

Z20K14xM MCAL Release Notes RTM

1.2.1

Table of contents

1. Getting Started	1
1.1. Package content	1
1.2. Installation	1
2. Release Specifics	2
2.1. Supported Derivatives	2
2.2. Drivers Delivered	2
2.3. Module Configuration	3
2.4. Build options	4
3. Changes List for Z20K14xM MCAL	7
3.1. RTM 1.2.1	7
3.2. RTM 1.2.0	11
3.3. RTM 1.1.0	14
3.4. RTM 1.0.0	22
4. Version Tracking	25

1. Getting Started

1.1. Package content

This package contains ZHIXIN Z20K14xM MCAL for AUTOSAR R20-11 version:

- "eclipse/plugins/<Module>_ZX_14xM" directories - Tresos Plugins, 1 per module
- "QuickStartGuide/Z20K14xM_MCAL_Quick_Start_Guide.pdf" - A document which can help you to get started quickly with Z20K14xM MCAL
- "ReleaseNotes/Z20K14xM_MCAL_ReleaseNotes.pdf" - This file

1.2. Installation

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 "<Module>_ZX_14xM" directories to the "<Tresos_Install_Path>/plugins" directory.

2. Release Specifics

The Z20K14xM MCAL RTM 1.2.1 is AUTOSAR R20-11 compliant. The Z20K14xM MCAL drivers take AUTOSAR SRS and SWS Specifications as requirements.

2.1. Supported Derivatives

This section lists supported derivatives by the MCAL package.

- Z20K148M_lqfq100
- Z20K148M_lqfq144
- Z20K146M_lqfq64
- Z20K146M_lqfq100
- Z20K146M_lqfq144
- Z20K144M_lqfq64
- Z20K144M_lqfq100

2.2. Drivers Delivered

This section lists drivers delivered:

Module	Version	Description
Adc	1.2.1	Analog Digital Converter Driver
Base	1.2.1	Base Module, General AUTOSAR files
Can	1.2.1	Controller Area Network Driver
CanIf	1.2.1	Stub module, Controller Area Network Interface
Crypto	1.2.1	Crypto Driver
CryptoIf	1.2.1	Stub module, Crypto Interface
Csm	1.2.1	Stub module, Crypto Service Manager
Dem	1.2.1	Stub module, Diagnostic Event Manager
Det	1.2.1	Stub module, Development Error Tracer
Dio	1.2.1	Digital Input Output Driver
Dma	1.2.1	Direct Memory Access Driver
EcuC	1.2.1	Stub module, ECU Configuration
EcuM	1.2.1	Stub module, ECU State Manager
Fls	1.2.1	Flash Driver
Gpt	1.2.1	General Purpose Timer Driver
I2c	1.2.1	Inter-Integrated Circuit Driver

Module	Version	Description
Icu	1.2.1	Input Capture Unit Driver
Lin	1.2.1	Local Interconnect Network Driver
LinIf	1.2.1	Stub module, Local Interconnect Network Interface
McalLib	1.2.1	McaLib module, MCAL libraries and Hardware register files
Mcu	1.2.1	Microcontroller Unit Driver
MemIf	1.2.1	Stub module, Memory Abstraction Interface
Nvm	1.2.1	Stub module, NVRAM Manager
Ocu	1.2.1	Output Control Unit Driver
Os	1.2.1	Stub module, Operating System
Platform	1.2.1	Platform module, Provides Interrupt and TMU(TrigMux Unit) drivers
Port	1.2.1	Port Driver
Pwm	1.2.1	Pulse Width Modulation Driver
Resource	1.2.1	Resource Module, Required by all other modules to select MCU derivative
Rte	1.2.1	Stub module, Run-Time Environment, only used for Schedule Manager
Spi	1.2.1	Serial Peripheral Interface Driver
Ewdt	1.2.1	External Watchdog Driver
Wdg	1.2.1	Watchdog Driver
WdgIf	1.2.1	Stub module, Watchdog Interface

2.3. Module Configuration

Modules configurations were developed and tested using the Tresos Configuration Tool version "EB tresos Studio 29.2.0".

A folder named "<Module>_ZX_14xM" exists for each delivered module. It is called a Tresos plugin for the module. A plugin contains:

- AUTOSAR module definition file (epd)
- Tresos Xpath Data Model module definition file (xdm)
- The module driver source files
- The module configuration generation template source files
- The module user manual
- The module example
- Additional necessary Tresos specific tooling files.

2.4. Build options

This section describes compiler, assembler and linker options used for building the MCAL drivers.

The MCAL drivers are compiled with:

- IAR 8.40.3
- Green Hills Multi 7.1.4 / Compiler 2022.1.4
- Keil MDK-ARM Version 5.34

The details of compiler, assembler, and linker options are explained below.

2.4.1. IAR Compiler/Assembler/Linker Options

IAR Compiler Options

Compiler Option	Description
--cpu=Cortex-M4	Use this option to specify the Arm core.
--fpu=VFPv4_sp	Use this option to generate code that carries out floating-point operations using a Floating Point Unit (FPU). Single-precision variant.
--cpu_mode=thumb	Use this option, alternatively the --arm or --thumb options, to specify which instruction set should be used for your project.
--endian=little	Use this option to specify the byte order for your project.
-e	Use this option to enable all IAR C language extensions.
-On	Use this option to set the optimization level to be used by the compiler when optimizing the code. If no optimization option is specified, the optimization level is used by default. n presents None(Best debug support).
--debug or -r	Use the --debug or -r option to make the compiler include information in the object modules required by the IAR C-SPY® Debugger and other symbolic debuggers.

IAR Assembler Options

Assembler Option	Description
--cpu Cortex-M4	Use this option to specify the Arm core.
--fpu VFPv4_sp	Use this option to generate code that carries out floating-point operations using a Floating Point Unit (FPU). Single-precision variant.
--cpu_mode thumb	Use this option, alternatively the --arm or --thumb options, to specify which instruction set should be used for your project.
-r	Use this option to make the assembler generate debug information, which means the generated output can be used in a symbolic debugger such as IAR C-SPY® Debugger.

IAR Linker Options

Linker Option	Description
--cpu=Cortex-M4	Use this option to specify the Arm core.
--fpu=VFPv4_sp	Use this option to generate code that carries out floating-point operations using a Floating Point Unit (FPU). Single-precision variant.
--entry Reset_Handler	Use this option to make a symbol be treated as a root symbol and the start label of the application. This is useful for loaders.
--map filename	Use this option to produce a linker memory map file. The map file has the default filename extension map.
--config filename	Use this option to specify the configuration file to be used by the linker (the default filename extension is icf). If no configuration file is specified, a default configuration is used. This option can only be used once on the command line.

2.4.2. GHS Compiler/Assembler/Linker Options

GHS Compiler Options

Compiler Option	Description
-cpu=cortexm4f	Specify a target processor with the driver.
-G	Generate MULTI debugging information.
-thumb	Select the generation of Thumb code.
-thumb_lib	Enables linking with Thumb libraries. This option is the default for Thumb-only processors (the Cortex M family).
--gnu_asm	Select GNU extended asm syntax support.
-dual_debug	Generated DWARF information conforms to the DWARF 2.0 standard.
-keeptempfiles	Prevents the deletion of temporary files after they are used.
-Wimplicit-int	Controls a warning that is issued if the return type of a function is not declared before it is called (thus causing the return type to be implicitly declared int).
-Wshadow	Controls a warning that is issued if the declaration of a local variable shadows the declaration of a variable of the same name declared at the global scope, or at an outer scope.
-Wtrigraphs	Controls a warning that is issued for any use of trigraphs.
-Wundef	Controls a warning that is issued for undefined symbols in preprocessor expressions. Such undefined symbols are always given the value of 0.
--no_commons	Disables commons support for denser code.
--no_exceptions	Controls support for exception handling. When enabled, code size and speed may be impacted even when exception handling is not directly used.

Compiler Option	Description
--prototype_errors	Controls the treatment of functions referenced or called when no prototype has been provided.
--incorrect_pragma_warnings	Controls the treatment of valid #pragma directives that use the wrong syntax.

GHS Assembler Options

Assembler Option	Description
-cpu=cortexm4f	Specify a target processor with the driver.
-preprocess_assembly_files	Controls whether assembly files with standard extensions such as .s and .asm are preprocessed.
-list	Creates a listing by using the name and directory of the object file with the .lst extension.

GHS Linker Options

Linker Option	Description
-e Reset_Handler	Specifies the program entry point or start address as Reset_Handler
-map	Creates a map file with the name of the object file plus a .map extension.
-keepmap	Controls the retention of the map file in the event of a link error.
-Mn	Generates a listing of symbols sorted numerically by address.
-delete	Controls the removal of unused and unreferenced functions from an executable.
-ignore_debug_references	Ignores relocations from DWARF debug sections when using -delete.
-nostartfile	Controls the start files to be linked into the executable.
-T file.ld	Reads additional options, memory maps, and section maps from the linker directives file file.ld.

2.4.3. KEIL Compiler/Assembler/Linker Options

KEIL Compiler Options

Compiler Option	Description
-mcpu=cortex-m4	Specify a target processor.
-mfpv4-sp-d16	Specifies the target FPU architecture.
-mfloat-abi=hard	Specifies to use hardware instructions for floating-point operations.
-mlittle-endian	Use this option to specify the byte order for your project.

KEIL Assembler Options

Assembler Option	Description
-mcpu=cortex-m4	Specify a target processor.
-mfpv4-sp-d16	Specifies the target FPU architecture. Enable the Armv7 FPUv4-SP-D16 Floating-point Extension for single precision only.
-mfloat-abi=hard	Specifies to use hardware instructions for floating-point operations.
-masm=auto	Automatically detect the correct assembler from the syntax of the input assembly source file.
-Wa,armasm, -pd,"START_FROM_SRAM SETA 1"	Define variable START_FROM_SRAM as 1 when start program from ram.

KEIL Linker Options

Linker Option	Description
--entry Reset_Handler	Specifies the unique initial entry point of the image.
--scatter	Specifies the scatter file path.

3. Changes List for Z20K14xM MCAL

3.1. RTM 1.2.1

Module	Description
Adc	• Bugfix: Fix incorrect limit check issue when left align is selected. ◦ Detailed Description: If limit check is enabled and left align is selected, the result of limit check is not correct. • Bugfix: Incorrect group is started in Adc_StopGroupConversion() when queue is enabled. ◦ Detailed Description: When multiple groups are started and not all group conversions are completed, then current group in conversion is stopped, it will cause incorrect conversion start of left groups. • Bugfix: Modify generation of symbolic names according to AUTOSAR specification. • Example: Set value of McuLVDUnderActiveEn and McuLVDResetEn parameters as true for 144/146/148 examples. • Example: Update initialization of MCU module with unified method for 144/146/148 examples.

Module	Description
Can	- Bugfix: Fix no acknowledge error. - Detailed Description: In case of high bus load, there is possibility that CAN messages can't get replied. This will cause no acknowledge error. - Bugfix: Modify generation of symbolic names according to AUTOSAR specification. - Example: Set value of McuLVDUnderActiveEn and McuLVDResetEn parameters as true for 144/146/148 examples. - Example: Update initialization of MCU module with unified method for 144/146/148 examples.
Crypto	- Bugfix: Modify generation of symbolic names according to AUTOSAR specification. - Optimization: Remove undefined CRYPTO_ALGOFAM_ECIES in enumeration Crypto_AlgorithmFamilyType. - Example: Set value of McuLVDUnderActiveEn and McuLVDResetEn parameters as true for 144/146/148 examples. - Example: Update initialization of MCU module with unified method for 144/146/148 examples.
Dio	- Bugfix: Modify generation of symbolic names according to AUTOSAR specification. - Example: Set value of McuLVDUnderActiveEn and McuLVDResetEn parameters as true for 144/146/148 examples. - Example: Update initialization of MCU module with unified method for 144/146/148 examples.
Dma	- Example: Set value of McuLVDUnderActiveEn and McuLVDResetEn parameters as true for 144/146/148 examples. - Example: Update initialization of MCU module with unified method for 144/146/148 examples.
Fls	- Bugfix: Modify generation of symbolic names according to AUTOSAR specification. - Example: Set value of McuLVDUnderActiveEn and McuLVDResetEn parameters as true for 144/146/148 examples. - Example: Update initialization of MCU module with unified method for 144/146/148 examples.
Gpt	- Bugfix: Modify generation of symbolic names according to AUTOSAR specification. - Example: Set value of McuLVDUnderActiveEn and McuLVDResetEn parameters as true for 144/146/148 examples. - Example: Update initialization of MCU module with unified method for 144/146/148 examples.

Module	Description
I2c	• Example: Set value of McuLVDUnderActiveEn and McuLVDRResetEn parameters as true for 144/146/148 examples. • Example: Update initialization of MCU module with unified method for 144/146/148 examples.
Icu	• Bugfix: Modify generation of symbolic names according to AUTOSAR specification. • Example: Set value of McuLVDUnderActiveEn and McuLVDRResetEn parameters as true for 144/146/148 examples. • Example: Update initialization of MCU module with unified method for 144/146/148 examples.
Lin	• Bugfix: Enumeration Definition of Lin_FrameCsModelType is not consistent with AUTOSAR. ◦ Detailed Description: If user's Lin_FrameCsModelType is different with ZhiXin, wrong checksum model of LIN frame is configured. • Bugfix: Modify generation of symbolic names according to AUTOSAR specification. • Optimization: Update handling of Lin_StatusType by removing type casting. • Example: Set value of McuLVDUnderActiveEn and McuLVDRResetEn parameters as true for 144/146/148 examples. • Example: Update initialization of MCU module with unified method for 144/146/148 examples.
McalLib	• Optimization: Change CounterId type from uint32 to CounterType when AUTOSAR OS is used.
Mcu	• Bugfix: Modify generation of symbolic names according to AUTOSAR specification. • Example: Set value of McuLVDUnderActiveEn and McuLVDRResetEn parameters as true for 144/146/148 examples. • Example: Update initialization of MCU module with unified method for 144/146/148 examples.
Ocu	• Bugfix: Modify generation of symbolic names according to AUTOSAR specification. • Example: Set value of McuLVDUnderActiveEn and McuLVDRResetEn parameters as true for 144/146/148 examples. • Example: Update initialization of MCU module with unified method for 144/146/148 examples.

Module	Description
Port	• Bugfix: Modify generation of symbolic names according to AUTOSAR specification. • Example: Set value of McuLVDUnderActiveEn and McuLVDResetEn parameters as true for 144/146/148 examples. • Example: Update initialization of MCU module with unified method for 144/146/148 examples.
Pwm	• Bugfix: Modify generation of symbolic names according to AUTOSAR specification. • Example: Set value of McuLVDUnderActiveEn and McuLVDResetEn parameters as true for 144/146/148 examples. • Example: Update initialization of MCU module with unified method for 144/146/148 examples.
Spi	• Bugfix: Configuration parameter SpiPhyUnitDmaPatch is added to switch the workaround of Errata DMA.1 on or off. ◦ Detailed Description: The workaround of Errata DMA.1 is added when SpiPhyUnitDmaPatch is checked, and the workaround of Errata DMA.1 is not added when SpiPhyUnitDmaPatch is unchecked. • Bugfix: Modify generation of symbolic names according to AUTOSAR specification. • Example: Set value of McuLVDUnderActiveEn and McuLVDResetEn parameters as true for 144/146/148 examples. • Example: Update initialization of MCU module with unified method for 144/146/148 examples.
Wdg	• Bugfix: Modify generation of symbolic names according to AUTOSAR specification. • Optimization: Update handling of WdgIf_ModeType by removing type casting. • Example: Set value of McuLVDUnderActiveEn and McuLVDResetEn parameters as true for 144/146/148 examples. • Example: Update initialization of MCU module with unified method for 144/146/148 examples.
Ewdt	• Bugfix: Modify generation of symbolic names according to AUTOSAR specification. • Example: Set value of McuLVDUnderActiveEn and McuLVDResetEn parameters as true for 144/146/148 examples. • Example: Update initialization of MCU module with unified method for 144/146/148 examples.

3.2. RTM 1.2.0

3.2.1. New Features

Module	Description
Adc	• Add AdcGroupTriggerDelayMode parameter to configure trigger delay for each group channel.
Can	• Add error injection support.
Crypto	• Crypto module is newly added.
Dma	• Modify nested configuration structure to structure pointer. • Add new API services Dma_GetChannelTransferConfig() and Dma_GetChannelGlobalParam(). • Move callback function from driver layer to autosar layer and add callback parameter LogicChIndex • Add configured logic id macro definition, such as "DMA_LOGIC_CH_0" • Move MajorIntEn from TransferConfig to GlobalConfig and move DmaChannelMajorInterruptEn to DmaChannelInterruptConfig in EB tresos
Ewdt	• Ewdt module is newly added.
Gpt	• Add rtc periodic interrupt support.
Lin	• Add a new feature regarding the polling mode of lin, where interrupts are turned off.
I2c	• I2c module is newly added.
Mcu	• Add PARCC_I2C1 option of McuParccName in EB configuration for Z20K144M. • Add McuGetHwTestStateFunc parameter to configure API to get current module self test state in SafetyLib. • Add McuLVDInterruptCallbackFunc and McuLVWInterruptCallbackFunc to configure callback function when LVD interrupt and LVW interrupt occurred. • Add interrupt handler to process LVD interrupt and LVW interrupt. • Add McuFOSCClockLossCallbackFunc and McuFIRC64MClockLossCallbackFunc to configure callback function when loss of FOSC clock and loss of FIRC clock interrupt occurred. • Add interrupt handler of loss of FIRC clock and loss of FOSC clock. • Support to configure multiple clock settings. • Support to get new reset reason(Reset caused by a standby wakeup).
Platform	• Remove Drvw layer.

3.2.2. Bug Fix

Module	Description
Adc	• Fix notifications are invoked correctly when group queuing is enabled. ◦ Detailed Description: If ADC group queuing is enabled, ADC group notification cannot be invoked correctly when ADC group conversion is completed. • Add workaround for Errata ADC.1. ◦ Detailed Description: According to Errata ADC.1, Adc_SelfTest function is added to perform first conversion and discard the conversion result.
Can	• Fix DET error type wrongly reported. ◦ Detailed Description: Fix DET error type reported wrong in Can_Deinit, Can_GetControllerTxErrorCounter and Can_GetControllerRxErrorCounter services. • Fix CanFIFODmaErrorNotification fail to be called • Remove CAN_MSG_ID_MIXED in type CanIdMessageType • Optimize the judgment when statement changes from sleep to stop
Dma	• Fix unexpected enable I2S0 request source error. ◦ Detailed Description: when DmaChannelRequestEn is enabled and DmaChannelMuxRequestSource is selected as DMA_REQ_SOFTWARE, I2S0 request source is unexpectedly enabled. this situation is not allowed now. • Fix physical id and priority level unique check error. ◦ Detailed Description: Physical id and priority level of multiple channels must be unique in EB tresos. • Fix channel related registers possible write failure problem. ◦ Detailed Description: When anyone of 16 channels is in busy status, attempting to write channel related registers will not take effect. Attempt to write and read back until success as a workaround.
McalLib	• Remove McalLibSystemTickTimerClockSource parameter. ◦ Detailed Description: Since clock source of system tick timer support core clock only, related configuration shall be removed • Incorrect service id is reported from DET. ◦ Detailed Description: Originally fixed service id is reported from DET, it is changed to use correct service id now.

Module	Description
Mcu	• Change configurable parameters default value according to RM 1.1. ◦ Detailed Description: Change configurable parameters (incl. system default clock source, bus clock divider, core clock divider, slow clock divider, LVD detect under active mode...) default value according to RM 1.1. • Add condition check of McuLVDUnderActiveEn status when enable McuLVDUnderLowPowerEn. ◦ Detailed Description: According to RM 1.1, LVD detect is effective under low power mode only when both VDD_LVD_ACT_EN VDD_LVD_LP_EN are set to 1. • Set LOCKWIN with 3 when enable PLL clock. ◦ Detailed Description: It is recommended to set LOCKWIN with 3 when enable PLL clock.
Ocu	• Fix Ocu Ocu_ChannelType symbolic name. • Fix TIM_OUTSWCR bit operation error problem.
Platform	• Correct definition of TIM0 Match Trigger in Tmu_Drv_InputChannelType ◦ Detailed Description: definition of TIM0 Match Trigger in Tmu_Drv_InputChannelType is inconsistent of resource file.
Pwm	• Fix duty cycle setting and notification recovery issues. ◦ Detailed Description: Fix the issue of setting the 100% duty cycle in edge alignment or complementary edge alignment mode. Add a duty cycle check for TIM channels in center aligned mode. Notification function recovery call, when exiting idle state or with a duty cycle between 0% and 100%. • Fix issues with channel duty cycle dither and period dither settings. ◦ Detailed Description: Fix the issue of resetting the period dither and duty cycle dither values of the MCPWM channel after setting the period and duty cycle.
Spi	• Fix Spi transmit issue when using Dma. ◦ Detailed Description: Add Dma errata workaround to fix Spi transmit issue when using Dma. • Fix Spi transmit failed when using GPIO to control CS signal. ◦ Detailed Description: Whatever user select for SpiEnableCs or SpiCsSelection in EB, PCS is available based on SpiCsIdentifier

Module	Description
Gpt	• Fix timer tick counting issue. ◦ Detailed Description: The time takes one tick to go from 65535 to 0. • Fix timer multi-channel stop issue. ◦ Detailed Description: The timer one channel of timer instance is disabled, the other channels of timer instance will also stop running. • Fix channel Id Out-of-order problem. ◦ Detailed Description: The EB tool out of order configuration channel, generated code channel Id out of order can not match.

3.2.3. Examples

- Add Keil project in Port 144/146/148 examples.
- Add Makefile project of ghs compiler in Port 148 example.

3.2.4. Docs

- Add quick start guide.

3.3. RTM 1.1.0

3.3.1. New Features

Module	Description
Adc	• Add hardware average selection support. Configuration parameter AdcGroupHardwareAverageSelect is added to support this feature. • Remove AdcEnableCalibration parameter.
Base	• Remove CanIf_MemMap.h, Dem_MemMap.h, Det_MemMap.h, EcuM_MemMap.h memory map header files of stubbed modules. • Remove Gcc support from memory map header files. • Add Keil support from memory map header files.
Icu	• Add port IP support gpio wakes up in stop mode.
Mcu	• Add Mcu module initialization check in Non-AUTOSAR services.
Platform	• Remove Gcc support files, including startup.s and linker files. • Add Keil support files, including startup.s and linker files. • Change unused IrqHandler name to undefined_handler.

3.3.2. Bug Fix

Module	Description
Adc	• Fix ADC read data problem when Hardware trigger and DMA is used. ◦ Detailed Description: When DMA transfer is selected and ADC Group is configured as hardware trigger, one-shot, streaming mode, ADC data read from result buffer is not correct. • Fix result buffer access out of range problem. ◦ Detailed Description: When hardware trigger is used and group is configured as streaming access, linear buffer mode, result buffer access out of range will occur if trigger event is continuously generated. • Fix incorrect generated result of Adc_HWUnitConfigList ◦ Detailed Description: In case of disordered HwUnit Configuration, incorrect Adc_HWUnitConfigList is generated. • Fix limit check problem of ADC0 Channel-28,29,30 ◦ Detailed Description: When global limit check is enabled and limit check of Channel-28,29,30 are configured, results of limit check are always failed.

Module	Description
Can	• Fix compile error when CanWakeupFunctionalityAPI set to true and CanWakeupSupport set to false. ◦ Detailed Description: There is compile error when CanWakeupFunctionalityAPI set to true and CanWakeupSupport set to false. • Fix CAN driver donot disable automatic busoff recovery issue, correct manual recovery from busoff implementation. ◦ Detailed Description: Automatic busoff recovery is not disabled in CAN driver. And the condition check of enter busoff is not correct. • Correct Can and CanFd baudrate configuration parameters range. ◦ Detailed Description: Range of baudrate configuration parameter(CanControllerPropSeg, CanControllerSeg1, CanControllerSeg2 and CanControllerSyncJumpWidth) is not correct. • Fix CAN controller cannot receive standard message if CanIdType configured with MIXED in non-fifo mode. ◦ Detailed Description: If CanIdType of HRH configured with MIXED while HRH configured with normal mode, it cannot receive standard frame. • Fix CAN controller cannot receive standard type frame if CanIdType configured with MIXED in FIFO mode. ◦ Detailed Description: If CanIdType of HRH configured with MIXED while HRH configured with FIFO mode, it cannot receive standard frame. • Fix cannot configure CAN controller with reject all frame. ◦ Detailed Description: CanIdAcceptanceMode cannot be configured with REJECT_ALL, and it leads to the HRH in FIFO mode cannot reject all frames. • Fix parameter CanPropDelayTranceiver cannot change when CanTimeSegmentAutomaticCalculation set to true. ◦ Detailed Description: CanPropDelayTranceiver value cannot be changed if CanTimeSegmentAutomaticCalculation set to true. • Fix if more than two errors created at the same time, CAN driver cannot report error to CAN interface issue. ◦ Detailed Description: If two types and above errors occurred at the same time, errors cannot be reported to CanIf correctly. • Fix Can_Init det report to init failed. ◦ Detailed Description: If input parameter of Can_Init() is NULL_PTR in PostBuild or is not NULL_PTR in PreCompile, CAN_E_PARAM_POINTER is wrongly reported to DET module.

Module	Description
Dio	- macro name for channel group configuration are incorrect. - Detailed Description: The channel group macro name does not match what the user expects, resulting in the failure to perform the channel group function
Dma	- Add set source&dest addr interface layer function. - Detailed Description: Setting source & destination address by Dma_ConfigChannelTransfer() will overwrite other parameters configured in EB. There is no other way to set source & destination address independently and they are not configurable in EB. - Fix error status check bug when error interrupt disabled. - Detailed Description: When dma error interrupt disabled, reading wrong error status register leads to channel-0 always in error state. If error occurs on multiple channels, error state will not update. - Fix error channel&type mismatch issue - Detailed Description: if dma error occurs between Dma_Drv_GetLastErrorChannel() and Dma_Drv_GetLastErrorType(), error channel and error type is from different error event.
Fls	- Generation result deviates for Fls_FlsConfigSet. - Detailed Description: Declaration of the structural variable Fls_FlsConfigSet deviates from its definition under post-build configuration. - Handle multi-bit ecc error and flash read error. - Detailed Description: Multi-bit ecc error cannot be detected and handled normally when it occurs during flash read, compare or blank check.

Module	Description
Gpt	- Update the order of variables in Gpt_Drvw_HwPredefChannelConfigType. Update the order of variables in Gpt_Drvw_HwPredefChannelConfigType. - Detailed Description: The generated Gpt_Drvw_HwPredefChannelConfigType struct variables are not in the same order as defined in Gpt_Drvw_Types.h. - Fix error that Gpt Predef Timer 1us count could not be implemented. The cause is prescaler is not enabled in Stim_Drv_StartPredefTimer - Detailed Description: The counting period of the Gpt Predef Timer is not 1 us. - Fix error about Predef Timer clock source generation. The cause is that the generation clock source judgment condition name is incorrect. - Detailed Description: The clock source selected by the Predef Timer is not generated. - Fix channels with relatively large CV values in the same instance cannot generate interrupts normally. - Detailed Description: Disable one channel also stops the other channels in instance, so channels with a larger CV value cannot generate interrupt. - Fix an issue where the timer interrupt processing time increased. - Detailed Description: Because the interrupt is not closed after the timer is disabled, the interrupt handler is entered again if CHF is set. - Fix Gpt_GetElapsedTime return value error. This is because an interrupt occurred between getting the counter value and the compare value. - Detailed Description: Because an interrupt will cause the two values not to be in the same period. - Fix an issue where the interrupt period error of the Gpt timer channel became larger after it was running for a long time. - Detailed Description: Because flipping to 0 after reaching 0xFFFF requires 1 tick. - Fix an error in obtaining remaining value of Gpt. - Detailed Description: This is because the increment 1 operation is performed when obtaining the CV value in the Gpt_Drvw_RtcGetTimeRemaing function. - Fix an issue where STIM_DRV_ENABLE is not enabled when only Predef timer is configured. - Detailed Description: Because STIM_DRV_ENABLE is missing a condition. - Fixed an issue with channel calling Callback functions that did not meet expectations. - Detailed Description: This is because the interrupt handler calls the Callback without determining whether the channel is running.

Module	Description
Icu	• Fix Line buffer mode Icu_GetTimestampIndex returns a value that exceeds the notification call limit in enable dma. ◦ Detailed Description: Because when the capture functionality reaches the end of the buffer dma is not closed • Fix in post-build generate code,The generated dma callback code is overwritten. ◦ Detailed Description: Because the callback function is defined in a Cfg file, and the previous configuration is overridden • Fix Icu_GetInputState api service id definition error. • Fix Icu_Config is defined static in Variant PostBuild from VECTOR. • Fix "Icu_MemMap.h" duplicate definition from VECTOR.
Ocu	• Fix OCU EB configuration error. ◦ Detailed Description: build error with different EB configurations of pb,pc, error detect and apis. • Fix parameter attributes to fix warnings from VECTOR.
Lin	• Fix get lin status error by calling Lin_GetStatus() ◦ Detailed Description: When lin status is sleep and lin config as master,user get lin status by calling function Lin_GetStatus(),that will return error. • Fix LIn node is master,some function cause compilation error ◦ Detailed Description: When lin config as master ,some Linif functon will causing compilation errors.
McalLib	• Fix incorrect generated CounterId of McalLib_Config structure. ◦ Detailed Description: When MCALLIB_OS_AUTOSAR is selected, OsCounter adds an "U" to the generated result in McalLib_Config structure. • Fix McalLib_OsCounter_SetCounterFreq() undefined problem ◦ Detailed Description: When MCALLIB_OS_AUTOSAR is selected, McalLib_OsCounter_SetCounterFreq() is not defined. • Macro MCALLIB_DEV_ERROR_DETECT is missing for McalLib_DetReportError(). ◦ Detailed Description: Macro MCALLIB_DEV_ERROR_DETECT is missing for McalLib_DetReportError(). Macro MCALLIB_DEV_ERROR_DETECT is added to enable/disable McalLib_DetReportError().

Module	Description
Mcu	• Fix calculated slow clock frequency and bus clock frequency are not correct issue. ◦ Detailed Description: Formular of McuSysBusClockFrequency and McuSysSlowClockFrequency are not correct. • Fix reset reason register is not cleared after read out issue. ◦ Detailed Description: After MCU last reset reason is read out, data in reset reason register(SRMC_SSRS) is not been cleared. • Fix clockout output frequency is not correct if core clock selected as clockout source and core clock source changed issue. ◦ Detailed Description: If core source is changed during running, the frequency of clockout output is not changed accordingly. • Fix LPO32K cannot be disabled issue. ◦ Detailed Description: If LPO32K clock is configured with disable in EB, LPO_CLK_DIS is not set with 1(disabled). • Fix mcu cannot perform reset when FIRC64M clock is not in ready state. ◦ Detailed Description: When FIRC64M clock is not in ready state, the MCU cannot perform reset. • Fix polarity of wakeup source and enabled status is not set correctly issue. ◦ Detailed Description: If enable specify wakeup source, the related field in register is 0(disabled), if set active polarity of specify wakeup is high the related field in register is 0(active low). • Fix MCU_E_ALLREADY_INITIALIZED error reported wrongly issue. ◦ Detailed Description: If input parameter is wrong while calling Mcu_Init(), DET error MCU_E_ALLREADY_INITIALIZED is wrongly reported. • Remove Mcu_DistributePllClock switched by MCU_INIT_CLOCK macro. ◦ Detailed Description: If MCU_INIT_CLOCK is configured with STD_OFF, Mcu_DistributePllClock() API cannot be used.
Platform	• Fix CMP0_SAPMLE spelling mistake. ◦ Detailed Description: Spelling of TMU_DRV_OUTPUT_CHANNEL_CMP0_SAMPLE in Tmu_Drv_Types.h is not correct. • Remove the disabling of watchdog in Startup.s ◦ Detailed Description: Since the disabling of watchdog is not allowed for functional safety requirements, remove disable watchdog implementation in Startup.s.

Module	Description
Pwm	- Fix the error where PWM_E_PARAM_CHANNEL issues cannot be reported correctly. - Detailed Description: Separate the error detection of PWM_E_PARAM_CHANNEL and PWM_E_PARAM_CONFIG, and if the previous detection is incorrect, do not proceed to the next detection.
Resource	- Fix incorrect name "IO_TMU_OUT05" of Platform.TmuOutputChannelList - Detailed Description: "IO_TMU_OUT05" of Platform.TmuOutputChannelList in all properties files is inconsistent with type definition "IO_TMU_OUT5" in Tmu_Drv_Types.h. - Fix comma is missing in Platform.TmuInputChannelList - Detailed Description: Comma after I2C0_TX_TRIG of Platform.TmuInputChannelList is missing in Z20K144M_lqfp100 properties file. - Fix incorrect default IrqHandler name for CAN6 and CAN7. - Detailed Description: Default IrqHandler name of Can_Drv_6_Mb0To31IrqHandler, Can_Drv_6_Mb32To63IrqHandler, Can_Drv_7_Mb0To31IrqHandler, Can_Drv_7_Mb32To63IrqHandler is not correct.
Spi	- When set SpiChannelBuffersAllowed == 0, it is also set spi channel type to EB - Detailed Description: When set SpiChannelBuffersAllowed == 0 in EB,it is should not set spi channel type to EB. - Spi_CheckReadIB set wrong det parameter. - Detailed Description: Spi_CheckReadIB should report SPI_E_PARAM_CHANNEL when channel type is not equal to EB.

Module	Description
Wdg	• Fix NULL_PTR is used with unexpected. ◦ Detailed Description: When WdgDisableAllowed==true and default mode is WDGIF_OFF_MODE,Wdg_StartGpt() will access NULL_PTR with unexpected. • WdgAllowUpdates is not work if it is configured disable. ◦ Detailed Description: When WdgAllowUpdates is configured to disable,the wdog register should not be writted again. • Refresh watchdog should use asm language. ◦ Detailed Description: Wdog_Drv_Refresh() should be code with asm language. • All Wdog_Drv_WaitForConfigComplete should check the result. ◦ Detailed Description: Fixed the return value of Wdog_Drv_WaitForConfigComplete() is not checked. • Wdg_Init() should no check Wdg_mode == OFF mode. ◦ Detailed Description: Delete Wdg_mode == OFF code,because it is not used anymore. • Wdg_Init() should not disable WDOG. ◦ Detailed Description: Fixed WDOG will be disabled during Wdg_Init(). • before Wdg_Init(),it is should not assert wdg is disabled. ◦ Detailed Description: Wdg_Init() should handle wdg based on EB configuration. • before call SuspendAllInterrupts(), it should check PriMaskReg. ◦ Detailed Description: before call SuspendAllInterrupts(), it should check whether the interrupts have been already suspended. • it is not need unlock before refresh watchdog hardware. ◦ Detailed Description: Delete the unlock action before refresh watchdog. • Fix PB build is change To S0 with s0 in value name ◦ Detailed Description: When PB build is enabled and variant is set to s0, related variable name should have suffix s0 instead of S0.

3.3.3. Examples

- Examples of all modules are updated to follow coding style.
- Readme.txt is added for all modules.

3.4. RTM 1.0.0

3.4.1. New Features

Module	Description
Adc	• Add SWCD arxml • Add clock reference points to MCU • Add post-build support • Remove driver wrapper layer to improve performance
Can	• Add SWCD arxml • Add post-build support
Dio	• Add SWCD arxml
Dma	• Add SWCD arxml • Add configuration parameter of clock reference points to MCU • Add post-build support
Fls	• Add SWCD arxml • Add configuration parameter of clock reference points to MCU • Support double bit fault detect interrupt enable/disable • Add post-build support
Gpt	• Add SWCD arxml • Add TIM IP support • Add post-build support
Icu	• Add SWCD arxml • Add configuration parameter of clock reference points to MCU • Add CMP IP support • Add post-build support
Lin	• Add SWCD arxml • Add post-build support • Use software simulated LIN slave to received header
Mcu	• Add SWCD arxml • Add post-build support
Ocu	• Add SWCD arxml • Add post-build support • Add OCU pin DMA trigger feature

Module	Description
Port	• Add SWCD arxml • Add post-build support • Add clock reference check when configuration • Fix version check macro in Port_Cfg.h • Add AutoSAR parameter PortPinInitialMode in xdm
Pwm	• Add SWCD arxml • Add post-build support • Add TIM IP support
Spi	• Add SWCD arxml • Add post-build support
Wdg	• Add SWCD arxml • Add post-build support

3.4.2. Bug Fix

Module	Description
Adc	• Fix incorrect conversion result after calibration • To avoid external interference, IREF_SEL is configured as internal reference. • Fix incorrect conversion result when streaming access mode is used in DMA transfer mode. • Parameter AdcChannelPhysicalId is renamed as AdcChannelId
Can	• Set CanObjectPayloadLength feature to not editable • Delete fault injection and fix sleep to sleep state flow • Add CanTTController container • Fix parameter definition errors and warnings of xdm file • Fix generate macro CAN_DRV_FD_MODE_ENABLE definition in Can_Drv_Cfg.h • Fix symbolic name generated for can controller and hardware object in Can_Cfg.h
Dma	• Fix Register offset error
Fls	• Fix parameter definition errors and warnings of xdm file
Icu	• Fix parameter definition errors and warnings of xdm file

Module	Description
Mcu	• Correct definitions of reset reasons • Correct definition of Mcu_Config variable • Limit to support only one clock setting • Clear pmu isolation clear register before initializing mcu • Fix issue of wakeup source polarity cannot set to low. • Fix parameter definition errors and warnings of xdm file
Ocu	• Fix Ocu bug that different instance set different max counter value fail • Fix parameter definition errors and warnings of xdm file • Fix ocu set pin action error • Fix channel state check function
Port	• Fix version check macro in Port_Cfg.h • Fix port variant issue when only one variant is selected • Fix parameter definition errors and warnings of xdm file
Pwm	• Modify the pwm logical channel ID limit • Modify Pwm_Instances generation template • Fix parameter definition errors and warnings of xdm file • Modify the minimum and default values of PwmMcpwmDeadTime and PwmMcpwmPairedChPhaseShift in the xdm file

3.4.3. Docs

- User manuals are added for all modules.

4. Version Tracking

Date	Version	Comments
2023/1/2	Beta0.6.0	Beta0.6.0 for Z20K14xM MCAL
2023/2/17	Beta0.7.0	Beta0.7.0 for Z20K14xM MCAL
2023/4/28	RTM1.0.0	RTM1.0.0 for Z20K14xM MCAL
2023/7/10	RTM1.1.0	RTM1.1.0 for Z20K14xM MCAL
2023/11/10	RTM1.2.0	RTM1.2.0 for Z20K14xM MCAL
2023/12/10	RTM1.2.1	RTM1.2.1 for Z20K14xM MCAL

How to Reach Us:

Home Page:

zhixin-semi.com/

Information in this document is provided solely to enable system and software implementers to use Zhixin products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. Zhixin reserves the right to make changes without further notice to any products herein. Zhixin makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Zhixin assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in Zhixin data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including “typicals,” must be validated for each customer application by customer’s technical experts. Zhixin does not convey any license under its patent rights nor the rights of others. Zhixin sells products pursuant to standard terms and conditions of sale, which can be found at the following address: <http://www.zhixin-semi.com>.

Zhixin and the Zhixin logo are trademarks of Zhixin Semiconductor Co., Ltd. All rights reserved.

©2023 Zhixin Semiconductor Co., Ltd.

Document Number: Z20K14xM-MCAL-RN

Revision 1.2.1, December, 2023

