MCUXpresso SDK Release Notes Supporting evkmimx8mq

Change Logs

Contents

Contents

Driver Change Log

CLOCK

Current CLOCK driver version is 2.3.2

- 2.3.2
  - Bug Fixes
    - Corrected and added clock information for IOMUX and IPMUX.
- 2.3.1
  - Bug Fixes
    - Fixed MISRA C-2012 rule 10.1,rule 10.4,rule 10.8,rule 20.7 and so on.
- 2.3.0
  - New feature:
    - Moved SDK_DelayAtLeastUs function from clock driver to common driver.
- 2.2.0
  - New feature
    - Added new API CLOCK_DelayAtLeastUs() implemented by DWT to allow users set delay in unit of microsecond.
- 2.1.2
  - Bug Fix:
    - Fixed build issue with GCC compiler when include header from C++ file.
- 2.1.1
  - Improvement:
    - Changed reserved bit fields in _scg_sys_clk_config struct into unnamed bit fields.
- 2.1.0
  - Other Changes:
    - Merged fsl_scg and fsl_osc into fsl_clock.
- 2.0.0
  - Initial version.

IOMUXC

The current IOMUXC driver version is 2.0.1.

- 2.0.1
  - Doxygen improvement.
- 2.0.0
  - Initial version.

ECSPI_CMSIS

Current ecspi_cmsis driver version is 2.1

- 2.1
  - Bug Fixes
    - Fixed the bug that, the parameter num of APIs ARM_SPI_Transfer, ARM_SPI_Send and ARM_SPI_Receive, and the return value of API ARM_SPI_GetDataCount should be the number of data item defined by datawidth, rather than the number of byte.
- 2.0
  - Initial version.

I2C

Current I2C CMSIS driver version is 2.1

- 2.1
  - Bug Fixes
    - Fixed the bug that in PowerControl, module should be reset first by calling I2C_MasterInit and I2C_SlaveInit.
- 2.0
  - Initial version.

UART

The current UART CMSIS driver version is 2.0.

- 2.0
  - Initial version.

COMMON

The current COMMON driver version is 2.2.4.

- 2.2.4
  - Bug Fixes
    - Fixed MISRA C-2012 rule-10.4.
- 2.2.3
  - New Features
    - Provided better accuracy of SDK_DelayAtLeastUs with DWT, use macro SDK_DELAY_USE_DWT to enable this feature.
    - Modified the Cortex-M7 delay count divisor based on latest tests on RT series boards, this setting let result be more close to actual delay time.
- 2.2.2
  - New Features
    - Added include RTE_Components.h for CMSIS pack RTE.
- 2.2.1
  - Bug Fixes

        - Fixed violation of MISRA C-2012 Rule 3.1, 10.1, 10.3, 10.4, 11.6, 11.9.
- 2.2.0
  - New Features
    - Moved SDK_DelayAtLeastUs function from clock driver to common driver.
- 2.1.4
  - New Features
    - Added OTFAD into status group.
- 2.1.3
  - Bug Fixes
    - MISRA C-2012 issue fixed.
      - Fixed the rule: rule-10.3.
- 2.1.2
  - Improvements
    - Add SUPPRESS_FALL_THROUGH_WARNING() macro for the usage of suppressing fallthrough warning.
- 2.1.1
  - Bug Fixes
    - Deleted and optimized repeated macro.
- 2.1.0
  - New Features
    - Added IRQ operation for XCC toolchain.
    - Added group IDs for newly supported drivers.
- 2.0.2
  - Bug Fixes
    - MISRA C-2012 issue fixed.
      - Fixed the rule: rule-10.4.
- 2.0.1
  - Improvements
    - Removed the implementation of LPC8XX Enable/DisableDeepSleepIRQ() function.
    - Added new feature macro switch "FSL_FEATURE_HAS_NO_NONCACHEABLE_SECTION" for specific SoCs which have no noncacheable sections, that helps avoid an unnecessary complex in link file and the startup file.
    - Updated the align(x) to **attribute**(aligned(x)) to support MDK v6 armclang compiler.
- 2.0.0
  - Initial version.

ECSPI

The current eCSPI driver version is 2.1.0.

- 2.1.0
  - Improvements
    - Added timeout mechanism when waiting certain states in transfer driver.
- 2.0.2

  - Bug Fixes
    - Fixed violations of MISRA C-2012 rules: 10.1, 10.3, 10.4
- 2.0.1
  - Bug Fixes
    - Memset local variable SDMA transfer configuration structure to make sure unused members in structure are cleared.
    - Fixed sign-compare warning in ECSPI_SendTransfer.
- 2.0.0
  - Initial version.

GPT

The current GPT driver version is 2.0.2.

- 2.0.2
  - Bug Fixes
    - Fixed violations of the MISRA C-2012 rules 17.7.
- 2.0.1
  - Bug Fixes
    - Fixed violations of the MISRA C-2012 rules 10.1, 10.3, 10.4, 10.6, 10.8, 17.7.
- 2.0.0
  - Initial version.

GPIO

The current GPIO driver version is 2.0.3.

- 2.0.3
  - Bug Fixes
    - MISRA C-2012 issue fixed.
      - Fixed rules, containing: rule-10.3, rule-14.4, and rule-15.5.
- 2.0.2
  - Bug Fixes
    - Fixed the bug of enabling wrong GPIO clock gate in initial API. Since some GPIO instances may not have a clock gate enabled, it checks the clock gate number and makes sure the clock gate is valid.
- 2.0.1
  - Improvements
    - API interface changes:
      - Refined naming of the API while keeping all original APIs, marking them as deprecated. Original APIs will be removed in next release. The main change is to update the API with prefix of _PinXXX() and _PortXXX().
- 2.0.0
  - Initial version.

I2C

The current I2C driver version is 2.0.6.

- 2.0.6
  - Bug Fixes
    - Fixed the bug that, in I2C_MasterStop after the stop command is issued, the IBB flag should be cleared rather than set.
    - Fixed the bug that to clear kI2C_ArbitrationLostFlag and kI2C_IntPendingFlag, their bits should be written '0' rather than '1'.
- 2.0.5
  - Bug Fixes
    - Fixed Coverity issue of unchecked return value in I2C_RTOS_Transfer.
    - Fixed MISRA issues.
      - Fixed rules 10.1, 10.3, 10.4, 11.9, 14.4, 15.7, 16.4, 17.7.
  - Improvements
    - Updated the I2C_WAIT_TIMEOUT macro to unified name I2C_RETRY_TIMES.
- 2.0.4
  - Bug Fixes
    - Fixed the issue that I2C Master transfer APIs(blocking/non-blocking) did not support the situation that master transfer with subaddress and transfer data size being zero, which means no data followed by the subaddress.
- 2.0.3
  - Improvements
    - Improved code readability, added new static API I2C_WaitForStatusReady for the status flag wait, and changed to call I2C_WaitForStatusReady instead of polling flags with reading register.
- 2.0.2
  - Improvements
    - Added I2C_WATI_TIMEOUT macro to allow users to specify the timeout times for waiting flags in functional API and blocking transfer API.
- 2.0.1
  - Bug Fixes
    - Added a proper handle for transfer config flag kI2C_TransferNoStartFlag to support transmit with kI2C_TransferNoStartFlag flag. Only supports write only or write+read with no start flag; does not support read only with no start flag.
- 2.0.0
  - Initial version.

PWM

The current PWM driver version is 2.0.1.

- 2.0.1
  - Bug Fixes

        * Fixed violations of the MISRA C-2012 rules 17.7.
- 2.0.0
  - Initial version.

UART

The current UART driver version is 2.1.0.

- 2.1.0
  - Improvements
    * Added timeout mechanism when waiting for certain states in transfer driver.
- 2.0.2
  - Improvements
    * Added check for transmission complete in UART_WriteBlocking, UART_TransferHandleIRQ and UART_SendSDMACallback to ensure all the data would be sent out to bus.
    * Modified UART_ReadBlocking so that if more than one receiver errors occur, all status flags will be cleared and the most severe error status will be returned.
  - Bug Fixes
    * Fixed MISRA issues.
      · Fixed rules 10.1, 10.3, 10.4, 10.6, 10.7, 10.8, 11.9, 14.4.
- 2.0.1
  - Bug Fixes
    * Memset local variable SDMA transfer configuration structure to make sure unused members in structure are cleared.
- 2.0.0
  - Initial version.

MU

The Current MU driver version is 2.0.5.

- 2.0.5
  - Bug Fixes
    * Fixed violations of the MISRA C-2012 rules 14.4, 15.5.
- 2.0.4
  - Improvements
    * Improved for the platforms which don't support reset assert interrupt and get the other core power mode.
- 2.0.3
  - Bug fixes
    * MISRA C-2012 issue fixed.
      · Fixed rules, containing: rule-10.3, rule-14.4, rule-15.5.
- 2.0.2

  - Improvements
    - Added support for MIMX8MQx.
- 2.0.1
  - Improvements
    - Added support for MCIMX7Ux_M4.
- 2.0.0
  - Initial version.

QSPI

The current QSPI driver version is 2.2.1.

- 2.2.1
  - Bug Fixes
    - Fixed violations of MISRA C-2012 rule 10.1, 10.3, 10.4, 10.6, 10.8, 11.3, 11.6, 11.8, 11.9, 14.4, 16.1, 16.4, 17.7.
- 2.2.0
  - New Features
    - Added new API QSPI_ClearCache to clear cache for new IP feature FSL_FEATURE_QSPI_SOCCR_HAS_CLR_LPCAC.
  - Bug Fixes
    - Fixed the QSPI_WriteBlocking API programming issue for low watermark, caused by previous improvement change of using TX watermark signal to fill the TX FIFO. Reverted change to previous implementation to use TX FIFO full flag for filling the FIFO. Improved previous API by accessing TX data register directly.
    - Fixed the issue that QSPI_SetIPCommandSize incorrectly triggered a transaction.
    - Fixed clock divider accurate issue when using internal QSPI internal divider.
    - Fixed build fail issue for some devices' not supporting API QSPI_SetDqsConfig for DQS configuration.
- 2.1.0
  - New Features
    - Added new API QSPI_SetDqsConfig for DQS configuration.
  - Improvements
    - Updated the QSPI_WriteBlocking API to fill the TX FIFO once there are bytes of TX watermark room in the FIFO. This will improve the performance of filling TX FIFO when watermark is high.
- 2.0.2
  - Improvements
    - New Macro function:
      - Added QSPI_LUT_SEQ() function for users to set LUT table easily.
      - Added LUT command macros for users to easy use.
    - Comment update:
      - Added the comments for the limitation of QSPI_ReadBlocking and QSPI_TransferReceiveBlocking.

- 2.0.1
  - Improvements
    - New API:
      - QSPI_SetReadArea to set the read area.
  - Bug Fixes
    - Fixed the issue that QSPI_UpdateLUT function only updated first LUT.
    - Fixed issue that some function that hardcode QSPI0 as base.
- 2.0.0
  - Initial version.

RDC

The current RDC driver version is 2.1.1.

- 2.1.1
  - Bug Fixes
    - Fixed violations of the MISRA C-2012 rules 10.6.
- 2.1.0
  - Improvements
    - Enhanced to support memory region larger than 32-bit address.
- 2.0.2
  - Bug Fixes
    - Fixed violations of the MISRA C-2012 rules 10.3, 10.4, 11.3, 11.8, 17.7.
- 2.0.1
  - Bug Fixes:
    - Added __DSB after new configuration is set to ensure the new configuration takes effect.
- 2.0.0
  - Initial version.

RDC_SEMA42

The current RDC_SEMA42 driver version is 2.0.3.

- 2.0.3
  - Improvements:
    - Supported the RDC_SEMAPHORE_Type structure whose gate registers are defined as an array.
- 2.0.2
  - Bug Fixes
    - Fixed violations of the MISRA C-2012 rules 10.3, 10.4, 10.8, 14.3, 14.4, 18.1.
- 2.0.1
  - Improvements:
    - Added support for the platforms that don't have dedicated RDC_SEMA42 clock gate.
- 2.0.0

  - Initial version.

SAI

The current SAI driver version is 2.3.1

- 2.3.1
  - Bug Fixes
    - Corrected the peripheral name in function SAI0_DriverIRQHandler.
    - Fixed violations of MISRA C-2012 rule 17.7.
- 2.3.0
  - Bug Fixes
    - Fixed the build error caused by the SOC has no fifo feature.
- 2.2.3
  - Bug Fixes
    - Corrected the peripheral name in function SAI0_DriverIRQHandler.
- 2.2.2
  - Bug Fixes
    - Fixed the issue of MISRA 2004 rule 9.3.
    - Fixed sign-compare warning.
    - Fixed the PA082 build warning.
    - Fixed sign-compare warning.
    - Fixed violations of MISRA C-2012 rule 10.3,17.7,10.4,8.4,10.7,10.8,14.4,17.7,11.6,10.-1,10.6,8.4,14.3,16.4,18.4.
    - Allow to reset Rx or Tx FIFO pointers only when Rx or Tx is disabled.
  - Improvements
    - Added 24bit raw audio data width support in sai sdma driver.
    - Disabled the interrupt/DMA request in the SAI_Init to avoid generates unexpected sai FIFO requests.
- 2.2.1
  - Improvements
    - Added mclk post divider support in function SAI_SetMasterClockDivider.
    - Removed useless configuration code in SAI_RxSetSerialDataConfig.
  - Bug Fixes
    - Fixed the SAI SDMA driver build issue caused by the wrong structure member name used in the function SAI_TransferRxSetConfigSDMA/SAI_TransferTxSetConfigSDMA.
    - Fixed BAD BIT SHIFT OPERATION issue caused by the FSL_FEATURE_SAI_CHA-NNEL_COUNTn.
    - Applied ERR05144: not set FCONT = 1 when TMR $>$ 0, otherwise the TX may not work.
- 2.2.0
  - Improvements
    - Added new APIs for parameters collection and simplified user interfaces:
      - SAI_Init
      - SAI_SetMasterClockConfig

        - SAI_TxSetBitClockRate
        - SAI_TxSetSerialDataConfig
        - SAI_TxSetFrameSyncConfig
        - SAI_TxSetFifoConfig
        - SAI_TxSetBitclockConfig
        - SAI_TxSetConfig
        - SAI_TxSetTransferConfig
        - SAI_RxSetBitClockRate
        - SAI_RxSetSerialDataConfig
        - SAI_RxSetFrameSyncConfig
        - SAI_RxSetFifoConfig
        - SAI_RxSetBitclockConfig
        - SAI_RXSetConfig
        - SAI_RxSetTransferConfig
        - SAI_GetClassicI2SConfig
        - SAI_GetLeftJustifiedConfig
        - SAI_GetRightJustifiedConfig
        - SAI_GetTDMConfig
- 2.1.9
  - Improvements
    - Improved SAI driver comment for clock polarity.
    - Added enumeration for SAI for sample inputs on different edges.
    - Changed FSL_FEATURE_SAI_CHANNEL_COUNT to FSL_FEATURE_SAI_CHANNEL_COUNTn(base) for the difference between the different SAI instances.
  - Added new APIs:
    - SAI_TxSetBitClockDirection
    - SAI_RxSetBitClockDirection
    - SAI_RxSetFrameSyncDirection
    - SAI_TxSetFrameSyncDirection
- 2.1.8
  - Improvements
    - Added feature macro test for the sync mode2 and mode 3.
    - Added feature macro test for masterClockHz in sai_transfer_format_t.
- 2.1.7
  - Improvements
    - Added feature macro test for the mclkSource member in sai_config_t.
    - Changed "FSL_FEATURE_SAI5_SAI6_SHARE_IRQ" to "FSL_FEATURE_SAI_SAI5_SAI6_SHARE_IRQ".
    - Added #ifndef #endif check for SAI_XFER_QUEUE_SIZE to allow redefinition.
  - Bug Fixes
    - Fixed build error caused by feature macro test for mclkSource.
- 2.1.6
  - Improvements
    - Added feature macro test for mclkSourceClockHz check.
    - Added bit clock source name for general devices.

  - Bug Fixes
    - Fixed incorrect channel numbers setting while calling RX/TX set format together.
- 2.1.5
  - Bug Fixes
    - Corrected SAI3 driver IRQ handler name.
    - Added I2S4/5/6 IRQ handler.
    - Added base in handler structure to support different instances sharing one IRQ number.
  - New Features
    - Updated SAI driver for MCR bit MICS.
    - Added 192 KHZ/384 KHZ in the sample rate enumeration.
    - Added multi FIFO interrupt/SDMA transfer support for TX/RX.
    - Added an API to read/write multi FIFO data in a blocking method.
    - Added bclk bypass support when bclk is same with mclk.
- 2.1.4
  - New Features
    - Added an API to enable/disable auto FIFO error recovery in platforms that support this feature.
    - Added an API to set data packing feature in platforms which support this feature.
- 2.1.3
  - New Features
    - Added feature to make I2S frame sync length configurable according to bitWidth.
- 2.1.2
  - Bug Fixes
    - Added 24-bit support for SAI eDMA transfer. All data shall be 32 bits for send/receive, as eDMA cannot directly handle 3-Byte transfer.
- 2.1.1
  - Improvements
    - Reduced code size while not using transactional API.
- 2.1.0
  - Improvements
    - API name changes:
      - SAI_GetSendRemainingBytes -> SAI_GetSentCount.
      - SAI_GetReceiveRemainingBytes -> SAI_GetReceivedCount.
      - All names of transactional APIs were added with "Transfer" prefix.
      - All transactional APIs use base and handle as input parameter.
      - Unified the parameter names.
  - Bug Fixes
    - Fixed WLC bug while reading TCSR/RCSR registers.
    - Fixed MOE enable flow issue. Moved MOE enable after MICS settings in SAI_TxInit/SAI_RxInit.
- 2.0.0
  - Initial version.

SEMA4

The current SEMA4 driver version is 2.0.2.

- 2.0.2
  - Improvements:
    - Supported the SEMA4_Type structure whose gate registers are defined as an array.
- 2.0.1
  - Bug Fixes
    - Fixed violations of the MISRA C-2012 rules 10.3, 10.4, 15.5, 18.1, 18.4.
- 2.0.0
  - Initial version.

TMU

The current TMU driver version is 2.0.3.

- 2.0.3
  - Bug Fixes
    - Fixed the violations of MISRA 2012 rules:
      - Rule 10.1 10.3 10.4 17.7.
- 2.0.2
  - Bug Fixes
    - Fixed missing right pair definition for extern C.
- 2.0.1
  - New Features
    - Added control macro to enable/disable the CLOCK code in current driver.
- 2.0.0
  - Initial version.
  - This module was first developed on i.MX 8MQuad.

WDOG

The current WDOG driver version is 2.1.1.

- 2.1.1
  - Bug Fixes
    - MISRA C-2012 issue fixed: rule 10.1, 10.3, 10.4, 10.6, 10.7 and 11.9.
    - Fixed the issue of the inseparable process interrupted by other interrupt source.
      - WDOG_Init
      - WDOG_Refresh
- 2.1.0
  - New Features
    - Added new API "WDOG_TriggerSystemSoftwareReset()" to allow users to reset the system by software.

    - Added new API "WDOG_TriggerSoftwareSignal()" to allow users to trigger a WDOG_B signal by software.
    - Removed the parameter "softwareAssertion" and "softwareResetSignal" out of the wdog_config_t structure.
    - Added new parameter "enableTimeOutAssert" to the wdog_config_t structure. With this parameter enabled, when the WDOG timeout occurs, a WDOG_B signal will be asserted. This signal can be routed to external pin of the chip. Note that WDOG_B signal remains asserted until a power-on reset (POR) occurs.
- 2.0.1
  - New Features
    - Added control macro to enable/disable the CLOCK code in current driver.
- 2.0.0
  - Initial version.

Middleware Change Log

Multicore SDK

The current version of Multicore SDK is 2.8.0.

- 2.8.0
  - Multicore SDK component versions:
    - embedded Remote Procedure Call (eRPC) v1.7.4
    - eRPC generator (erpcgen) v.1.7.4
    - Multicore Manager (MCMgr) v4.1.0
    - RPMsg-Lite v3.1.0
  - New features:
    - eRPC: Unit test code updated to handle service add and remove operations.
    - eRPC: Several MISRA issues in rpmsg-based transports addressed.
    - eRPC: Support MU transport unit testing.
    - eRPC: Adding mbed os support.
    - eRPC: Fixed Linux/TCP acceptance tests in release target.
    - eRPC: Minor documentation updates, code formatting.
    - erpcgen: Whitespace removed from C common header template.
    - RPMsg-Lite: MISRA C-2012 violations fixed (7.4).
    - RPMsg-Lite: Fix missing lock in rpmsg_lite_rx_callback() for QNX env.
    - RPMsg-Lite: Correction of rpmsg_lite_instance structure members description.
    - RPMsg-Lite: Address -Waddress-of-packed-member warnings in GCC9.
    - RPMsg-Lite: Clang update to v10.0.0, code re-formatted.
- 2.7.0
  - Multicore SDK component versions:
    - embedded Remote Procedure Call (eRPC) v1.7.3
    - eRPC generator (erpcgen) v.1.7.3
    - Multicore Manager (MCMgr) v4.1.0
    - RPMsg-Lite v3.0.0
  - New features:
    - eRPC: Improved the test_callbacks logic to be more understandable and to allow requested callback execution on the server side.
    - eRPC: TransportArbitrator::prepareClientReceive modified to avoid incorrect return value type.
    - eRPC: The ClientManager and the ArbitratedClientManager updated to avoid performing client requests when the previous serialization phase fails.
    - erpcgen: Generate the shim code for destroy of statically allocated services.
    - MCMgr: Code adjustments to address MISRA C-2012 Rules
    - RPMsg-Lite: MISRA C-2012 violations fixed, incl. data types consolidation.
    - RPMsg-Lite: Code formatted
- 2.6.0
  - Multicore SDK component versions:

    - embedded Remote Procedure Call (eRPC) v1.7.2
    - eRPC generator (erpcgen) v.1.7.2
    - Multicore Manager (MCMgr) v4.0.3
    - RPMsg-Lite v2.2.0
  - New features:
    - eRPC: Improved support of const types.
    - eRPC: Fixed Mac build.
    - eRPC: Fixed serializing python list.
    - eRPC: Documentation update.
    - eRPC: Add missing doxygen comments for transports.
    - RPMsg-Lite: Added configuration macro RL_DEBUG_CHECK_BUFFERS.
    - RPMsg-Lite: Several MISRA violations fixed.
    - RPMsg-Lite: Added environment layers for QNX and Zephyr.
    - RPMsg-Lite: Allow environment context required for some environments (controlled by the RL_USE_ENVIRONMENT_CONTEXT configuration macro).
    - RPMsg-Lite: Data types consolidation.
    - MCMgr: Documentation updated to describe handshaking in a graphic form.
    - MCMgr: Minor code adjustments based on static analysis tool findings
- 2.5.0
  - Multicore SDK component versions:
    - embedded Remote Procedure Call (eRPC) v1.7.1
    - eRPC generator (erpcgen) v.1.7.1
    - Multicore Manager (MCMgr) v4.0.2
    - RPMsg-Lite v2.0.2
  - New features:
    - RPMsg-Lite, MCMgr: Align porting layers to the updated MCUXpressoSDK feature files.
    - eRPC: Fixed semaphore in static message buffer factory.
    - erpcgen: Fixed MU received error flag.
    - erpcgen: Fixed tcp transport.
- 2.4.0
  - Multicore SDK component versions:
    - embedded Remote Procedure Call (eRPC) v1.7.0
    - eRPC generator (erpcgen) v.1.7.0
    - Multicore Manager (MCMgr) v4.0.1
    - RPMsg-Lite v2.0.1
  - New features:
    - eRPC: Improved code size of generated code.
    - eRPC: Generating crc value is optional.
    - eRPC: Fixed CMSIS Uart driver. Removed dependency on KSDK.
    - eRPC: List names are based on their types. Names are more deterministic.
    - eRPC: Service objects are as a default created as global static objects.
    - eRPC: Added missing doxygen comments.
    - eRPC: Forbid users use reserved words.
    - eRPC: Removed outByref for function parameters.
    - eRPC: Added support for 64bit numbers.

    - eRPC: Added support of program language specific annotations.
    - eRPC: Optimized code style of callback functions.
    - RPMsg-Lite: New API rpmsg_queue_get_current_size()
    - RPMsg-Lite: Fixed bug in interrupt handling for lpc5411x, lpc5410x
    - RPMsg-Lite: Code adjustments based on static analysis tool findings
- 2.3.1
  - Multicore SDK component versions:
    - embedded Remote Procedure Call (eRPC) v1.6.0
    - eRPC generator (erpcgen) v.1.6.0
    - Multicore Manager (MCMgr) v4.0.0
    - RPMsg-Lite v1.2.0
  - New features:
    - eRPC: Improved code size of generated code.
    - eRPC: Improved eRPC nested calls.
    - eRPC: Improved eRPC list length variable serialization.
    - eRPC: Added @nullable support for scalar types.
    - MCMgr: Added new MCMGR_TriggerEventForce() API.
- 2.3.0
  - Multicore SDK component versions:
    - embedded Remote Procedure Call (eRPC) v1.5.0
    - eRPC generator (erpcgen) v.1.5.0
    - Multicore Manager (MCMgr) v3.0.0
    - RPMsg-Lite v1.2.0
  - New features:
    - eRPC: Added support for unions type non-wrapped by structure.
    - eRPC: Added callbacks support.
    - eRPC: Added support @external annotation for functions.
    - eRPC: Added support @name annotation.
    - eRPC: Added Messaging Unit transport layer.
    - eRPC: Added RPMSG Lite RTOS TTY transport layer.
    - eRPC: Added version verification and IDL version verification between eRPC code and eRPC generated shim code.
    - eRPC: Added support of shared memory pointer.
    - eRPC: Added annotation to forbid generating const keyword for function parameters.
    - eRPC: Added python matrix multiply example.
    - eRPC: Added nested call support.
    - eRPC: Added struct member "byref" option support.
    - eRPC: Added support of forward declarations of structures
    - eRPC: Added Python RPMsg Multiendpoint kernel module support
    - eRPC: Added eRPC sniffer tool
    - MCMgr: Unused API removed
    - MCMgr: Added the ability for remote core monitoring and event handling
    - RPMsg-Lite: Several source files renamed to avoid conflicts with other middleware sw components
    - RPMsg-Lite: Added the ability to use Multicore Manager (MCMGR) as the IPC interrupts

router

- 2.2.0
  - Multicore SDK component versions:
    - embedded Remote Procedure Call (eRPC) v1.4.0
    - eRPC generator (erpcgen) v.1.4.0
    - Multicore Manager (MCMgr) v2.0.1
    - RPMsg-Lite v1.1.0
  - New features:
    - eRPC: win_flex_bison.zip for windows updated.
    - eRPC: Use one codec (instead of inCodec outCodec).
    - eRPC: New RPMsg-Lite Zero Copy (RPMsgZC) transport layer.
    - MCMgr: code updated to be Misra compliant.
    - RPMsg-Lite: Added macros for packed structures (compiler.h).
    - RPMsg-Lite: Improved interrupt handling in platform layer.
    - RPMsg-Lite: Changed RL_BUFFER_SIZE definition.
    - RPMsg-Lite: Fix of double initialization of vring shared data structure.
    - RPMsg-Lite: Support for the multi-instance.
- 2.1.0
  - Multicore SDK component versions:
    - embedded Remote Procedure Call (eRPC) v1.3.0
    - eRPC generator (erpcgen) v.1.3.0
  - New features:
    - eRPC: New annotation types introduced (@length, @max_length, ...).
    - eRPC: Support for running both erpc client and erpc server on one side.
    - eRPC: New transport layers for (LP)UART, (D)SPI.
    - eRPC: Error handling support.
- 2.0.0
  - Multicore SDK component versions:
    - embedded Remote Procedure Call (eRPC) v1.2.0
    - eRPC generator (erpcgen) v.1.2.0
    - Multicore Manager (MCMgr) v2.0.0
    - RPMsg-Lite v1.0.0
  - New features:
    - Multicore SDK support for lpcxpresso54114 board added.
    - RPMsg component of the Open-AMP framework re-implemented and the RPMsg-Lite version introduced.
    - eRPC source directory organization changed.
    - Many eRPC improvements.
- 1.1.0
  - Multicore SDK component versions:
    - embedded Remote Procedure Call (eRPC) v1.1.0
    - Multicore Manager (MCMgr) v1.1.0
    - Open-AMP / RPMsg based on SHA1 ID 44b5f3c0a6458f3cf80 rev01
  - New features:
    - Multicore SDK 1.1.0 ported to KSDK 2.0.0.

        - Python support added into eRPC.
- 1.0.0
    - Multicore SDK component versions:
        - embedded Remote Procedure Call (eRPC) v1.0.0
        - Multicore Manager (MCMgr) v1.0.0
        - Open-AMP / RPMsg based on SHA1 ID 44b5f3c0a6458f3cf80 rev00

Component Change Log

SERIAL_MANAGER

The current Serial_Manager component version is 1.0.0.

- 1.0.0
  - Initial version

How to Reach Us:

Home Page:
nxp.com

Web Support:
nxp.com/support

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

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including “typicals,” must be validated for each customer application by customer’s technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.

While NXP has implemented advanced security features, all products may be subject to unidentified vulnerabilities. Customers are responsible for the design and operation of their applications and products to reduce the effect of these vulnerabilities on customer’s applications and products, and NXP accepts no liability for any vulnerability that is discovered. Customers should implement appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, Freescale, the Freescale logo, Kinetis, Processor Expert, and Tower are trademarks of NXP B.V. All other product or service names are the property of their respective owners. Arm, Cortex, Keil, Mbed, Mbed Enabled, and Vision are trademarks or registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved. Oracle and Java are registered trademarks of Oracle and/or its affiliates. The Power Architecture and Power.org word marks and the Power and Power.org logos and related marks are trademarks and service marks licensed by Power.org.

© 2018 NXP B.V.