #define INST_SAI1 1U Requirement source: Usability improvement Proposed solution optional: name proposal (caps of configuration name)_(functional group name)_INSTANCE !image-2021-05-25-15-50-56-879.png|thumbnail! [...]"
ARTD-10970	Bug	[FLS] Initialization sequence needs to comply with HW timing restrictions.<*> Per attached discussions with design, the AHB domain / Serial Flash domain reset must meet certain timing restrictions. Also after this reset the DLL chain must be reconfigured.
ARTD-11025	Bug	[MCU] Wrong frquence of all peripheral module when disable node "McuPeripheralClockEnable" in EB Tresos ,,"Detailed description (how to reproduce it): Wrong frequency of clock Peripheral LPUART0 when disable node ""McuPeripheralClockEnable"" in EB Tresos. The same issue with other Peripheral module Preconditions: Step1 : cfg clock Peripheral LPUART0 source SPLL Step2 : Disable node McuPeripheralClockEnable Step3 : Build code Step4 : Run test Test Case ID (internal TC that caught the defect) optional: Mcu_TC_FCT_0554 Observed behavior: frequency of LPUART0 equal 48Mhz Expected behavior: frequency of LPUART0 must be equal to 0 Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)

ID	Subtype	Headline and Description
		Proposed solution optional: [...]
ARTD-11057	Bug	[Wdg] There is some information that does not match EB and S32CT when comparing code generation<*> Detailed description (how to reproduce it): Step 1: clean generate Wdg_TS_COT_001 for EB and Wdg_TS_COT_101 for CT Step 2: Compare include folder and src folder of S32CT with EB Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: There is some information that does not match EB and S32CT when comparing code generation. Detail in *share point link*: [https://nxp1.sharepoint.com/:x:/r/sites/Zebra/_layouts/15/Doc.aspx?sourcedoc=%7B4580B8F9-4A6B-4CE5-B74B-56E052438238%7D&file=Compare%20Config%20S32CT%20and%20EB%20Tresos_S32K146.xlsx&action=default&mobileredirect=true] Expected behavior: EB and S32CT need to generate the same file.
ARTD-11073	Bug	[SAI] Driver does not support disabling clock after transmission (BCE)<*> Detailed description (how to reproduce it): the bit clock continues to be generated even after transmission has been complete this has impact on power consumption. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Proposed solution optional: Possible solution would be to clear BCE from last interrupt (need to decide how to implement when using DMA) Add support for BCE disable clock after transmission To decide if parameter is per instance or transmission
ARTD-11168	Bug	[FLS]Header files found in src folder<*> For the components mentioned in the issue title, there are some header files misplaced under src folder and not under include folder. In Model-Based Design Toolbox for MATLAB and Simulink, we generate the projects automatically and need to copy all the dependencies. We explect to find only header files in the include and src files in src folder. This pattern is satisfied for all other modules but not for the mentioned ones. Also, the RTD veresion is 0.9.0 released on 26th of March, but in Jira I was not able to find this version marked as "Released". !image-2021-04-19-11-50-22-549.png!!image-2021-04-19-11-53-11-912.png! !image-2021-04-19-11-57-04-010.png! !image-2021-04-19-11-58-12-914.png!
ARTD-11182	New Feature	

ID	Subtype	Headline and Description
		New Feature [SAI] S32ConfiguratorTool and Tresos remove .members from structs (for C90 compliance), keep generated files bit-exact with Tresos" ,,"Tresos and S32CT file generation remove .members from structs (for C90 compliance) Validate dev_tests after changes Validate that IPL example is still compiling after changes.
ARTD-11204	New Feature	New Feature [I2C] Lpi2c should receive more than 256 bytes per transfer ,,Lpi2c receive function is limited to receive 256 bytes per transfer. The number of bytes that could be received in one transfer should be increased.
ARTD-11213	New Feature	New Feature [LIN] - Create new examples for Slave Nodes ,,"NewWorkDescription: LIN driver provides examples for Master Mode communication. Requirement source: posed solution optional: In order to ilustrate the slave node functionality, we need to create examples for that. "
ARTD-11220	New Feature	New Feature [RTE] Reduce size of exclusive area to save ram on S32K1XX ,,"NewWorkDescription: On s32K1xx there is only one core, so the size the exclusive area array should be reduce to 1 to save memory. When there are some modules to be used at the same time, exclusive area arrays occupy much memory that probably cause issue ""memory doesn't fit .."" !image-2021-06-03-18-30-09-410.png! Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: [...]"
ARTD-11254	Bug	[Base]hardfault when using sys_registerIsrHandler / usermode / K116/k118<*> Detailed description (how to reproduce it): !image-2021-05-26-15-08-28-570.png! Preconditions: Call OsIf_ResumeAllInterrupts function in the user mode support Test Case ID (internal TC that caught the defect) optional: test case use interrupt functional Observed behavior: harfault when using sys_registerIsrHandler / usermode / K116/k118 Expected behavior: Not hard fault Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:

ID	Subtype	Headline and Description
		need to resume all interrupts on S32K11X and not get hardfault when running project with user mode support
ARTD-11305	Bug	[MCU] The value of SCG_CLKOUT, SPLL_CLK, SPLLDIV1_CLK, SPLLDIV2_CLK, RTC_CLK, LPO_CLK are always equal 0 when using Mcu_GetClockFrequency(); Wrong clock freq for disabled periphs; wrong LPTMR clock frequency ,,"Detailed description (how to reproduce it): The value of SCG_CLKOUT, SPLL_CLK, SPLLDIV1_CLK, SPLLDIV2_CLK, RTC_CLK, LPO_CLK always equal 0 when using Mcu_GetClockFrequency() function; Wrong clock freq for disabled periphs; wrong LPTMR clock frequency The same issue when configure CLKOUT's input source to *SCG_CLKOUT_CLK, HCLK, LPO_CLK, LPO_128K_CLK, RTC_CLK, SPLLDIV2_CLK Clock is not calculated correctly when peripheral is disabled. LPTMR clock frequency is not calculated correctly. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Mcu_TC_FCT_0505 Mcu_TC_FCT_0507 Mcu_TC_FCT_0590 Mcu_TC_FCT_0553 Mcu_TC_FCT_0554 Mcu_TC_FCT_0549 Mcu_TC_FCT_0552 Observed behavior: The returned frequency is 0 for SCG_CLKOUT, SPLL_CLK, SPLLDIV1_CLK, SPLLDIV2_CLK, RTC_CLK, LPO_CLK The returned frequency is not 0 for disabled peripherals (ADC0_CLK ...) LPTMR_CLK frequency is not divided and multiplied as configured. Expected behavior: The returned frequency corresponding with input clock sources Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-11320	Bug	[ICU] Some IPL functions name are different from requirements and driver<*> Detailed description (how to reproduce it): pls see attached file Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-11324	Bug	[S32K1XX][S32K1xx] Compare difference between EB and CT<*>

ID	Subtype	Headline and Description
		Detailed description (how to reproduce it): 1. Creating new project for S32K1xx on EB, generate any config on EB, After that import config to S32CT. 2, On S32CT After that import config (epc file) from EB, Update code. 3, Compare config generate from EB and CT. Preconditions: PVT_S32K1XX_ARTD_10722_007 Test Case ID (internal TC that caught the defect) optional: Case compare between EB and CT Observed behavior: 1. Checking Platform_Ipw_Cfg.c file. Platform_Ipw_Cfg.c file gene from CT have line of code: #if (INT_CTRL_IP_CORETEXM4 == STD_ON) #endif EB have not !image-2021-06-07-16-16-52-638.png! 2. Checking System_Ip_CfgDefines.h !image-2021-06-07-16-20-19-341.png! 3. Checking Platform_CfgDefines.h file. On S32CT: missing check version and define S32K1XX_ROUTING_CONTROL_REGISTER !image-2021-06-07-16-24-13-190.png! 4. Checking IntCtrl_Ip_Cfg.c file. On S32CT IntCtrl_Ip_IrqConfigType aIrqConfig[] and IntCtrl_Ip_IrqRouteConfigType aIrqRouteConfig[] array difference IRQ indexnumber. !image-2021-06-07-16-29-10-334.png! 5. Checking IntCtrl_Ip_Cfg.h On S32CT missing include for each divertive. !image-2021-06-07-16-32-25-641.png! 6. Disable Generic Interuppt settings config checking Platform_Ipw_Cfg.c file. !image-2021-06-07-16-44-36-348.png! Expected behavior: Same config between EB and CT Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-11328	Bug	[MCU] Generate failure after disabled QSPI clock gate<*> Detailed description (how to reproduce it): Create .mex for S32K148, add Mcu Disabled QSPI gate in clock tool and synchronize it in Mcu Update code (Update code sussessful, code review ok) Use this .mex to generate by cywin comand line Test Case ID (internal TC that caught the defect) optional: Mcu_TC_FCT_0557 Observed behavior: Generate failure, can not generate mcu in generate folder Import this .mex to S32K148 project available, S32CT report error QSPI gate not synchronize betwen clock tool and Mcu componet Expected behavior: Generate successful [...]
ARTD-11330	Bug	

ID	Subtype	Headline and Description
		[lin] Errors and missing exclusive areas in Bswmd.arxml<*> Detailed description (how to reproduce it): Exclusive areas not available in <Module>_Bswmd.arxml Incorrectly formatted <Module>_Bswmd.arxml Preconditions: N.A. Test Case ID (internal TC that caught the defect) optional: N.A. Observed behavior: Missing exclusive areas in Davinci after integrating RTD !image-2021-04-12-19-29-09-321.png|thumbnail! Expected behavior: All exclusive areas are included in <Module>_Bswmd.arxml <Module>_Bswmd.arxml is valid Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution Fix the following issues and perform all the verification steps*: Issue #1 Some exclusive areas are missing because "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" files are not using the pattern expected by bswmd_creator script: "is used in function" Solution #1 From https://jira.sw.nxp.com/browse/AMPT-5710, bswmd_creator is parsing files "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" and it's looking up for the "is used in function" pattern. Therefore, "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" files shall use the following pattern for describing each exclusive area: <MODULE>_EXCLUSIVE_AREA_09 is used in function <Module>_StartChannel ... Issue #2 Missing exclusive areas mapped to IPL functions 2.1 Exclusive areas not referenced by any function in Adc_Bswmd.arxml, missing: <CAN-ENTER-EXCLUSIVE-AREA-REF DEST="EXCLUSIVE-AREA"> E.g.:{code:java} ADC_EXCLUSIVE_AREA_12 is used in function Adc_Sar_Ip_DoCalibration 2.2 Exclusive areas missing completely from Adc_Bswmd.arxml E.g.:{code:java} ADC_EXCLUSIVE_AREA_73 is used in function Ctu_Ip_SetFifoThreshold Solution #2 The integration manual shall map exclusive areas to public HLD functions (*no internal functions should be mentioned*). This is done by adding in "exclusive_areas_to_be_defined_in_bsw_scheduler.dox" all the public HLD functions that are using an exclusive area via calls to IPL functions. For example: ADC_EXCLUSIVE_AREA_12 is used in function Adc_Calibrate to protect the ADC_MCR register from read/modify/write operation in Adc_Sar_Ip_Powerup. ADC_EXCLUSIVE_AREA_13 is used in function Adc_Init to protect the ADC_MCR register from read/modify/write operation in Adc_Sar_Ip_Powerup. ADC_EXCLUSIVE_AREA_13 is used in function Adc_SelfTest to protect the ADC_MCR register from read/modify/write operation in Adc_Sar_Ip_Powerup. [...] <b>Exclusive Areas implemented in Low level driver layer (IPL)</b> ADC_EXCLUSIVE_AREA_12 is used in function Adc_Sar_Ip_DoCalibration to protect the updates for: ADC_MCR register ADC_EXCLUSIVE_AREA_13 is used in function Adc_Sar_Ip_Powerup to protect the updates for: ADC_MCR register

ID	Subtype	Headline and Description
		This adds in "Adc_Bswmd.arxml" the ADC Exclusive Area 12 for Adc_Calibrate and ADC Exclusive Area 13 for Adc_Init and Adc_SelfTest. It also keeps the note in Integration Manual about the EAs being used by IPL API functions Adc_Sar_Ip_Powerup and Adc_Sar_Ip_DoCalibration. Issue #3 Using uppercase hexadecimals for non-autosar API service IDs. Solution #3 Non-autosar API service IDs shall be converted to lowercase in "*generic/doc/NonASR_ServiceID.xml*" to be compliant with AUTOSAR schema file (otherwise, the file is not valid and cannot be loaded). For example: <SERVICE-ID>0x2C</SERVICE-ID> must be changed to <SERVICE-ID>0x2c</SERVICE-ID> Verification #1 After fixing, please generate plugin with GENERATE_BSWMD_FILE=ON (more details in https://nxp1.sharepoint.com/:p:/s/Zebra/Ecuug60PLRlFv7JI1xLByAABrR3E2qa0txuqVbp4II2-IQ?e=GeSbhP) there must not be any errors generated by the script. Verification #2 Make sure exclusive areas are available in generate_swcd\swcd\<Driver>_Bswmd.arxml: <EXCLUSIVE-AREA> <SHORT-NAME>ADC_EXCLUSIVE_AREA_13</SHORT-NAME> </EXCLUSIVE-AREA> and referred by functions <BSW-CALLED-ENTITY> <SHORT-NAME>Adc_Init</SHORT-NAME> <CAN-ENTER-EXCLUSIVE-AREA-REFS> <CAN-ENTER-EXCLUSIVE-AREA-REF DEST="EXCLUSIVE-AREA">/AUTOSAR_Adc/BswModuleDescriptions/Adc/InternalBehavior_0/ADC_EXCLUSIVE_AREA_13</CAN-ENTER-EXCLUSIVE-AREA-REF> </CAN-ENTER-EXCLUSIVE-AREA-REFS> <MINIMUM-START-INTERVAL>0.0</MINIMUM-START-INTERVAL> <IMPLEMENTED-ENTRY-REF DEST="BSW-MODULE-ENTRY">/AUTOSAR_Adc/BswModuleEntrys/Adc_Init</IMPLEMENTED-ENTRY-REF> </BSW-CALLED-ENTITY> Verification #3 Validate the <Driver>_Bswmd.arxml file following steps from: https://confluence.sw.nxp.com/display/AUTORD/BSWMD Verification #4 Regenerate IM if task for updating documentation has been already resolved.
ARTD-11361	Bug	[ICU] The first timestamp value in the buffer always changes when there is an input pulse in the LPIT module<*> !image-2021-06-30-10-40-15-082.png! As the result in debug, the index current value of timestamp buffer always equal the first index value. Because in the function Lpit_Icu_Ip_IrqHandler, the value of first index in timestamp buffer is always get the value of CVAL, then this value is passed to the function Lpit_Icu_Ip_TimestampHandler to assign the value of current index in buffer. As you can see in the picture below: !image-2021-06-30-10-48-40-937.png! So the problem is the timestamp buffer are using both to get the value of CVAL and get the value of timestamp, this is making the value of each elements in bufffer always is same the first one. We should use the other array to get the value of CVAL as the following: !image-2021-06-30-10-55-18-854.png!

ID	Subtype	Headline and Description
ARTD-11366	Bug	[ICU] Lpit_Icu_Ip_GetInputState function is wrong<*> Detailed description (how to reproduce it): when read TIE bit for channel, need read in MIER register. pls see attached file Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: Ip_Lpit_TC_FCT_7000 Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-11374	Bug	[SPI] Jump to hardfault in Spi_JobTransferFinished is called twice<*> Detailed description (how to reproduce it): If Job result is failed but the Spi_Cancel is called in case don't know the job result. An hard fault will available because AsyncCrtSequenceState is assigned to null (The Spi_JobTransferFinished is called twice in both interrupt and Spi_Cancel) Preconditions: First Sequence is pending, after that call Spi_Cancel. But in this time, the tranferring job is failed. Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-11377	New Feature	New Feature [Wdg] Exclusive areas analysis for Wdog and EWM IP ,,"NewWorkDescription: [...] Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: [...]"
ARTD-11382	Bug	[PWM] Remove ASR specific validation from NonAsr component<*> Detailed description (how to reproduce it): build the plugins and s32ct component for PWM Create a new example for K3XX and add Emios_pwm_ip compoenent to the project

ID	Subtype	Headline and Description
		A lot of errors are thrown in the error log because validation of some nodes require the HLD nodes to be present. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: Errors are thrown in the error log of S32DS from invalid validation constraints. !image-2021-06-08-16-39-47-179.png|thumbnail! After a look in the template file it seems that this impacts all IPs of PWM driver. Expected behavior: No errors are present for any of the IPL components. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Guard in the template file the validations that are using nodes from ASR mode. This should be done for all available IPs.
ARTD-11405	Bug	[MCL] The function Mcl_CacheEnable can not set enable cache bit at some time<*> Detailed description (how to reproduce it): When run test i saw that: 1, At the some time when run over Mcl_CacheEnable but the bit enable cache is not set, this dosen't happen when debug step by step. I think the root cause from Timeout.So this funtion and Mcl_CacheDisable function need to support the delay time out. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Mcl_TC_FCT_0804 Observed behavior: At the some time when run over Mcl_CacheEnable but the bit enable cache is not set, Expected behavior: the enable cache bit always set when call Mcl_CacheEnable Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Mcl_CacheDisable and Mcl_CacheEnable function need to support the delay time out.
ARTD-11413	Bug	[ICU] Port_Ci_Icu_Ip_GetInputState function gets FALSE when interrupt is not enabled and signal is present<*> Detailed description (how to reproduce it): Port_Ci_Icu_Ip_GetInputState gets FALSE when interrupt is not enabled and signal is present. In this case the value should be TRUE. This issue apply for ip layer. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: Ip_Port_TC_FCT_6000 Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:

ID	Subtype	Headline and Description
		[...]
ARTD-11406	Bug	[FEE] Fix compiler warnings<*> Detailed description (how to reproduce it): See the attached file for more details Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: iar warnings Expected behavior: no warnings Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-11442	Bug	[GPT] The returned value of the TSR register is incorrectly after each interrupt<*> Detailed description (how to reproduce it): After each interrupt, the value of the TAR register returns incorrectly because after each interrupt, we have added a value for it after the statement: / TIF cleared by writing the TSR register when the time counter is disabled.*/ Srtc_Ip_SetTimeSecondsRegister(base, 1U); check this same with functions Srtc_Ip_Init,Srtc_Ip_StopCounter Preconditions: using SRTC channel Test Case ID (internal TC that caught the defect) optional: Ip_SRtc_TC_FCT_0004 Observed behavior: The value of the TSR register returned incorrectly after each interrupt, and also when stopping the counter,initialize counter Expected behavior: TAR register value returns exactly after each interrupt, and also when stopping the counter,initialize counter Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: checks and returns the correct value of the TAR register after interrupts and also when stopping the counter,initialize counter
ARTD-11443	Bug	[LIN] Different generated code between EBT and S32DS<*> Detailed description (how to reproduce it): There are some difference in generated code by S32DS and EBT. Please check the attached files. Test suite to compare: Lin_TS_D01 (*cfg_1*) Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: Lin_TS_D01 (cfg_1) Observed behavior: [...] Expected behavior:

ID	Subtype	Headline and Description
		[...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-11472	Bug	[ETH] Function Eth_GetCounterValues calculate wrongly<*> Detailed description (how to reproduce it): According to IETF RFC 1757 document, the value of UndersizePkt, OversizePkt should be counted on packets received only meanwhile our driver counts on both transmitted and received packets. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-11481	Bug	[S32K1 EAR][CRYPTO]Csec_Ip_GenerateMacAddrMode and Csec_Ip_VerifyMacAddrMode should be to Ram code<*> Detailed description (how to reproduce it): Csec_Ip_GenerateMacAddrMode and Csec_Ip_VerifyMacAddrMode when run on S32K118 and S32K116 with gcc compiler ,function return error Preconditions: S32K118 and S32K116 with gcc compiler Test Case ID (internal TC that caught the defect) optional: Ip_Csec_TC_FCT_0104 Observed behavior: function return error Expected behavior: Move Csec_Ip_GenerateMacAddrMode and Csec_Ip_VerifyMacAddrMode to ram code Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-11528	New Feature	New Feature [FEE] If these conditions are not satisfied. The Fee_Init() return FEE_E_INIT_FAILED ,,"NewWorkDescription: CPR_RTD_00563.fee:If the parameter checking for the driver's initialization function is enabled, the Configuration pointer argument shall be checked with the following conditions: In the supportedConfigVariants VariantPreCompile and VariantLinkTime if only one configuration variant set is used, the initialization function does not need nor evaluate the passed argument. Thus the Configuration pointer shall have a NULL_PTR value.

ID	Subtype	Headline and Description
		In the supportedConfigVariant VariantPostBuild or if multiple configuration variant sets are used, the initialization function requires the passed argument. Thus the Configuration pointer shall be different from NULL_PTR. If these conditions are not satisfied, a Development error with type <MIP>_E_INIT_FAILED shall be reported to Development Error Tracer (Det) Requirement source: cPRD (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: NA"
ARTD-11517	Bug	[ICU] The macro Overflow Interrput is not generated in S32K118 and S32K116 derivatives<*> Detailed description (how to reproduce it): The macro Overflow Interrput is not generated in S32K118 and S32K116 derivatives Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: Icu_TC_WBT_8088 Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-11519	Bug	[BASE] Compiler warning on IAR when using option --do_explicit_zero_opt_in_named_sections<*> Detailed description (how to reproduce it): There is an compiler warning on IAR compiler as below: !image-2021-06-11-09-32-50-290.png|thumbnail! Those variables are declared in Fee.c: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / zero initialized / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" #define FEE_START_SEC_VAR_INIT_UNSPECIFIED #include "Fee_MemMap.h" static Fee_JobType Fee_eJob = FEE_JOB_DONE; / initialized other than zero / #define FEE_STOP_SEC_VAR_INIT_UNSPECIFIED #include "Fee_MemMap.h" 2 variables were pushed into the same section .mcal_data as specified by FEE_START_SEC_VAR_INIT_BOOLEAN and FEE_START_SEC_VAR_INIT_UNSPECIFIED. The compiler option "--do_explicit_zero_opt_in_named_sections" is already added into build_cfg.mak. So, it leads to the compiler consider bSwapToBePerformed variable is not copy initialized and raised the warning message as above. To compiler considers bSwapToBePerformed variable as copy initialized. The compiler option "--do_explicit_zero_opt_in_named_sections" should be removed. Preconditions: Build test Fee with IAR compiler

ID	Subtype	Headline and Description
		Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Compiler warning message raised due to both variables bSwapToBePerformed and Fee_eJob psuhed into the same section .mcal_data when option "--do_explicit_zero_opt_in_named_sections" used. Expected behavior: NA Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: The compiler option "--do_explicit_zero_opt_in_named_sections" should be removed.
ARTD-11523	Bug	[ETH] Buffers should be unlocked after enable controller<*> Detailed description (how to reproduce it): Buffers should get unlocked after cotroller enable again by function Eth_SetControllerMode with parameter ETH_MODE_ACTIVE. Sequent : Eth_Init > Eth_SetControllerMode(ETH_MODE_ACTIVE) > Eth_SetControllerMode(ETH_MODE_DOWN) > Eth_SetControllerMode(ETH_MODE_ACTIVE) Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: Eth_TC_FCT_0131.c Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-11530	Bug	[S32K1 EAR] SPI: Wrong received data in RX Buffer when using slave mode with FLEXIO at both interrupt and DMA mode<*> Detailed description (how to reproduce it): Wrong received data in RX Buffer when using slave mode with FLEXIO at both interrupt and DMA mode: The data transmit on the bus is fine, but received data is unexpected. Expected data is \{1, 2, 3, 4} !image-2021-06-11-14-11-04-841.png! More details: Cause of error because error flag is not clear before starting a transmission Did not disable error interrupts after having an error occurred or a transfer completed Rx shifter flag is set from previous transfer Preconditions: NA Test Case ID (internal TC that caught the defect) optional: TS_105 Observed behavior: [...] Expected behavior: [...]

ID	Subtype	Headline and Description
		Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Clear error flags after a transmission is started Disable error interrupts when a transfer finished
ARTD-11531	Bug	[LIN] Function Lin_GoToSleepInternal can't set RIE bit with IPV: LPUART<*> Detailed description (how to reproduce it): Call function Lin_Wakeup > call Lin_GoToSleepInternal > receiver wakeup signal from other node. Driver can't wakeup Preconditions: Function Lin_Wakeup call 2 function: Lpuart_Lin_Ip_SendWakeupSignal and Lpuart_Lin_Ip_GotoIdleState. But function Lpuart_Lin_Ip_GotoIdleState must be call in IRQ after wakeup signal send complete. If I call function Lin_GoToSleepInternal after Lin_Wakeup function Lpuart_Lin_Ip_GotoIdleState make bit RIE not set in Lin_GoToSleepInternal Test Case ID (internal TC that caught the defect) optional: Lin_TC_FCT_0211 Observed behavior: NA Expected behavior: NA Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Remove function Lpuart_Lin_Ip_GotoIdleState in Lin_Wakeup
ARTD-11537	Bug	[LIN] Function Lin_Wakeup go to callback function and set Lin_Ipw_WakeupFlag = TRUE<*> Detailed description (how to reproduce it): Have sequence: Lin_Wakeup > Lin_GoToSleepInternal > Lin_CheckWakeup. Driver still found a wakeup event. Preconditions: As implemented in ticket: ARTD-11423: Callback will only call when current node detects a wake-up signal sent by other. Driver check it with conditions : FALSE == LinCurrentState->IsBusBusy. But before that, driver call function Lin_Ip_GotoIdleState. In this function, driver always set LinCurrentState->IsBusBusy = FALSE. This bug apply for IPV: LPUART and FLEXIO. Test Case ID (internal TC that caught the defect) optional: Lin_TC_FCT_0018 Observed behavior: NA Expected behavior: NA Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Move function Lin_Ip_GotoIdleState down.
ARTD-11541	New Feature	New Feature [PORT] The Exclusive Areas should be reviewed and updated for K1 platform ,,"NewWorkDescription:

ID	Subtype	Headline and Description
		The Exclusive Areas for K1 platform are incorrect. Some of them are redundant (need to be removed), and the rest need to be renamed. Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Remove/Add/Renaming the Exclusive Areas in code for K1. From IPW to IPL. All exclusives should be existed in IPL. The ones in IPW should be removed. Update the UM/IM if needed"
ARTD-11548	New Feature	New Feature [ADC] SIM Internal supply monitoring channel support ,,"NewWorkDescription: Add support for ADC internal supply monitoring channel enable and source select. They are controlled by CHIPCTL register of SIM module. Add config check in CT/Tresos that only a single supply monitoring channel is configured per group. Add requirement that specifies the usage of supply monitoring. Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: [...]"
ARTD-11549	New Feature	New Feature [ADC] SIM PDB back to back mode selection support ,,"NewWorkDescription: Add support for PDB back to back mode selection. The selection is controlled by PDB_BB_SEL fields found in CHIPCTL register of the SIM module. Presence of these fields is derivative dependent. Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: [...]"
ARTD-11550	New Feature	New Feature [ADC] SIM ADC trigger/pretrigger/software source selection support ,,"NewWorkDescription: Implement support for trigger/pretrigger/software source select. This functionality is controlled per ADC instance by the ADCOPT register of the SIM module. Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: [...]"
ARTD-11546	New Feature	New Feature [WDG] Add check that the Gpt callback function is Wdg_Cbk_GptNotificationX

ID	Subtype	Headline and Description
		,,"Detailed description (how to reproduce it): In GPT triggered mode, the Wdg does not issue an error if the registered callback function of the Gpt channel used by the Wdg is not Wdg_Cbk_GptNotificationX, where X is the Wdg instance number. Observed behavior: No warning is issued if the Gpt callback is not Wdg_Cbk_GptNotificationX. Expected behavior: There should be an error reported so that the user configures the used Gpt channel's callback notification to be Wdg_Cbk_GptNotificationX. Proposed solution optional: For EBT and S32CT add a verification either in .xdm / .template or at generation time to check the Gpt channel's configured callback function name."
ARTD-11547	New Feature	New Feature [WDG] S32ConfiguratorTool and Tresos remove .members from structs (for C90 compliance), keep generated files bit-exact with Tresos" ,,"Detailed description (how to reproduce it): Code generated by EBT and S32CT is not the same. EBT uses .member initialisation ,while S32CT typecast. Expected behavior: Generated code should be the same. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Modify code generation so that struct members are generated with typecast by both configurators.
ARTD-11564	Bug	[S32K1 EAR] SPI: Transmit failed when using transmission with large data via Flexio SPI Master<*> Detailed description (how to reproduce it): PCS pin not keep asserted during transmission when using transmission with large data via Flexio SPI Master !image-2021-06-14-11-37-30-115.png! I think because the TIMCMP is not reset after setting to 0 so it will enable again when timer of clk reset Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: Spi_TC_FTC_1014 Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-11581	Bug	[S32K1 RTM][CRYPTO]Fix compiler warning for example<*> Detailed description (how to reproduce it): Example have compiler warning from cryif driver Test Case ID (internal TC that caught the defect) optional: NA

ID	Subtype	Headline and Description
		Observed behavior: CryIf_TS_T40D2M8I0R0/src/CryIf.c: In function 'CryIf_CallbackNotification': CryIf_TS_T40D2M8I0R0/src/CryIf.c:116:50: warning: unused parameter 'job' [-Wunused-parameter] 116 void CryIf_CallbackNotification (Crypto_JobType job, Std_ReturnType result) CryIf_TS_T40D2M8I0R0/src/CryIf.c:116:70: warning: unused parameter 'result' [-Wunused-parameter] 116 void CryIf_CallbackNotification (Crypto_JobType job, Std_ReturnType result) Expected behavior: No compiler waning during build example Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)
ARTD-11613	New Feature	New Feature [WDG] Verify MISRA comments and check accepted deviations xlsx ,,"Verify logged MISRA comments and check if they are in the Accepted Deviations list or cand be added there. Modify any unclear comments."
ARTD-11622	Bug	[MCU] Wrong clock frequency and divider of ENET module in derivative S32K148<*> Detailed description (how to reproduce it): ENET module divider is 4 but writing 8 to register so in wrong clock frequency Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: Mcu_TC_FCT_0552 Observed behavior: Frequency of ENET module is 24 Mhz Expected behavior: Frequency of ENET module is 12 Mhz Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-11638	Bug	[S32K1][User manuals] Add missing sub-chapters for Module documentation<*> Detailed description (how to reproduce it): Some sub-chapters are missing in Module Documentation part of UM. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Some sub-chapters are missing in Module Documentation part of UM. Expected behavior: There shall not be any missing sub-chapters in Module Documentation part of UM. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update UM
ARTD-11643	Bug	[S32K1 RTM][RM] fix misra violation<*>

ID	Subtype	Headline and Description
		Detailed description (how to reproduce it): misra violation folow file excel attacted Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: misra violation fixed Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-11644	New Feature	New Feature [I2C] CPR_RTD_00190 regarding disabling Dem_SetEventStatus should be implemented ,,"CPR_RTD_00190.i2c should be implemented. All modules which need to call Dem module shall provide a configuration parameter for disabling all calls of Dem_SetEventStatus. If this parameter is activated, no call of Dem_SetEventStatus must be performed. Per default, the call of Dem_SetEventStatus shall be allowed."
ARTD-11646	New Feature	New Feature [ADC] Streaming results reorder ,,"NewWorkDescription: CPR_RTD_00490.adc:The adc driver shall support streaming access mode functionality with the possibility to arrange the adc results as multiple sets of group result buffer. E.g.: for a group with channels {CH1 Ch5 CH7} the resulting stream buffer shall be: {CH1, Ch5, CH7, CH1, Ch5, CH7, CH1, Ch5, CH7} instead of {CH1,CH1,CH1,CH5,CH5,CH5,CH7,CH7,CH7} like supported by autosar standard. This feature extension shall be enabled by using a parameter named Stream Result Grouping at group level. The default value of this parameter shall be False and shall only be editable when streaming access mode is selected for its group. Mark requirement as fulfilled in Requirement source: Port from S32G/K3 Proposed solution optional: N.A."
ARTD-11649	Bug	[CAN] Missing constraint for ICOM on K1<*> Detailed description (how to reproduce it): CanIcomMessageIdType: STANDARD CanIcomIdOperation: GREATER_MINNUM => input invalid value of standard id (> 0x7FF) => fail to check invalid configuration parameter Preconditions: [...]

ID	Subtype	Headline and Description
		Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: !image-2021-06-16-08-04-52-501.png! Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-11650	Bug	[EEP][S32K1] Follow up ticket for Misra violations<*> Detailed description (how to reproduce it): Some Misra violations are not fixed Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Some Misra violations are not fixed Expected behavior: Misra violations shall be fixed Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update code
ARTD-11663	Bug	[WDG] Some functions missing in UM chapter 6.2<*> Detailed description (how to reproduce it): Some functions missing in UM chapter 6.2 Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-11664	Bug	[I2C] Channel ID and channel index are not match on CT configuration<*> Detailed description (how to reproduce it): Channel ID and channel index are not match on CT configuration, it cause some problem if user config like that Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...]

ID	Subtype	Headline and Description
		Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Add a constraint to check match between channel id and channel index
ARTD-11692	Bug	[LIN] DS FLEXIO Header break byte is longer than 13 bits<*> Detailed description (how to reproduce it): DS FLEXIO Header break longer than 13 bits Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: DS FLEXIO Examples Observed behavior: DS FLEXIO Header break longer than 13 bits Expected behavior: DS FLEXIO Header break is correct Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-11700	Bug	[PWM] Mismatch the macro definition<*> Detailed description (how to reproduce it): Mismatch the macro definition in Flexio IP layer #if (defined(FLEXIO_PWM_IP_HAS_PIN_OVERRIDE) && (FLEXIO_PWM_IP_HAS_PIN_OVERRIDE == STD_ON)) #if (defined(FLEXIO_PWM_IP_HAS_PIN_OVERRIDE) && (FLEXIO_PWM_IP_HAS_PIN_OVERRIDE == TRUE)) There are drivers that use '#if' preprocessor directives like follow: '#if (COMPILING_CONDITION == TRUE)' or '#if (COMPILING_CONDITION == STD_ON)', or both See also the discussions on this https://teams.microsoft.com/l/message/19:664e0ee3787240d2b4cbd1f568eaab8b@thread.tacv2/1623815330575?tenantId=686ea1d3-bc2b-4c6f-a92c-d99c5c301635&groupId=f9decd68-7b74-4101-a4d2-ad54e86f727f&parentMessageId=1623815330575&teamName=Zebra&channelName=Group%206&createdTime=1623815330575 Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: Need to align the macro definition Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Replace in the code 'COMPILING_CONDITION == TRUE' by 'COMPILING_CONDITION == STD_ON' for consistency; this will also help us to fix

ID	Subtype	Headline and Description
		violations of 'MISRA Rule 10.3' and to avoid the issues reported in this PR https://bitbucket.sw.nxp.com/projects/ARTD/repos/base/pull-requests/520/overview?commentId=861452
ARTD-11757	Bug	[ICU] The prescaler register of LPTMR is written wrong value compared to config value<*> When user configure the value 4 for prescaler: !image-2021-06-20-22-05-23-624.png! The register in debug interface is wrong of value !image-2021-06-20-22-04-17-114.png!
ARTD-11761	New Feature	New Feature [FLS] Investigate to remove unused configurations in the driver ,,"NewWorkDescription: Investigate the two unused nodes in FlsSectorList: *FlsProgrammingSize & *FlsPhysicalSectorUnlock If they are temporarily unused, they should be greyed out to notice to the users. In case of removing, these items should be considered: Check and remove the related constraint with other nodes Check and remove the corresponding resources in the resource files Remove generated code in *Fls_PBcfg.c and the corresponding structures in *Fls_Types.h*: paSectorUnlock Fls_ProgSizeType Remove the related CRC calculation in both generated code and the driver code Requirement source: (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Add list to select"
ARTD-11764	Bug	[CAN] wrong guard macro for single-core platform on CPR_RTD_00420.can implementation<*> Detailed description (how to reproduce it): !image-2021-06-21-11-33-17-228.png! With above condition the implementation (in Can_Init API) for multicore-platform still be visible on single core platform (S32K1XX) => this will cause unreachable code B*> bValidCoreID seem always be used, => so it should always be declared?! !image-2021-06-21-11-42-51-919.png! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]

ID	Subtype	Headline and Description
ARTD-11767	Bug	[PORT] Build fail when using incorrect name with boolean type<*> Detailed description (how to reproduce it): with BLN_PORT_044, build test Port_TS_COT_005 cfg 28 Preconditions: Node PORT_CI_IP_PORT_DEV_ERR_DETECT true, add NO_STDINT in make file Test Case ID (internal TC that caught the defect) optional: Port_TC_WBT_0004 Observed behavior: This test will build fail during incorrect input of function bellow picture: !image-2021-06-21-13-43-46-671.png! Expected behavior: The driver can be built sucessfully when PORT_CI_IP_PORT_DEV_ERR_DETECT is enabled and NO_STDINT in make file is added. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update from lower case to UPPER CASE: false > FALSE
ARTD-11769	Bug	[LIN] In Slave mode, S32DS allows the value BL_10 for the DetectedBreakLength field. ,,"Detailed description (how to reproduce it): In Slave mode, S32DS allows the value BL 10 for the DetectedBreakLength field. Preconditions: Following Lin Spec 2.1 : !image-2021-06-21-14-24-11-836.png|width=584,height=304! Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: NA Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-11772	Bug	[PORT] Re-check all the bitfields in MSCR registers<*> Detailed description (how to reproduce it): Re-check all the bitfield in MSCR registers. Check one by one if that bitfield can be enabled/disabled when user using Port_Init and other functions. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Re-check all the bitfield in MSCR registers. Check one by one if that bitfield can be enabled/disabled when user using Port_Init and other functions. Expected behavior: All bitifields in MSCRs/PCRs should work correctly Proposed solution optional: All bitifields in MSCRs/PCRs should check again in the code

ID	Subtype	Headline and Description
ARTD-11802	Bug	[ETH] Implement the requirement CPR_RTD_00211 for Eth driver<*> Detailed description (how to reproduce it): Please implement the requirement CPR_RTD_00211 for Eth driver: Any while loop shall be constructed not without at least one dedicated escape for avoidance of endless loops. The while loop in Enet_Ip_ReadTimerValue function should be updated. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...] - Please implement the requirement CPR_RTD_00211 for Eth driver: Any while loop shall be constructed not without at least one dedicated escape for avoidance of endless loops. The while loop in Enet_Ip_ReadTimerValue function should be updated.
ARTD-11804	Bug	[ADC] Adc cannt work with Dma when number of configured channels is greater than number of SC registers<*> Detailed description (how to reproduce it): Adc cannt work with Dma when number of configured channels is greater than number of SC registers Adc_init modifies the values of registers behind the address of highest SC register that causes unexpected behaviors Sw oneshot group can not get status stream complete if if number of configuration channels is greater than ADC_MAX_CHAN_COUNT and using dma Preconditions: Configuration channels is greater than ADC_MAX_CHAN_COUNT Transfering by DMA Test Case ID (internal TC that caught the defect) optional: Adc_TC_FCT_0121 Adc_TS_047 cfg 2 Observed behavior: Number of configuration channels is 17 > 16 max sc channel of S32K144w Start sw trigger oneshot group Expected group status is stream complete without timeout error Stop sw trigger oneshot group Real status: timeout error occurred and can not complete group because dma need to reconfigure for final channel Expected behavior: Group status is stream complete without timeout error Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-11805	Bug	[GPT] Function Srtc_Ip_ConfigureSecondsInterrupt is not working as expected<*> Detailed description (how to reproduce it):

ID	Subtype	Headline and Description
		After calling the function Srtc_Ip_ConfigureSecondsInterrupt, I expected the value TSIC=64 Hz but this did not happen Function Srtc_Ip_ConfigureSecondsInterrupt is not working as expected Preconditions: use the function Srtc_Ip_ConfigureSecondsInterrupt Test Case ID (internal TC that caught the defect) optional: Ip_SRtc_TC_FCT_0006 Observed behavior: the value of TSIC is incorrectly logged through the function Srtc_Ip_ConfigureSecondsInterrupt Expected behavior: the value of TSIC is correctly logged through the function Srtc_Ip_ConfigureSecondsInterrupt Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: double check the values written by the function Srtc_Ip_SetSecondsInterruptConfiguration
ARTD-11824	Bug	[ADC] AdcChannelDelay restriction does not match with ADC_MAX_CHAN_COUNT on S32K148<*> Detailed description (how to reproduce it): AdcChannelDelay restriction does not match with ADC_MAX_CHAN_COUNT on S32K148 Preconditions: Derivative is S32K148 AdcGroupInBacktoBackMode is disable AdcGroupUsesChannelDelays is enable Number of configuration channels is 33 (0->32) Test Case ID (internal TC that caught the defect) optional: Adc_TC_FCT_0121 Adc_TS_047 cfg 3 Observed behavior: AdcChannelDelay tab configure as attachment: Channel delay 16 < Channel delay 15 but ADC_MAX_CHAN_COUNT is 32 for S32K148 Real status: Generation pass Expected behavior: Generate fail because channel delays need to increase in block of 32 Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-12013	New Feature	New Feature [S32K1 RTM] Driver activities for BASE ,,"IPs list: [see SOW for details|https://nxp1.sharepoint.com/:w:/r/sites/Zebra/_layouts/15/Doc.aspx?sourcedoc=%7B0170E3D5-3012-4515-A8A0-4B646688F9C1%7D&file=S32K1%20RTD%20ASR%204.4%201.0.0%20SOW.docx&action=default&mobileredirect=true] "
ARTD-12052	New Feature	New Feature [S32K1 RTM] Dependency Driver activities for S32K1XX

ID	Subtype	Headline and Description
		,,"IPs list: [see SOW for details|https://nxp1.sharepoint.com/:w:/r/sites/Zebra/_layouts/15/Doc.aspx?sourcedoc=%7B0170E3D5-3012-4515-A8A0-4B646688F9C1%7D&file=S32K1%20RTD%20ASR%204.4%201.0.0%20SOW.docx&action=default&mobileredirect=true] "
ARTD-12101	New Feature	New Feature [RM][S32K1 RTM] Mpu_Ip_SetRegionConfig_Privileged need to check regionNumber range ,,"NewWorkDescription: Mpu_Ip_SetRegionConfig_Privileged need to check regionNumber range Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Check regionNumber range in Mpu_Ip_SetRegionConfig_Privileged function"
ARTD-12105	Bug	[ETH] The Error callback is not invoked<*> Detailed description (how to reproduce it): The error callback is not invoked because the EIR register value is clear before checking spurious interrupt. Please check the screen-shot: !image-2021-06-23-11-23-18-500.png! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-12108	Bug	[LIN] Function Flexio_Lin_Ip_AbortTransferData make driver suspended for a period of time<*> Detailed description (how to reproduce it): Test case call function Lin_SendFrame, while header is transmitting test call function Lin_GoToSleepInternal. After that function Lin_Wakeup is called. Wakeup signal don't send. Preconditions: In function Flexio_Lin_Ip_AbortTransferData driver will check SHIFTSTAT register equal 1 to wait until the data is completely. But SHIFTSTAT only equivalent to shiftbuff load data to shifter. So when data not send completed, function Flexio_Lin_Ip_GotoIdleState is called => disable timer => driver suspend. Test Case ID (internal TC that caught the defect) optional: Lin_TC_FCT_1007 Observed behavior: TC failed Expected behavior: TC pass

ID	Subtype	Headline and Description
		Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Check TIMSTAT change to SHIFTSTAT
ARTD-12112	Bug	[ADC] Driver need to check complete flag before reading data in ISR<*> Detailed description (how to reproduce it): Driver need to check complete flag before reading data in ISR Preconditions: AdcGroupUsesChannelDelays is enable AdcGroupInBacktoBackMode is disable Number of configuration channels is 9 (PDB0-channel0 (pre trigger 0->7) and PDB0-channel1 (pre trigger8)) Delay time between configuration channels is too close make PDB0 ch0 error Test Case ID (internal TC that caught the defect) optional: Adc_TC_FCT_0121 Adc_TS_047 cfg3 Observed behavior: Start sw oneshot group Wait until stream complete Stop group Real status: Group status is stream complete but wrong data because some incomplete channels were read Expected behavior: time out occurred because delay time too close make PDB0 has errors and some channels are incomplete Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-12117	Bug	[S32K1 RTM][SPI][FLEXIO] Unsupported SpiDataShiftEdge=TRAILING for multiple transfer<*> Detailed description (how to reproduce it): Relate to multiple transfers when using FLEXIO, RM notes that: !image-2021-06-23-17-26-11-523.png! This means, SpiDataShiftEdge must be LEADING for multiple transfers over FLEXIO But driver has no warning in EB configuration or mention in UM limitation Preconditions: NA Test Case ID (internal TC that caught the defect) optional: Spi_TS_109 Observed behavior: NA Expected behavior: NA Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-12115	Bug	[S32K1 RTM][S32CT] FLS: Abort timeout should get from OsIf_MicrosToTicks()<*> Detailed description (how to reproduce it):

ID	Subtype	Headline and Description
		[Currently: #if (FTFC_TIMEOUT_SUPERVISION_ENABLED == STD_ON) uint32 u32TimerCounterAbort = FTFC_ABORT_TIMEOUT; #endif / FTFC_TIMEOUT_SUPERVISION_ENABLED == STD_ON /] Observed behavior: [The timeout didn't converts to ticks units.] Expected behavior: [#if (FTFC_TIMEOUT_SUPERVISION_ENABLED == STD_ON)}} uint32 u32ElapsedAbortTicks = 0UL;}} uint32 u32TimeoutAbortTicks = OsIf_MicrosToTicks(FTFC_ABORT_TIMEOUT, (OsIf_CounterType)FTFC_TIMEOUT_TYPE);}} uint32 u32CurrentAbortTicks = OsIf_GetCounter((OsIf_CounterType)FTFC_TIMEOUT_TYPE);}}]
ARTD-12116	Bug	[UART][FLEXIO_UART] Timer Status Flag should be clear if there is no more data to transfer, transmission after the last byte are sent ,,"Detailed description (how to reproduce it): !screenshot-1.png|thumbnail! Link: https://community.nxp.com/t5/Kinetis-Microcontrollers/Understanding-FlexIO/ta-p/1115419 As above information Timer Status Flag (TIMSTAT) should be clear in Flexio_Uart_Ip_CheckTxOperation function to write new data to the SHITBUF register to start the transaction. Preconditions: [...]N/A Test Case ID (internal TC that caught the defect) optional: [...]N/A Observed behavior: [...] Timer Status Flag was not cleared in Flexio_Uart_Ip_CheckTxOperation function Expected behavior: [...] Timer Status Flag should be clear in Flexio_Uart_Ip_CheckTxOperation function by Flexio_Mcl_Ip_ClearTimerStatus Proposed solution optional: [...]
ARTD-12119	Bug	[S32K1 RTM] PWM: FTM outputs wasn't placed into expect values when fault events in ongoing and Fault interrupts wasn't generated<*> Detailed description (how to reproduce it): FTM outputs wasn't placed into expect values when fault events in ongoing !image-2021-06-23-17-35-00-635.png|width=811,height=192! Fault interrupts wasn't generated Preconditions: configuring FTM outputs and Fault inputs as attached files

ID	Subtype	Headline and Description
		Test Case ID (internal TC that caught the defect) optional: Pwm_TS_107 Observed behavior: FTM outputs wasn't placed into expect values when fault events in ongoing Fault interrupts wasn't generated Expected behavior: FTM outputs will be placed into expect values when fault events in ongoing Fault interrupts will be generated Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-12121	New Feature	New Feature [S32K1 RTM] Driver activities for EEP ,,"IPs list: [see SOW for details|https://nxp1.sharepoint.com/:w:/r/sites/Zebra/_layouts/15/Doc.aspx?sourcedoc=%7B0170E3D5-3012-4515-A8A0-4B646688F9C1%7D&file=S32K1%20RTD%20ASR%204.4%201.0.0%20SOW.docx&action=default&mobileredirect=true] "

ARTD-12141	Bug	[WDG] Incorrect Det Error when checking Configuration pointer<*>
		
		Detailed description (how to reproduce it):
		If the parameter checking for the driver's initialization function is enabled, the Configuration pointer argument shall be checked with the following conditions:
		In the supportedConfigVariants VariantPreCompile and VariantLinkTime if only one configuration variant set is used, the initialization function does not need nor evaluate the passed argument. Thus the Configuration pointer shall have a NULL_PTR value.
		In the supportedConfigVariant VariantPostBuild or if multiple configuration variant sets are used, the initialization function requires the passed argument. Thus the Configuration pointer shall be different from NULL_PTR.
		If these conditions are not satisfied, a Development error with type <MIP>_E_INIT_FAILED shall be reported to Development Error Tracer (Det)
		ID: 134
		Req ID: CPR_RTD_00563.wdg
		About this requirement when the conditions are not satisfied, driver will raise error WDG_E_INIT_FAILED.
		Preconditions:
		[...]
		Test Case ID (internal TC that caught the defect) optional:
		[...]
		Observed behavior:
		Now, driver raises WDG_E_PARAM_POINTER, not match with requirement
		#if (WDG_PRECOMPILE_SUPPORT == STD_ON)
		if (NULL_PTR != ConfigPtr)
		#else
		if (NULL_PTR == ConfigPtr)
		#endif /*#if (WDG_PRECOMPILE_SUPPORT == STD_ON)*/
		{
		(void)Det_ReportError((uint16)WDG_CHANNEL_MODULE_ID, (uint8)Wdg_Instance, (uint8)WDG_INIT_ID, (uint8)WDG_E_PARAM_POINTER);
		valid = (Std_ReturnType)E_NOT_OK;
		}
		Expected behavior:

ID	Subtype	Headline and Description
		driver will raise error WDG_E_INIT_FAILED Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-12179	Bug	[fr] Fix inconsistency related to using 'TRUE' or 'FALSE' instead of 'STD_ON' or 'STD_OFF' for '#if' preprocessor directives<*> Detailed description : There are drivers that use '_*#if*_' preprocessor directives like follow: '_#if (COMPILING_CONDITION == TRUE*)_' or '_#if (COMPILING_CONDITION == STD_ON*)_', or both See also the discussions on this [post.|https://teams.microsoft.com/l/message/19:664e0ee3787240d2b4cbd1f568eaab8b@thread.tacv2/1623815330575?tenantId=686ea1d3-bc2b-4c6f-a92c-d99c5c301635&groupId=f9decd68-7b74-4101-a4d2-ad54e86f727f&parentMessageId=1623815330575&teamName=Zebra&channelName=Group%206&createdTime=1623815330575] Proposed solution*: Replace in the code '_COMPILING_CONDITION == TRUE*_' by '_COMPILING_CONDITION == STD_ON*_' for consistency; this will also help us to fix violations of '_MISRA Rule 10.3_' and to avoid the issues reported in this [PR|https://bitbucket.sw.nxp.com/projects/ARTD/repos/base/pull-requests/520/overview?commentId=861452]
ARTD-12142	Bug	[RM][S32K1 RTM] Range of Start Address, End Address on CT, EB incorrect ,,"Detailed description (how to reproduce it): Start Address defines the most significant bits of the 0-modulo-32 byte start address of the memory region. End Address defines the most significant bits of the 31-modulo-32 byte end address of the memory region. Test Case ID (internal TC that caught the defect) optional: NA Preconditions: NA Observed behavior: no error when config Start Address on CT, End Address on CT EB different with value in register Expected behavior: error when config Start Address on CT, End Address on CT EB different with value in register Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-12145	Bug	[LIN] While loop does not end in dummy timer<*> Detailed description (how to reproduce it): In dummy TimeoutTick = timeoutUs !image-2021-06-24-16-46-20-356.png!

ID	Subtype	Headline and Description
		And ElaspseTime cannot lager or equal to TimeoutTicks [!image-2021-06-24-16-47-05-151.png|width=1065,height=204,id=x_0-weu-d9-48f34d87019cb538a178648310d33971!|https://eu-prod.asyncgw.teams.microsoft.com/v1/objects/0-weu-d9-48f34d87019cb538a178648310d33971/views/imgo] because ElapsedTime always assigned to 1 [!image-2021-06-24-16-47-25-926.png|width=845,height=117,id=x_0-weu-d6-fb6611fd73346c7fc5b2caf3c1de9e5e!|https://eu-prod.asyncgw.teams.microsoft.com/v1/objects/0-weu-d6-fb6611fd73346c7fc5b2caf3c1de9e5e/views/imgo] Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-12146	Bug	[ICU] FTM Overflow calculation is wrong when overflow notification is enabled<*> Detailed description (how to reproduce it): [This problem is explained in [ARTD-10896]] Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-12144	Bug	[LIN] Driver failed in case : current frame is aborted if a new frame is requested with IPV LPUART<*> Detailed description (how to reproduce it): While header a frame is sending , a new frame is requested. Driver need abort old frame and send new frame. But driver only send header of new frame and a frame error interrupt created. Preconditions: After aborted header old frame, driver call function Lpuart_Lin_Ip_StartSendFrame to setup header of new frame. It will set Break char detect length as 13 bits minimum and enable LIN Break Detect Interrupt. But after that, a header break interrupt of old frame is requested (old was send header completed). In function interrupt handler driver will set Break char detect length as 10 bits minimum and disable LIN Break Detect Interrupt. So 1 frame error interrupt created. Test Case ID (internal TC that caught the defect) optional:

ID	Subtype	Headline and Description
		Lin_TC_FCT_0038 Observed behavior: NA Expected behavior: NA Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Must check 2 flags: receive and transmit in function Lpuart_Lin_Ip_AbortTransferData.
ARTD-12189	Bug	[PORT] Generated file was incorrect in CTHL<*> Detailed description (how to reproduce it): In S32K1XX, use S32DS, compare Port_TS_004 in EB with CT for HL, derivative S32K148 Preconditions: all config output of EB and CT for HL, which need to be same with all derivative Test Case ID (internal TC that caught the defect) optional: Port_TS_004 Observed behavior: packet for s32k148_lqfp144 not available some pins as define in left column in picture below(Port_Cfg.h in EB): !image-2021-06-25-14-30-07-961.png! !image-2021-06-25-14-32-59-166.png! Expected behavior: All config need to generate correctly in CT for HL Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-12197	Bug	[ADC] Driver is not initialized, when interrupt occurs, ISR does not clear COCO bit interrupt status flag. ,,"Detailed description (how to reproduce it): ISR shall check whether its respective driver is initialized. When the driver is not initialized, the ISR do not clear interrupt status flag COCO bit. => always jumps into interrupt Adc_Ip_Handler() and can't get out. Preconditions: Initializes the ADC hardware unit and the driver. Fault injection point to set init static global value to uninit state. Start ADC group conversion. When the interrupt happens, the end of chain conversion callback function is not called, the interrupt flag is not cleared. Test Case ID (internal TC that caught the defect) optional: Adc_TC_FLT_2101.c Observed behavior: The interrupt flag is not cleared. Expected behavior: The interrupt flag is cleared when the interrupt happens. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-12203	New Feature	

ID	Subtype	Headline and Description
		New Feature [MCU] Support SRAM retention ,,"Detailed description (how to reproduce it): Support to to retain and access SRAM contents across functional resets. The description for the implementation is describe in ""SRAM retention: power modes and resets"" chapter. Preconditions: NA Test Case ID (internal TC that caught the defect) optional NA Observed behavior: NA Expected behavior: NA Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution (Optional): Add function support for SRAM retention"
ARTD-12206	Bug	[Wdg] Some DEM error nodes are redundant and Remove information SWT into Channel.c file<*> Detailed description (how to reproduce it): Step 1: Run command: Clean generate any TS in list test of WDG Step 2: Check interface Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: Some DEM error nodes are redundant: WDG_E_CORRUPT_CONFIG, WDG_E_UNLOCKED, WDG_E_INVALID_PARAMETER, WDG_E_FORBIDDEN_INVOCATION, WDG_E_INVALID_CALL Remove all information of SWT in Wdg_Channel.c file Please see the attached file for more detail. Expected behavior: Remove all Dem node not use and all information in Wdg_Channel.c file Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-12226	New Feature	New Feature [adc] Replace "NO_INIT" with "CLEARED" in memory section macros" ,,"1. Replace "NO_INIT" with "CLEARED" in the memory section identifiers of both source code files and documentation. For example: <MIP>_START_SEC_VAR_NO_INIT_UNSPECIFIED > <MIP>_START_SEC_VAR_CLEARED_UNSPECIFIED <MIP>_STOP_SEC_VAR_NO_INIT_32 > <MIP>_STOP_SEC_VAR_CLEARED_32 ... 2. In file "sections_to_be_defined_in_memmaph.dox", perform the following replacement in the description field of the "CLEARED" (formerly "NO_INIT") sections:

ID	Subtype	Headline and Description
		"These variables are never cleared and never initialized by start-up code." > "These variables are cleared to zero by start-up code." 3. Move all zero-initialized variables into the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" 4. Remove all explicit zero-initializers in the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed; / implicit zero initialization*/ #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"
ARTD-12230	New Feature	New Feature [crypto] Replace "NO_INIT" with "CLEARED" in memory section macros" ,,"1. Replace "NO_INIT" with "CLEARED" in the memory section identifiers of both source code files and documentation. For example: <MIP>_START_SEC_VAR_NO_INIT_UNSPECIFIED > <MIP>_START_SEC_VAR_CLEARED_UNSPECIFIED <MIP>_STOP_SEC_VAR_NO_INIT_32 > <MIP>_STOP_SEC_VAR_CLEARED_32 ... 2. In file "sections_to_be_defined_in_memmaph.dox", perform the following replacement in the description field of the "CLEARED" (formerly "NO_INIT") sections: "These variables are never cleared and never initialized by start-up code." > "These variables are cleared to zero by start-up code." 3. Move all zero-initialized variables into the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" 4. Remove all explicit zero-initializers in the "CLEARED" memory sections.

ID	Subtype	Headline and Description
		For example: #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed; / implicit zero initialization*/ #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"
ARTD-12231	New Feature	New Feature [dem] Replace "NO_INIT" with "CLEARED" in memory section macros" ,,"1. Replace "NO_INIT" with "CLEARED" in the memory section identifiers of both source code files and documentation. For example: <MIP>_START_SEC_VAR_NO_INIT_UNSPECIFIED > <MIP>_START_SEC_VAR_CLEARED_UNSPECIFIED <MIP>_STOP_SEC_VAR_NO_INIT_32 > <MIP>_STOP_SEC_VAR_CLEARED_32 ... 2. In file "sections_to_be_defined_in_memmaph.dox", perform the following replacement in the description field of the "CLEARED" (formerly "NO_INIT") sections: "These variables are never cleared and never initialized by start-up code." > "These variables are cleared to zero by start-up code." 3. Move all zero-initialized variables into the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" 4. Remove all explicit zero-initializers in the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed; / implicit zero initialization*/ #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"
ARTD-12232	New Feature	New Feature

ID	Subtype	Headline and Description
		[det] Replace "NO_INIT" with "CLEARED" in memory section macros" ,,"1. Replace "NO_INIT" with "CLEARED" in the memory section identifiers of both source code files and documentation. For example: <MIP>_START_SEC_VAR_NO_INIT_UNSPECIFIED > <MIP>_START_SEC_VAR_CLEARED_UNSPECIFIED <MIP>_STOP_SEC_VAR_NO_INIT_32 > <MIP>_STOP_SEC_VAR_CLEARED_32 ... 2. In file "sections_to_be_defined_in_memmaph.dox", perform the following replacement in the description field of the "CLEARED" (formerly "NO_INIT") sections: "These variables are never cleared and never initialized by start-up code." > "These variables are cleared to zero by start-up code." 3. Move all zero-initialized variables into the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" 4. Remove all explicit zero-initializers in the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed; / implicit zero initialization*/ #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"
ARTD-12234	New Feature	New Feature [ecum] Replace "NO_INIT" with "CLEARED" in memory section macros" ,,"1. Replace "NO_INIT" with "CLEARED" in the memory section identifiers of both source code files and documentation. For example: <MIP>_START_SEC_VAR_NO_INIT_UNSPECIFIED > <MIP>_START_SEC_VAR_CLEARED_UNSPECIFIED <MIP>_STOP_SEC_VAR_NO_INIT_32 > <MIP>_STOP_SEC_VAR_CLEARED_32 ... 2. In file "sections_to_be_defined_in_memmaph.dox", perform the following replacement in the description field of the "CLEARED" (formerly "NO_INIT") sections: "These variables are never cleared and never initialized by start-up code." > "These variables are cleared to zero by start-up code." 3. Move all zero-initialized variables into the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h"

ID	Subtype	Headline and Description
		static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" 4. Remove all explicit zero-initializers in the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed; / implicit zero initialization*/ #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"
ARTD-12235	New Feature	New Feature [eep] Replace "NO_INIT" with "CLEARED" in memory section macros" ,,"1. Replace "NO_INIT" with "CLEARED" in the memory section identifiers of both source code files and documentation. For example: <MIP>_START_SEC_VAR_NO_INIT_UNSPECIFIED > <MIP>_START_SEC_VAR_CLEARED_UNSPECIFIED <MIP>_STOP_SEC_VAR_NO_INIT_32 > <MIP>_STOP_SEC_VAR_CLEARED_32 ... 2. In file "sections_to_be_defined_in_memmaph.dox", perform the following replacement in the description field of the "CLEARED" (formerly "NO_INIT") sections: "These variables are never cleared and never initialized by start-up code." > "These variables are cleared to zero by start-up code." 3. Move all zero-initialized variables into the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" 4. Remove all explicit zero-initializers in the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"

ID	Subtype	Headline and Description
		> #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed; / implicit zero initialization*/ #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"
ARTD-12236	New Feature	New Feature [eth] Replace "NO_INIT" with "CLEARED" in memory section macros" ,,"1. Replace "NO_INIT" with "CLEARED" in the memory section identifiers of both source code files and documentation. For example: <MIP>_START_SEC_VAR_NO_INIT_UNSPECIFIED > <MIP>_START_SEC_VAR_CLEARED_UNSPECIFIED <MIP>_STOP_SEC_VAR_NO_INIT_32 > <MIP>_STOP_SEC_VAR_CLEARED_32 ... 2. In file "sections_to_be_defined_in_memmaph.dox", perform the following replacement in the description field of the "CLEARED" (formerly "NO_INIT") sections: "These variables are never cleared and never initialized by start-up code." > "These variables are cleared to zero by start-up code." 3. Move all zero-initialized variables into the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" 4. Remove all explicit zero-initializers in the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed; / implicit zero initialization*/ #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"
ARTD-12210	Bug	[ICU][S32K1XX] Lptmr is not generated notification callback for IP layer.<*> The callback is always NULL_PTR in this output configuration for the struct below: typedef struct { boolean chInit; /**< chInit state / Lptmr_Icu_MeasurementModeType measMode; /**< measurement mode /

ID	Subtype	Headline and Description

		Lptmr_Icu_Ip_NotifyType lptmrChannelNotification; /*!< notification function for LPTMR IPL. /
		Lptmr_Icu_Ip_CallbackType callback{color}; /*!< interrupt callback function. /
		uint16 callbackParam; /*!< Logic channel for which callback is executed. /
		boolean notificationEnable; /*!< Notification status /
		}Lptmr_Icu_Ip_ChStateType;
		But in the interrupt function, the callback is checked:
		switch (measMode)
		{
		case LPTMR_ICU_MODE_SIGNAL_EDGE_DETECT:
		{
		if(NULL_PTR != Lptmr_Icu_aChConfig[instance].callback)
		{
		Lptmr_Icu_aChConfig[instance].{color:#de350b}callback{color}
		(Lptmr_Icu_aChConfig[instance].{color:#de350b}callbackParam{color},
		(boolean)FALSE);
		}
		}
		break;
		default: / case LPTMR_ICU_EDGE_COUNT:*/
		{
		/ Do nothing /
		}
		break;
		}

ARTD-12211	New Feature	New Feature [CAN] Enable Timestamp for K1xx ,,"NewWorkDescription: Update Code and Add Timestamp timer feature. Merge it with the HR timestamp which is not supported. Enable Tresos & CT configurator for this feature. Enable for IP and HLD driver level. Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: [...]"
ARTD-12213	Bug	[S32k1xx][Platform] Using test_EqClass tools can't gene isq name and isqhardler from Platform.component<*> Detailed description (how to reproduce it): ,Using test_eqclass tool gene config for test eq cot on s32ct. 1. Creating the mex file with as many configs as possible on S32CT, 2. Using test_eqclass tools gene config ramdom for test eq cot. Isq name, name, and isqhardler can't gene plaform.component. Maybe platform.component aren't allow read them. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Platform_TS_COT_CT_002 Observed behavior: !image-2021-06-28-14-22-28-319.png! Expected behavior:

ID	Subtype	Headline and Description
		Update platform.component allow read name isq Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: !image-2021-06-28-14-26-54-461.png! for ex: line of code 912 and 918, Add enable = "false". !image-2021-06-28-14-29-24-764.png!
ARTD-12218	New Feature	New Feature [PORT] Align the naming of DSE upon unused pins and PortPin list ,,"NewWorkDescription: Please re-align the naming of DSE field in both unused pins and portPin list. Those should be ""High_Drive_Strength"" and ""Low_Driver_strength"" rather than the previous ones. Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Those should be ""High_Drive_Strength"" and ""Low_Driver_strength"" rather than the previous ones."
ARTD-12242	New Feature	New Feature [gpt] Replace "NO_INIT" with "CLEARED" in memory section macros" ,,"1. Replace "NO_INIT" with "CLEARED" in the memory section identifiers of both source code files and documentation. For example: <MIP>_START_SEC_VAR_NO_INIT_UNSPECIFIED > <MIP>_START_SEC_VAR_CLEARED_UNSPECIFIED <MIP>_STOP_SEC_VAR_NO_INIT_32 > <MIP>_STOP_SEC_VAR_CLEARED_32 ... 2. In file "sections_to_be_defined_in_memmaph.dox", perform the following replacement in the description field of the "CLEARED" (formerly "NO_INIT") sections: "These variables are never cleared and never initialized by start-up code." > "These variables are cleared to zero by start-up code." 3. Move all zero-initialized variables into the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" 4. Remove all explicit zero-initializers in the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"

ID	Subtype	Headline and Description
		> #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed; / implicit zero initialization*/ #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"
ARTD-12243	New Feature	New Feature [i2c] Replace "NO_INIT" with "CLEARED" in memory section macros" ,,"1. Replace "NO_INIT" with "CLEARED" in the memory section identifiers of both source code files and documentation. For example: <MIP>_START_SEC_VAR_NO_INIT_UNSPECIFIED > <MIP>_START_SEC_VAR_CLEARED_UNSPECIFIED <MIP>_STOP_SEC_VAR_NO_INIT_32 > <MIP>_STOP_SEC_VAR_CLEARED_32 ... 2. In file "sections_to_be_defined_in_memmaph.dox", perform the following replacement in the description field of the "CLEARED" (formerly "NO_INIT") sections: "These variables are never cleared and never initialized by start-up code." > "These variables are cleared to zero by start-up code." 3. Move all zero-initialized variables into the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" 4. Remove all explicit zero-initializers in the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed; / implicit zero initialization*/ #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"
ARTD-12245	New Feature	New Feature [icu] Replace "NO_INIT" with "CLEARED" in memory section macros" ,,"1. Replace "NO_INIT" with "CLEARED" in the memory section identifiers of both source code files and documentation. For example: <MIP>_START_SEC_VAR_NO_INIT_UNSPECIFIED > <MIP>_START_SEC_VAR_CLEARED_UNSPECIFIED

ID	Subtype	Headline and Description
		<MIP>_STOP_SEC_VAR_NO_INIT_32 > <MIP>_STOP_SEC_VAR_CLEARED_32 ... 2. In file "sections_to_be_defined_in_memmaph.dox", perform the following replacement in the description field of the "CLEARED" (formerly "NO_INIT") sections: "These variables are never cleared and never initialized by start-up code." > "These variables are cleared to zero by start-up code." 3. Move all zero-initialized variables into the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" 4. Remove all explicit zero-initializers in the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed; / implicit zero initialization*/ #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"
ARTD-12246	New Feature	New Feature [lin] Replace "NO_INIT" with "CLEARED" in memory section macros" ,,"1. Replace "NO_INIT" with "CLEARED" in the memory section identifiers of both source code files and documentation. For example: <MIP>_START_SEC_VAR_NO_INIT_UNSPECIFIED > <MIP>_START_SEC_VAR_CLEARED_UNSPECIFIED <MIP>_STOP_SEC_VAR_NO_INIT_32 > <MIP>_STOP_SEC_VAR_CLEARED_32 ... 2. In file "sections_to_be_defined_in_memmaph.dox", perform the following replacement in the description field of the "CLEARED" (formerly "NO_INIT") sections: "These variables are never cleared and never initialized by start-up code." > "These variables are cleared to zero by start-up code." 3. Move all zero-initialized variables into the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"

ID	Subtype	Headline and Description
		static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" 4. Remove all explicit zero-initializers in the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed; / implicit zero initialization*/ #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"
ARTD-12249	New Feature	New Feature [ocu] Replace "NO_INIT" with "CLEARED" in memory section macros" ,,"1. Replace "NO_INIT" with "CLEARED" in the memory section identifiers of both source code files and documentation. For example: <MIP>_START_SEC_VAR_NO_INIT_UNSPECIFIED > <MIP>_START_SEC_VAR_CLEARED_UNSPECIFIED <MIP>_STOP_SEC_VAR_NO_INIT_32 > <MIP>_STOP_SEC_VAR_CLEARED_32 ... 2. In file "sections_to_be_defined_in_memmaph.dox", perform the following replacement in the description field of the "CLEARED" (formerly "NO_INIT") sections: "These variables are never cleared and never initialized by start-up code." > "These variables are cleared to zero by start-up code." 3. Move all zero-initialized variables into the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" 4. Remove all explicit zero-initializers in the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed; / implicit zero initialization*/ #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"

ID	Subtype	Headline and Description
ARTD-12251	New Feature	New Feature [port] Replace "NO_INIT" with "CLEARED" in memory section macros" ,,"1. Replace "NO_INIT" with "CLEARED" in the memory section identifiers of both source code files and documentation. For example: <MIP>_START_SEC_VAR_NO_INIT_UNSPECIFIED > <MIP>_START_SEC_VAR_CLEARED_UNSPECIFIED <MIP>_STOP_SEC_VAR_NO_INIT_32 > <MIP>_STOP_SEC_VAR_CLEARED_32 ... 2. In file "sections_to_be_defined_in_memmaph.dox", perform the following replacement in the description field of the "CLEARED" (formerly "NO_INIT") sections: "These variables are never cleared and never initialized by start-up code." > "These variables are cleared to zero by start-up code." 3. Move all zero-initialized variables into the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" 4. Remove all explicit zero-initializers in the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed; / implicit zero initialization*/ #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"
ARTD-12252	New Feature	New Feature [pwm] Replace "NO_INIT" with "CLEARED" in memory section macros" ,,"1. Replace "NO_INIT" with "CLEARED" in the memory section identifiers of both source code files and documentation. For example: <MIP>_START_SEC_VAR_NO_INIT_UNSPECIFIED > <MIP>_START_SEC_VAR_CLEARED_UNSPECIFIED <MIP>_STOP_SEC_VAR_NO_INIT_32 > <MIP>_STOP_SEC_VAR_CLEARED_32 ... 2. In file "sections_to_be_defined_in_memmaph.dox", perform the following replacement in the description field of the "CLEARED" (formerly "NO_INIT") sections: "These variables are never cleared and never initialized by start-up code." > "These variables are cleared to zero by start-up code." 3. Move all zero-initialized variables into the "CLEARED" memory sections.

ID	Subtype	Headline and Description
		For example: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" 4. Remove all explicit zero-initializers in the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed; / implicit zero initialization*/ #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"
ARTD-12255	New Feature	New Feature [rte] Replace "NO_INIT" with "CLEARED" in memory section macros" ,,"1. Replace "NO_INIT" with "CLEARED" in the memory section identifiers of both source code files and documentation. For example: <MIP>_START_SEC_VAR_NO_INIT_UNSPECIFIED > <MIP>_START_SEC_VAR_CLEARED_UNSPECIFIED <MIP>_STOP_SEC_VAR_NO_INIT_32 > <MIP>_STOP_SEC_VAR_CLEARED_32 ... 2. In file "sections_to_be_defined_in_memmaph.dox", perform the following replacement in the description field of the "CLEARED" (formerly "NO_INIT") sections: "These variables are never cleared and never initialized by start-up code." > "These variables are cleared to zero by start-up code." 3. Move all zero-initialized variables into the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" 4. Remove all explicit zero-initializers in the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"

ID	Subtype	Headline and Description
		static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed; / implicit zero initialization*/ #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"
ARTD-12256	New Feature	New Feature [sai] Replace "NO_INIT" with "CLEARED" in memory section macros" ,,"1. Replace "NO_INIT" with "CLEARED" in the memory section identifiers of both source code files and documentation. For example: <MIP>_START_SEC_VAR_NO_INIT_UNSPECIFIED > <MIP>_START_SEC_VAR_CLEARED_UNSPECIFIED <MIP>_STOP_SEC_VAR_NO_INIT_32 > <MIP>_STOP_SEC_VAR_CLEARED_32 ... 2. In file "sections_to_be_defined_in_memmaph.dox", perform the following replacement in the description field of the "CLEARED" (formerly "NO_INIT") sections: "These variables are never cleared and never initialized by start-up code." > "These variables are cleared to zero by start-up code." 3. Move all zero-initialized variables into the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" 4. Remove all explicit zero-initializers in the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed; / implicit zero initialization*/ #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"
ARTD-12259	New Feature	New Feature [spi] Replace "NO_INIT" with "CLEARED" in memory section macros" ,,"1. Replace "NO_INIT" with "CLEARED" in the memory section identifiers of both source code files and documentation.

ID	Subtype	Headline and Description
		For example: <MIP>_START_SEC_VAR_NO_INIT_UNSPECIFIED > <MIP>_START_SEC_VAR_CLEARED_UNSPECIFIED <MIP>_STOP_SEC_VAR_NO_INIT_32 > <MIP>_STOP_SEC_VAR_CLEARED_32 ... 2. In file "sections_to_be_defined_in_memmaph.dox", perform the following replacement in the description field of the "CLEARED" (formerly "NO_INIT") sections: "These variables are never cleared and never initialized by start-up code." > "These variables are cleared to zero by start-up code." 3. Move all zero-initialized variables into the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" 4. Remove all explicit zero-initializers in the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed; / implicit zero initialization*/ #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"
ARTD-12261	New Feature	New Feature [uart] Replace "NO_INIT" with "CLEARED" in memory section macros" ,,"1. Replace "NO_INIT" with "CLEARED" in the memory section identifiers of both source code files and documentation. For example: <MIP>_START_SEC_VAR_NO_INIT_UNSPECIFIED > <MIP>_START_SEC_VAR_CLEARED_UNSPECIFIED <MIP>_STOP_SEC_VAR_NO_INIT_32 > <MIP>_STOP_SEC_VAR_CLEARED_32 ... 2. In file "sections_to_be_defined_in_memmaph.dox", perform the following replacement in the description field of the "CLEARED" (formerly "NO_INIT") sections: "These variables are never cleared and never initialized by start-up code." > "These variables are cleared to zero by start-up code." 3. Move all zero-initialized variables into the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h"

ID	Subtype	Headline and Description
		> #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" 4. Remove all explicit zero-initializers in the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed; / implicit zero initialization*/ #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"
ARTD-12262	New Feature	New Feature [wdg] Replace "NO_INIT" with "CLEARED" in memory section macros" ,,"1. Replace "NO_INIT" with "CLEARED" in the memory section identifiers of both source code files and documentation. For example: <MIP>_START_SEC_VAR_NO_INIT_UNSPECIFIED > <MIP>_START_SEC_VAR_CLEARED_UNSPECIFIED <MIP>_STOP_SEC_VAR_NO_INIT_32 > <MIP>_STOP_SEC_VAR_CLEARED_32 ... 2. In file "sections_to_be_defined_in_memmaph.dox", perform the following replacement in the description field of the "CLEARED" (formerly "NO_INIT") sections: "These variables are never cleared and never initialized by start-up code." > "These variables are cleared to zero by start-up code." 3. Move all zero-initialized variables into the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_INIT_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" 4. Remove all explicit zero-initializers in the "CLEARED" memory sections. For example: #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" static boolean bSwapToBePerformed = (boolean)FALSE; / explicit zero initialization / #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h" > #define FEE_START_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"

ID	Subtype	Headline and Description
		static boolean bSwapToBePerformed; / implicit zero initialization*/ #define FEE_STOP_SEC_VAR_CLEARED_BOOLEAN #include "Fee_MemMap.h"
ARTD-12265	Bug	[EEP] Missing extern when enabling Notif in IPL<*> Detailed description (how to reproduce it): Missing extern functions acCallBackPtr, startEepromAccessNotifPtr, finishedEepromAccessNotifPtr in generation files in IP layer. See more in picture: !image-2021-06-29-09-09-31-222.png|thumbnail! !image-2021-06-29-09-09-57-495.png|thumbnail! Preconditions: enable acCallBackPtr, startEepromAccessNotifPtr, finishedEepromAccessNotifPtr in IPL Test Case ID (internal TC that caught the defect) optional: Ip_Ftfc_TC_FCT_1005.c Observed behavior: build failed Expected behavior: build successfully Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: add extern as picture below:
ARTD-12266	Bug	[EEP] Got hardfault if write with start address was un-aligned In IPL<*> Detailed description (how to reproduce it): EU_ASSERT(FTFC_EEP_IP_STATUS_OK == Ftfc_Eep_Ip_Write(0U 1U, dataWrTestPatt, 4U, FALSE)); if start address was un-aligned then driver write 4 buytes to eep memory then one hardfault ocured. See more in picure Pls check erase/read/compare if it also is same issue. ? !image-2021-06-29-09-36-08-822.png|thumbnail! Preconditions: Ftfc_Eep_Ip_Write(0U 1U, dataWrTestPatt, 4U, FALSE) Test Case ID (internal TC that caught the defect) optional: Ip_Ftfc_TC_FCT_1005.c Observed behavior: hardfault occurs Expected behavior: do not have any hardfault Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Checking align before writing/read/compare/erase.
ARTD-12268	Bug	[WDG] Wdg_Ipw_Deinit should be removed or guarded by macro in S32K1XX<*> Detailed description (how to reproduce it): Wdg_Ipw_Deinit is call by HLD function Wdg_ChannelClearResetRequest which is not available in S32K1XX Wdg_ChannelClearResetRequest guarded by #if (WDG_DISABLE_ALLOWED == STD_ON) #if (WDG_CLEAR_RESET_REQUEST == STD_ON) but

ID	Subtype	Headline and Description
		Wdg_Ipw_Deinit is not guarded by any marcro => This function still build in test, that make affect to CCOV index Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: Wdg_Ipw_Deinit still can build although it is not available in S32K1 Expected behavior: Wdg_Ipw_Deinit should be removed or guarded by macro in S32K1XX Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-12290	Bug	[LIN] - Flexio irq handler channel parameter is not correct<*> Detailed description (how to reproduce it): A flexio lin channel contains 2 hw flexio channels (pairs of shiftertimer) Configure a flexio lin channel. Use another Mcl Flexio channel than 0 and 1. Driver don't check driver is initialized in ISR of IPV Flexio Preconditions: Follow req CPR_RTD_00011.lin: ISR shall check whether its respective driver is initialized. If the driver is not initialized, the ISR shall only clear interrupt status flag and return immediately. This requirement is implemented from Lpuart !image-2021-07-09-09-39-22-271.png! Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: In this case, the Mcl Flexio Irq is called using the shifter+timer pair channel number. Inside Flexio Lin driver, there is different scheme of channels which are using the pairs of shifter timer. This is why the wrong channel is addressed. Expected behavior: In Flexio Lin Irq, the correct channel must be addressed. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Add an easing mapping between the Mcl Flexio channels (hardware channels) and Flexio Lin channels (driver channels)
ARTD-12311	Bug	[OCU] Can't set clock mode for FTM module without configuring consecutive modules<*> Detailed description (how to reproduce it): When test with tag ocu: OCU_091 we discovered if config the modules with with discontinuous order function Ocu_SetClockMoke not working properly. Example I config with FTM0, FTM2 and FTM3 then only ftm0 and ftm2 are set correctly and ftm3 will not be set. Test Case ID (internal TC that caught the defect) optional: Ocu_TC_FCT_0219 in Ocu_TS_101 and Ocu_TS_DS_101 (S32K144W)
ARTD-12312	Bug	[OCU] When choose PinUsed is false but can still use Ocu_SetPinAction function to set action<*> Detailed description (how to reproduce it): When choose PinUsed is false but can still use Ocu_SetPinAction function to set action after.

ID	Subtype	Headline and Description
		Preconditions: OCU_091 Test Case ID (internal TC that caught the defect) optional: Ocu_TS_DS_100
ARTD-12325	Bug	[PWM] The DetApiID value passed in ValidateParamDeadTime and ValidateDeadTime functions is incorrect<*> Detailed description (how to reproduce it): The DetApiID value passed in Pwm_ValidateParamDeadTime and Pwm_ValidateDeadTime functions is incorrect. Both function check det error for SetChannelDeadtime function, but the DetApiID value passed in both function is PWM_SETTRIGGERDELAY_ID. It is incorrect Preconditions: Enable set channel deadtime Test Case ID (internal TC that caught the defect) optional: Pwm_TC_FCT_2020 Observed behavior: N/A Expected behavior: N/A Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-12332	Bug	[EEP] Different output CT and EB<*> Detailed description (how to reproduce it): See more attachments !image-2021-06-30-14-33-42-492.png|thumbnail! !image-2021-06-30-14-34-19-229.png|thumbnail! !image-2021-06-30-14-35-01-161.png|thumbnail! !image-2021-06-30-14-36-21-572.png|thumbnail! Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: N/A Expected behavior: N/A Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-12337	Bug	[S32K1 RTM] PWM: FTM outputs wasn't generated when using the Ftm_Pwm_Ip_UpdatePwmChannel function<*> Detailed description (how to reproduce it): Using 2 functions with parameters as bellow: Ftm_Pwm_Ip_UpdatePwmPeriod(FTM_INSTANCE_4, 0x5000, TRUE); Ftm_Pwm_Ip_UpdatePwmChannel(FTM_INSTANCE_4, FTM_CH_1, 0x5000, 0x000, TRUE); FTM outputs wasn't generated due to condition if ((firstEdge <= ftmPeriod) && (secondEdge <= ftmPeriod)) in Ftm_Pwm_Ip_UpdatePwmChannel was failed (the ftmPeriod variable was updated with value in Ftm_Pwm_Ip_UpdatePwmPeriod function)

ID	Subtype	Headline and Description
		Preconditions: Period default is 0x4000 Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: FTM outputs wasn't generated Expected behavior: FTM outputs will be generated Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: update value of Ftm_Pwm_Ip_Period[] in Ftm_Pwm_Ip_UpdatePwmPeriod function
ARTD-12342	Bug	[ETH] Remove redundant a close bracket for ENET_WAKEUP_INTERRUPT in Enet_Ip_PBcfg.c<*> Detailed description (how to reproduce it): Please remove redundant a close bracket for ENET_WAKEUP_INTERRUPT in Enet_Ip_PBcfg.c file. Please check the attached file for detail. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-12380	Bug	[CRYPTO] Disable CCIE for spurious interrupts<*> strong text_*Detailed description (how to reproduce it): Trigger an interrupt when Csec_Ip_pState is not set or bCmdInProgress signals that a command is not in progress. Preconditions: Csec_Ip_pState is null pointer or Csec_Ip_pState->bCmdInProgress is false. Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Infinite loop due to the fact that CCIE is not cleared. Expected behavior: For spurious interrupt when Csec_Ip_pState is not set or bCmdInProgress signals that a command is not in progress clear CCIE and exit Csec_Ip_IrqHandler. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-12379	Bug	[S32K1XX] Crypto driver doesn't raise an error when CryptoPrimitives configuration fails<*> Detailed description (how to reproduce it):

ID	Subtype	Headline and Description
		When config CryptoPrimitiveService is decrypt or encrypt , CryptoPrimitiveAlgorithmSecondaryFamily should be CRYPTO_ALGOFAM_NOT_SET or CRYPTO_ALGOFAM_CUSTOM but acctually ,config is CRYPTO_ALGOFAM_RNG or CRYPTO_ALGOFAM_AES When config CryptoPrimitiveService is decrypt or encrypt , driver EB not raise generate error . I try config same parameter on CT and Driver gen fail CRYPTO_ALGOFAM_RNG or CRYPTO_ALGOFAM_AES. You can see picture i cap screen bellow attachments site Preconditions: NA Observed behavior: Driver not generate fail when config CryptoPrimitiveService is encrypt or decrypto and CryptoPrimitiveAlgorithmSecondaryFamily is CRYPTO_ALGOFAM_RNG or CRYPTO_ALGOFAM_AES at the same time Expected behavior: Driver raise gen fail when config fail Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-12988	Bug	[ICU][S32K1XX] The overflow callback FTM is not called for ip layer<*> In debug interface, i see this condition for overflow ip layer is not satisfied: !image-2021-07-02-09-06-07-970.png!
ARTD-12989	Bug	[ETH] Can not get timestamp for outcoming frame<*> Detailed description (how to reproduce it): Can not get timestamp of of outcoming frame due to TS bit in BD is not be enabled and interrupt for timestamp not enable as well Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: Eth_TC_FCT_0187, Eth_TC_FCT_0188 Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-12990	Bug	[UART] While loop is infinity in dummy timer<*> Detailed description (how to reproduce it): In Osif Dummy counter mode, OsIf_GetCounter always returns 0, then Uart_Ip_CheckTimeout always return FALSE Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: Fix issue

ID	Subtype	Headline and Description
		Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-12996	New Feature	New Feature [RM] Update code after performing code review ,,"NewWorkDescription: Update code after performing code review Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: NA"
ARTD-12999	New Feature	New Feature [GPT] Some nodes need to switch to readonly when the platform doesn't support this feature ,,"NewWorkDescription: On tresos EB interface, there are buttons which platform does not support, we need to change their properties to read-only GptTimeoutMethod GptTimeoutDuration GptPredefTimer100us32bitEnable Requirement source: N/A (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Change the properties of these nodes to readonly"
ARTD-13000	Bug	[S32K1XX RTM] SPI: DMA Fast mode transmit with wrong default data<*> Detailed description (how to reproduce it): DMA Fastmode transmit with wrong default data When setup EB TX buffer with NULL_PTR for transmit with default data which configured at channel transferred on EB tresos. The data transmit from master is not correct. I have solution to fix, please take a look on that and fix this bug: in Lpspi_Ip_DmaFastConfig() function: !image-2021-07-02-17-56-45-437.png! in Lpspi_Ip_TxDmaTcdSGConfig() function: !image-2021-07-02-17-58-03-914.png! !image-2021-07-02-17-57-51-916.png! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: Spi_TC_FCT_10134 Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)

ID	Subtype	Headline and Description
		Proposed solution optional: [...]
ARTD-13006	New Feature	New Feature [FLS] Add support to use FlsNumberOfSectors for internal flash sectors ,,"Investigate to remove the constraint of FlsNumberOfSectors that is applying for internal flash sectors: {{}}{{}}{{}}<a:tst expr=""not(contains(../FlsPhysicalSector, 'FLS_EXT_SECTOR')) and (node:fallback(., 0) > 1)"" true=""Multiple of internal sectors are not allowed.""/> This AUTOSAR configuration (*ECUC_Fls_00280*) allows the users to configure multiple continuous physycal sectors"
ARTD-13049	Bug	[eth] Fix file version checking<*> Detailed description (how to reproduce it): Fix all violation version checking Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: [...] Expected behavior: No file version checking errors. Please ensure all violations fix for your driver, reference link: timor.ea.freescale.net/0/project/custom_file_verchecking/details Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...] - NewWorkDescription: Fix all file version checking for RTD drivers Requirement source: Attachment baseline report on 2021.07.05 Proposed solution optional: [...]
ARTD-13053	Bug	[i2c] Fix file version checking<*> Detailed description (how to reproduce it): Fix all violation version checking Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: [...] Expected behavior: No file version checking errors. Please ensure all violations fix for your driver, reference link: timor.ea.freescale.net/0/project/custom_file_verchecking/details Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...] - NewWorkDescription: Fix all file version checking for RTD drivers Requirement source: Attachment baseline report on 2021.07.05

ID	Subtype	Headline and Description
		Proposed solution optional: [...]
ARTD-13055	Bug	[lin] Fix file version checking<*> Detailed description (how to reproduce it): Fix all violation version checking Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: [...] Expected behavior: No file version checking errors. Please ensure all violations fix for your driver, reference link: timor.ea.freescale.net/0/project/custom_file_verchecking/details Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...] - NewWorkDescription: Fix all file version checking for RTD drivers Requirement source: Attachment baseline report on 2021.07.05 Proposed solution optional: [...]
ARTD-13059	Bug	[platform] Fix file version checking<*> Detailed description (how to reproduce it): Fix all violation version checking Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: [...] Expected behavior: No file version checking errors. Please ensure all violations fix for your driver, reference link: timor.ea.freescale.net/0/project/custom_file_verchecking/details Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...] - NewWorkDescription: Fix all file version checking for RTD drivers Requirement source: Attachment baseline report on 2021.07.05 Proposed solution optional: [...]
ARTD-13091	Bug	[rm] Fix VSMD violation<*> Create VSMD report for driver Fix any VSMD violation if any Reference: [http://bohai.ea.freescale.net/1/project/custom_vsmd/details]
ARTD-13153	Bug	

ID	Subtype	Headline and Description
		[PORT] Generated file was incorrect when enable LK register in HL<*> Detailed description (how to reproduce it): Execute test Port_TS_006 with Port have tag: BLN_PORT_048 by EB tresos, Pin configuration was enable LK bit Preconditions: Generate configuration MSCR value in HL need to be correct with EB config Test Case ID (internal TC that caught the defect) optional: Port_TC_WBT_0018 Observed behavior: MSCR value was not enable value LK in Port_PBcfg, although this pin was enable in Port_Ci_Port_Ip_VS_0_PBcfg. !image-2021-07-05-15-46-42-671.png! !image-2021-07-05-15-47-02-779.png! !image-2021-07-05-15-48-51-979.png! Expected behavior: Driver need to be update MACRO GetMSCR with LK reg (don't have update field for LK) Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-13154	New Feature	New Feature [DIO] The Exclusive Areas should be reviewed and updated for K1 platform ,,"NewWorkDescription: The Exclusive Areas for K1 platform are incorrect. Some of them are redundant (need to be removed), and the rest need to be renamed. Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Remove/Add/Renaming the Exclusive Areas in code for K1. From IPW to IPL. All exclusives should be existed in IPL. The ones in IPW should be removed. #include SchM_Dio.h should be moved from IPW to IPL Update the UM/IM if needed"
ARTD-13166	Bug	[ADC] Number of external channels is not correct on some derivatives<*> Detailed description (how to reproduce it): There are different number of external channels on each derivatives of K1xx Also the external channels on the same derivative is not identical also Should check again IOMux and SDK legacy for reference Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: number of channels are not correct Expected behavior: number of channels are not correct for all derivatives Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]

ID	Subtype	Headline and Description
ARTD-13165	Bug	[CAN] Controller remains bus-off state and cannot participate on CAN bus after restarting engine<*> Detailed description (how to reproduce it): After Bus-off event occurred, ESR1[FLTCONF] will be set to 11b (indicates bus-off state), Can_MainFunction_Busoff() or bus-off ISR will set the CAN controller to stop mode. Call Can_SetControllerMode(CAN_CS_STARTED) to exit bus-off state and starting transmit a new message but the data cannot be transmitted. ESR1[FLTCONF] still remains 11b even Can_SetControllerMode(CAN_CS_STARTED) has been executed successfully. !image-2021-07-06-11-16-02-886.png|width=1306,height=695! I have tried to request soft reset by asserting MCR[SOFTRST] before set controller to start mode and then the data can be transmitted successfully. It means that we should have a soft reset in restart the CAN controller after bus-off event occurred (as known is manual recover bus-off). Preconditions: Don't use auto bus-off recovery. Test Case ID (internal TC that caught the defect) optional: Can_TC_FCT_1012 Observed behavior: NA Expected behavior: NA Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-13169	Bug	[CAN] ERR050443 errata implementation should be applied for K1<*> Detailed description (how to reproduce it): https://jira.sw.nxp.com/browse/ARTD-13013 as the errata documentation on above ticket, ERR050443 was implemented for other platform, and now it should be enabled for K1. !image-2021-07-06-14-23-03-950.png! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: ERR050443: rx mailboxes can be inactivated in specific case and impacting to functionality of driver Expected behavior: ERR050443 implemetation should be applied for K1 Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-13199	Bug	[Uart] Flexio Uart Can not generate Declare callback functions in EB Tresos<*> Detailed description (how to reproduce it):

ID	Subtype	Headline and Description
		[When configure function callback for FlexIO channel, Declare callback functions cannot generate into header files output of Ipw layer (Uart_Ipw_PBcfg.h) which generated by EB Tresos] Preconditions: [N/A] Test Case ID (internal TC that caught the defect) optional: [Uart_TS_3006] Observed behavior: [Building failed because Flexio Callback is use but not declare error occurred] Expected behavior: [Flexio Callback function is declared on Ipw layer in Uart_Ipw_PBcfg.h file and Building success] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-13344	Bug	[LIN] Driver don't go IDLE state after function Lin_Wakeup called with IPV FLEXIO<*> Detailed description (how to reproduce it): Driver don't go IDLE state after function Lin_Wakeup called with IPV FLEXIO Preconditions: After Wakeup signal was transmitted driver need go to function Flexio_Lin_Ip_WakeupInterruptHandler to handler wakeup interrupt. In this function driver check: if FALSE == LinCurrentState->IsBusBusy then function Flexio_Lin_Ip_GotoIdleState(Channel) is called. But in this time, driver have value LinCurrentState->IsBusBusy is TRUE. Thus, after Wakeup signal was transmitted driver still in SLEEP state. Test Case ID (internal TC that caught the defect) optional: Lin_TC_FCT_0201 Observed behavior: After Wakeup driver have status is SLEEP state Expected behavior: After Wakeup driver have status is OPERATION state Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Call function Flexio_Lin_Ip_GotoIdleState after Wakeup signal was transmitted.
ARTD-13713	Bug	[GPT] Correct and update Srtc_Ip_ConfigureLockReg API<*> Detailed description (how to reproduce it): when user ticks Enable Register Lock API button on Design Studio interface to use function Srtc_Ip_ConfigureLockRegister, macro value SRTC_IP_ENABLE_LOCK_REGISTER_API was generated = STD_OFF At the same time many other errors are generated: "d:/Workspace/Git_work/S32K1XX_4.4/output/S32K1XX_S32K144W_ghs/gpt/../../eclipse/plugins/Gpt_TS_T40D2M8I0R0/src/SRtc_Ip.c", line 651: warning #223-D: function Srtc_Ip_ConfigureStatuskRegLock declared implicitly status = Srtc_Ip_ConfigureStatuskRegLock(base);(Wrong name function) "d:/Workspace/Git_work/S32K1XX_4.4/output/S32K1XX_S32K144W_ghs/gpt/../../eclipse/plugins/Gpt_TS_T40D2M8I0R0/src/SRtc_Ip.c", line 1405: error #44: expression must have pointer type if (lockRegister->controlRegisterLock) "d:/Workspace/Git_work/S32K1XX_4.4/output/S32K1XX_S32K144W_ghs/gpt/../../eclipse/plugins/Gpt_TS_T40D2M8I0R0/src/SRtc_Ip.c", line 1410: error #44:

ID	Subtype	Headline and Description
		expression must have pointer type if (lockRegister->statusRegisterLock) "d:/Workspace/Git_work/S32K1XX_4.4/output/S32K1XX_S32K144W_ghs/gpt/../../eclipse/plugins/Gpt_TS_T40D2M8I0R0/src/SRtc_Ip.c", line 1415: error #44: expression must have pointer type if (lockRegister->timeCompensationRegisterLock) "d:/Workspace/Git_work/S32K1XX_4.4/output/S32K1XX_S32K144W_ghs/gpt/../../eclipse/plugins/Gpt_TS_T40D2M8I0R0/src/SRtc_Ip.c", line 1420: error #44: expression must have pointer type if (lockRegister->lockRegisterLock) redesigned Srtc_Ip_ConfigureLockReg function: we need to Check if the Lock Register Lock (LRL) is locked, if it is, any other register lock status cannot edited. If the Lock Register Lock(LRL) register is not locked, we can configure the entire lock register Status Register Lock(SRL) Control Register Lock(CRL) TCL(Time Compensation Lock) Lock Register Lock(LRL) when the user does not tick the button Enable compensation but the value marco SRTC_IP_ENABLE_COMPENSATION_SUPPORT =STD_ON Preconditions: use the Enable Register Lock API button on the DS interface user does not tick the button Enable compensation Test Case ID (internal TC that caught the defect) optional: Ip_SRtc_TC_FCT_0007 Observed behavior: Marco SRTC_IP_ENABLE_LOCK_REGISTER_API always generated =STD_OFF even if it is enabled on DS interface Expected behavior: Marco SRTC_IP_ENABLE_LOCK_REGISTER_API always generated =STD_ON if it is enabled on DS interface Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: check the SRTC_IP_ENABLE_LOCK_REGISTER_API value generated on the DS interface and other errors arising The lockRegister input parameter of the Srtc_Ip_ConfigureLockRegister function needs to be converted to a pointer
ARTD-13715	Bug	[CAN] XOR vs in-range in CanIcomSignalOperation<*> Detailed description (how to reproduce it): !image-2021-07-07-10-50-02-058.png! !image-2021-07-07-10-47-32-582.png! !image-2021-07-07-10-47-59-928.png! Currently, when XOR is chosen on EB, driver will generate configuration as inside-range of payloads. I see two definitions are not identical. Please reanalysis this feature B> please add a note for bytes-order for configuring CanIcomSignalValue C> payload length error is not supported ? !image-2021-07-07-17-10-08-481.png! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional:

ID	Subtype	Headline and Description
		None because XOR behavior could not be tested now Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-13719	Bug	[S32K1 RTM][FEE] fix misra violation<*> Detailed description (how to reproduce it): misra violation folow file excel attacted Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: misra violation fixed Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-13714	Bug	[GPT] Implement errata workaround for ERR010716<*> NewWorkDescription: implement ERR010716: RTC: Timer Alarm Flag can assert erroneously Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: in base need define ERR_IPV_SRTC_E05010716. Workaround in code : Write the Time Alarm Register (RTC_TAR) when the RTC Seconds Register is not incrementing. This can be when Time Counter Enable (RTC_SR[TCE]) bit in the RTC Status Register is clear or within the RTC_SR[TAF] interrupt routine. Alternatively, if the RTC_SR[TAF] is asserted following a write to the RTC_TAR, then write the RTC_TAR again.
ARTD-13716	New Feature	New Feature [BASE][ERR010716] Enable errata workaround define ,,"Add definition for enabling ERR010716. Expected define should look like: #define ERR_IPV_RTC_ERR010716 (STD_ON) Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior:

ID	Subtype	Headline and Description
		NA Expected behavior: NA Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: add define for enabling ERR010716; architecture document (section 3.3.12)"
ARTD-13730	Bug	 [CAN] wrong mask for clearing bit field in FlexCAN_SetPNDlcFilter<*> Detailed description (how to reproduce it): !image-2021-07-07-15-37-49-263.png! expected behavior: tmp &= (*FLEXCAN_FLT_DLC_FLT_DLC_LO_MASK FLEXCAN_FLT_DLC_FLT_DLC_HI_MASK); Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-13736	Bug	 [CAN] update description for implementation that does not follow exactly autosar srs<*> Detailed description (how to reproduce it): A>* The description for CanIcomSignalOperation is very spare when comparing to can-spec's one !image-2021-07-07-17-17-22-389.png! expectation: **_the functional description should be stated clearly in um/im (as the one in specification) !image-2021-07-07-17-20-20-696.png! B>* **_Current implementation does not follow exactly autosar srs: autosar: +support '*strictly greater/smaller' mask is always be supported driver: support 'greater/smaller or equal can not* support mask in two cases => the description is totally required, also a constraint for CanIcomSignalMask can be implemented (disable this node when it is not used) !image-2021-07-07-17-27-00-106.png! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior:

ID	Subtype	Headline and Description
		[...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-13734	Bug	[WDG] Ip_Wdog Node 'Wdog Disable Allowed' in S32CT can't prevent user disable module at runtime<*> Detailed description (how to reproduce it): Configure node 'Wdog Disable Allowed' = false in S32CT (IPL) and try to disable Wdog at runtime. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: Ip_Wdog_TS_COT_001 Observed behavior: When user configure node 'Wdog Disable Allowed' = false, macro WDOG_IP_DEINIT = OFF are generated. But no where in driver use this macro to prevent user from calling function Wdog_Ip_DeInit() to disable Wdog Expected behavior: Node 'Wdog Disable Allowed' in S32CT can prevent user from disabling Wdog Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-13770	Bug	[WDG] Must remove some unused wdg det error reports in generation file<*> Detailed description (how to reproduce it): Step 1: Run command clean generate any TS of WDG Step 2: Check code generate Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: Must remove some unused wdg det error reports in generation file Expected behavior: Remove some unused wdg det error reports in generation file Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-13807	New Feature	New Feature [SPI] Add support for half duplex mode on CT ,,"NewWorkDescription: Add support for half duplex mode on CT Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional:

ID	Subtype	Headline and Description
		[...]"
ARTD-13815	Bug	[PWM] Implement errata workaround for ERR010856<*> Detailed description (how to reproduce it): Implement the workaround for errata. ERR010856 FTM: Safe state is not removed from channel outputs after fault condition ends if SWOCTRL is being used to control the pin Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: NA Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: base need to add the macro definition : ERR_IPV_FTM_E010856. Workaround in code: If fault control is enabled while the software output control register is also being used (FTM_SWOCTRL), then the FTM should be configured as follows: FTM_MODE[FAULTM] configured for manual fault clearing (0b10) For devices that include the FTM_CONF[NUMTOF] field, it must be cleared to 0b00000 (TOF set for each counter overflow). For FTM versions that don't include the FTM_CONF[NUMTOF] field this doesn't apply. The procedure below must be used in the TOF interrupt handler when a fault is detected to ensure that the outputs return to the value configured by FTM_SWOCTRL. 1. Check the value of FTM_FMS[FAULTF]. If FTM_FMS[FAULTF] = 1 (fault occurred or is occurring), then set a variable to indicate that a fault was detected and continue to step 2. If FTM_FMS[FAULTF] = 0 but the fault variable is set (fault is not active, but was previously detected), continue to step 6. 2. Write the FTM_OUTMASK register to set the bit or bits corresponding to any channels that are controlled by FTM_SWOCTRL to temporarily inactivate the channel output. 3. Clear fault conditions by reading the FTM_FMS register and then writing FTM_FMS with all zeroes. 4. Clear the FTM_SC[TOF] bit by reading the FTM_SC register, then writing a 0 to FTM_SC[TOF]. 5. Exit the interrupt handler to skip following steps (they will execute the next time the TOF handler is called). 6. Clear the FTM_SWOCTRL by writing all zeroes to it. 7. Write FTM_SWOCTRL with the desired value again. 8. Clear the FTM_OUTMASK bits that were set in step 2. 9. Clear the fault variable that was set in step 1 when the fault condition was originally detected. 10. Clear the FTM_SC[TOF] bit by reading the FTM_SC register, then writing a 0 to FTM_SC[TOF].
ARTD-13811	Bug	

ID	Subtype	Headline and Description
		[ETH] Some functions for IPL are missing input parameter validation<*> Detailed description (how to reproduce it): Some functions for IPL are missing input parameter validation (ENET_DEV_ASSERT) Enet_Ip_TimerGet Enet_Ip_ReadTimerValue Enet_Ip_SetSysTimeCorr Enet_Ip_ConfigCounters Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-13816	Bug	[ETH] Promiscuous mode shall be enabled if the HW does not support filtering<*> Detailed description (how to reproduce it): In the current implement for Eth_UpdatePhysAddrFilter, the Promiscuous are not be enabled in case HW does not support that is not meet requirement SWS_Eth_00146 Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-13824	Bug	[S32K1XX][PORT] Port pin level value can be configured on CTHL without GPIO mode<*> Detailed description (how to reproduce it): Install S32DS 3.4 and EB Tresos 27.1.0 test_port uses tag PVT_TEST_PORT_S32K1XX_RTM_100_012, TS: Port_TS_004 to compare configuration between EB and CT Preconditions: Config ADC mode on CTHL then try to config PortPinvalue. It should not config able without GPIO mode Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: On EB tresos: !image-2021-07-08-17-45-13-790.png|width=399,height=216! On CTHL: !image-2021-07-08-17-46-57-666.png|width=364,height=284!

ID	Subtype	Headline and Description
		Expected behavior: No thing difference between EB and CTHL. PortPin Level Value on CT could be configurable with port pin mode is ADC Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-13829	Bug	[I2C] Driver code of i2c should follow the coding rule guideline<*> Detailed description (how to reproduce it): Driver code of i2c didn't follow the coding rule, refer code review checklist to see details Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: [...] Expected behavior: All ** violations should be fixed Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Fixing according code checklist at the document of i2c development repo
ARTD-13833	Bug	[PORT] Some function in IPL did not rise Error detection / reporting when Lock bit register enable in PCR<*> Detailed description (how to reproduce it): When check the operation of LK bit enable, the PCR was lock cannot modify. So some function in ipl effect to PCR register should rise error and block function. Preconditions: if LK bit enable in hardware,using any API try to modify PCR Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: according Radu's email and the discussion thread mail in attach file Expected behavior: Driver need to be update the error detect when user config with LK reg Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-13835	Bug	[LIN] Fix LPUART Auto baud rate feature<*> Detailed description (how to reproduce it): Fix LPUART Auto baud rate feature. Undefine Lpuart_Lin_Ip_InputCaptureType in extern callback Lpuart_Lin_Ip_Cfg.h Lpuart_Lin_Ip_InputCaptureType need to move to Lpuart_Lin_Ip_Types.h Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: Ip_Lpuart_Lin_TC_FCT_0021 Observed behavior: Fix LPUART Auto baud rate feature

ID	Subtype	Headline and Description
		Expected behavior: Fix LPUART Auto baud rate feature Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]

ARTD-13837	Bug	[LIN][DS][IP][LPUART] LinTimeoutMethod error when choose OSIF_COUNTER_SYSTEM<*>
		
		Detailed description (how to reproduce it):
		LinTimeoutMethod error when choose OSIF_COUNTER_SYSTEM.
		Lpuart_Lin_Ip_Hw_Access.c Lpuart_Lin_Ip_TimeoutExpired() is wrong, need to use ElapsedTime =
		Lpuart_Lin_Ip_Init No need to check timeout the case Enable Transmitter and Receiver > remove the check.
		Lpuart_Lin_Ip_Cfg.h LPUART_LIN_IP_NUMBER_OF_INSTANCES must be the used number of instances not the maximum number of instances.
		Lpuart_Lin_Ip_StatusType LPUART_LIN_IP_STATUS_TIMEOUT is not used.
		Lpuart_Lin_Ip.c Lpuart_Lin_Ip_Deinit can not return LPUART_LIN_IP_STATUS_SUCCESS:
		if (!TimeoutOccuredTx && !TimeoutOccuredRx)
		{
		RetStatus = LPUART_LIN_IP_STATUS_SUCCESS;
		}
		Preconditions:
		[...]
		Test Case ID (internal TC that caught the defect) optional:
		Ip_Lpuart_Lin_TC_FCT_0012
		Observed behavior:
		LinTimeoutMethod error when choose OSIF_COUNTER_SYSTEM
		Expected behavior:
		LinTimeoutMethod error when choose OSIF_COUNTER_SYSTEM
		Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)
		Proposed solution optional:
		[...]

ARTD-13847	Bug	[LIN] Add label fault so that the test can cover some requirement that can't be covered by normal test cases<*> Detailed description (how to reproduce it): We have some reqs: SWS_Lin_00097, SWS_Lin_00218, LPUART_LIN_013_001, FLEXIO_LIN_003_001 It can't be covered by normal test cases. Preconditions: Case cover this reqs is controlled by hardware. Test Case ID (internal TC that caught the defect) optional: Lin_TC_FLT_0101 Observed behavior: NA Expected behavior: NA Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:

ID	Subtype	Headline and Description
		Need add label fault MCAL_FAULT_INJECTION_POINT(LPUART_FIP_1_T_TIME_OUT_1) to function Lpuart_Lin_Ip_AbortTransferData and MCAL_FAULT_INJECTION_POINT(FLEXIO_FIP_1_T_TIME_OUT_1) to function Flexio_Lin_Ip_AbortTransferData.
ARTD-13887	Bug	[crypto] Fix and comment static analysis violations (HIS metric)<*> Create HIS report. Fix all violations if any are found. Any violation cannot fix, need to communicate with PMs, Project Tech lead, Product Architect as soon. RTM Target: !image-2021-07-12-16-27-49-714.png! Guideline following: [[https://confluence.sw.nxp.com/pages/viewpage.action?pageId=133734230] [https://confluence.sw.nxp.com/display/AUTORD/Coverity+HIS]
ARTD-13894	Bug	[i2c] Fix and comment static analysis violations (HIS metric)<*> Create HIS report. Fix all violations if any are found. Any violation cannot fix, need to communicate with PMs, Project Tech lead, Product Architect as soon. RTM Target: !image-2021-07-12-16-27-49-714.png! Guideline following: [[https://confluence.sw.nxp.com/pages/viewpage.action?pageId=133734230] [https://confluence.sw.nxp.com/display/AUTORD/Coverity+HIS]
ARTD-13901	Bug	[port] Fix and comment static analysis violations (HIS metric)<*> Create HIS report. Fix all violations if any are found. Any violation cannot fix, need to communicate with PMs, Project Tech lead, Product Architect as soon. RTM Target: !image-2021-07-12-16-27-49-714.png! Guideline following: [[https://confluence.sw.nxp.com/pages/viewpage.action?pageId=133734230] [https://confluence.sw.nxp.com/display/AUTORD/Coverity+HIS]
ARTD-13907	Bug	[WDG] Fix and comment static analysis violations (HIS metric)<*> Create HIS report. Fix all violations if any are found. Any violation cannot fix, need to communicate with PMs, Project Tech lead, Product Architect as soon. RTM Target: !image-2021-07-12-16-27-49-714.png! Guideline following: [[https://confluence.sw.nxp.com/pages/viewpage.action?pageId=133734230]

ID	Subtype	Headline and Description
		[https://confluence.sw.nxp.com/display/AUTORD/Coverity+HIS]

ARTD-13861	Bug	[SAI] Wrong ServiceId in DetErrorReport for Sai_AsyncTransmit function<*>
		
		Detailed description (how to reproduce it):
		Wrong ServiceId in DetErrorReport for Sai_AsyncTransmit function
		Driver reports error for SAI_SYNCTRANSMIT_ID when using Sai_AsyncTransmit.
		It should be SAI_ASYNCTRANSMIT_ID
		Std_ReturnType Sai_AsyncTransmit(Sai_LogicalChannel LogicChn,Sai_RequestType * pRequest)
		{
		...
		case (Std_ReturnType)E_NOT_OK:
		(void)Det_ReportError((uint16)SAI_MODULE_ID, (uint8)0, (uint8)SAI_SYNCTRANSMIT_ID, (uint8)status);
		break;
		...
		}
		
		Preconditions:
		[...]
		Test Case ID (internal TC that caught the defect) optional:
		[...]
		Observed behavior:
		Wrong service ID for DET error report
		Expected behavior:
		[...]
		Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)
		Proposed solution optional:
		[...]

ARTD-13877	Bug	[LIN] Driver can't abort old frame when new frame is requested send.<*> Detailed description (how to reproduce it): Driver is MASTER node and in mode LIN_DISABLE_FRAME_TIMEOUT = ON*: In case driver send a header and expect a response from slave node. But slave will don't transmit any thing. So, driver will return status LIN_RX_NO_RESPONSE. In this time, I request a new frame, driver need abort old frame and transmit new frame. But status I received when call function Lin_SendFrame is E_NOT_OK. This bug applied with IPVs: LPUART and FLEXIO*. Preconditions: When the driver transmitted header and waiting response from slave (old frame). Value LinCurrentState->IsBusBusy still is TRUE, although no data is transmitting from the slave. With the current implementation of the driver, it only abort old frame when have data byte transmitting in bus. But with case slave don't send any thing, status return on function Lin_Ip_AbortTransferData is LPUART_LIN_IP_STATUS_ERROR. Thus, function Lin_SendFrame will return E_NOT_OK. !Untitled.png|width=1209,height=530! Test Case ID (internal TC that caught the defect) optional: Lin_TC_FCT_0209 Observed behavior: NA

ID	Subtype	Headline and Description
		Expected behavior: NA Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-13909	Bug	[LIN] Driver doesn't call functions LinLpuartStopTimerNotification or LinFlexioStopTimerNotification when abort the old frame.<*> Detailed description (how to reproduce it): Driver in mode* *LIN_DISABLE_FRAME_TIMEOUT = OFF**: Driver is transmitting a frame and a new frame is requested. Driver can return timeout error. Preconditions: When the driver is transmitting or receiving a frame and in mode *LIN_DISABLE_FRAME_TIMEOUT = OFF* the driver use a timer such as GPT to count time transmit or receive to make sure receive time of a frame not too long. But when a frame aborted for the purpose of send a new frame. Function LinLpuartStopTimerNotification or LinFlexioStopTimerNotification don't called. Thus, after aborted old frame driver can go to function handler timeout error if value count of GPT is over. Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: NA Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Call function LinLpuartStopTimerNotification when abort tranfer data of old frame.

ARTD-13910	New Feature	New Feature
		
		[S32K1XX] ERR009005 errata implementation can be applied for K1
		,,"NewWorkDescription:
		ERR009005: Core: Store immediate overlapping exception return operation might vector to incorrect interrupt
		Workaround suggest by errata
		: For software not using the memory protection unit, this erratum can be worked around by
		setting DISDEFWBUF in the Auxiliary Control Register.
		In all other cases, the erratum can be avoided by ensuring a DSB occurs between the store
		and the BX instruction. For exception handlers written in C, this can be achieved by inserting
		the appropriate set of intrinsics or inline assembly just before the end of the interrupt function,
		for example:
		ARMCC:
		...
		schedule_barrier();
		asm\{DSB};
		schedule_barrier();
		}

ID	Subtype	Headline and Description

		GCC:
		...
		asm volatile ("dsb 0xf" ::: "memory");
		}
		Requirement source:
		N/A
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		Proposed solution optional:
		errata require to add a data sync barrier before the end of interrupt function
		For Base please add definition for:
		#define EXIT_INTERRUPT() MCAL_DATA_SYNC_BARRIER() / DSB sy full system /
		For other module please add macro EXIT_INTERRUPT() at the end of isr handler, for example
		ISR(PIT_0_ISR)
		{
		uint8 instance = 0U;
		uint8 channel;
		uint32 temp = 0;
		for (channel = 0U; channel < PIT_0_CHANNELS_NUMBER; channel++)
		{
		temp = Pit_Ip_GetInterruptFlags(instance, channel);
		if (1U == temp)
		{
		Pit_Ip_ProcessCommonInterrupt(instance, channel);
		}
		}
		EXIT_INTERRUPT()
		}
		"

ARTD-13919	Bug	[GPT] GptChannelTickFrequency need auto calculate by the prescaler and McuClockReferencePointFrequency<*> Detailed description (how to reproduce it): GptChannelTickFrequency can not calculate correct the frequency if have 2 instance of ipv (like FTM STM Emios) Preconditions: always get the prescaler of the 1st instance of the list. Test Case ID (internal TC that caught the defect) optional: GPT_TS_001 Observed behavior: GptChannelTickFrequency incorrect Expected behavior: Auto calculate the GptChannelTickFrequency by the prescaler and the mcu reference Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: need correct the node GptChannelTickFrequency to calculate the frequency
ARTD-13929	Bug	[LIN][DS][IP][FLEXIO] Error when install an user callback<*> Detailed description (how to reproduce it): When using the LIN FLEXIO module, there is an error when install an user callback Preconditions: [...] Test Case ID (internal TC that caught the defect) optional:

ID	Subtype	Headline and Description
		Ip_Lpuart_Lin_TC_FCT_0011 Observed behavior: FLEXIO error when install an user callback Expected behavior: For Lin FLEXIO ds component, an user callback should be able to configure without errors. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-13930	Bug	[CAN] sw version should be hardcode and reflect plugins version instead the user's configuration<*> !image-2021-07-13-10-20-32-729.png! This issue only happen on CT / HLD (tests for EB or CT / IPL are ok). This comes from following implementation, the sw version now is being interpreted as the info from .mex: The sw can be changed and the old .mex should be re-used without buildfail error (if the .mex is obsolete => driver should report generate-error instead buildfail-error, no invalid generated code should be allowed) The sw version should refect the version of plugins/driver itself instead of the version of user's .mex file. Expectation: the implementation for HLD should be as similar the the IPL's one HLD/CT: !image-2021-07-13-10-21-39-339.png! IPL/CT: !image-2021-07-13-10-28-06-698.png!
ARTD-13951	Bug	[CRYPTO] Remove descriptor feeding feature code for the platforms without HSE firmware<*> Detailed description (how to reproduce it): At build time for S32K1XX platform warnings will be reported for {color:#172b4d}CRYPTO_ENABLE_FEED_HSE_DESC_SUPPORT macro because it is not defined. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: CRYPTO_ENABLE_FEED_HSE_DESC_SUPPORT is undefined. Expected behavior: CRYPTO_ENABLE_FEED_HSE_DESC_SUPPORT usage and the code incapsulated by the macro will be removed from plugin code. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: CRYPTO_ENABLE_FEED_HSE_DESC_SUPPORT usage and the code incapsulated by the macro should be removed from plugin code with M4 tags.
ARTD-13955	Bug	[MCL]FTM_MCL still occurs hardfault error in user mode<*> Detailed description (how to reproduce it): Run MC_TS_500

ID	Subtype	Headline and Description
		Hardfault error happened > due to FTM is not support user mode on K1XX Test Case ID (internal TC that caught the defect) optional: Mcl_TC_FCT_0501 Observed behavior: Hardfault error occured Expected behavior: FTM driver works well
ARTD-13966	Bug	[CRYPTO]Exclusive Area analysis and update<*> Detailed description (how to reproduce it): Exclusive Areas analysis was not done for S32K1XX. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Missing Exclusive Areas analysis Expected behavior: Exclusive Areas report should be present and EAs should be added in code if necessary. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: The following should be done on this ticket: 1. Create RTD_CRYPTO_EXCLUSIVE_AREAS.xlsx document 2. Based on analysis document add in code necessary Exclusive Areas. 3. Update in IM the chapter "Exclusive areas to be defined in BSW scheduler" An example of Exclusive Areas report was attached.
ARTD-13970	Bug	[PORT] Function Port_Ci_Port_Ip_SetGlobalPinControl did not set LK bit<*> Detailed description (how to reproduce it):*_ Declare function: Port_Ci_Port_Ip_SetGlobalPinControl(PORT_Type *const *base*, uint16 *pins*, uint16 0x00008103*, Port_Ci_Port_Ip_PortGlobalControlPins *halfPort*) Try to set LK bit on PCR register then check the value of PCR of pins (multi pins was chosen) Preconditions: all pin have same configuration. Test Case ID (internal TC that caught the defect) optional: IP_Port_Ci_TC_0001 Observed behavior: PCR's value of multi pins after set = 0x00000103 ==> bit LK not set to enable or disable in this function. Expected behavior: Update Port_Ci_Port_Ip_SetGlobalPinControl for setting all bit support in PCR PCR's value should be: 0x00008103. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-13979	Bug	

ID	Subtype	Headline and Description
		[EEP] Eep should not accept to be reinitialized while a job is pending<*> Detailed description (how to reproduce it): Call Eep_Init(). Preconditions: Eep is correctly initialized and a job is pending. Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: Eep is reinitialized. Expected behavior: Det EEP_E_BUSY is raised. Eep should only accept a reinitialization when Eep_GetStatus returns MEMIF_IDLE. SWS_Eep_00000: API service called while driver still busy Development EEP_E_BUSY Proposed solution optional: Reinitialize the global variables only after verifying that Eep_eJobResult is not MEMIF_JOB_PENDING.
ARTD-13982	Bug	[S32K1XX][S32KXX_100] S32CT interface can't config beacause irq interrupt was read only<*> Detailed description (how to reproduce it): Creating new project on s32ct add platform component to project User can't not config on Generic interrupt settings interface Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: Platfrom_TS_CT_001 Observed behavior: !image-2021-07-14-15-53-07-972.png! Expected behavior: N/A Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Can config ireq interrupt
ARTD-13974	Bug	[Uart][FlexIO] Need to update convert error status for Flexio<*> Detailed description (how to reproduce it): [Cannot get RX_OVERUN when use Flexio !image-2021-07-14-14-55-51-432.png!] Preconditions: [NA] Test Case ID (internal TC that caught the defect) optional: [Uart_TC_WBT_0006] Observed behavior: [NA] Expected behavior: [Update convert status for UART_RX_OVERUN when use FlexIO on HLD] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [None]
ARTD-13975	Bug	

ID	Subtype	Headline and Description
		[FEE] Fix compiler warnings on IAR compiler<*> Detailed description (how to reproduce it): See the attached file for more details Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: iar warnings Expected behavior: no warnings Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-13983	Bug	 [Platform] Compiler Warnings in Platform driver<*> Detailed description (how to reproduce it): While creating the example for the release, I saw there are some compiler warnings in Platform driver {code:c} Building file: ../RTD/src/OsIf_Timer.c ../RTD/src/IntCtrl_Ip.c: In function 'IntCtrl_Ip_InstallHandlerPrivileged': ../RTD/src/IntCtrl_Ip.c:196:6: warning: "INT_CTRL_IP_CORTEXR" is not defined, evaluates to 0 [-Wundef] 196 #if (INT_CTRL_IP_CORTEXR == STD_ON) ../RTD/src/IntCtrl_Ip.c: In function 'IntCtrl_Ip_EnableIrqPrivileged': ../RTD/src/IntCtrl_Ip.c:247:6: warning: "INT_CTRL_IP_CORTEXR" is not defined, evaluates to 0 [-Wundef] 247 #if (INT_CTRL_IP_CORTEXR == STD_ON) ../RTD/src/IntCtrl_Ip.c: In function 'IntCtrl_Ip_DisableIrqPrivileged': ../RTD/src/IntCtrl_Ip.c:262:5: warning: "INT_CTRL_IP_CORTEXR" is not defined, evaluates to 0 [-Wundef] 262 #if(INT_CTRL_IP_CORTEXR == STD_ON) ../RTD/src/IntCtrl_Ip.c: In function 'IntCtrl_Ip_ClearPendingPrivileged': ../RTD/src/IntCtrl_Ip.c:339:6: warning: "INT_CTRL_IP_CORTEXR" is not defined, evaluates to 0 [-Wundef] 339 #if (INT_CTRL_IP_CORTEXR == STD_ON) ../RTD/src/IntCtrl_Ip.c: In function 'IntCtrl_Ip_Group1Privileged': ../RTD/src/IntCtrl_Ip.c:356:6: warning: "INT_CTRL_IP_CORTEXR" is not defined, evaluates to 0 [-Wundef] 356 #if (INT_CTRL_IP_CORTEXR == STD_ON) ../RTD/src/IntCtrl_Ip.c: In function 'IntCtrl_Ip_ConfigureSpiPrivileged': ../RTD/src/IntCtrl_Ip.c:378:6: warning: "INT_CTRL_IP_CORTEXR" is not defined, evaluates to 0 [-Wundef] 378 #if (INT_CTRL_IP_CORTEXR == STD_ON) ../RTD/src/IntCtrl_Ip.c:369:18: warning: unused variable 'tmp' [-Wunused-variable] 369 uint32 bank, tmp; ../RTD/src/IntCtrl_Ip.c:369:12: warning: variable 'bank' set but not used [-Wunused-but-set-variable] 369 uint32 bank, tmp; ../RTD/src/IntCtrl_Ip.c: In function 'IntCtrl_Ip_SetIntTargetPrivileged': ../RTD/src/IntCtrl_Ip.c:393:6: warning: "INT_CTRL_IP_CORTEXR" is not defined, evaluates to 0 [-Wundef] 393 #if (INT_CTRL_IP_CORTEXR == STD_ON)

ID	Subtype	Headline and Description
		../RTD/src/IntCtrl_Ip.c:386:104: warning: unused parameter 'target' [-Wunused-parameter] 386 static inline void IntCtrl_Ip_SetIntTargetPrivileged(IRQn_Type eIrqNumber, IntCtrl_Ip_Routing_ModeType target) ../RTD/src/IntCtrl_Ip.c: In function 'IntCtrl_Ip_SetPendingPrivileged': ../RTD/src/IntCtrl_Ip.c:407:6: warning: "INT_CTRL_IP_CORTEXR" is not defined, evaluates to 0 [-Wundef] 407 #if (INT_CTRL_IP_CORTEXR == STD_ON) ../RTD/src/IntCtrl_Ip.c: In function 'IntCtrl_Ip_GetPendingPrivileged': ../RTD/src/IntCtrl_Ip.c:424:6: warning: "INT_CTRL_IP_CORTEXR" is not defined, evaluates to 0 [-Wundef] 424 #if (INT_CTRL_IP_CORTEXR == STD_ON) Preconditions: NA Test Case ID (internal TC that caught the defect) optional: Adc_Example Observed behavior: Some warnings appeared while building the example Expected behavior: All warnings will be gone Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-13984	Bug	[fee] Fix misra violations<*> Fix misra violations if any are found.
ARTD-14018	Bug	[S32K1XX]hardfault when using sys_registerIsrHandler / usermode / K116/k118<*> Detailed description (how to reproduce it): !image-2021-05-26-15-08-28-570.png! Preconditions: Call OsIf_ResumeAllInterrupts function in the user mode support Test Case ID (internal TC that caught the defect) optional: test case use interrupt functional Observed behavior: harfault when using sys_registerIsrHandler / usermode / K116/k118 Expected behavior: Not hard fault Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: need to resume all interrupts on S32K11X and not get hardfault when running project with user mode support
ARTD-14026	New Feature	New Feature [BASE][ERR010856] Enable errata workaround define ,,"Add definition for enabling ERR010856. Expected define should look like: #define ERR_IPV_FTM_ERR010856 (STD_ON) Preconditions: NA

ID	Subtype	Headline and Description
		Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: NA Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: add define for enabling ERR010856; architecture document (section 3.3.12)"
ARTD-14028	Bug	[S32K1 RTM] Crypto: "Crypto_Ipw_ProcessAsyncCsecResponse" is undefined when no configure key ,,"Detailed description (how to reproduce it): ""Crypto_Ipw_ProcessAsyncCsecResponse"" is undefined when no configure key Preconditions: CRYPTO_074 Test Case ID (internal TC that caught the defect) optional: Crypto_TS_Eq_Cot_01 Observed behavior: !image-2021-07-15-14-10-49-418.png! Expected behavior: NA Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-14031	New Feature	New Feature [S32K1 RTM] SPI: Improve IPL code to improve decision coverage when using DevAssert() ,,"NewWorkDescription: As IPL function below, we can not test with cases which go into conditions such as ExternalDevice=NULL_PTR. So, we need to change the implement way for these sketch. !image-2021-07-15-15-19-01-921.png! Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: [...]"
ARTD-14035	Bug	[Platform] Support feature retention for RAM<*> Detailed description (how to reproduce it): RAM should be enable when start code read SRAMU_RETEN and SRAML_RETEN in the Chip Control register (CHIPCTL) and write 1 to them to allow accesses to SRAM. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...]

ID	Subtype	Headline and Description
		Observed behavior: [...] Expected behavior: RAM should be enable when start code Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-14042	Bug	[EEP]Bulid fail when configure inconsistent config relate to enable compare feature<*> Detailed description (how to reproduce it): configure "*EepCompareApi*" node equal to true and "*EepFtfcEnableCompareApi*" node equal to false then build project with basic sequence erase-->write-->compare and read operation > result: false build !image-2021-07-15-18-09-58-531.png|thumbnail! Preconditions: Eep is correctly initialized Test Case ID (internal TC that caught the defect) optional: Eep_TC_COT_0001.c Observed behavior: Fail build Expected behavior: both of two node should be same value(true or false) Proposed solution optional: N/A
ARTD-14043	New Feature	New Feature [BASE]GetcoreID should be simplified in case multicore support is off ,,"NewWorkDescription: Getcoreid function has time execution is extremely high, this is potential problem. After analyze, i recognize that GetcoreID alway invoke context switch to request supervisor mode. Please optimize Getcore_Id api in this case. Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Add define for macro OsIf_GetCoreID incase multicore support is set alway return as zero #define OsIf_GetCoreID() (0U) "
ARTD-14118	Bug	[ADC] Need to add constraint for sw trigger, streaming group and AdcGroupConversionMode is ONESHOT as AUTOSAR document ,,"Detailed description (how to reproduce it): Need to add constraint for sw trigger, streaming group and AdcGroupConversionMode is ONESHOT as AUTOSAR document Preconditions: AdcGroupTriggSrc: ADC_TRIGG_SRC_SW AdcGroupAccessMode: ADC_ACCESS_MODE_STREAMING AdcGroupConversionMode: ADC_CONV_MODE_ONESHOT Test Case ID (internal TC that caught the defect) optional:

ID	Subtype	Headline and Description
		Adc_TS_COT_011 cfg 8 Observed behavior: Generate completely when configuring: AdcGroupTriggSrc: ADC_TRIGG_SRC_SW AdcGroupAccessMode: ADC_ACCESS_MODE_STREAMING AdcGroupConversionMode: ADC_CONV_MODE_ONESHOT Expected behavior: Generate fail because sw trigger group using streaming access mode can run only with continuous conversion mode as mention in chapter 7.3 State Diagrams of adc specification Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-14128	Bug	[ADC] Build fail: "undefined reference to `Adc_Ipw_Adc0EndConversionNotification'" when all AdcInterruptEnable nodes are disable ,,"Detailed description (how to reproduce it): Build fail: ""undefined reference to `Adc_Ipw_Adc0EndConversionNotification'"" when all AdcInterruptEnable nodes are disable Preconditions: All AdcInterruptEnable nodes are disable Compiler gcc Test Case ID (internal TC that caught the defect) optional: Adc_TS_COT_011 cfg 32 Observed behavior: Build fail log: ""e:/S32K1XX/output/S32K1XX_S32K148_gcc/adc/Adc_TS_COT_011_cfg32/out/Adc_Ip_VS_0_PBcfg_c.o: (.mcal_const_cfg+0x20): undefined reference to `Adc_Ipw_Adc0EndConversionNotification'"" Expected behavior: Build done Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-14130	Bug	[EEP] Eep_Cancel does not reset the internal async job in IPW<*> Detailed description (how to reproduce it): Start a job which requires more than one MainFunction calls. After 1 MainFunction execution cancel the job. Request a new job. Preconditions: EEP_ASYNC_ERASE_OPERATIONS_ENABLED == STD_ON}} Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: The second job starts with the internal parameters of the last job. Expected behavior: The second job starts with the parameters received by the corresponding API. SWS_Eep_00215 The function Eep_Cancel shall cancel an ongoing EEPROM read, write, erase or compare job.

ID	Subtype	Headline and Description
		Proposed solution optional: Reset the internal job of Eep_IPW.c from Eep_Ipw_Cancel().

ARTD-14158	Bug	[SAI] FMEA Driver does not clear interrupt flag when the interrupt is spurious<*>
		
		Detailed description (how to reproduce it):
		[SAI] FMEA Driver does not clear interrupt flag the interrupt is spurious
		[CPR_RTD_00028] ISR shall check whether its respective interrupt status flag and interrupt enable flag are set; i.e. whether the interrupt is not spurious. If at least one of the flag is not set, then the interrupt is spurious and the ISR shall only clear interrupt status flag and return immediately.
		Test setup:
		The interrupt is spurious, case 3: Interrupt enable flag is clear and interrupt status flag is set
		Sai_Init // master mode
		sys_disableAllInterrupts()
		Sai_SyncTransmit() // send request
		// Clear the interrupt enable flag
		t_g_pSaiBase[t_u8SaiInsIdx]->TCSR &= ~(SAI_TCSR_FEIE_MASK
		SAI_TCSR_WSIE_MASK
		SAI_TCSR_SEIE_MASK
		SAI_TCSR_WSF_MASK
		SAI_TCSR_SEF_MASK
		SAI_TCSR_FEF_MASK);
		SAI0_IRQHandler() // Call the ISR directly
		/* [CPR_RTD_00028.sai] Verification Point: The interrupt status flag is clear */
		EU_ASSERT(0 == (t_g_pSaiBase[t_u8SaiInsIdx]->TCSR & SAI_TCSR_WSF_MASK));
		EU_ASSERT(0 == (t_g_pSaiBase[t_u8SaiInsIdx]->TCSR & SAI_TCSR_SEF_MASK));
		EU_ASSERT(0 == (t_g_pSaiBase[t_u8SaiInsIdx]->TCSR & SAI_TCSR_FEF_MASK));
		EU_ASSERT(0 == (t_g_pSaiBase[t_u8SaiInsIdx]->TCSR & SAI_TCSR_FWF_MASK));
		EU_ASSERT(0 == (t_g_pSaiBase[t_u8SaiInsIdx]->TCSR & SAI_TCSR_FRF_MASK));
		=> FAILED, the WSF, FWF, FRF flags were not cleared
		Preconditions:
		[...]
		Test Case ID (internal TC that caught the defect) optional:
		[...]
		Observed behavior:
		[...]
		Expected behavior:
		[...]
		Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)
		Proposed solution optional:
		[...]

ARTD-14165	Bug	[ADC] COCO flags of queued group are cleared inside ISR of previous group<*> Detailed description (how to reproduce it): ISR of group 0 starts conversions of group 1 which was in queue, but the COCO flags of group 1 are cleared in ISR of group 0 right after that. Preconditions: Dev test Adc_TS_006, cfg 2 & 3 Test Case ID (internal TC that caught the defect) optional: Adc_TC_DVT_0601.c Adc_TC_DVT_0602.c Observed behavior:

ID	Subtype	Headline and Description
		COCO flags of queued group are cleared before ISR of that group occurs Expected behavior: COCO flags are remained until ISR of that group is called Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Explicitly clearing all COCO flags when notification is called but no group is currently converting (spurious) if ((AdcIpState[_Instance_].Init == FALSE) (AdcIpState[_Instance_].ConversionCompleteNotification == NULL_PTR)) (spurious)
ARTD-14172	Bug	[UART] - ISR does not check if driver is initialized for Flexio Ip<*> Detailed description (how to reproduce it): Preconditions: Follow req CPR_RTD_00011.lin: ISR shall check whether its respective driver is initialized. If the driver is not initialized, the ISR shall only clear interrupt status flag and return immediately. Eventhough the Flexio Mcl Common Irq checks if the macro corresponding the Uart channels configured before the Flexio_Uart_Ip_Irq function is called, the Flexio_Uart_Ip_Init() may not have been initialized on the channel in use, This must be checked in the Flexio_Uart_Ip_IrqHandler. Lpuart driver approach can be applied. Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: Check if driver has been initialized for the channel in use. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Check driver state structure to be different than NULL_PTR

ARTD-14261	New Feature	New Feature
		
		[adc] Add EXIT_INTERRUPT() for ISR handler due to ERR009005 errata
		,,"NewWorkDescription:
		ERR009005: Core: Store immediate overlapping exception return operation might vector to incorrect interrupt
		Workaround suggest by errata
		: For software not using the memory protection unit, this erratum can be worked around by
		setting DISDEFWBUF in the Auxiliary Control Register.
		In all other cases, the erratum can be avoided by ensuring a DSB occurs between the store
		and the BX instruction. For exception handlers written in C, this can be achieved by inserting
		the appropriate set of intrinsics or inline assembly just before the end of the interrupt function,
		for example:
		ARMCC:
		...
		schedule_barrier();
		asm\{DSB};
		schedule_barrier();
		}

ID	Subtype	Headline and Description

		GCC:
		...
		asm volatile ("dsb 0xf" ::: "memory");
		}
		Requirement source:
		N/A
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		Proposed solution optional:
		errata require to add a data sync barrier before the end of interrupt function
		For Base please add definition for:
		#define EXIT_INTERRUPT() MCAL_DATA_SYNC_BARRIER() / DSB sy full system /
		For other modules please add macro EXIT_INTERRUPT() at the end of isr handler, for example
		ISR(PIT_0_ISR)
		{
		uint8 instance = 0U;
		uint8 channel;
		uint32 temp = 0;
		for (channel = 0U; channel < PIT_0_CHANNELS_NUMBER; channel++)
		{
		temp = Pit_Ip_GetInterruptFlags(instance, channel);
		if (1U == temp)
		{
		Pit_Ip_ProcessCommonInterrupt(instance, channel);
		}
		}
		EXIT_INTERRUPT();
		}
		"

ARTD-14262	New Feature	New Feature
		
		[can] Add EXIT_INTERRUPT() for ISR handler due to ERR009005 errata
		,,"NewWorkDescription:
		ERR009005: Core: Store immediate overlapping exception return operation might vector to incorrect interrupt
		Workaround suggest by errata
		: For software not using the memory protection unit, this erratum can be worked around by
		setting DISDEFWBUF in the Auxiliary Control Register.
		In all other cases, the erratum can be avoided by ensuring a DSB occurs between the store
		and the BX instruction. For exception handlers written in C, this can be achieved by inserting
		the appropriate set of intrinsics or inline assembly just before the end of the interrupt function,
		for example:
		ARMCC:
		...
		schedule_barrier();
		asm\{DSB};
		schedule_barrier();
		}
		GCC:
		...
		asm volatile ("dsb 0xf" ::: "memory");
		}

ID	Subtype	Headline and Description

		Requirement source:
		N/A
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		Proposed solution optional:
		errata require to add a data sync barrier before the end of interrupt function
		For Base please add definition for:
		#define EXIT_INTERRUPT() MCAL_DATA_SYNC_BARRIER() / DSB sy full system /
		For other modules please add macro EXIT_INTERRUPT() at the end of isr handler, for example
		ISR(PIT_0_ISR)
		{
		uint8 instance = 0U;
		uint8 channel;
		uint32 temp = 0;
		for (channel = 0U; channel < PIT_0_CHANNELS_NUMBER; channel++)
		{
		temp = Pit_Ip_GetInterruptFlags(instance, channel);
		if (1U == temp)
		Unknown macro: \{ Pit_Ip_ProcessCommonInterrupt(instance, channel); }
		}
		EXIT_INTERRUPT()
		}
		"

ARTD-14264	New Feature	
		New Feature
		
		[crypto] Add EXIT_INTERRUPT() for ISR handler due to ERR009005 errata
		,,"NewWorkDescription:
		ERR009005: Core: Store immediate overlapping exception return operation might vector to incorrect interrupt
		Workaround suggest by errata
		: For software not using the memory protection unit, this erratum can be worked around by
		setting DISDEFWBUF in the Auxiliary Control Register.
		In all other cases, the erratum can be avoided by ensuring a DSB occurs between the store
		and the BX instruction. For exception handlers written in C, this can be achieved by inserting
		the appropriate set of intrinsics or inline assembly just before the end of the interrupt function,
		for example:
		ARMCC:
		...
		schedule_barrier();
		asm\{DSB};
		schedule_barrier();
		}
		GCC:
		...
		asm volatile ("dsb 0xf" ::: "memory");
		}
		Requirement source:
		N/A
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		Proposed solution optional:
		errata require to add a data sync barrier before the end of interrupt function
		For Base please add definition for:

ID	Subtype	Headline and Description

		#define EXIT_INTERRUPT() MCAL_DATA_SYNC_BARRIER() / DSB sy full system /
		For other modules please add macro EXIT_INTERRUPT() at the end of isr handler, for example
		ISR(PIT_0_ISR)
		{
		uint8 instance = 0U;
		uint8 channel;
		uint32 temp = 0;
		for (channel = 0U; channel < PIT_0_CHANNELS_NUMBER; channel++)
		{
		temp = Pit_Ip_GetInterruptFlags(instance, channel);
		if (1U == temp)
		Unknown macro: \{ Pit_Ip_ProcessCommonInterrupt(instance, channel); }
		}
		EXIT_INTERRUPT()
		}
		"

ARTD-14267	New Feature	New Feature
		
		[eth] Add EXIT_INTERRUPT() for ISR handler due to ERR009005 errata
		,,"NewWorkDescription:
		ERR009005: Core: Store immediate overlapping exception return operation might vector to incorrect interrupt
		Workaround suggest by errata
		: For software not using the memory protection unit, this erratum can be worked around by
		setting DISDEFWBUF in the Auxiliary Control Register.
		In all other cases, the erratum can be avoided by ensuring a DSB occurs between the store
		and the BX instruction. For exception handlers written in C, this can be achieved by inserting
		the appropriate set of intrinsics or inline assembly just before the end of the interrupt function,
		for example:
		ARMCC:
		...
		schedule_barrier();
		asm\{DSB};
		schedule_barrier();
		}
		GCC:
		...
		asm volatile ("dsb 0xf" ::: "memory");
		}
		Requirement source:
		N/A
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		Proposed solution optional:
		errata require to add a data sync barrier before the end of interrupt function
		For Base please add definition for:
		#define EXIT_INTERRUPT() MCAL_DATA_SYNC_BARRIER() / DSB sy full system /
		For other modules please add macro EXIT_INTERRUPT() at the end of isr handler, for example
		ISR(PIT_0_ISR)
		{
		uint8 instance = 0U;

ID	Subtype	Headline and Description

		uint8 channel;
		uint32 temp = 0;
		for (channel = 0U; channel < PIT_0_CHANNELS_NUMBER; channel++)
		{
		temp = Pit_Ip_GetInterruptFlags(instance, channel);
		if (1U == temp)
		Unknown macro: \{ Pit_Ip_ProcessCommonInterrupt(instance, channel); }
		}
		EXIT_INTERRUPT()
		}
		"

ARTD-14270	New Feature	New Feature
		
		[gpt] Add EXIT_INTERRUPT() for ISR handler due to ERR009005 errata
		,,"NewWorkDescription:
		ERR009005: Core: Store immediate overlapping exception return operation might vector to incorrect interrupt
		Workaround suggest by errata
		: For software not using the memory protection unit, this erratum can be worked around by
		setting DISDEFWBUF in the Auxiliary Control Register.
		In all other cases, the erratum can be avoided by ensuring a DSB occurs between the store
		and the BX instruction. For exception handlers written in C, this can be achieved by inserting
		the appropriate set of intrinsics or inline assembly just before the end of the interrupt function,
		for example:
		ARMCC:
		...
		schedule_barrier();
		asm\{DSB};
		schedule_barrier();
		}
		GCC:
		...
		asm volatile ("dsb 0xf" ::: "memory");
		}
		Requirement source:
		N/A
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		Proposed solution optional:
		errata require to add a data sync barrier before the end of interrupt function
		For Base please add definition for:
		#define EXIT_INTERRUPT() MCAL_DATA_SYNC_BARRIER() / DSB sy full system /
		For other modules please add macro EXIT_INTERRUPT() at the end of isr handler, for example
		ISR(PIT_0_ISR)
		{
		uint8 instance = 0U;
		uint8 channel;
		uint32 temp = 0;
		for (channel = 0U; channel < PIT_0_CHANNELS_NUMBER; channel++)
		{
		temp = Pit_Ip_GetInterruptFlags(instance, channel);
		if (1U == temp)

ID	Subtype	Headline and Description

		Unknown macro: \{ Pit_Ip_ProcessCommonInterrupt(instance, channel); }
		}
		EXIT_INTERRUPT()
		}
		"

ARTD-14271	New Feature	New Feature
		
		[i2c] Add EXIT_INTERRUPT() for ISR handler due to ERR009005 errata
		,,"NewWorkDescription:
		ERR009005: Core: Store immediate overlapping exception return operation might vector to incorrect interrupt
		Workaround suggest by errata
		: For software not using the memory protection unit, this erratum can be worked around by
		setting DISDEFWBUF in the Auxiliary Control Register.
		In all other cases, the erratum can be avoided by ensuring a DSB occurs between the store
		and the BX instruction. For exception handlers written in C, this can be achieved by inserting
		the appropriate set of intrinsics or inline assembly just before the end of the interrupt function,
		for example:
		ARMCC:
		...
		schedule_barrier();
		asm\{DSB};
		schedule_barrier();
		}
		GCC:
		...
		asm volatile ("dsb 0xf" ::: "memory");
		}
		Requirement source:
		N/A
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		Proposed solution optional:
		errata require to add a data sync barrier before the end of interrupt function
		For Base please add definition for:
		#define EXIT_INTERRUPT() MCAL_DATA_SYNC_BARRIER() / DSB sy full system /
		For other modules please add macro EXIT_INTERRUPT() at the end of isr handler, for example
		ISR(PIT_0_ISR)
		{
		uint8 instance = 0U;
		uint8 channel;
		uint32 temp = 0;
		for (channel = 0U; channel < PIT_0_CHANNELS_NUMBER; channel++)
		{
		temp = Pit_Ip_GetInterruptFlags(instance, channel);
		if (1U == temp)
		Unknown macro: \{ Pit_Ip_ProcessCommonInterrupt(instance, channel); }
		}
		EXIT_INTERRUPT()
		}
		"

ID	Subtype	Headline and Description

ARTD-14273	New Feature	New Feature
		
		[lin] Add EXIT_INTERRUPT() for ISR handler due to ERR009005 errata
		,,"NewWorkDescription:
		ERR009005: Core: Store immediate overlapping exception return operation might vector to incorrect interrupt
		Workaround suggest by errata
		: For software not using the memory protection unit, this erratum can be worked around by
		setting DISDEFWBUF in the Auxiliary Control Register.
		In all other cases, the erratum can be avoided by ensuring a DSB occurs between the store
		and the BX instruction. For exception handlers written in C, this can be achieved by inserting
		the appropriate set of intrinsics or inline assembly just before the end of the interrupt function,
		for example:
		ARMCC:
		...
		schedule_barrier();
		asm\{DSB};
		schedule_barrier();
		}
		GCC:
		...
		asm volatile ("dsb 0xf" ::: "memory");
		}
		Requirement source:
		N/A
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		Proposed solution optional:
		errata require to add a data sync barrier before the end of interrupt function
		For Base please add definition for:
		#define EXIT_INTERRUPT() MCAL_DATA_SYNC_BARRIER() / DSB sy full system /
		For other modules please add macro EXIT_INTERRUPT() at the end of isr handler, for example
		ISR(PIT_0_ISR)
		{
		uint8 instance = 0U;
		uint8 channel;
		uint32 temp = 0;
		for (channel = 0U; channel < PIT_0_CHANNELS_NUMBER; channel++)
		{
		temp = Pit_Ip_GetInterruptFlags(instance, channel);
		if (1U == temp)
		Unknown macro: \{ Pit_Ip_ProcessCommonInterrupt(instance, channel); }
		}
		EXIT_INTERRUPT()
		}
		"

ARTD-14276	New Feature	New Feature [ocu] Add EXIT_INTERRUPT() for ISR handler due to ERR009005 errata ,,"NewWorkDescription: ERR009005: Core: Store immediate overlapping exception return operation might

ID	Subtype	Headline and Description

		vector to incorrect interrupt
		Workaround suggest by errata
		: For software not using the memory protection unit, this erratum can be worked around by
		setting DISDEFWBUF in the Auxiliary Control Register.
		In all other cases, the erratum can be avoided by ensuring a DSB occurs between the store
		and the BX instruction. For exception handlers written in C, this can be achieved by inserting
		the appropriate set of intrinsics or inline assembly just before the end of the interrupt function,
		for example:
		ARMCC:
		...
		schedule_barrier();
		asm\{DSB};
		schedule_barrier();
		}
		GCC:
		...
		asm volatile ("dsb 0xf" ::: "memory");
		}
		Requirement source:
		N/A
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		Proposed solution optional:
		errata require to add a data sync barrier before the end of interrupt function
		For Base please add definition for:
		#define EXIT_INTERRUPT() MCAL_DATA_SYNC_BARRIER() / DSB sy full system /
		For other modules please add macro EXIT_INTERRUPT() at the end of isr handler, for example
		ISR(PIT_0_ISR)
		{
		uint8 instance = 0U;
		uint8 channel;
		uint32 temp = 0;
		for (channel = 0U; channel < PIT_0_CHANNELS_NUMBER; channel++)
		{
		temp = Pit_Ip_GetInterruptFlags(instance, channel);
		if (1U == temp)
		Unknown macro: \{ Pit_Ip_ProcessCommonInterrupt(instance, channel); }
		}
		EXIT_INTERRUPT()
		}
		"

ARTD-14278	New Feature	New Feature [pwm] Add EXIT_INTERRUPT() for ISR handler due to ERR009005 errata ,,"NewWorkDescription: ERR009005: Core: Store immediate overlapping exception return operation might vector to incorrect interrupt Workaround suggest by errata : For software not using the memory protection unit, this erratum can be worked around by setting DISDEFWBUF in the Auxiliary Control Register.

ID	Subtype	Headline and Description

		In all other cases, the erratum can be avoided by ensuring a DSB occurs between the store
		and the BX instruction. For exception handlers written in C, this can be achieved by inserting
		the appropriate set of intrinsics or inline assembly just before the end of the interrupt function,
		for example:
		ARMCC:
		...
		schedule_barrier();
		asm\{DSB};
		schedule_barrier();
		}
		GCC:
		...
		asm volatile ("dsb 0xf" ::: "memory");
		}
		Requirement source:
		N/A
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		Proposed solution optional:
		errata require to add a data sync barrier before the end of interrupt function
		For Base please add definition for:
		#define EXIT_INTERRUPT() MCAL_DATA_SYNC_BARRIER() / DSB sy full system /
		For other modules please add macro EXIT_INTERRUPT() at the end of isr handler, for example
		ISR(PIT_0_ISR)
		{
		uint8 instance = 0U;
		uint8 channel;
		uint32 temp = 0;
		for (channel = 0U; channel < PIT_0_CHANNELS_NUMBER; channel++)
		{
		temp = Pit_Ip_GetInterruptFlags(instance, channel);
		if (1U == temp)
		Unknown macro: \{ Pit_Ip_ProcessCommonInterrupt(instance, channel); }
		}
		EXIT_INTERRUPT()
		}
		"

ARTD-14280	New Feature	New Feature [sai] Add EXIT_INTERRUPT() for ISR handler due to ERR009005 errata ,,"NewWorkDescription: ERR009005: Core: Store immediate overlapping exception return operation might vector to incorrect interrupt Workaround suggest by errata : For software not using the memory protection unit, this erratum can be worked around by setting DISDEFWBUF in the Auxiliary Control Register. In all other cases, the erratum can be avoided by ensuring a DSB occurs between the store and the BX instruction. For exception handlers written in C, this can be achieved by inserting the appropriate set of intrinsics or inline assembly just before the end of the interrupt function,

ID	Subtype	Headline and Description

		for example:
		ARMCC:
		...
		schedule_barrier();
		asm\{DSB};
		schedule_barrier();
		}
		GCC:
		...
		asm volatile ("dsb 0xf" ::: "memory");
		}
		Requirement source:
		N/A
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		Proposed solution optional:
		errata require to add a data sync barrier before the end of interrupt function
		For Base please add definition for:
		#define EXIT_INTERRUPT() MCAL_DATA_SYNC_BARRIER() / DSB sy full system /
		For other modules please add macro EXIT_INTERRUPT() at the end of isr handler, for example
		ISR(PIT_0_ISR)
		{
		uint8 instance = 0U;
		uint8 channel;
		uint32 temp = 0;
		for (channel = 0U; channel < PIT_0_CHANNELS_NUMBER; channel++)
		{
		temp = Pit_Ip_GetInterruptFlags(instance, channel);
		if (1U == temp)
		Unknown macro: \{ Pit_Ip_ProcessCommonInterrupt(instance, channel); }
		}
		EXIT_INTERRUPT()
		}
		"

ARTD-14281	New Feature	New Feature [spi] Add EXIT_INTERRUPT() for ISR handler due to ERR009005 errata ,,"NewWorkDescription: ERR009005: Core: Store immediate overlapping exception return operation might vector to incorrect interrupt Workaround suggest by errata : For software not using the memory protection unit, this erratum can be worked around by setting DISDEFWBUF in the Auxiliary Control Register. In all other cases, the erratum can be avoided by ensuring a DSB occurs between the store and the BX instruction. For exception handlers written in C, this can be achieved by inserting the appropriate set of intrinsics or inline assembly just before the end of the interrupt function, for example: ARMCC: ... schedule_barrier(); asm\{DSB}; schedule_barrier();

ID	Subtype	Headline and Description

		}
		GCC:
		...
		asm volatile ("dsb 0xf" ::: "memory");
		}
		Requirement source:
		N/A
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		Proposed solution optional:
		errata require to add a data sync barrier before the end of interrupt function
		For Base please add definition for:
		#define EXIT_INTERRUPT() MCAL_DATA_SYNC_BARRIER() / DSB sy full system /
		For other modules please add macro EXIT_INTERRUPT() at the end of isr handler, for example
		ISR(PIT_0_ISR)
		{
		uint8 instance = 0U;
		uint8 channel;
		uint32 temp = 0;
		for (channel = 0U; channel < PIT_0_CHANNELS_NUMBER; channel++)
		{
		temp = Pit_Ip_GetInterruptFlags(instance, channel);
		if (1U == temp)
		Unknown macro: \{ Pit_Ip_ProcessCommonInterrupt(instance, channel); }
		}
		EXIT_INTERRUPT()
		}
		"

ARTD-14282	New Feature	New Feature
		
		[uart] Add EXIT_INTERRUPT() for ISR handler due to ERR009005 errata
		,,"NewWorkDescription:
		ERR009005: Core: Store immediate overlapping exception return operation might vector to incorrect interrupt
		Workaround suggest by errata
		: For software not using the memory protection unit, this erratum can be worked around by
		setting DISDEFWBUF in the Auxiliary Control Register.
		In all other cases, the erratum can be avoided by ensuring a DSB occurs between the store
		and the BX instruction. For exception handlers written in C, this can be achieved by inserting
		the appropriate set of intrinsics or inline assembly just before the end of the interrupt function,
		for example:
		ARMCC:
		...
		schedule_barrier();
		asm\{DSB};
		schedule_barrier();
		}
		GCC:
		...
		asm volatile ("dsb 0xf" ::: "memory");
		}
		Requirement source:

ID	Subtype	Headline and Description

		N/A
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		Proposed solution optional:
		errata require to add a data sync barrier before the end of interrupt function
		For Base please add definition for:
		#define EXIT_INTERRUPT() MCAL_DATA_SYNC_BARRIER() / DSB sy full system /
		For other modules please add macro EXIT_INTERRUPT() at the end of isr handler, for example
		ISR(PIT_0_ISR)
		{
		uint8 instance = 0U;
		uint8 channel;
		uint32 temp = 0;
		for (channel = 0U; channel < PIT_0_CHANNELS_NUMBER; channel++)
		{
		temp = Pit_Ip_GetInterruptFlags(instance, channel);
		if (1U == temp)
		Unknown macro: \{ Pit_Ip_ProcessCommonInterrupt(instance, channel); }
		}
		EXIT_INTERRUPT()
		}
		"

ARTD-14283	New Feature	New Feature
		[wdg] Add EXIT_INTERRUPT() for ISR handler due to ERR009005 errata
		,,"NewWorkDescription:
		ERR009005: Core: Store immediate overlapping exception return operation might vector to incorrect interrupt
		Workaround suggest by errata
		: For software not using the memory protection unit, this erratum can be worked around by
		setting DISDEFWBUF in the Auxiliary Control Register.
		In all other cases, the erratum can be avoided by ensuring a DSB occurs between the store
		and the BX instruction. For exception handlers written in C, this can be achieved by inserting
		the appropriate set of intrinsics or inline assembly just before the end of the interrupt function,
		for example:
		ARMCC:
		...
		schedule_barrier();
		asm\{DSB};
		schedule_barrier();
		}
		GCC:
		...
		asm volatile ("dsb 0xf" ::: "memory");
		}
		Requirement source:
		N/A
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		Proposed solution optional:
		errata require to add a data sync barrier before the end of interrupt function
		For Base please add definition for:
		#define EXIT_INTERRUPT() MCAL_DATA_SYNC_BARRIER() / DSB sy full system /

ID	Subtype	Headline and Description

		For other modules please add macro EXIT_INTERRUPT() at the end of isr handler, for example
		ISR(PIT_0_ISR)
		{
		uint8 instance = 0U;
		uint8 channel;
		uint32 temp = 0;
		for (channel = 0U; channel < PIT_0_CHANNELS_NUMBER; channel++)
		{
		temp = Pit_Ip_GetInterruptFlags(instance, channel);
		if (1U == temp)
		Unknown macro: \{ Pit_Ip_ProcessCommonInterrupt(instance, channel); }
		}
		EXIT_INTERRUPT()
		}
		"

ARTD-14177	New Feature	New Feature [FLS][S32K1XX] Allow to allocate function InvalidPrefetchBuffer in FLASH ,,"Regarding the requirement: CPR_RTD_00515.fls*|Driver shall ensure that all the functionalities that are executable from RAM, can be disabled. All limitations for execution from RAM shall be stated in the Integration manual. Rationale: On some platforms, the RAM may not be executable due to security restrictions. The background: Due to the information from Reference manual, the Ftfc_Fls_Ip_InvalidPrefetchBuff_Ram function in *Ftfc_Fls_Ip.c* should be placed and executed from RAM, because it accesses the flash prefetch buffers, which should not be modified while reading code instructions. But, in some situations, this function still works normally from FLASH because it might be loaded to I_CACHE (Cache instruction) and executed from there. Therefore, the driver could allow to allocate it on both RAM or FLASH from the configurations GUI. Proposal solution; Create a new boolean node called: FlsInvalidatePrefetchBufferFromRam* (with default value is TRUE) in the container AutosarExt to toggle the macro: Ftfc_Fls_Ip_Cfg.h #define FTFC_FLS_INVALIDATE_PREFETCH_BUFFER_FROM_RAM (STD_ON) Ftfc_Fls_Ip.c*: check the macro and use the corresponding Memap syntax #if (FTFC_FLS_INVALIDATE_PREFETCH_BUFFER_FROM_RAM == STD_ON) #define FLS_START_SEC_RAMCODE #else #define FLS_START_SEC_CODE #endif #include ""Fls_MemMap.h"" static void Ftfc_Fls_Ip_InvalidPrefetchBuff_Ram(void); #if (FTFC_FLS_INVALIDATE_PREFETCH_BUFFER_FROM_RAM == STD_ON) #define FLS_STOP_SEC_RAMCODE #else #define FLS_STOP_SEC_CODE #endif #include ""Fls_MemMap.h"""

ID	Subtype	Headline and Description
ARTD-14182	Bug	[PORT] Port_Init incorrect data for Digital Filter configuration<*> Detailed description (how to reproduce it): use driver BLN_PORT_055, config digital filter in Port A, channel 0 9 31, Clock 1, Width 3; declare Port_Init==> fail Preconditions: some channel don't config digital filter but still be have .digitalFilter = true Test Case ID (internal TC that caught the defect) optional: Port_TC_FCT_0031 Observed behavior: Channel 5 in Port A was not config digital filter, but driver still enable !image-2021-07-20-11-39-41-228.png!!image-2021-07-20-11-40-11-950.png! Expected behavior: Driver need to be update correctly value of digital filter of channel in Port Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-14204	Bug	[EEP][S32K1 RTM] Update memmap for variable in example due to base memmap changes<*> Detailed description (how to reproduce it): Current examples use EEP_START_SEC_VAR_NO_INIT_8_NO_CACHEABLE for read and write buffer. But this region is removed in base for S32K1 Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: EEP_START_SEC_VAR_NO_INIT_8_NO_CACHEABLE for read and write buffer. Expected behavior: EEP_START_SEC_VAR_NO_INIT_8_NO_CACHEABLE shall not be used for read and write buffer. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Switch to EEP_START_SEC_VAR_INIT_8_NO_CACHEABLE
ARTD-14207	New Feature	New Feature [MCU] SleepOnExit incompatible with current approach on supervisor/user mode ,,"NewWorkDescription: Problem: Sleep On Exit bit from System Control Register can only be written in supervisor mode The behavior is that at the exit from the next interrupt the core goes to sleep !image-2021-07-21-08-44-11-091.png|width=816,height=126! When the driver is called in user mode context writing this bit will do the following: SVC_GoToSupervisor Write SLEEPONEXIT SVC_GoToUser Because SVC_GoToUser is currently implemented using the svc handler (which is an interrupt) the result is that the core goes in sleep immediately after going back to user mode

ID	Subtype	Headline and Description
		For detail please the email attached. Requirement source: CPR_RTD_00447.mcu Req Text*: The MCU driver shall offer support to control the exception return behavior of the currently executing core on a wake-up event triggered during SLEEP. The function void Mcu_SleepOnExit (boolean enableSleepOnExit) will enable/disable the automatic sleep entry on exception return. Requirement ticket: AAI-498 (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: 2 step must be done in Base, platform and mcu driver to avoid this issue 1. Base and Platform will implement the transition from supervisor mode to user mode without using the SVC Handler? Instead, just writing the CONTROL bit? !image-2021-07-21-08-46-09-719.png|width=800,height=151! 2. Mcu driver implementation: Add another function to the API to enable/disable calling Power_Ip_EnableSleepOnExit in the driver right before the WFI. Note that a pre-compile switch might not be enough since the application could choose not to enable SleepOnExit before every sleep cycle."
ARTD-14209	Bug	[S32K1XX][S32k1xx] Duplicate IRQ interrupt on EB<*> Detailed description (how to reproduce it): 1. Creating new project on EB for S32k116 2. Add Platform and Resource module for the project. 3. Choses resource is K32K116, and add all irq intrrupt of K116 to config generic interrupt setting . After change resource to S32K118, keeping all interrupt of K116 on generic intrrupt setting , Click add more irq interrupt. dulicate interrupt occur. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: review accpect Observed behavior: !image-2021-07-21-09-39-57-436.png! Expected behavior: the error will appear when duplication occurs Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-14211	Bug	[PWM] Update and correct the exclusive areas for Flexio<*> Detailed description (how to reproduce it): Update and correct the EAs for Flexio Review and update the EAs for another IPV (Ftm, Emios) Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: the EAs are available in Flexio Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)

ID	Subtype	Headline and Description
		Proposed solution optional: Update the Eas
ARTD-14217	Bug	[PORT] Fix compiler warning<*> Detailed description (how to reproduce it): Check file compiler warning report after all test was build successfully Preconditions: Driver need to have no warning Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: check file [http://ceram.ea.freescale.net/0/project/custom_compilerwarning/details] for Port or in attach file Expected behavior: Driver need to be update to resolve some warnings Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-14222	Bug	[BASE] Enable errata ERR010777 SCG Corrupted status when the system clock is switching<*> Detailed description (how to reproduce it): Implement errata: ERR010777: SCG: Corrupted status when the system clock is switching. Description: The SCG_RCCR[SCS] and SCG_HCCR[SCS] may have a corrupted status during the interval when the system clock is switching Workaround: The SCS field should be read twice by the software to ensure the system clock switch has completed The macro ERR_IPV_SCG_ERR010777 must be added in Soc_Ips.h for derivatives: S32K116, S32K118, S32K142, S42K144, S32K144W, S32K146, S32K148 Preconditions: S32K116_0N96V, S32K118_0N97V, S32K142_0N33V, S42K144_0N57U, S32K144W_0P64A, S32K146_0N73V, S32K148_0N20V Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: errata is implemented in driver code Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: The SCS field should be read twice by the software to ensure the system clock switch has completed
ARTD-14225	Bug	[LIN] Driver can't transmit frame success in case 10 bit break field generate with IPV: LPUART and FlexIO<*> Detailed description (how to reproduce it): Driver can't transmit frame success in case 10 bit break field generate with IPV: LPUART and the number of bits that are generated in the UserConfig->MasterBreakLength ** we have to sent AT LEAST 11 bits for any configuration.

ID	Subtype	Headline and Description
		Preconditions: Detected Break Length always is 11 bit Test Case ID (internal TC that caught the defect) optional: Lin_TC_FCT_0018 Observed behavior: NA Expected behavior: NA Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-14237	Bug	[BASE] Some compiler options are not the same between SOW and S32CT<*> Detailed description (how to reproduce it): Step 1: Create new project. Step 2: Select project > right click > Properties > C/C Build > Settings Step 3: Check compiler options and compare with SOW file Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: Some compiler options are not the same between SOW and S32CT : S32K11X*: Linker: Duplicate Option on *CT: -mfpu=auto *S32K1XX*: ** Compiler: Missing Option on *CT: -fno-short-enums Expected behavior: Update compiler option for CT the same with compiler option in SOW file. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-14240	Bug	[UART] Conflict if using common callback in the same channel<*> Detailed description (how to reproduce it): If user uses the common callback in channel for his program, it will cause conflict, bcs the declaration of callback functions will generate several times in EB/CT generated code. It can cause the compiler warning for dupplicate declaration. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: Fix this issue Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-14242	Bug	[UART] In Abort case, Complete Send/Receive internal functions need to wait for finishing of current byte transmission

ID	Subtype	Headline and Description
		,,"Detailed description (how to reproduce it): In Abort function, Complete Send/Receive internal functions are called to finish the current transmission. But they need to wait for finishing of current byte transmission and afterward disable TE and RE. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: Fix this issue Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-14243	Bug	[ICU] missing memmap.h in file Lptmr_Icu_Ip.c<*> Detailed description (how to reproduce it): In Lptmr_Icu_Ip.c, the inlcude of the Icu_MemMap is missing for ICU_START_SEC_VAR_INIT_UNSPECIFIED and ICU_STOP_SEC_VAR_INIT_UNSPECIFIED Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: In Lptmr_Icu_Ip.c, the inlcude of the Icu_MemMap is missing for ICU_START_SEC_VAR_INIT_UNSPECIFIED and ICU_STOP_SEC_VAR_INIT_UNSPECIFIED Expected behavior: Add the #include Icu_MemMap.h for ICU_START_SEC_VAR_INIT_UNSPECIFIED and ICU_STOP_SEC_VAR_INIT_UNSPECIFIED Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-14245	Bug	[PWM] PWM configuration cannot generate with EB command because of attribute READONLY<*> Detailed description (how to reproduce it): Pwm and and all modules referencing it cannot be generated on the command line because of the error: ERROR 21-07-20,11:02:46 (1803) Invalid XPath-expression for Attribute "READONLY" of node "/AUTOSAR/TOP-LEVEL-PACKAGES/Pwm/ELEMENTS/Pwm/PwmChannelConfigSet/PwmFlexio/PwmFlexio_0/PwmFlexioChannels/PwmFlexioChannels_0/FlexioChPrescaler": (35016) Cannot get a resource file for target CORTEXM and derivate S32K14X. Maybe this is due to module restrictions. My commandline call is: call tresos_cmd.bat Dtarget=CORTEXM Dderivate=S32K14X legacy generate .\CLT_SW32K1_RTD_4.4_0.8.0_Tresos.arxml@asc:4.4.0 o .\GenData g Pwm_TS_T40D2M8I0R0

ID	Subtype	Headline and Description
		The solution has been defined in the ticket AAI-585 Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Cannot generate PWM with EB by command Expected behavior: Can generate PWM with EB by command Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Change READONLY to EDITABLE
ARTD-14247	Bug	[WDG] All functions in Wdog_Ip should check re-configuration complete after unlocking and updating registers<*> Detailed description (how to reproduce it): In Ip_Wdog test: Initialize Wdog Use Wdog_Ip_StopTimer() to stop Wdog counter Read counter value Delay for a period of time Read counter again Compare the two counter values of before and after delay. Make sure that they are equal When i debug the test, it worked fine but when i run report, the test is fail. When add delay time after calling function Wdog_Ip_StopTimer(). The test is pass. !Capture.JPG|thumbnail! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: Ip_Wdog_TC_FCT_0001 Observed behavior: Currently, only functions Wdog_Ip_Init, Wdog_Ip_DeInit, Wdog_Ip_Config have checked reconfiguration complete after unlocking and updating registers by invoking function Wdog_Ip_IsReconfigurationComplete(). Other functions like: Wdog_Ip_SetTimeout, Wdog_Ip_StartTimer, Wdog_Ip_StopTimer, ... don't have this verification. I think these functions should use Wdog_Ip_IsReconfigurationComplete() to ensure that all new configurations are updated and available for following functions. Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-14244	New Feature	New Feature [BASE]Add EXIT_INTERRUPT() for ISR handler due to ERR009005 errata ,,"NewWorkDescription: ERR009005: Core: Store immediate overlapping exception return operation might vector to incorrect interrupt Workaround suggest by errata

ID	Subtype	Headline and Description

		: For software not using the memory protection unit, this erratum can be worked around by
		setting DISDEFWBUF in the Auxiliary Control Register.
		In all other cases, the erratum can be avoided by ensuring a DSB occurs between the store
		and the BX instruction. For exception handlers written in C, this can be achieved by inserting
		the appropriate set of intrinsics or inline assembly just before the end of the interrupt function,
		for example:
		ARMCC:
		...
		schedule_barrier();
		asm\{DSB};
		schedule_barrier();
		}
		GCC:
		...
		asm volatile ("dsb 0xf" ::: "memory");
		}
		Requirement source:
		N/A
		(e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...)
		Proposed solution optional:
		errata require to add a data sync barrier before the end of interrupt function
		For Base please add definition for:
		#define EXIT_INTERRUPT() MCAL_DATA_SYNC_BARRIER() / DSB sy full system /
		For other module please add macro EXIT_INTERRUPT() at the end of isr handler, for example
		ISR(PIT_0_ISR)
		{
		uint8 instance = 0U;
		uint8 channel;
		uint32 temp = 0;
		for (channel = 0U; channel < PIT_0_CHANNELS_NUMBER; channel++)
		{
		temp = Pit_Ip_GetInterruptFlags(instance, channel);
		if (1U == temp)
		{
		Pit_Ip_ProcessCommonInterrupt(instance, channel);
		}
		}
		EXIT_INTERRUPT()
		}
		"

ARTD-14248	Bug	[S32K1XX] Failure at building on S32K11x<*> Detailed description (how to reproduce it): N/A Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: CRC_TC_0001 Observed behavior: Failure at building on S32K118(Pls see attached file) !image-2021-07-22-13-55-52-710.png|thumbnail!

ID	Subtype	Headline and Description
		Expected behavior: Update driver code to fix build failed Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: N/A
ARTD-14254	Bug	[S32K1XX] Compiler error in Platform_CfgDefines.h S32DS<*> Detailed description (how to reproduce it): Some examples have compiler error in Platform_CfgDefines.h !image-2021-07-22-14-52-39-964.png! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: Compiler error in Platform_CfgDefines.h Expected behavior: No compiler error Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Change '\' to '\\\' #ifndef DISABLE_MCAL_INTERMODULE_ASR_CHECK / Check if source file and StandardTypes header file are of the same Autosar version */ #if ((S32K1XX_CFGDEFINES_AR_RELEASE_MAJOR_VERSION != STD_AR_RELEASE_MAJOR_VERSION) || \\\ (S32K1XX_CFGDEFINES_AR_RELEASE_MINOR_VERSION != STD_AR_RELEASE_MINOR_VERSION) \\\ ) #error "AutoSar Version Numbers of Platform_CfgDefines.h and StandardTypes.h are different" #endif #endif
ARTD-14260	Bug	[S32K1XX] Syntax for version checking command is wrong in file Platform_CfgDefines.h<*> Detailed description (how to reproduce it): Syntax for version checking command is wrong in file Platform_CfgDefines.h. pls see attached file. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-14288	Bug	

ID	Subtype	Headline and Description
		[S32K1XX][FLS] Module documentation chapter in UM was missing generated<*> Detailed description (how to reproduce it): Some sub-chapters are missing in Module Documentation part of UM such as: 6.1.3 Macro Definition Documentation, 6.1.6 Function Reference Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: Some sub-chapters are missing in Module Documentation part of UM Expected behavior: There shall not be any missing sub-chapters in Module Documentation part of UM Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update UM
ARTD-14426	Bug	[OCU] Check and update the functions that don't match the requirements<*> Detailed description (how to reproduce it): Some function in "ReqExport.txt" file but does not include in "*.h" files of driver code pls see attached files for more details Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: NA Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Check and update for all APIs
ARTD-14428	Bug	[GPT] Check and update the functions that don't match the requirements<*> Detailed description (how to reproduce it): Some function in "ReqExport.txt" file but does not include in "*.h" files of driver code pls see attached files for more details Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: NA Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Check and update for all APIs
ARTD-14306	New Feature	New Feature

ID	Subtype	Headline and Description
		[CAN] re-analysis Exclusive Area ,,"NewWorkDescription: some code changed, need to re-analyze EA Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: [...]"

ARTD-14305	Bug	[CRYPTO]Crypto_KeyElementSet return error when does not configure to store M4 and M5 values.<*>
		
		Detailed description (how to reproduce it):
		Call Crypto_KeyElementSet() to load SHE key, function return error The address parameters are invalid when doesn't configure to store M4, M5 value
		Preconditions:
		Call Crypto_KeyElementSet() to load SHE key, but user doesn't configure key to store M4, M5 value.On funtion Crypto_Ipw_SheLoadEncryptedKey:
		pM4 = NULL_PTR;
		pM5 = NULL_PTR;
		
		Test Case ID (internal TC that caught the defect) optional:
		Crypto_TC_FCT_070
		Observed behavior:
		Crypto_KeyElementSet return error the address parameters are invalid
		Expected behavior:
		Function Crypto_KeyElementSet must return no error in both case
		Proposed solution optional:
		Declare on stack temporary buffers for M4 and M5

ARTD-14308	Bug	[ADC] There is some information that does not match EB and S32CT when comparing code generation<*> Detailed description (how to reproduce it): Step 1: clean generate Adc_TS_COT_001 for EB and Adc_TS_COT_101_CT for CT Step 2: Compare include folder and src folder of S32CT with EB Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: There is some information that does not match EB and S32CT when comparing code generation. Detail in *share point link*: [https://nxp1.sharepoint.com/sites/Zebra/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2Fsites%2FZebra%2FShared%20Documents%2FGroup%204%2FEBvsS32CT%2FADC] File: "Compare Config S32CT and EB Tresos_EPC_S32K148.xlsx", "Adc_Compare_Config_S32CT_and_EB_Tresos_define_S32K1xx.xlsx" Expected behavior: EB and S32CT need to generate the same file. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-14321	Bug	

ID	Subtype	Headline and Description
		[S32K1XX] CMU gating is not enabled in sys_init()<*> Detailed description (how to reproduce it): CMU gating is not enabled in sys_init() !image-2021-07-23-16-17-33-078.png! Preconditions: NA Test Case ID (internal TC that caught the defect) optional: Mcu_TC_FCT_0585 Observed behavior: CMU gating is disabled after calling sys_init() Expected behavior: CMU gating is enabled after calling sys_init() Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Add these code into sys_init() #if defined(S32K116)||defined(S32K118) IP_PCC->PCCn[PCC_CMU0_INDEX]= 0x43000000; IP_PCC->PCCn[PCC_CMU1_INDEX]= 0x43000000; #endif
ARTD-14326	Bug	[WDG] Different function prototype between requirement and driver<*> Detailed description (how to reproduce it): In requirement SWS_Wdg_00155 prototype Syntax: void Wdg_SetTriggerCondition(uint16 timeout) In driver, now the prototype is : void Wdg_43_Instance0_SetTriggerCondition(uint16 u16Timeout*) The parameter timeout is different between requirement and driver now. SWS_Wdg_00155 is Autosar requirement so I think driver should has same prototype with requirement. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: Different function prototype between requirement and current driver Expected behavior: Function prototype should be same with requirement Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-14328	New Feature	New Feature [FLS][S32G] Wrong SFDP configuration for Quad flashes ,,Qspi_Ip_ReadSfdp generates wrong configuration for a MX66U1G45G quads flash. The Fls_Init function fails.
ARTD-14334	Bug	[CAN] redundant mode-switching inside FlexCAN_Ip_GetTDCFail<*> Detailed description (how to reproduce it): During ccov analysis, i had an observation:

ID	Subtype	Headline and Description
		FlexCAN_Ip_*GetTDCFail now support to handler the timeout (when switching mode) case but: This case should not be handled because read-TDCFAIL does not required mode-switching => there is no timeout case for this api, it should always be invoked successfully with correct returned value (as similar as FlexCAN_Ip_GetBitrate) Unnecessary mode-switching cause redundant code and impacting to driver performance (Unnecessarily enter/exit start-mode for reading bit-value), operation-mode of module !image-2021-07-24-09-40-22-329.png! additional check: FlexCAN_Ip_GetTDCValue, FlexCAN_Ip_ClearTDCFail Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: Function don't need to enable flexcan clock to access registers. Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-14343	Bug	[I2C] S32K118 example need to be updated according new EVB<*> Detailed description (how to reproduce it): [...] Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update example description according new EVB was used
ARTD-14349	Bug	[S32K1xx][RTM_100][PORT] PortPinPFE field lacks some mux mode configurations<*> Detailed description (how to reproduce it): Use DS layout and EB of NB_210726 On both DS side and EB side, add Port component (High layer on DS) with the following configuration: PCR5: any mode except RESET_b PCR99: any mode except NMI_b Observed behavior: The "PortPin Passive Filter Enable" on EB and CTHL was unable to configurate unless the users set those 2 pins in RESET_b (for PCR5) or NMI_b (for PCR99) as below: !image-2021-07-26-11-42-40-926.png! !image-2021-07-26-11-43-01-818.png! Expected behavior:

ID	Subtype	Headline and Description
		Following Reference manual in chip-specific PORT information, this PFE field can be configurable for all mode of these 2 pins
ARTD-14356	Bug	[CRYPTO]Add a macro cancel time out for Csec_Ip_CancelCommand<*> Detailed description (how to reproduce it): When set timeout for test is minimum (Csec_Ip_pState->u32Timeout = 0) the program will be error half fault Preconditions: Call a Job different CSEC_IP_CMD_ENC_ECB and CSEC_IP_CMD_DEC_ECB command and set Csec_Ip_pState->u32Timeout = 0 Test Case ID (internal TC that caught the defect) optional: Crypto_TC_FCT_0075 Observed behavior: The program is processing a job not done but still write the command header , it will have error half fault Expected behavior: Should add cancel time out for Csec_Ip_CancelCommand Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: NA
ARTD-14359	Bug	[EEP][Example] Update board information in readme.txt and description.txt files<*> Detailed description (how to reproduce it): Current readme and description files contain out of date information about EVB boards Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Current readme and description files contain out of date information about EVB boards Expected behavior: Current readme and description files shall contain up to date information about EVB boards Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update readme and description files.
ARTD-14368	Bug	[Platform][S32K1XX] Check invalid parameters to Platform_SetIrq, this function is working incorrect ,,"Detailed description (how to reproduce it): 1. Creating project Check if invalid parameters are passed to Platform_SetIrq.. 2. Calling Platform_SetIrq function with invalid parameter, (Ex: INT_CTRL_IP_IRQ_MIN 1 and INT_CTRL_IP_IRQ_MAX 1), This function will return an error. Preconditions: Parameter is invalid dev_error is enable Test Case ID (internal TC that caught the defect) optional: Platfrom_TS_001 Observed behavior:

ID	Subtype	Headline and Description
		!image-2021-07-26-17-07-51-068.png! !image-2021-07-26-17-09-28-765.png! After calling Platform_SetIrq function with invalid parameter, the driver checked and return error, but Platform_Ipw_SetIrq still runs with invalid parameters resulting in hard_fault. Because, In debug mode, when pointer point to line of code on the picture. It will write r3 valid to r2 address, however r2 address is flash address, So test hard_fault. Expected behavior: This Platform_SetIrq function doesn't call Platform_Ipw_SetIrq function when chek with invalid parameter and dev_error enable. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: You can refer to the image below. !image-2021-07-26-17-26-24-239.png!
ARTD-14375	Bug	[ADC] ADC_VALIDATE_APP_BUFFER_ALIGNMENT is never defined<*> Detailed description (how to reproduce it): ADC_VALIDATE_APP_BUFFER_ALIGNMENT is never defined, resulting in Adc_ValidateSetupBufferAlignment() never being actually called. Proposed solution optional: Check if this check is still needed/working, also in MCAL If needed define should be generated by config. Otherwise function code removed
ARTD-14396	Bug	[EEP] Ftfc_Eep_Ip_Erase uses an unaligned variable<*> Detailed description (how to reproduce it): Invoke Ftfc_Eep_Ip_Erase() with a PageSize of FTFC_EEP_IP_PAGE_LONGWORD (4 bytes). Preconditions: The CPU does not accept unaligned accesses to RAM. EraseValue is stored at an unaligned address (in .rodata). Test Case ID (internal TC that caught the defect) optional: example test S32K118 Observed behavior: If FTFC_EEP_IP_ALIGNED_RAM_ACCESS == STD_ON a DevAssert trap is caught. Else, a hard fault is generated. Expected behavior: Erase operation executes successfully. SWS_Eep_00020 The Eep module shall execute the erase job asynchronously within the Eep module's job processing function. The Eep module shall erase an EEPROM block starting from EepromAddress EEPROM base address of size Length.The function Eep_Erase checks the API parameters according to requirements SWS_Eep_00016, SWS_Eep_00017, SWS_Eep_00018.The function Eep_Erase checks the EEPROM state according to requirement SWS_Eep_00033. Proposed solution optional: EraseValue variable should be moved in an aligned memory section.
ARTD-14406	Bug	[ETH] Eth_ProvideTxBuffer returns BUFREQ_E_OVFL in a specific case<*> Detailed description (how to reproduce it):

ID	Subtype	Headline and Description
		In case, configure one buffer, release this buffer after transmitting it then using Eth_ProvideTxBuffer to provide another buffer with a length longer than the length transmitted, this function will return BUFREQ_E_OVFL Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: Eth_TC_FCT_0156 Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-14411	New Feature	New Feature [ADC] Add note in documentation that users must read results from callback to clear the interrupt flag ,,"Detailed description (how to reproduce it): Init Ip_Adc with SW trigger source Call function Adc_Ip_EnableChannelNotification() set AIEN bit. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: Ip_Adc_TC_0101.c Observed behavior: When config SW trigger source. Configure SC1A register. AIEN bit is set. COCO bit is set. => always run Adc_Ip_IRQHandler(). COCO bit not clear. Expected behavior: COCO bit clear when config SC1A register. Read data in notification when use SW trigger Ip_Adc Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]"
ARTD-14412	New Feature	New Feature [CRYPTO]Note in UM about main function called during a asynchronous command with interrupt enabled ,,"NewWorkDescription: A note must be added in the User Manual to make the user aware of the fact that the main function (Csec_Ip_MainFunction) must not be called after requesting asynchronous service with interrupt enabled because it may lead to requesting a CSEc command during the execution of another CSEc command which is not allowed by the CSEc. Requirement source: NA (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: Add a note in UM about main function usage."
ARTD-14416	Bug	[RM] Test Example K1xx : Fix typo in readme.txt and description.txt<*>

ID	Subtype	Headline and Description
		Detailed description (how to reproduce it): 1. Please fix the typo in the file: readme.txt and description.txt in example (more detail in attached file). 2. Wrong resource s32k144_lqfp64 => Change to s32k144_lqfp100 Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: [...] Expected behavior: Fix the typo in the file readme.txt and description.txt Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-14417	Bug	[ADC] Disable ADC interrupt when streaming linear hardware trigger group is in ADC_STREAM_COMPLETED state<*> Detailed description (how to reproduce it): Streaming linear hardware trigger group might have further notifications after ADC_STREAM_COMPLETED since ADC interrupt is not disabled Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: NA Expected behavior: NA Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Disable AIEN when hardware streaming linear group is in ADC_STREAM_COMPLETED state
ARTD-14430	Bug	[RM] Test Example S32K1xx : Fix typo in src/main.c<*> Detailed description (how to reproduce it): Please fix typo in src/main.c (All examples). Detail in attachment. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-14435	Bug	[Example][i2c] Fix example run fail on S32K118 EvB<*>

ID	Subtype	Headline and Description
		Detailed description (how to reproduce it): [...] Stuck in Port_Ip_Init function !image-2021-07-28-14-19-08-534.png! !image-2021-07-28-14-18-43-312.png! Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: [...] Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-14434	New Feature	New Feature [FLS][S32K148_RTM100] Fls driver shall check type of sector prior to call Cache_Ip_InvalidateByAddr ,,"NewWorkDescription: After (might be) some update from MCL for Cache_Ip_InvalidateByAddr. It make different at least in timing aspect to Invalidating Cache, it lead to unexpected result if call Cache_Ip_InvalidateByAddr in case sectors to test is DATA SECTOR, which is non-cacheable . Requirement source: [...] (e.g.cPRD,gMRD, Customer Request, Quality, ASR SWS, RM.pdf, Errata.pdf...) Proposed solution optional: It 's better if driver can be check type of Sector to test , and allow Invalidate Cache whether node FlsSynchronizeCache is being set or not."
ARTD-14455	Bug	[EEP] Eep_Write is affected by EEP_ASYNC_ERASE_OPERATIONS_ENABLED<*> Detailed description (how to reproduce it): Request a write job then wait for it to complete. Preconditions: EEP_ASYNC_ERASE_OPERATIONS_ENABLED == STD_OFF EEP_ASYNC_WRITE_OPERATIONS_ENABLED == STD_ON Observed behavior: Eep_GetJobResult always returns MEMIF_JOB_PENDING. Expected behavior: The Write operation ends within a determinate time period. Req ID:_ SWS_Eep_00015 The Eep module shall execute the write job asynchronously within the Eep module's job processing function. During job processing the Eep module shall write a data block of size Length from DataBufferPtr to EepromAddress EEPROM base address.The function Eep_Write checks the API parameters according to requirements SWS_Eep_00016, SWS_Eep_00017, SWS_Eep_00018.The function Eep_Write checks the EEPROM state according to requirement SWS_Eep_00033. Proposed solution optional: Fix the typo in the Eep_ProcessWriteJob function.
ARTD-14446	Bug	

ID	Subtype	Headline and Description
		[EEP][Example] Update .mex and .xdm file for example S32K118<*> Detailed description (how to reproduce it): Current .mex and .xdm file for S32K118 example is applied for LQFP64 CPU. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: Current .mex and .xdm file for S32K118 example is applied for LQFP64 CPU. Expected behavior: .mex and .xdm file for S32K118 example shall be applied for LQFP48 CPU. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Update configuration files
ARTD-14456	Bug	[EEP] Update generate code to remove warning checkDef<*> Detailed description (how to reproduce it): CheckDef reports EEP_CONFIG_PB multi defined. Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: CheckDef reports EEP_CONFIG_PB multi defined. Expected behavior: CheckDef shall not report EEP_CONFIG_PB multi defined. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: #define EEP_CONFIG_EXT \ [!IF "var:defined('postBuildVariant')"!][!// [!LOOP "variant:all()"!][!// EEP_CONFIG_[!"."!]_PB \ [!ENDLOOP!][!// [!ELSE!][!// EEP_CONFIG_PB [!ENDIF!]
ARTD-14465	Bug	[Platform] Variable do not used in Platform_Ipw.c<*> Detailed description (how to reproduce it): To build the example Wdg_example_S32K118, execute the following command to build: make build In Platform_Ipw.c: warning: variable 'ret' set but not used [-Wunused-but-set-variable] 151 Std_ReturnType ret = (Std_ReturnType)E_OK; Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: Update Platform_Ipw.c !image-2021-07-29-10-25-57-142.png! Expected behavior:

ID	Subtype	Headline and Description
		Building code hasn't warning Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...] - aa
ARTD-14469	Bug	[S32K1xx][RTM_100][Port] Examples have warnings after generating source configuration<*> Detailed description (how to reproduce it): Use EBT with license for examples code, then install +SW32K1_RTD_4.4_1.0.0_D2107.exe of B210726. Access Port examples in Port_plugins, configurate file +project_parameters.mk with your pc's paths and use cmd command to clean and generate the source configuration code Observed behavior: After generating source configuration for the example, this warning occurs: !image-2021-07-29-10-39-31-627.png|width=853,height=434! Expected behavior: Neither errors nor warnings occur after finishing generating
ARTD-14555	Bug	[CAN][example] Wrong pin package for CT examples on S32K118<*> Detailed description (how to reproduce it): The example CAN_Example_S32K118 and FlexCAN_Example_S32K118 have wrong pin package. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: CAN_Example_S32K118 FlexCAN_Example_S32K118 Observed behavior: Currently the pin package for these example is S32K118_LQFP*64 Expected behavior: Expected pin pakage: S32K118_LQFP*48 Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: [...]
ARTD-14557	Bug	[LIN] - Examples on s32k118 have the wrong pin package for EBT<*> Detailed description (how to reproduce it): For the ebt example the pin package is wrong. Preconditions: [...] Test Case ID (internal TC that caught the defect) optional: [...] Observed behavior: [...] Expected behavior: Use the S32K118_LQFP48 pin package for these examples Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.)

ID	Subtype	Headline and Description
		Proposed solution optional: [...]
ARTD-14653	Bug	[CRYPTO]Cancel timeout is not counter dependent<*> Detailed description (how to reproduce it): Preprocessor if evaluates the check on dummy counter true even if the selected counter is different. !image-2021-08-02-18-20-14-170.png! Because the values for counter types are enums at pre processing time, OSIF_COUNTER_DUMMY and CSEC_IP_TIMEOUT_OSIF_COUNTER_TYPE are treated as defines with value 0. !image-2021-08-02-18-20-54-370.png! The check over the dummy counter always returns true so CSEC_IP_CANCEL_CMD_TIMEOUT_U32 always takes the bigger value. Preconditions: NA Test Case ID (internal TC that caught the defect) optional: NA Observed behavior: CSEC_IP_CANCEL_CMD_TIMEOUT_U32 always takes the bigger value. Expected behavior: CSEC_IP_CANCEL_CMD_TIMEOUT_U32 depends on the counter type. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional: Move the definition of CSEC_IP_CANCEL_CMD_TIMEOUT_U32 in the Csec_Ip_Cfg.h at generation time. Use strCounterType varaible to determine which counter was configured and define the CSEC_IP_CANCEL_CMD_TIMEOUT_U32 with the proper value based on the configured counter.
ARTD-14699	Bug	[EEP]Warning when generate example for EBT<*> Detailed description (how to reproduce it): Generate project example project for s32k118, s32k144, s32k144w and s32k148 on EBT use cmd : "make generate " Preconditions: N/A Test Case ID (internal TC that caught the defect) optional: N/A Observed behavior: warning happened: WARNING 21-08-04,14:58:55 (1055) Removed node "EepCompareApi": The schema does not define a parameter with name "EepCompareApi" in container "/TS_T40D2M10I0R0/Eep/AutosarExt". Expected behavior: No warning wen generate , build also. Note: in the "Expected behavior" field, please mention also the requirement source too (cPRD, Errata version, RM, CR number etc.) Proposed solution optional:

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at https://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Suitability for use in automotive applications — This NXP product has been qualified for use in automotive applications. If this product is used by customer in the development of, or for incorporation into, products or services (a) used in safety critical applications or (b) in which failure could lead to death, personal injury, or severe physical or environmental damage (such products and services hereinafter referred to as "Critical Applications"), then customer makes the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, safety, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP. As such, customer assumes all risk related to use of any products in Critical Applications and NXP and its suppliers shall not be liable for any such use by customer. Accordingly, customer will indemnify and hold NXP harmless from any claims, liabilities, damages and associated costs and expenses (including attorneys' fees) that NXP may incur related to customer's incorporation of any product in a Critical Application.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

HTML publications — An HTML version, if available, of this document is provided as a courtesy. Definitive information is contained in the applicable document in PDF format. If there is a discrepancy between the HTML document and the PDF document, the PDF document has priority.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP B.V. — NXP B.V. is not an operating company and it does not distribute or sell products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

Contents

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.