
MCUXpresso SDK Release Notes Supporting Ipcxpresso55s69

Change Logs

NXP Semiconductors



Contents

Driver Change Log

CLOCK	1
RESET	1
RESET	1
EDMA_SOC	1
TRDC_SOC	1
COMMON	2
LPADC	4
PRINCE	6
RNG	7
SDIF	7

Middleware Change Log

emWin library	10
FatFs for MCUXpresso SDK	10
FreeMASTER Communication Driver	11
LVGL for KSDK	12
lwIP for MCUXpresso SDK	13
mbedTLS for MCUXpresso SDK	19
MOTOR_CONTROL for KSDK	23
Multicore SDK	24
NTAG I2C plus library	33

Title

Page No.

RTCESL for KSDK 34

SAFETY_IEC60730B for KSDK 34

Host SDIF driver for MCUXpresso SDK 34

MMC Card driver for MCUXpresso SDK 35

SD Card driver for MCUXpresso SDK 37

SDIO Card driver for MCUXpresso SDK 40

Secure Element hostlib for KSDK 42

Trusted Firmware M (TF-M) for MCUXpresso SDK 43

USB stack for MCUXpresso SDK 44

NXP WiFi 50

RTOS Change Log

kernel for MCUXpresso SDK 76

Component Change Log

CODEC 78

WM8904 79

SGTL5000 82

DA7212 82

CS42888 83

SERIAL_MANAGER 84

1 Driver Change Log

CLOCK

The current CLOCK driver version is 1.0.1.

- 1.0.1
 - Improvements
 - * Added Clock ip name array for SINC.
- 1.0.0
 - initial version.

RESET

The current RESET driver version is 2.4.0.

- 2.4.0
 - Improvements
 - * Add RESET_ReleasePeripheralReset API.
- 1.0.0
 - Initial version.

RESET

The current RESET driver version is 2.4.0.

- 2.4.0
 - Improvements
 - * Add RESET_ReleasePeripheralReset API.
- 1.0.0
 - Initial version.

EDMA_SOC

The current EDMA_SOC driver version is 1.0.0.

- 1.0.0
 - Initial version.

TRDC_SOC

The current TRDC_SOC driver version is 1.0.0.

- 1.0.0
 - Initial version.

COMMON

The current COMMON driver version is 2.4.0.

- 2.4.0
 - New Features
 - * Added EnableIRQWithPriority, IRQ_SetPriority, and IRQ_ClearPendingIRQ for ARM.
 - * Added MSDK_EnableCpuCycleCounter, MSDK_GetCpuCycleCount for ARM.
- 2.3.3
 - New Features
 - * Added NETC into status group.
- 2.3.2
 - Improvements
 - * Make driver aarch64 compatible
- 2.3.1
 - Bug Fixes
 - * Fixed MAKE_VERSION overflow on 16-bit platforms.
- 2.3.0
 - Improvements
 - * Split the driver to common part and CPU architecture related part.
- 2.2.10
 - Bug Fixes
 - * Fixed the ATOMIC macros build error in cpp files.
- 2.2.9
 - Bug Fixes
 - * Fixed MISRA C-2012 issue, 5.6, 5.8, 8.4, 8.5, 8.6, 10.1, 10.4, 17.7, 21.3.
 - * Fixed SDK_Malloc issue that not allocate memory with required size.
- 2.2.8
 - Improvements
 - * Included stddef.h header file for MDK tool chain.
 - New Features:
 - * Added atomic modification macros.
- 2.2.7
 - Other Change
 - * Added MECC status group definition.
- 2.2.6
 - Other Change
 - * Added more status group definition.
 - Bug Fixes
 - * Undef __VECTOR_TABLE to avoid duplicate definition in cmsis_clang.h
- 2.2.5

- Bug Fixes
 - * Fixed MISRA C-2012 rule-15.5.
- 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 lets result be closer 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.

LPADC

The current LPADC driver version is 2.8.4.

- 2.8.4
 - Bug Fixes
 - * Remove function 'LPADC_SetOffsetValue' assert statement, this statement may cause runtime errors in existing code.
- 2.8.3
 - Bug Fixes
 - * Fixed SDK lpadc driver examples compile issue, move condition 'commandId < ADC_CV_COUNT' to a more appropriate location.
- 2.8.2
 - Bug Fixes
 - * Fixed the violations of MISRA C-2012 rule 18.1, 10.3, 10.1 and 10.4.
- 2.8.1
 - Bug Fixes
 - * Fixed LPADC sample mode enum name mistake.
- 2.8.0
 - Improvements
 - * Release peripheral from reset if necessary in init function.
 - Bug Fixes
 - * Fixed function LPADC_GetConvResult() issue.
 - * Fixed function LPADC_SetConvCommandConfig() bugs.
- 2.7.2
 - Improvements
 - * Use feature macros instead of header file macros.
 - Bug Fixes
 - * Fixed the violations of MISRA C-2012 rule 10.1, 10.3, 10.4 and 14.3.
- 2.7.1
 - Improvements
 - * Corrected descriptions of several functions.
 - * Improved function LPADC_GetOffsetValue and LPADC_SetOffsetValue.
 - * Revert changes of feature macros for lpadc.
 - * Use feature macros instead of header file macros. -Bug Fixes
 - * Fixed the violations of MISRA C-2012 rule 10.8.
 - * Fixed the violations of MISRA C-2012 rule 10.1, 10.3, 10.4 and 14.3.

- 2.7.0
 - Improvements
 - * Added supports of CFG2 register.
 - * Removed some useless macros.
- 2.6.2
 - Bug Fixes -Fixed the violations of MISRA C-2012 rules. -Fixed LPADC driver code compile error issue.
- 2.6.1
 - Improvements
 - * Updated the use of macros in the driver code.
- 2.6.0
 - Improvements
 - * Added the API LPADC_SetOffset12BitValue() to configure 12bit ADC conversion offset trim value manually.
 - * Added the API LPADC_SetOffset16BitValue() to configure 16bit ADC conversion offset trim value manually.
 - * Added API to set offset calibration mode.
 - * Added configuration of alternate channel.
 - * Updated auto calibration API and added calibration value conversion API.
 - New feature
 - * Added API LPADC_EnableHardwareTriggerCommandSelection() to enable trigger commands controlled by ADC_ETC.
 - * Updated LPADC_DoAutoCalibration() to allow doing something else before the ADC initialization to be totally complete. Enhance initialization duration time of the ADC.
 - * Added two new APIs to get/set calibration value.
- 2.5.2
 - Improvements
 - * Added while loop, LPADC_GetConvResult() will return only when the FIFO will not be empty.
- 2.5.1
 - Bug Fixes
 - * Fixed some typos in Lpadc driver comments.
- 2.5.0
 - Improvements
 - * Added missing items to enable trigger interrupts.
- 2.4.0
 - New features
 - * Added APIs to get/clear trigger status flags.
- 2.3.0
 - Improvements
 - * Removed LPADC_MeasureTemperature() function for the LPADC supports different temperature sensor calculation equations.
- 2.2.1
 - Improvements
 - * Optimized LPADC_MeasureTemperature() function to support the specific series with

- flash solidified calibration value.
- * Clean doxygen warnings.
- Bug Fixes
 - * Fixed violations of MISRA C-2012 rule 10.3, rule 10.8 and rule 17.7.
- 2.2.0
 - New Feature
 - * Added API LPADC_MeasureTemperature() to get correct temperature from the internal sensor.
 - Improvements
 - * Separated lpadc_conversion_resolution_mode_t with related feature macro.
 - Bug Fixes
 - * Fixed the violations of MISRA C-2012 rules:
 - Rule 10.3, 10.4, 10.6, 10.7 and 17.7.
- 2.1.1
 - Improvements
 - * Updated the gain calibration formula.
 - * Used feature to segregate the new item kLPADC_TriggerPriorityPreemptSubsequently.
- 2.1.0
 - New Features
 - * Added the API LPADC_SetOffsetValue() to support configure offset trim value manually.
 - * Added the API LPADC_DoOffsetCalibration() to do offset calibration independently.
 - Improvements
 - * Improved the usage of macros and removed invalid macros.
- 2.0.2
 - Improvements
 - * Added support for platforms with 2 FIFOs and different calibration measures.
- 2.0.1
 - Bug Fixes
 - * Ensured the API LPADC_SetConvCommandConfig configure related registers correctly.
- 2.0.0
 - Initial version.

PRINCE

The current PRINCE driver version is 2.5.1.

- Version 2.6.0
 - Renamed CSS to ELS.
- Version 2.5.1
 - Fix build error due to renamed symbols.
- Version 2.3.2
 - Fix documentation of enumeration.
 - Extend PRINCE example.
- Version 2.3.1

- Fix MISRA-2012 issues.
- Add support for LPC55S0x series
- Version 2.3.0
 - Add support for LPC55S1x and LPC55S2x series
- Version 2.2.0
 - Add runtime checking of the A0 and A1 rev. of LPC55Sxx serie to support both silicone revisions.
- Version 2.1.0
 - Update for the A1 rev. of LPC55Sxx serie.
- 2.0.0
 - Initial version.

RNG

The current RNG driver version is 2.0.3.

- 2.0.3
 - Modified RNG_Init and RNG_GetRandomData functions, added rng_accumulateEntropy and rng_readEntropy functions. These changes are reflecting recommended usage of RNG according to device UM
- 2.0.2
 - Add RESET_PeripheralReset function inside RNG_Init and RNG_Deinit functions.
- 2.0.1
 - Fix MISRA C-2012 issue.
- 2.0.0
 - Initial version.

SDIF

The current SDIF driver version is 2.1.0

- 2.1.0
 - Improvements
 - * Removed redundant member endianMode in sdif_config_t.
 - * Added error status check in function SDIF_WaitCommandDone.
 - * Fixed the read fifo data incomplete issue in interrupt non-dma mode.
- 2.0.15
 - Bug Fixes
 - * Cleared the interrupt status before enable the interrupt to avoid interrupt generate unexpectedly.
 - * Fixed the SDIF_ReadDataPortBlocking blocking at wrong condition issue.
 - Improvements
 - * Enabled the functionality of timeout parameter in SDIF_SendCommand.
 - * Added the error recovery while sending sync clock command timeout.

- 2.0.14
 - Improvements
 - * Used different status code for command and data interrupt callback.
 - Bug Fixes
 - * Fixed the DMA descriptor attribute field unreset when configuring the current transfer DMA descriptor issue which may cause the transfer terminate unexpected.
- 2.0.13
 - Improvements
 - * Disabled redundant interrupt per different transfer request.
 - * Disabled interrupt and reset command/data pointer in handle when transfer completes.
 - Bug Fixes
 - * Fixed the PA082 build warning.
 - * Fixed violations of the MISRA C-2012 rules 14.4, 17.7, 10.4, 10.3, 10.8, 14.3, 10.1, 16.4, 15.7, 12.2, 11.3, 11.9.
- 2.0.12
 - Bug Fixes
 - * Fixed the issue that SDIF_ConfigClockDelay didn't reset the delay field before write.
 - * Removed useless fifo reset code in transfer function.
 - * Fixed the divider overflow issue in function SDIF_SetCardClock.
- 2.0.11
 - Improvements
 - * Added API SDIF_GetEnabledInterruptStatus/SDIF_GetEnabledDMAInterruptStatus and used in SDIF_TransferHandleIRQ.
 - * Removed useless members interruptFlags/dmaInterruptFlags in the sdif_handle_t.
 - * Improved SDIF_SendCommand with return success directly when timeout is 0.
 - * Added timeout error check when sending update clock command in SDIF_SetCardClock.
 - * Removed START_CMD status polling for normal command sending in SDIF_TransferBlocking/SDIF_TransferNonBlocking.
 - * Disabled timeout parameter in function SDIF_SendCommand.
 - Bug Fixes
 - * Added delay cycle for the default speed mode(400 K and 25 M) to fix the timing issue when different AHB clocks are configured.
- 2.0.10
 - Bug Fixes
 - * Fixed the issue that API SDIF_EnableCardClock could not clear the clock enable bit.
- 2.0.9
 - Bug Fixes
 - * Fixed MDK 66-D warning.
- 2.0.8
 - New Features
 - * Added control macro to enable/disable the RESET and CLOCK code in current driver.
 - * Disabled useless interrupt while DMA is used.
 - * Updated SDIF driver for one instance support two cards.
- 2.0.7
 - Bug Fixes

- * Enlarged the timeout value to avoid a command conflict issue.
- 2.0.6
 - Bug Fixes
 - * Removed assert(srcClock_Hz <= FSL_FEATURE_SDIF_MAX_SOURCE_CLOCK).
 - * Used hardware reset instead of software reset during initialization.
- 2.0.5
 - New Features
 - * Added non-word aligned data address and DMA descriptor address transfer support. Once one of the above addresses is not aligned, switch to host transfer mode.
 - Bug Fixes
 - * Fixed the issue that DMA suspended during initialization.
 - * Removed useless memset function call.
- 2.0.4
 - Improvements
 - * Added cardInserted/cardRemoved callback function.
 - * Added host base address/user data parameter for all call back functions.
- 2.0.3
 - Improvements
 - * Improved Clock Delay macro to allow the user to redefine and remove useless delay for clock below 25 MHz.
- 2.0.2
 - Bug Fixes
 - * Fixed the issue that the status flag could not be cleared entirely after transfer complete.
- 2.0.1
 - New Features
 - * Improved interrupt transfer callback.
 - Bug Fixes
 - * Added assert to limit the SDIF source clock below 52 MHz.
- 2.0.0
 - Initial version.

2 Middleware Change Log

emWin library

The currently supported version is 6.34c

- v6.34c
 - upgraded to v6.34c
- v6.28_rev1
 - add cm33_nodsp_fpu libraries for Cortec M33 without DSP extension with SP FPU
- v6.28
 - upgraded to v6.28
- v6.24_rev2
 - add cm33_nodsp libraries for Cortex M33 without DSP extension
- v6.24_rev1
 - recompiled cm33 library with fpu single precision
 - added cm7_sp library for Cortex M7 with sp fpu for IAR
- v6.24
 - upgraded to v6.24
- v6.16c
 - upgraded to v6.16c
 - updated temperature_control demo generated by AppWizard
- v6.14d
 - upgraded to v6.14d
- v6.10f
 - upgraded to v6.10f

FatFs for MCUXpresso SDK

Current version is FatFs R0.15_rev0.

- R0.15_rev0
 - Upgraded to version 0.15
 - Applied patches from <http://elm-chan.org/fsw/ff/patches.html>
- R0.14b_rev1
 - Applied patches from <http://elm-chan.org/fsw/ff/patches.html>
- R0.14b_rev0
 - Upgraded to version 0.14b
- R0.14a_rev0
 - Upgraded to version 0.14a
 - Applied patch ff14a_p1.diff and ff14a_p2.diff
- R0.14_rev0
 - Upgraded to version 0.14
 - Applied patch ff14_p1.diff and ff14_p2.diff

- R0.13c_rev0
 - Upgraded to version 0.13c
 - Applied patches ff_13c_p1.diff,ff_13c_p2.diff, ff_13c_p3.diff and ff_13c_p4.diff.
- R0.13b_rev0
 - Upgraded to version 0.13b
- R0.13a_rev0
 - Upgraded to version 0.13a. Added patch ff_13a_p1.diff.
- R0.12c_rev1
 - Add NAND disk support.
- R0.12c_rev0
 - Upgraded to version 0.12c and applied patches ff_12c_p1.diff and ff_12c_p2.diff.
- R0.12b_rev0
 - Upgraded to version 0.12b.
- R0.11a
 - Added glue functions for low-level drivers (SDHC, SDSPI, RAM, MMC). Modified diskio.c.
 - Added RTOS wrappers to make FatFs thread safe. Modified syscall.c.
 - Renamed ffconf.h to ffconf_template.h. Each application should contain its own ffconf.h.
 - Included ffconf.h into diskio.c to enable the selection of physical disk from ffconf.h by macro definition.
 - Conditional compilation of physical disk interfaces in diskio.c.

FreeMASTER Communication Driver

Current version is 3.0.6. Visit <https://www.nxp.com/freemaster> for more information. Reach out for a support at <https://community.nxp.com/community/freemaster>.

- 3.0.0
 - Initial version of FreeMASTER driver reworked from a standalone package to MCUXpresso SDK middleware.
 - This driver version supports new version V4 of FreeMASTER serial communication protocol.
 - Supports UART, LPUART, USART, MINIUSART, FlexCAN, USB-CDC and JTAG/BDM communication.
 - Initial version was tested with the following boards: evkmimxrt1060, frdmk64f, frdmke15z, frdmkl28z, lpcxpresso54628 lpcxpresso55s69, lpcxpresso845max and twrk64f120m.
 - Use with FreeMASTER PC Host tool version 2.5 or later.
- 3.0.1
 - FreeMASTER driver extended to support wide range of Kinetis, LPC and i.MX-RT platforms.
 - Low-level communication drivers also available for few non-SDK NXP platforms like S12Z, S32x and more.
 - Use with FreeMASTER PC Host tool version 3.0 or later.
- 3.0.2
 - FreeMASTER driver support of DSC56F800EX and S12 platforms extended.
 - Removed dependency on C99 compiler features.
 - Use with FreeMASTER PC Host tool version 3.0.2 or later.

- 3.0.3
 - General update for SDK 2.9.0
 - fmstr_any demo added to selected platforms - use with MCUXpresso SDK and FreeMASTER peripheral configuration tool.
 - New example.pmp project file embedded into application flash storage.
 - USB-CDC implementation fixed, new JTAG EOnCE communication interface added to DSC 56F800E family.
 - Use with FreeMASTER PC Host tool version 3.0.3 or later. Version 3.1.x is recommended.
- 3.0.4
 - Fixed component dependency logic of FreeMASTER driver.
 - Use with FreeMASTER PC Host tool version 3.1.x
- 3.0.5
 - General update for SDK 2.11 and 2.12
 - New TCP and UDP support with lwIP stack
 - New communication over Segger RTT interface
 - Add fmstr_net and fmstr_wifi examples for selected i.MX-RT platforms
 - Add fmstr_rtt example for selected platforms
 - Fixed negative recorder threshold trigger processing
- 3.0.6
 - General update for SDK 2.13
 - Use of new Ethernet MDIO driver concept.
 - Support of ENET and NETC Ethernet modules in the fmstr_net example application.
- 3.0.7
 - General update for SDK 2.14

LVGL for KSDK

- 8.3.10_rev1
 - Integrate LVGL 8.3.10 to SDK.
- 8.3.9_rev1
 - Integrate LVGL 8.3.9 to SDK.
- 8.3.5_rev1
 - Integrate LVGL 8.3.5 to SDK.
- 8.3.2_rev1
 - Integrate LVGL 8.3.2 to SDK.
- 8.3.0_rev1
 - Integrate LVGL 8.3.0 to SDK.
- 8.2.0_rev1
 - Integrate LVGL 8.2.0 to SDK.
- 8.0.2_rev1
 - Integrate LVGL 8.0.2 to SDK.
- 7.10.1_rev1
 - Integrate LVGL 7.10.1 to SDK.
 - Added PXP, VGLite hardware acceleration.

- 7.4.0_rev1
 - Integrate LVGL 7.4.0 to SDK.
- 7.0.0_rev1
 - Integrate LVGL 7.0.0 to SDK.
 - Added PXP hardware acceleration initial version.
- 6.1.1_rev1
 - Integrate LVGL 6.1.1 to SDK.
- 5.3_rev1
 - Integrate LVGL 5.3 to SDK.

lwIP for MCUXpresso SDK

Lightweight IP (lwIP) is a small independent implementation of the TCP/IP protocol suite. Source code included in this SDK is based on version 2.2.0 taken from 3rd party lwIP GIT repository. The webpage <https://git.savannah.nongnu.org/cgit/lwip.git> allows to browse the repository and also contains URLs for its cloning. The development versions (X.Y.Z.dev) do not refer to a single source code snapshots. To avoid ambiguity, change log below contains SHA-1 hashes of GIT commits used when importing the code into the SDK.

- 2.2.0_rev10
 - New features:
 - * Ported lwIP 2.2.0 (2023-09-25, branch: master, SHA-1: 0a0452b2c39bdd91e252aef045c115f88f6ca7 tag: STABLE-2_2_0_RELEASE) to MCUXpresso SDK.
 - * Enabled hardware-accelerated CRC computation and verification (MAC, IPv4, TCP, UDP, ICMPv4, ICMPv6) for ENET Kinetis, ENET QoS and ENET LPC.
 - * Enabled link state detection based on PHY interrupts. The ETH_LINK_POLLING_INTERVAL_MS macro controls this - setting it to 0 and specifying ethernetif_config_t->phy-IntGpio enables it, setting it to a value greater than zero enables polling instead. Supported only under an RTOS (NO_SYS == 0). By default, the link state is polled.
 - * ND6: Implemented RFC 4191 type C host, which means default router list (learned from Router Advertisement messages) has been replaced with routing table, which contains default route records for each router and also routes learned from received Route Information Options. Changes partially based on <https://savannah.nongnu.org/patch/?10114>. The option LWIP_ND6_NUM_ROUTERS has been removed, and the new option LWIP_ND6_NUM_ROUTES has been added to configure the size of the routing table.
 - * IPv6: Implemented a new hook - LWIP_HOOK_IP6_CANFORWARD. This hook can be used, for example, for multicast forwarding between netifs. Defining this hook enables multicast traffic forwarding, thus the hook is also invoked for multicast traffic.
 - * MLD6: Multicast Listener Discovery v1 replaced by v2 (RFC 3810) but without support of source specific multicast.
 - * port/enet_ethernetif_kinetis.c: Added check to generate/validate ICMPv6 checksum in SW as the Kinetis ENET peripheral does not do it.
 - * Added disabling of Rx interrupt when the port is out of Rx buffers. See port/README.md

for more details.

- Bug fixes:
 - * src/apps/lwiperf: Fixed access to invalid data when UDP report is to be sent from a timer but abort has been called before.
 - * src/apps/lwiperf: Fixed deallocation of TCP server started by client (in reverse or dual modes) which failed to connect.
 - * port/netc_ethernetif.c: Fixed cache control enablement macro (FSL_SDK_ENABLE_DRIVER_CACHE_CONTROL > FSL_ETH_ENABLE_CACHE_CONTROL).
 - * port/sys_arch.c: The function sys_assert does not call portENTER_CRITICAL when called from an interrupt.
 - * src/core/ipv4/ip4.c: Fixed checksum reset condition.
 - * ND6:
 - Lladdr length is now taken from netif->hwaddr_len so ND6 works properly regardless of NETIF_MAX_HWADDR_LEN.
 - Added check of sufficient length of lladdr options from incoming messages.
 - * src/apps/httpsrv/httpsrv.c: Fixed hangup in HTTPSrv_release if caller's task has higher priority than server task.
- port/arch/cc.h: LWIP_PLATFORM_DIAG is defined (and can be overridden) independently of the LWIP_DEBUG setting. Removed printing extra newline symbols from LWIP_PLATFORM_DIAG.
- src/apps/lwiperf: The "end of test" UDP datagram is resent more often. This increases the probability of the server to receive it and end the test when datagrams are getting lost.
- Added port/README.md describing possible settings and helper functions in the port layer.
- 2.2.0_rev9
 - New features:
 - * Ported lwIP 2.2.0.dev (2023-01-03, branch: master, SHA-1: 3fe8d2fc43a9b69f7ed28c63d44a7744f9c) to MCUXpresso SDK.
 - * Applied patch to allow sending IPv6 router advertisement. Improved to allow selection of interface and router life time and to allow sending route information options.
 - * src/apps/lwiperf: Support for reverse test (client receives, server sends). Requires iperf version 2.1.0 or newer.
 - Bug fixes:
 - * src/apps/httpsrv: Fixed operation with LWIP_IPV6 enabled. Server can be also accessed using both IPv4 and IPv6 at the same time if compiled with both LWIP_IPV4=1 and LWIP_IPV6=1. Note the type of the field struct httpsrv_param_struct.address has changed from struct sockaddr to struct sockaddr_storage.
- 2.2.0_rev8
 - New features:
 - * src/apps/lwiperf: Added new parameter "buffer_len" to functions lwiperf_start_tcp_client() and lwiperf_start_udp_client() to configure TCP/UDP packet size.
 - * src/apps/lwiperf: Added new parameter "tos" to functions lwiperf_start_tcp_client() to configure TCP packet priority.
 - * NETC adaptation layer: Not forcing the RX/TX buffers placement in non-cacheable memory. Requires the symbol FSL_ETH_ENABLE_CACHE_CONTROL to be defined on project level if the memory region, where the buffers are placed by a linker, has cache

- enabled.
- Bug fixes:
 - * src/apps/httpsrv: Added missing includes.
 - * src/apps/lwiperf: Fixed TCP client to send settings at the beginning of each 128 KB block like the PC iperf 2.0.x application does.
 - * src/apps/lwiperf: Fixed validation of TCP received data (with LWIPERF_CHECK_RX_DATA enabled, works with iperf 2.0.x).
 - * src/apps/lwiperf: Fixed lwiperf_list_remove() to clear references to the removed item.
- src/apps/lwiperf: Program does not assert when buffer cannot be cloned in UDP test, only "can't clone buffer" message is printed.
- 2.2.0_rev7
 - New features:
 - * Ported lwIP 2.2.0.dev (2022-05-09, branch: master, SHA-1: 239918ccc173cb2c2a62f41a40fd893f571) to MCUXpresso SDK.
 - * Added function ethernetif_probe_link() which reads actual link, speed and duplex settings from phy and passes them to driver. Stack could be set to call this function periodically by setting ETH_LINK_POLLING_INTERVAL_MS to value higher than zero.
 - * Added helper functions ethernetif_wait_linkup() and ethernetif_wait_ipv4_valid() to allow blocking of RTOS task or bare metal application until link is up or IPv4 address becomes valid.
 - * Added NETC adaptation layer.
 - * Processing of rx packets under RTOS moved from ISR to a separate task to improve system reaction times. Switch back to old behavior can be done by setting ETH_DO_RX_IN_SEPARATE_TASK macro to 0.
 - Bug fixes:
 - * port: Fixed copying of pbuf contents. Previous code was using an incorrect end condition and could result in the overrun of the destination buffer if more packets were on the queue.
 - * port: Delegating pbuf_free calls to tcpip_thread via pbuf_free_callback where possible (RTOS), ensured pbuf_free is not called from interrupt context when LWIP_ALLOW_MEM_FREE_FROM_OTHER_CONTEXT is not set (bare metal).
 - * port/enet_ethernetif_qos.c - Fixed ENET_RXBD_NUM which was used instead of ENET_TXBD_NUM.
 - * port/enet_ethernetif_qos.c - Fixed buffer alignment to be at least 64.
 - * src/apps/lwiperf: Fixed IPv6 TCP TX throughput lower than IPv4 by modifying maximum segment size to avoid sending two segments instead of one.
 - * src/apps/lwiperf: Out-of-order datagrams in UDP RX server mode are counted to the throughput.
 - * src/apps/httpsrv: Implemented receive timeouts on sockets.
 - * src/apps/httpsrv: Don't assert on HTTP session task creation failure.
 - * src/apps/httpsrv: Fixed build with IPv6 enabled.
 - * src/apps/httpsrv: Updated endianness macros required for websocket SHA generation.
 - * src/apps/httpsrv: Added missing includes.
- 2.2.0_rev6
 - New features:
 - * Ported lwIP 2.2.0.dev (2022-03-25, branch: master, SHA-1: 124dc0a64ef5d7c14a27e3115e5888df65)

- to MCUXpresso SDK.
 - * Implemented leaving of multicast groups on ENET and ENET QOS.
- 2.2.0_rev5
 - New features:
 - * Ported lwIP 2.2.0.dev (2021-05-11, branch: master, SHA-1: 7ec4e9be304e7f8953740f10b2c810a292) to MCUXpresso SDK.
 - * LPC ENET adaptation layer allocates more buffers for frame reception now. Previously the number of receive buffers was determined by ENET_RXBD_NUM, which defaults to 5. It is determined by ENET_RXBUFF_NUM now, which is 2 * ENET_RXBD_NUM by default. Increase was needed because the actual version of LPC ENET driver always hold ENET_RXBD_NUM number of buffers and few additional buffers are needed for passing zero-copy frame data to lwIP. If this takes too much memory in your application, you can counteract by decreasing PBUF_POOL_SIZE, since PBUF_POOL is used only for transmission when LPC ENET, Kinetis ENET or ENET QOS is used.
- 2.2.0_rev4
 - New features:
 - * Ported lwIP 2.2.0.dev (2021-03-05, branch: master, SHA-1: 0056522cc974d2be2005c324f37187b5b3) to KSDK 2.0.0.
 - * LWIP_DHCP_DOES_ACD_CHECK option default changed to 0 (disabled):
 - Although the ACD check makes getting IP address from DHCP more robust, it added several seconds delay at startup of all applications which use DHCP.
 - This feature was not present in earlier versions of lwIP.
 - * ENET QOS adaptation layer - implemented zero-copy on receive.
 - * Kinetis ENET and ENET QOS adaptation layers allocate more buffers for frame reception now. Previously the number of receive buffers was determined by ENET_RXBD_NUM, which defaults to 5. It is determined by ENET_RXBUFF_NUM now, which is 2 * ENET_RXBD_NUM by default. Increase was needed because the actual version of Kinetis ENET and ENET QOS drivers always hold ENET_RXBD_NUM number of buffers and few additional buffers are needed for passing zero-copy frame data to lwIP. If this takes too much memory in your application, you can counteract by decreasing PBUF_POOL_SIZE, since PBUF_POOL is used only for transmission when Kinetis ENET or ENET QOS is used.
 - * Removed ethernetif_config_t.non_dma_memory field which was required to configure memory ranges unusable by ENET DMA on LPC devices. The setting has been replaced by BOARD_ENET_NON_DMA_MEMORY_ARRAY macro.
- 2.2.0_rev3
 - New features:
 - * Ported lwIP 2.2.0.dev (2020-07-07, branch: master, SHA-1: c385f31076b27efb8ee37f00cb5568783a) to KSDK 2.0.0.
- 2.2.0_rev2
 - New features:
 - * Kinetis ENET adaptation layer - implemented zero-copy on receive.
 - * lwiperf - counter of transferred bytes extended from 32 to 64 bit
 - Bug fixes:
 - * Fixed restarting Auto IP from DHCP.

- 2.2.0_rev1
 - New features:
 - * Ported lwIP 2.2.0.dev (2019-12-12, branch: master, SHA-1: 555812dcec38c9a2ef1ef9b31816291549) to KSDK 2.0.0.
 - * Implemented LWIP_ASSERT_CORE_LOCKED related functions in sys_arch.c. It can be enabled in lwipopts.h:


```

              · #define LWIP_ASSERT_CORE_LOCKED() sys_check_core_locking()
              · #define LWIP_MARK_TCPIP_THREAD() sys_mark_tcpip_thread()
              // if NO_SYS == 0
              · #define LOCK_TCPIP_CORE() sys_lock_tcpip_core() // if NO_SYS == 0 and LWIP_TCPIP_CORE_LOCKING == 1
              · #define UNLOCK_TCPIP_CORE() sys_unlock_tcpip_core() // if NO_SYS == 0 and LWIP_TCPIP_CORE_LOCKING == 1
              
```
- 2.1.2_rev5
 - New features:
 - * Implemented TCP_USER_TIMEOUT socket option.
 - * Implemented SIOCOUTQ ioctl.
- 2.1.2_rev4
 - New features:
 - * Ported lwIP 2.1.3.dev (2019-02-27, branch: STABLE-2_1_x, SHA-1: 1bb6e7f52de1cd86be0eed31e3) to KSDK 2.0.0.
 - * Updated sys_thread_new implementation and comment.
 - * Kinetis ENET adaptation layer - reading frames into a pbuf chain is conditionally compiled only when a single pbuf from pool cannot hold maximum frame size (PBUF_POOL_BUFSIZE >= maximum frame size). Avoiding this code also reduces stack size requirements by about 1.5 kilobytes.
 - Bug fixes:
 - * Fixes in ethernetif_linkoutput() in enet_ethernetif_lpc.c:
 - Removed access to possibly freed pbuf.
 - Call pbuf_free() when transmit buffers not available.
 - When copying pbuf chain, updating the number of necessary transmit buffers to wait for, which can be often smaller in the copy.
 - * When CGI script is reading POST data by chunks, the loop in httpsrv_read() may cause blocking in receive function waiting for more data at the end of the stream
 - HTTPSrv_cgi_read() - added limiting of the last chunk length according to content length to avoid undesired blocking
 - * Applied AUTOIP patch <https://savannah.nongnu.org/patch/?9847> - with modification to support multiple network interfaces.
 - * Fixed buffer overflow in httpsrv when application provided CGI script does not handle the whole content of POST request
 - Removed LwipMibCompiler contrib application as it contained LGPL licensed files in Sharp-SnmpLib.
- 2.1.2_rev3
 - New features:
 - * lwiperf updated with UDP client/server support from the patch 9751 (<https://github.com/lwip/lwip/pull/9751>)

[://savannah.nongnu.org/patch/?9751](http://savannah.nongnu.org/patch/?9751))

- 2.1.2_rev2
 - Bug fixes:
 - * Fixed lwiperf_abort() in lwiperf.c to correctly close connections and free resources
- 2.1.2_rev1
 - New features:
 - * Ported lwIP 2.1.2 (2018-11-22, SHA-1: 159e31b689577dbf69cf0683bbaffbd71fa5ee10) to KSDK 2.0.0.
 - * Ported lwIP-contrib 2.1.0 (2018-09-24, SHA-1: 35b011d4cf4c4b480f8859c456587a884ec9d287) to KSDK 2.0.0.
- 2.0.3_rev1
 - New features:
 - * Ported lwIP 2.0.3 (2017-09-15, SHA-1: 92f23d6ca0971a32f2085b9480e738d34174417b) to KSDK 2.0.0.
- 2.0.2_rev1
 - New features:
 - * Ported lwIP 2.0.2 (2017-03-13, SHA-1: c0862d60746e2d1ceae69af4c6f24e469570ecef) to KSDK 2.0.0.
- 2.0.0_rev3
 - New features:
 - * Ported lwIP 2.0.0 (2016-11-10, SHA-1: 216bf89491815029aa15463a18744afa04df58fe) to KSDK 2.0.0.
- 2.0.0_rev2
 - New features:
 - * Ported lwIP 2.0.0 RC2 (2016-08-08, SHA-1: b1dfd00f9233d124514a36a8c8606990016f2ad4) to KSDK 2.0.0.
- 2.0.0_rev1
 - New features:
 - * Ported lwIP 2.0.0 RC0 (2016-05-26) to KSDK 2.0.0.
 - * Changed lwIP bare-metal examples to use poll-driven approach instead of interrupt-driven one.
- 1.4.1_rev2
 - New features:
 - * Enabled critical sections in lwIP.
 - Bug fixes:
 - * Fixed default lwIP packet-buffer size to be able to accept a maximum size frame from the ENET driver.
 - * Fixed possible drop of multi-frame packets during transmission.
- 1.4.1_rev1
 - New features:
 - * Ported lwIP 1.4.1 to KSDK 2.0.0.

mbedTLS for MCUXpresso SDK

The current version of mbedTLS is based on mbed TLS 2.28.5 branch released 2023-10-05

- 2.28.5
 - New features:
 - * Ported mbedTLS 2.28.5 to SDK.
- 2.28.4
 - New features:
 - * Ported mbedTLS 2.28.4 to SDK.
- 2.28.3
 - New features:
 - * Ported mbedTLS 2.28.3 to SDK.
- 2.28.1
 - New features:
 - * Ported mbedTLS 2.28.1 to SDK.
- 2.28.0
 - New features:
 - * Ported mbedTLS 2.28.0 to SDK.
- 2.27.0
 - New features:
 - * Ported mbedTLS 2.27.0 to SDK.
- 2.26.0
 - New features:
 - * Ported mbedTLS 2.26.0 to SDK.
- 2.16.6_rev7
 - Bug fixes:
 - * Corrected definition of global variable `g_isCryptoHWInitialized` to be only internal static variable in `sssapi_mbedtls.c` file.
- 2.16.6_rev6
 - Bug fixes:
 - * Adding `#ifdef` in `ecdsa.c` to remove warning: "function "derive_mpi" was declared but never referenced", when alternative implementation of ECDSA sign and verify is used and not used Deterministic ECDSA, then was `derive_mpi` function never used.
- 2.16.6_rev5
 - New features:
 - * Changed return type of `CRYPTO_InitHardware()` from `void` to `status_t`. Added check of this return value in `selftest.c` and `benchmark.c` files.
- 2.16.6_rev4
 - New features:
 - * Added mutex for HW modules `HASHCRYPT` and `CASPER`. Enabled by `MBEDTLS_T-HREADING_C`
- 2.16.6_rev3
 - New features:
 - * Added support for KW45 device with latest Sentinel200. Port of SSS API `mbedtls` implementation to KW45.

- 2.16.6_rev2
 - New features:
 - * Added support for SW computing AES-192/256 while using DCP driver.
- 2.16.6_rev1
 - New features:
 - * Added support for NIST P-521 elliptic curve with CASPER driver.
 - * Added support for using multiple elliptic curves at once with CASPER driver.
- 2.16.6
 - New features:
 - * Ported mbedTLS 2.16.6 to SDK.
- 2.16.2_rev2
 - Bug fixes:
 - * Add support for HASHCRYPT context switch check, Hashcrypt without context switch is not able to calculate SHA in parallel with AES. HW acceleration of SHA is disabled by default in MbedTLS integration, enabled on chip with context switch.
- 2.16.2_rev1
 - Bug fixes:
 - * Add support for CTR_DRBG using AES-128 for crypto engines without AES-256 capability.
- 2.16.2
 - New features:
 - * Ported mbedTLS 2.16.2 to SDK.
- 2.13.1_rev5
 - Bug fixes:
 - * `ecp_alt_ksdk.c` fix CASPER port for ECJPAKE shortcut when points equal 1. This case is point addition and this shortcut follows original `mbedtls_ecp_muladd()` implementation which is required for `ecjpake_ecp_add3()`.
- 2.13.1_rev4
 - New features:
 - * Added support for NIST P-384 elliptic curve with CASPER driver.
- 2.13.1_rev3
 - Bug fixes:
 - * Force align AES_CCM and AES_GCM self-test keys to fix unaligned key issue when using HW acceleration.
- 2.13.1_rev2
 - Bug fixes:
 - * Disable default HW acceleration of SHA in parallel with AES.
- 2.13.1_rev1
 - Bug fixes:
 - * Fixed incorrect macro check when skipping AES-192 or AES-256
- 2.13.1
 - New features:
 - * Ported mbedTLS 2.13.1 to KSDK.
- 2.12.0_rev1
 - New features:

- * Added support for NIST P-256 elliptic curve with CASPER driver.
- 2.12.0
 - New features:
 - * Ported mbedTLS 2.12.0 to KSDK.
- 2.9.0_rev2
 - New features:
 - * Added support for Hashcrypt driver.
- 2.9.0_rev1
 - New features:
 - * Added support for CASPER driver.
- 2.9.0
 - New features:
 - * Ported mbedTLS 2.9.0 to KSDK.
- 2.6.0_rev2
 - Bug fixes:
 - * ssl_cookie.c now uses SHA256 for COOKIE_MD (instead of original SHA224). Some hw crypto acceleration (such as CAU3) don't support SHA224 but all support SHA256.
- 2.6.0_rev1
 - Bug fixes:
 - * ksdk_mbedtls.c bignum functions now read sign of input mbedtls_mpi at beginning of functions to properly support in place computations (when output bignum is the same as one of input bignums). Affected functions: mbedtls_mpi_mul_mpi(), mbedtls_mpi_mod_mpi(), ecp_mul_comb().
- 2.6.0
 - New features:
 - * Ported mbedTLS 2.6.0 to KSDK.
 - * Added MBEDTLS_FREESCALE_FREERTOS_CALLOC_ALT to allow alternate implementation of pvPortCalloc() when using /middleware/mbedtls/port/ksdk/ksdk_mbedtls.c.
- 2.5.1_rev1
 - New features:
 - * Added support for DCP driver.
- 2.5.1
 - New features:
 - * Ported mbedTLS 2.5.1 to KSDK.
- 2.4.2_rev2
 - New features:
 - * Added Curve25519 support for CAU3.
 - * Added MBEDTLS_ECP_MUL_MXZ_ALT configuration parameter enabling overloading of ecp_mul_mxz().
- 2.4.2_rev1
 - New features:
 - * Added support for CAU3 driver.
 - * Added new files:
 - * /middleware/mbedtls/port/ksdk/des_alt.c - contains regular software implementation of D-

- ES algorithm with added MBEDTLS_DES3_SETKEY_DEC_ALT and MBEDTLS_DES3_SETKEY_ENC_ALT config parameters.
- * /middleware/mbedtls/port/ksdk/des_alt.h - contains modified mbedtls_des_context and mbedtls_des3_context structures.
 - * Added MBEDTLS_DES3_SETKEY_DEC_ALT configuration parameter enabling reloading of mbedtls_des3_set2key_dec() and mbedtls_des3_set3key_dec().
 - * Added MBEDTLS_DES3_SETKEY_ENC_ALT configuration parameter enabling reloading of mbedtls_des3_set2key_enc() and mbedtls_des3_set3key_enc().
- 2.4.2
 - New features:
 - * Ported mbedtls 2.4.2 to KSDK 2.0.0.
 - * Added CRYPTO_InitHardware() function.
 - * Added new file:
 - /middleware/mbedtls/port/ksdk/ksdk_mbedtls.h - contains declaration of CRYPTO_InitHardware() function and should be included in applications.
 - 2.3.0_rev1
 - New features:
 - * Added support for CAAM driver.
 - * In LTC-specific wrapper, allocate temporary integers from heap in one large block.
 - 2.3.0
 - New features:
 - * Ported mbedtls 2.3.0 to KSDK 2.0.0.
 - 2.2.1
 - New features:
 - * Ported mbedtls 2.2.1 to KSDK 2.0.0.
 - * Added support of MMCAU cryptographic acceleration module. Accelerated MD5, SHA, AES, and DES.
 - * Added support of LTC cryptographic acceleration module. Accelerated AES, DES, and PKHA.
 - * Added new files:
 - * /middleware/mbedtls/port/ksdk/ksdk_mbedtls.c - alternative implementation of cryptographic algorithm functions using LTC and MMCAU module drivers.
 - * /middleware/mbedtls/port/ksdk/ksdk_mbedtls_config.h - configuration settings used by mbedtls KSDK bare metal examples.
 - * Added mbedtls KSDK bare-metal examples:
 - /boards/<board name>/demo_apps/mbedtls/mbedtls_benchmark - KSDK mbedtls benchmark application.
 - /boards/<board name>/demo_apps/mbedtls/mbedtls_selftest - KSDK mbedtls self-test application.
 - * Added MBEDTLS_GCM_CRYPT_ALT configuration parameter enabling reloading of mbedtls_gcm_crypt_and_tag().
 - * Added MBEDTLS_ECP_MUL_COMB_ALT to enable alternate implementation of ecp_mul_comb().
 - * Added MBEDTLS_ECP_ADD_ALT configuration parameter enabling reloading of ecp_add().

- * Added MBEDTLS_DES_SETKEY_DEC_ALT configuration parameter enabling reloading of mbedtls_des_setkey_dec(), mbedtls_des3_set2key_dec() and mbedtls_des3_set3key_dec().
- * Added MBEDTLS_DES_SETKEY_ENC_ALT configuration parameter enabling reloading of mbedtls_des_setkey_enc(), mbedtls_des3_set2key_enc() and mbedtls_des3_set3key_enc().
- * Added MBEDTLS_DES_CRYPT_CBC_ALT configuration parameter enabling reloading of mbedtls_des_crypt_cbc().
- * Added MBEDTLS_DES3_CRYPT_CBC_ALT configuration parameter enabling reloading of mbedtls_des3_crypt_cbc().
- * Added MBEDTLS_AES_CRYPT_CBC_ALT configuration parameter enabling reloading of mbedtls_aes_crypt_cbc().
- * Added MBEDTLS_AES_CRYPT_CTR_ALT configuration parameter enabling reloading of mbedtls_aes_crypt_ctr().
- * Added MBEDTLS_CCM_CRYPT_ALT configuration parameter enabling reloading of mbedtls_ccm_encrypt_and_tag() and mbedtls_ccm_auth_decrypt().
- * Added MBEDTLS_MPI_ADD_ABS_ALT configuration parameter enabling reloading of mbedtls_mpi_add_abs().
- * Added MBEDTLS_MPI_SUB_ABS_ALT configuration parameter enabling reloading of mbedtls_mpi_sub_abs().
- * Added MBEDTLS_MPI_EXP_MOD_ALT configuration parameter enabling reloading of mbedtls_mpi_exp_mod().
- * Added MBEDTLS_MPI_MUL_MPI_ALT configuration parameter enabling reloading of mbedtls_mpi_mul_mpi().
- * Added MBEDTLS_MPI_MOD_MPI_ALT configuration parameter enabling reloading of mbedtls_mpi_mod_mpi().
- * Added MBEDTLS_MPI_GCD_ALT configuration parameter enabling reloading of mbedtls_mpi_gcd().
- * Added MBEDTLS_MPI_INV_MOD_ALT configuration parameter enabling reloading of mbedtls_mpi_inv_mod().
- * Added MBEDTLS_MPI_IS_PRIME_ALT configuration parameter enabling reloading of mbedtls_mpi_is_prime().
- * Added encrypt/decrypt mode to mbedtls_des_context and mbedtls_des3_context structure.
- * Added carriage return '\r' for mbedtls_printf() in self test functions.

MOTOR_CONTROL for KSDK

Current version is 1.1.0

- 1.1.0
 - Initial version.

Multicore SDK

The current version of Multicore SDK is 2.15.0

- 2.15.0
 - Multicore SDK component versions:
 - * embedded Remote Procedure Call (eRPC) v1.12.0
 - * eRPC generator (erpcgen) v.1.12.0
 - * Multicore Manager (MCMgr) v4.1.5
 - * RMPMsg-Lite v5.1.1
 - New features:
 - * eRPC: Add dynamic/static option for transport init, GitHub PR #361.
 - * eRPC: Fix receive error value for spidev, GitHub PR #363.
 - * eRPC: UartTransport::init adaptation to changed driver.
 - * eRPC: Fix typo in assert, GitHub PR #371.
 - * eRPC,erpcgen: Move enums to enum classes, GitHub PR #379.
 - * eRPC: Fixed rpmsg tty transport to work with serial transport, GitHub PR #373.
 - * eRPC,erpcgen: Winsock2 support, GitHub PR #365.
 - * eRPC,erpcgen: Feature/support multiple clients, GitHub PR #271.
 - * eRPC,erpcgen: Feature/buffer head - Framed transport header data stored in Message-Buffer, GitHub PR #378.
 - * eRPC,erpcgen: Add experimental Java support.
 - * MCMgr: Added notification into MCMGR_EarlyInit and mcmgr_early_init_internal functions to avoid using uninitialized data in their implementations.
 - * RMPMsg-Lite: Minor changes in platform and env. layers, minor test code updates.
- 2.14.0
 - Multicore SDK component versions:
 - * embedded Remote Procedure Call (eRPC) v1.11.0
 - * eRPC generator (erpcgen) v.1.11.0
 - * Multicore Manager (MCMgr) v4.1.4
 - * RMPMsg-Lite v5.1.0
 - New features:
 - * eRPC: Makefiles update, GitHub PR #301.
 - * eRPC: Resolving warnings in Python, GitHub PR #325.
 - * eRPC: Python3.8 is not ready for usage of typing.Any type, GitHub PR #325.
 - * eRPC: Improved codec function to use reference instead of address, GitHub PR #324.
 - * eRPC: Fix NULL check for pending client creation, GitHub PR #341.
 - * eRPC: Replace sprintf with snprintf, GitHub PR #343.
 - * eRPC: Use MU_SendMsg blocking call in MU transport.
 - * eRPC: New LPSPI and LPI2C transport layers.
 - * eRPC: Freeing static objects, GitHub PR #353.
 - * eRPC: Fixed casting in deinit functions, GitHub PR #354.
 - * eRPC: Align LIBUSBSIO.GetNumPorts API use with libusbsio python module v. 2.1.11.
 - * erpcgen: Renamed temp variable to more generic one, GitHub PR #321.
 - * erpcgen: Add check that string read is not more than max length, GitHub PR #328.
 - * erpcgen: Move to g++ in pytest, GitHub PR #335.

- * erpcgen: Use build=release for make, GitHub PR #334.
- * erpcgen: Removed boost dependency, GitHub PR #346.
- * erpcgen: Mingw support, GitHub PR #344.
- * erpcgen: VS build update, GitHub PR #347.
- * erpcgen: Modified name for common types macro scope, GitHub PR #337.
- * erpcgen: Fixed memcpy for template, GitHub PR #352.
- * eRPC,erpcgen: Change default build target to release + adding artefacts, GitHub PR #334.
- * eRPC,erpcgen: Remove redundant includes, GitHub PR #338.
- * eRPC,erpcgen: Many minor code improvements, GitHub PR #323.
- * MCMgr: Avoid calling tx isr callbacks when respective Messaging Unit Transmit Interrupt Enable flag is not set in the CR/TCR register.
- * MCMgr: Messaging Unit RX and status registers are cleared after the initialization.
- * RPMsg-Lite: Resolved issues in ThreadX env. layer implementation.
- * RPMsg-Lite: Added aarch64 support.
- * RPMsg-Lite: Increased the queue size to (2 * RL_BUFFER_COUNT) to cover zero copy cases.
- 2.13.0_imxrt1180a0
 - Multicore SDK component versions:
 - * embedded Remote Procedure Call (eRPC) v1.10.0
 - * eRPC generator (erpcgen) v.1.10.0
 - * Multicore Manager (MCMgr) v4.1.3
 - * RPMsg-Lite v5.0.0
 - New features:
 - * MCMgr, RPMsg-Lite: Added porting layers for imxrt1180.
 - * MCMgr: mu_isr() updated to avoid calling tx isr callbacks when respective Transmit Interrupt Enable flag is not set in the CR/TCR register.
 - * RPMsg-Lite, eRPC: RPMsg_Lite queue size adjusted.
 - * eRPC: MU transport layer switched to blocking MU_SendMsg() API use.
- 2.13.0
 - Multicore SDK component versions:
 - * embedded Remote Procedure Call (eRPC) v1.10.0
 - * eRPC generator (erpcgen) v.1.10.0
 - * Multicore Manager (MCMgr) v4.1.3
 - * RPMsg-Lite v5.0.0
 - New features:
 - * eRPC: MUTransport adaptation to new supported SoCs.
 - * eRPC: Simplifying CI with installing dependencies using shell script, GitHub PR #267.
 - * eRPC: Using event for waiting for sock connection in TCP python server, formatting python code, C specific includes, GitHub PR #269.
 - * eRPC: Endianness agnostic update, GitHub PR #276.
 - * eRPC: Assertion added for functions which are returning status on freeing memory, GitHub PR #277.
 - * eRPC: Fixed closing arbitrator server in unit tests, GitHub PR #293.
 - * eRPC: Makefile updated to reflect the correct header names, GitHub PR #295.
 - * eRPC: Compare value length to used length() in reading data from message buffer, GitHub

- PR #297.
- * eRPC: Add TCP_NODELAY option to python, GitHub PR #298.
- * eRPC: Replace EXPECT_TRUE with EXPECT_EQ in unit tests, GitHub PR #318.
- * eRPC: Adapt rpmsg_lite based transports to changed rpmsg_lite_wait_for_link_up() API parameters.
- * eRPC, erpcgen: Better distinguish which file can and cannot be linked by C linker, GitHub PR #266.
- * eRPC, erpcgen: Stop checking if pointer is NULL before sending it to the erpc_free function, GitHub PR #275.
- * eRPC, erpcgen: Changed api to count with more interfaces, GitHub PR #304.
- * erpcgen: Check before reading from heap the buffer boundaries, GitHub PR #287.
- * erpcgen: Several fixes for tests and CI, GitHub PR #289.
- * erpcgen: Refactoring erpcgen code, GitHub PR #302.
- * erpcgen: Fixed assigning const value to enum, GitHub PR #309.
- * erpcgen: Enable runTesttest_enumErrorCode_allDirection, serialize enums as int32 instead of uint32.
- * MCMgr: mcmgr_mu_internal.c code adaptation to new supported SoCs.
- * RPMsg-Lite: Improved debug check buffers implementation - instead of checking the pointer fits into shared memory check the presence in the VirtIO ring descriptors list.
- * RPMsg-Lite: Timeout parameter added to rpmsg_lite_wait_for_link_up API function.
- * RPMsg-Lite: VRING_SIZE is set based on number of used buffers now (as calculated in vring_init) - updated for all platforms that are not communicating to Linux rpmsg counterpart.
- * RPMsg-Lite: Fixed wrong RL_VRING_OVERHEAD macro comment in platform.h files.
- * RPMsg-Lite: Misra corrections.
- 2.12.0_imx93
 - Multicore SDK component versions:
 - * embedded Remote Procedure Call (eRPC) v1.9.1
 - * eRPC generator (erpcgen) v.1.9.1
 - * Multicore Manager (MCMgr) v4.1.2
 - * RPMsg-Lite v4.0.1
 - New features:
 - * RPMsg-Lite: Added porting layers for i.mx93 device.
- 2.12.0
 - Multicore SDK component versions:
 - * embedded Remote Procedure Call (eRPC) v1.9.1
 - * eRPC generator (erpcgen) v.1.9.1
 - * Multicore Manager (MCMgr) v4.1.2
 - * RPMsg-Lite v4.0.0
 - New features:
 - * eRPC: Construct the USB CDC transport, rather than a client, GitHub PR #220.
 - * eRPC: Fix premature import of package, causing failure when attempting installation of Python library in a clean environment, GitHub PR #38, #226.
 - * eRPC: Improve python detection in make, GitHub PR #225.
 - * eRPC: Fix several warnings with deprecated call in pytest, GitHub PR #227.

- * eRPC: Fix freeing union members when only default need be freed, GitHub PR #228.
 - * eRPC: Fix making test under Linux, GitHub PR #229.
 - * eRPC: Assert costumizing, GitHub PR #148.
 - * eRPC: Fix corrupt clientList bug in TransportArbitrator, GitHub PR #199.
 - * eRPC: Fix build issue when invoking g++ with -Wno-error=free-nonheap-object, GitHub PR #233.
 - * eRPC: Fix inout cases, GitHub PR #237.
 - * eRPC: Remove ERPC_PRE_POST_ACTION dependency on return type, GitHub PR #238.
 - * eRPC: Adding NULL to ptr when codec function failed, fixing memcpy when fail is present during deserialization, GitHub PR #253.
 - * eRPC: MessageBuffer usage improvement, GitHub PR #258.
 - * eRPC: Get rid for serial and enum34 dependency (enum34 is in python3 since 3.4 (from 2014)), GitHub PR #247.
 - * eRPC: Several MISRA violations addressed.
 - * eRPC: Fix timeout for Freertos semaphore, GitHub PR #251.
 - * eRPC: Use of rpmsg_lite_wait_for_link_up() in rpmsg_lite based transports, GitHub PR #223.
 - * eRPC: Fix codec nullptr dereferencing, GitHub PR #264.
 - * erpcgen: Fix two syntax errors in erpcgen Python output related to non-encapsulated unions, improved test for union, GitHub PR #206, #224.
 - * erpcgen: Fix serialization of list/binary types, GitHub PR #240.
 - * erpcgen: Fix empty list parsing, GitHub PR #72.
 - * erpcgen: Fix templates for malloc errors, GitHub PR #110.
 - * erpcgen: Get rid of encapsulated union declarations in global scale, improve enum usage in unions, GitHub PR #249, #250.
 - * erpcgen: Fix compile error:UniqueIdChecker.cpp:156:104:'sort' was not declared, GitHub PR #265.
 - * MCMgr: Update mcmgr_stop_core_internal() implementations to set core state to kMCMGR_ResetCoreState.
 - * RPMsg-Lite: Introduce new rpmsg_lite_wait_for_link_up() API function - this allows to avoid using busy loops in rtos environments, GitHub PR #21.
 - * RPMsg-Lite: Adjust rpmsg_lite_is_link_up() to return RL_TRUE/RL_FALSE.
- 2.11.1
 - Multicore SDK component versions:
 - * embedded Remote Procedure Call (eRPC) v1.9.0
 - * eRPC generator (erpcgen) v.1.9.0
 - * Multicore Manager (MCMgr) v4.1.1
 - * RPMsg-Lite v3.2.1
 - New features:
 - * RPMsg-Lite: Add support for custom shared memory arrangement per the RPMsg_Lite instance.
 - 2.11.0
 - Multicore SDK component versions:
 - * embedded Remote Procedure Call (eRPC) v1.9.0

- * eRPC generator (erpcgen) v.1.9.0
- * Multicore Manager (MCMgr) v4.1.1
- * RPMsg-Lite v3.2.0
- New features:
 - * eRPC: Improving template usage, GitHub PR #153.
 - * eRPC: run_clang_format.py cleanup, GitHub PR #177.
 - * eRPC: Build TCP transport setup code into liberpc, GitHub PR #179.
 - * eRPC: Fix multiple definitions of g_client error, GitHub PR #180.
 - * eRPC: Fix memset past end of buffer in erpc_setup_mbf_static.cpp, GitHub PR #184.
 - * eRPC: Fix deprecated error with newer pytest version, GitHub PR #203.
 - * eRPC: Allow used LIBUSBSIO device index being specified from the Python command line argument.
 - * eRPC, erpcgen: Static allocation support and usage of rpmsg static FreeRTOSs related API, GitHub PR #168, #169.
 - * erpcgen: Remove redundant module imports in erpcgen, GitHub PR #196.
 - * RPMsg-Lite: Improve static allocations - allow OS-specific objects being allocated statically, GitHub PR #14.
 - * RPMsg-Lite: Minor Misra and typo corrections, GitHub PR #19, #20.
- 2.10.0
 - Multicore SDK component versions:
 - * embedded Remote Procedure Call (eRPC) v1.8.1
 - * eRPC generator (erpcgen) v.1.8.1
 - * Multicore Manager (MCMgr) v4.1.1
 - * RPMsg-Lite v3.1.2
 - New features:
 - * eRPC: Fix misra erpc c, GitHub PR #158.
 - * eRPC: Allow conditional compilation of message_loggers and pre_post_action.
 - * eRPC: New i2c_slave_transport transport introduced.
 - * eRPC: (D)SPI slave transports updated to avoid busy loops in rtos environments.
 - * erpcgen: Re-implement EnumMember::hasValue(), GitHub PR #159.
 - * erpcgen: Fixing several misra issues in shim code, erpcgen and unit tests updated, GitHub PR #156.
 - * erpcgen: Fix bison file, GitHub PR #156.
 - * RPMsg-Lite: Fixed incorrect description of the rpmsg_lite_get_endpoint_from_addr function.
 - * RPMsg-Lite: Updated RL_BUFFER_COUNT documentation.
 - * RPMsg-Lite: env_print macro adjusted to address MISRA 21.6 rule in MCUXpressoSDK projects.
- 2.9.0
 - Multicore SDK component versions:
 - * embedded Remote Procedure Call (eRPC) v1.8.0
 - * eRPC generator (erpcgen) v.1.8.0
 - * Multicore Manager (MCMgr) v4.1.1
 - * RPMsg-Lite v3.1.1
 - New features:

- * eRPC: Support win32 thread, GitHub PR #108.
 - * eRPC: Add mbed support for malloc() and free(), GitHub PR #92.
 - * eRPC: Update makefile.
 - * eRPC: Fixed warnings and error with using MessageLoggers, GitHub PR #127.
 - * eRPC: Extend error msg for python server service handle function, GitHub PR #132.
 - * eRPC: Update CMSIS UART transport layer to avoid busy loops in rtos environments, introduce semaphores.
 - * eRPC: Introduced pre and post callbacks for eRPC call, GitHub PR #131.
 - * eRPC: Introduced new USB CDC transport.
 - * eRPC: Introduced new Linux spidev-based transport.
 - * eRPC: SPI transport update to allow usage without handshaking GPIO.
 - * eRPC: Native *WIN32 erpc serial transport and threading*.
 - * *eRPC: Arbitrator deadlock fix, TCP transport updated, TCP setup functions introduced, GitHub PR #121.*
 - * *eRPC: Update of matrix_multiply.py example: Add -serial and -baud argument, GitHub PR #137.*
 - * *eRPC: Added formatting extension for VSC, GitHub PR #134.*
 - * *eRPC: Update of .clang-format, GitHub PR #140.*
 - * *eRPC: Update of erpc_framed_transport.cpp: return error if received message has zero length, GitHub PR #141.*
 - * *eRPC, erpcgen: Fixed error messages produced by -Wall -Wextra -Wshadow -pedantic-errors compiler flags, GitHub PR #136, #139.*
 - * *eRPC, erpcgen: Core re-formatted using Clang version 10.*
 - * *erpcgen: Enable deallocation in server shim code when callback/function pointer used as out parameter in IDL.*
 - * *erpcgen: Removed '\$' character from generated symbol name in '\$union' suffix, GitHub PR #103.*
 - * erpcgen: Resolved mismatch between C++ and Python for callback index type, GitHub PR #111.
 - * erpcgen: Python generator improvements, GitHub PR #100, #118.
 - * erpcgen: Fixed error messages produced by -Wall -Wextra -Wshadow -pedantic-errors compiler flags, GitHub PR #136.
 - * erpcgen: Introduce ustring type for unsigned char and force cast to char*, GitHub PR #125.
 - * RPMsg-Lite: Introduced `RL_ALLOW_CONSUMED_BUFFERS_NOTIFICATION` config option to allow opposite side notification sending each time received buffers are consumed and put into the queue of available buffers.
 - * RPMsg-Lite: Added environment layers for Threadx.
- 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).
 - * RMPmsg-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
 - * RMPmsg-Lite v2.0.2
 - New features:
 - * RMPmsg-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
 - * RMPmsg-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.
 - * RMPmsg-Lite: New API rmpmsg_queue_get_current_size()
 - * RMPmsg-Lite: Fixed bug in interrupt handling for lpc5411x, lpc5410x
 - * RMPmsg-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
 - * RMPmsg-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 RPMMSG 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 RPMMsg 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 (RPMMsgZC) 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

NTAG I2C plus library

The current version is 1.0.0.

- 1.0.0
 - initial release.

RTCESL for KSDK

Current version is 4.3

- 4.3
 - Initial version.

SAFETY_IEC60730B for KSDK

Current version is 1.1.0

- 1.1.0
 - Initial version.

Host SDIF driver for MCUXpresso SDK

The current driver version is 2.4.1.

- 2.4.1
 - Improvements
 - * Added macro SDMMCHOST_SUPPORT_VOLTAGE_CONTROL.
- 2.4.0
 - Improvements
 - * Removed deprecated api in SDIF host driver.
 - * Added SDMMCHOST_ConvertDataToLittleEndian api.
 - * Added capability/maxBlockCount/maxBlockSize in host decriptior.
 - * Added mutual exclusive access for function init/deinit/reset/transfer function.
 - * Fixed violations of MISRA C-2012 rule 10.1.
- 2.3.1
 - Improvements
 - * Added host instance capability macro.
 - * Added clear card inserted/removed event when card removed/inserted interrupt generated.
 - * Increased the reset timeout value to fix the data machine still busy after sdif reset issue.
 - * Enabled the error recovery function by adding host reset operations.
 - Bug Fixes
 - * Fixed violations of MISRA C-2012 rule 11.9, 15.7, 4.7, 16.4, 10.1, 10.3, 10.4, 11.3, 14.4, 10.6, 17.7, 16.1, 16.3.
- 2.3.0
 - Improvements
 - * Merged the host controller driver from polling/freertos/interrupt to non_blocking/blocking.
 - * Added SDMMC OSA layer to support muxtex access/event/delay.
- 2.2.14
 - Bug Fixes
 - * Fixed uninitialized value Coverity issue.
- 2.2.13

- Improvements:
 - * Added host reset after the card being powered on for host controller SDIF to fix the DATA_BUSY issue.
 - * Removed the SDIF_Reset from SDMMCHOST_Reset.
- 2.0.0
 - Initial version

MMC Card driver for MCUXpresso SDK

The current driver version is 2.5.0.

- 2.5.0
 - Improvements
 - * Added api MMC_SetSleepAwake to support enter/exit sleep state.
 - * Added new api MMC_PollingCardStatusBusy for application polling card status.
 - * Removed deprecated api in mmc driver and mark MMC_HostReset as deprecated.
 - * Improved the read/write/erase function flow.
 - * Added mutual exclusive access for init/deinit/read/write/erase function.
 - * Fixed violations of MISRA C-2012 rule 4.7, 17.7, 10.7, 10.4, 13.5, 14.4, 10.6.
- 2.4.1
 - Improvements
 - * Improved the voltage window argument of CMD1 according to host capability instead of use card ocr directly.
 - * Added host HS200/HS400/8bit bus width capability validation during card initialization.
 - * Used cache line size align buffer for MMC relate api.
 - * Increased the CMD13 timeout count to avoid polling CMD13 time out issue.
 - Bug Fixes
 - * Fixed violations of MISRA C-2012 rule 11.9, 15.7, 4.7, 16.4, 10.1, 10.3, 10.4, 11.3, 14.4, 10.6, 17.7, 16.1, 16.3.
- 2.4.0
 - Improvements
 - * Added new apis MMC_EnableCacheControl/MMC_FlushCache to support cache feature.
- 2.3.1
 - Improvements
 - * Removed the dead loop while polling DAT0 and CMD13 instead of using timeout mechanism.
 - * Added card state check before switching to HS400 to improve the emmc initialization stability.
 - * Removed the redundant operation of memset internal buffer in MMC_WriteBlocks function.
 - Bug Fixes
 - * Fixed the sandisk emmc always busy while sending CMD1 without supported voltage provide in argument.
- 2.3.0

- Improvements
 - * Deprecated api MMC_PowerOnCard/MMC_PowerOffCard by api MMC_SetCardPower.
 - * Added internalBuffer in mmc_card_t and removed rawCid/rawCsd/rawExtendedCsd.
 - * Added retuning support during data transfer under HS200 mode.
 - * Increased the read/write blocks failed retry times for stability.
 - * Added delay while retry the CMD1 for stability.
 - * Added legacy card support, the card not support CMD6, CMD8.
- 2.2.13
 - Improvements
 - * Used the boot mode value instead of boot mode mask value as the parameter of MMC_SetBootConfig to improve user experience.
 - * Removed dynamic voltage switch feature for mmc, according to JEDEC standard, the voltage should be fixed after power up.
- 2.2.12
 - Improvement
 - * Increased the CMD1 retry times in the MMC card driver to improve driver compatibility.
 - Bug Fixes
 - * Fixed the build warning by changing the old style function declaration static status_t inline to static inline status_t(found by adding -Wold-style-declaration in armgcc build flag).
 - * Fixed the fall through build warning by adding SUPPRESS_FALL_THROUGH_WARNING() in mmc driver.
- 2.2.7
 - Bug Fixes
 - * Fixed MDK 66-D warning.
- 2.2.6
 - Improvements
 - * Saved MMC OCR registers while sending CMD1 with argument 0.
 - Bug Fixes
 - * Added MMC_PowerOn function in which there is delay function after powerup sdcard. Otherwise, the card initialization by fail.
- 2.2.5
 - Improvements
 - * Added SDMMC_ENABLE_SOFTWARE_TUNING to enable/disable software tuning and it is disabled by default.
- 2.2.4
 - Bug Fixes
 - * Fixed DDR mode data sequence miss issue, which is caused by NIBBLE_POS.
 - Improvements
 - * Increased g_sdmmc 512byte to improve the performance when application use a non-word align data buffer address.
 - * Used OCR access mode bits to determine the mmccard high capacity flag.
- 2.2.3
 - Bug Fixes
 - * Added response check for send operation condition command. If not checked, the card may occasionally init fail.

- 2.2.1
 - Improvements
 - * Improved MMC Boot feature.
- 2.2.0
 - Improvements
 - * Optimized tuning/mmc switch voltage/mmc select power class/mmc select timing function.
 - * Added strobe dll for mmc HS400 mode.
 - * Added write complete wait operation for MMC_Write to fix command timeout issue.
- 2.1.2
 - Improvements
 - * Improved SDMMC to support eMMC v5.0.
 - Bug Fixes
 - * Fixed incorrect comparison between count and length in MMC_ReadBlocks/MMC_WriteBlocks.
- 2.1.1
 - Bug Fixes
 - * Fixed the block range boundary error when transferring data to MMC card.
- 2.1.0
 - Improvements
 - * Optimized the function of setting maximum data bus width for MMC card.
- 2.0.0
 - Initial version

SD Card driver for MCUXpresso SDK

The current driver version is 2.4.2.

- 2.4.2
 - Improvements
 - * Improved the erase timeout calculation logical in function SD_EraseBlocks according to SD specifications.
 - * Added polling erase done status after each erase operations.
- 2.4.1
 - Improvements
 - * Added macro SDMMCHOST_SUPPORT_VOLTAGE_CONTROL for the host which not support voltage control.
- 2.4.0
 - Improvements
 - * Removed deprecated api in sd driver.
 - * Added new api SD_PollingCardStatusBusy for application polling card status.
 - * Improved the read/write/erase function flow.
 - * Improved the signal line voltage switch flow.
 - * Added powerOnDelayMS/powerOffDelayMS in sd_usr_param_t to allow redefine the

- default power on/off delay.
- * Added mutual exclusive access for init/deinit/read/write/erase function.
- * Fixed the driver strength configurations missed when timing mode switch to non SDR50/-SDR104 mode.
- * Fixed violations of MISRA C-2012 rule 4.7, 17.7, 10.7, 10.4, 13.5, 14.4.
- 2.3.3
 - Improvements
 - * Added host SDR timing mode capability validation during card initialization.
 - * Added polling card ready for data status when transfer data failed.
 - * Used cache line size align buffer for SD initialization api.
 - Bug Fixes
 - * Fixed violations of MISRA C-2012 rule 11.9, 15.7, 4.7, 16.4, 10.1, 10.3, 10.4, 11.3, 14.4, 10.6, 17.7, 16.1, 16.3.
- 2.3.2
 - Improvements
 - * Moved power off function after card detect in SD_Init for DAT3 detect card feature.
- 2.3.1
 - Improvements
 - * Removed the dead loop while polling DAT0 and CMD13 instead of using timeout mechanism.
- 2.3.0
 - Improvements
 - * Marked api SD_HostReset/SD_PowerOnCard/SD_PowerOffCard/SD_WaitCardDetect-Status as deprecated.
 - * Added new api SD_SetCardPower/SD_PollingCardDetectStatus/SD_HostDoReset.
 - * Added internalBuffer in sd_card_t and removed rawCid/rawCsd/rawScr.
 - * Added retuning support during data transfer under SDR50/SDR104 mode.
 - * Increased the read/write blocks failed retry times for stability.
 - * Added delay while retry the ACMD41 for stability.
- 2.2.12
 - Improvements
 - * Increased the sd io driver strength for SD2.0 card.
 - Bug Fixes
 - * Fixed the build warning by changing the old style function declaration static status_t inline to static inline status_t(found by adding -Wold-style-declaration in armgcc build flag).
- 2.2.10
 - Bug Fixes
 - * Added event value check for all the FreeRTOS events to fix program hangs when a card event occurs before create.
- 2.2.7
 - Bug Fixes
 - * Fixed MDK 66-D warning.
- 2.2.5
 - Improvements
 - * Added SD_ReadStatus api to get 512bit SD status.

- * Added error log support in sdcard functions.
- * Added SDMMC_ENABLE_SOFTWARE_TUNING to enable/disable software tuning and it is disabled by default.
- 2.2.4
 - Bug Fixes
 - * Fixed DDR mode data sequence miss issue, which is caused by NIBBLE_POS.
 - Improvements
 - * Increased g_sdmmc 512byte to improve the performance when application use a non-word align data buffer address.
 - * Enabled auto cmd12 for SD read/write.
- 2.2.3
 - Bug Fixes
 - * Added response check for send operation condition command. If not checked, the card may occasionally init fail.
- 2.2.1
 - Improvements
 - * Kept SD_Init function for forward compatibility.
- 2.2.0
 - Improvements
 - * Separated the SD/MMC/SDIO init API to xxx_CardInit/xxx_HostInit.
 - * SD_Init/SDIO_Init will be deprecated in the next version.
- 2.1.6
 - Improvements
 - * Enhanced SD IO default driver strength.
- 2.1.5
 - Bug Fixes
 - * Fixed Coverity issue.
 - * Fixed SD v1.x card write fail issue. It was caused by the block length set error.
 - * Fixed card cannot detect dynamically.
- 2.1.3
 - Bug Fixes
 - * Fixed Non high-speed sdcard init fail at switch to high speed.
 - Improvements
 - * Added Delay for SDCard power up.
- 2.1.2
 - Improvements
 - * Improved SDMMC to support SD v3.0.
- 2.1.1
 - Bug Fixes
 - * Fixed the bit mask error in the SD card switch to high speed function.
 - Improvements
 - * Optimized the SD card initialization function.
- 2.1.0
 - Bug Fixes
 - * Changed the callback mechanism when sending a command.

- * Fixed the performance low issue when transferring data.
- Improvements
 - * Changed the name of some error codes returned by internal function.
 - * Merged all host related attributes to one structure.
- 2.0.0
 - Initial version.

SDIO Card driver for MCUXpresso SDK

The current driver version is 2.4.1.

- 2.4.1
 - Improvements
 - * Added macro SDMMCHOST_SUPPORT_VOLTAGE_CONTROL for the host which not support voltage control.
- 2.4.0
 - Improvements
 - * Removed deprecated api in sdio driver.
 - * Improved the signal line voltage switch flow.
 - * Added powerOnDelayMS/powerOffDelayMS in sdio_usr_param_t to allow redefine the default power on/off delay.
 - * Added mutual exclusive access for init/deinit/direct/extend function.
 - * Fixed violations of MISRA C-2012 rule 4.7, 17.7, 10.1, 12.2.
- 2.3.3
 - Bug Fixes
 - * Fixed logical dead code coverity issue.
 - Improvements
 - * Removed deprecated api in sdio driver.
- 2.3.2
 - Improvements
 - * Added host SDR timing mode capability validation during card initialization.
 - * Used cache line size align buffer for SDIO initialization api.
 - Bug Fixes
 - * Fixed violations of MISRA C-2012 rule 11.9, 15.7, 4.7, 16.4, 10.1, 10.3, 10.4, 11.3, 14.4, 10.6, 17.7, 16.1, 16.3.
- 2.3.1
 - Improvements
 - * Moved power off function after card detect in SD_Init for DAT3 detect card feature.
- 2.3.0
 - Improvements
 - * Marked api SDIO_HostReset/SDIO_PowerOnCard/SDIO_PowerOffCard/SDIO_Wait-CardDetectStatus as deprecated.
 - * Added new api SDIO_SetCardPower/SDIO_PollingCardDetectStatus/SDIO_HostDo-Reset.

- * Added internalBuffer in sdio_card_t for card register content extract and improve the data access efficiency.
- * Added retry function after switch to target timing failed in SDIO_SelectBusTiming.
- * Changed default bus clock from 400KHZ to 25MHZ.
- 2.2.13
 - Improvements
 - * Removed the sdio card interrupt from sdio host initialization, since the card interrupt enablement should be determined by application.
 - Bug Fixes
 - * Fixed Out-of-bounds write Coverity issue.
- 2.2.12
 - Improvements
 - * Added manual tuning function for looking for the tuning window automatically.
 - * Fixed the build warning by changing the old style function declaration static status_t inline to static inline status_t(found by adding -Wold-style-declaration in armgcc build flag).
 - * Fixed the fall through build warning by adding SUPPRESS_FALL_THROUGH_WARNING() in sdio driver.
- 2.2.11
 - Bug Fixes
 - * Added check card async interrupt capability in function SDIO_GetCardCapability.
 - * Fixed OUT OF BOUNDS access in function SDIO_IO_Transfer.
- 2.2.10
 - Bug Fixes
 - * Fixed SDIO card driver get an incorrect io number when the card io number is bigger than 2.
 - Improvements
 - * Added SDIO 3.0 support.
 - * Added API SDIO_IO_RW_Direct for direct read/write card register access.
- 2.2.9
 - Improvements
 - * Added API SDIO_SetIOIRQHandler/SDIO_HandlePendingIOInterrupt to handle multi io pending IRQ.
- 2.2.8
 - Improvements
 - * Updated sdmmc to support SDIO interrupt.
 - * Added API SDIO_GetPendingInterrupt to get the pending io interrupt.
- 2.2.7
 - Bug Fixes
 - * Fixed MDK 66-D warning.
- 2.2.6
 - Improvements
 - * Added an unify transfer interface for SDIO.
 - Bug Fixes
 - * Fixed Wrong pointer address used by SDMMCHOST_Init.
- 2.1.5

- Improvements
 - * Improved SDIO card init sequence and add retry option for SDIO_SwitchToHighSpeed function.
- 2.1.4
 - Improvements
 - * Added Go_Idle function for SDIO card.
- 2.0.0
 - Initial version.

Secure Element hostlib for KSDK

- For Detailed change log please download the Plug & Trust MW package from https://www.nxp.com/products/:SE050?tab=Design_Tools_Tab and refer to the Changes Present in the User guide.
- 3.01.00
 - Cleanup for heap management macros. Added support to redirect macros to FreeRTOS APIs.
- 3.00.03
 - smCom_Init: return type is now *UI6* instead of *void*. Return value indicates success/failure to create mutex/semaphore.
- 3.00.02
 - Fixed: potential null pointer dereference
- 2.16.00
 - MIMXRT1050 support replaced with MIMXRT1060
 - Examples renamed
- 2.14.00
 - Extensively revamped `fsl_sss_ftr.h` file for finer control of build configuration selection.
 - smCom Layer is refactored so that Application send down the connection handles/parameters to lower layer.
 - Define `T10I2C_UM1225_SE050` is no longer applicable, use `T10I2C_UM11225_SE05X` instead.
- 2.11.03
 - hostlib file/folder moved
- 2.11.0
 - Added Support for SE050 and SSS API's
- 1.6.0
 - Host Library version goes from 01.40 to 01.41 (A71CH Host API has not been extended)
 - Conditional translation of `i2c_Failed` into `i2c_NoAddrAck` removed (this translation is no longer required as a NACK is no longer lumped into `i2c_Failed` by the I2C driver wrapper)
 - Additional update of SCI2C implementation: Ensure multiple consecutive NACK's on address will trigger a return from function '`sci2c_SlaveToMasterDataTx`'.
 - `sci2c_Init` update: initial loop fetching `sci2c` status modified so a SoftReset is issued for all exception types.
 - Solved potential buffer overflow in the implementation of `smAduAppendCmdData`
 - Added additional explicit checks that pointer arguments passed at the `AX_API/A71CH_API`

level are not NULL pointers. In case this check was done through an ‘assert’ statement, the ‘assert’ statement has been removed.

- 1.5.0
 - Added i.MX RT1050 EVKB and LPC54018 IoT module projects
 - Added Watson IoT demo
 - GP Storage Lookup table is no longer restricted to 5 objects, up to 254 objects can be stored (use HLSE_MAX_OBJECTS_IN_TABLE (≥ 8) to limit support and save memory)
 - GP Storage Update Counter can be disabled by defining HLSE_DISABLE_UPDATE_COUNTER (not defined by default).
 - It’s possible to fetch the absolute storage offset of an object
 - Specific error code when attempting to update a locked Lookup table
 - Extended doxygen documentation
- 1.4.2
 - Upgraded rtos\amazon-freertos\lib\pkcs11\portable\nxp\se_hostlib\pkcs11_se.c
 - Added GCP (Google Cloud Platform Demo)
 - Readme updates for AWS JITR Demo
- 1.4.0.1
 - Examples: Fix time stamp printing on embedded platform. (Time stams are supported only for Linux/Windows)
- 1.0.0
 - New integration

Trusted Firmware M (TF-M) for MCUXpresso SDK

The current version is based on TF-M v1.8.0, released 2023-07-31 (42f052298bd0babd3484fad3befd2e77f395654d) on <https://git.trustedfirmware.org/TF-M/trusted-firmware-m.git>

- 1.8.0 Rev1
 - Ported TF-M v1.8.0 to MCUXpresso SDK. Based on the 2023-07-31 snapshot(42f052298bd0babd3484fad3befd2e77f395654d)
- 1.8.0
 - Ported TF-M v1.8.0 to MCUXpresso SDK. Based on the 2023-04-27 snapshot(35ac80c8581d483da7d18c)
- 1.7.0
 - Ported TF-M v1.7.0 to MCUXpresso SDK. Based on the 2023-04-04 snapshot(60c4936a7bb3f56bdb50b0)
- 1.6.0
 - Ported TF-M v1.6.0 to MCUXpresso SDK. Based on the 2022-08-29 snapshot(a5048fb687e4fbf28f9aac69)
 - Added lpcxpresso55s36 platform.
- 1.5.0
 - Ported TF-M v1.5.0 to MCUXpresso SDK. Based on the 2022-03-11 snapshot(47c26ecd0dec177fe2ddf82)
- 1.4.0
 - Ported TF-M v1.4.0 to MCUXpresso SDK. Based on the 2021-08-30 snapshot(00c1106624d733aade4486)
- 1.3.0
 - Ported TF-M v1.3.0 to MCUXpresso SDK. Based on the 2021-04-13 snapshot(cad01ab98c34bb3a13ab49)
 - Added evkmimxrt685 platform.
 - Added evkmimxrt595 platform.

- 1.1
 - Ported TF-M v1.1 to MCUXpresso SDK. Based on the 2020-08-19 snapshot(54507b1645087e92b5a11591)
- 1.0 Rev3
 - Ported TF-M v1.0 to MCUXpresso SDK. Based on the 2020-04-20 snapshot(1e089705899ff68c617cc0c01)
 - Added HUK derivation function.
- 1.0 Rev2
 - Ported TF-M v1.0 to MCUXpresso SDK. Based on the 2020-04-06 snapshot(d4ac5d14e74d1aa2a78526aa)
 - Added the LPC55S Flash module support for ITS and PS.
 - Added lpcxpresso55s16 platform.
- 1.0 Rev1
 - Ported TF-M v1.0-RC1 to MCUXpresso SDK. Based on the 2019-09-23 snapshot(011c0ad0e76d62bd6f2d)
 - Added MCUx and GCC demo applications.
 - Added PSA test suite application.
 - Used mbedCrypto instead of mbedTLS
- 1.0
 - Ported TF-M v1.0-beta to MCUXpresso SDK. Based on the 2019-03-13 snapshot(a5a2a5bc32bc50f4def8d)
 - Added lpcxpresso55s69 platform.
 - Added MDK demo applications.
 - Bug Fixes:
 - * Fixed compilation warnings and errors.
 - * Changes in the TF-M original source code are marked by the "NXP" comment.

USB stack for MCUXpresso SDK

The current version of USB stack is 2.9.1.

- 2.9.1
 - Improvement:
 - * Update EHCI controller driver for basic support of eUSB.
 - * Replace the hard code in audio cases with macro.
 - * Uniform the Chapter9 for device lite cases.
- 2.9.0
 - Improvement:
 - * Change ROOT2 as enabled by default in device stack.
 - * Implement independent frequency adjustment for speaker and recorder of composite audio unified demos.
 - * Fix vulnerability for host stack. CVE number: CVE-2023-38749
 - * Delete deprecated enet driver function for enet adapter.
- 2.8.4
 - Improvement:
 - * Add the new netc adatper for the new netc driver.
 - * Fix issues for USB device dfu and usb device msc when enable the macro USB_DEVICE_CONFIG_RETURN_VALUE_CHECK.
 - * Change the header file including order for usb.h header.

- * Update the USB host audio class driver to fix the wrong output log.
- * Add the workaround on dev_hid_mouse_bm case for the errata TN00071.
- * Enable ROOT2 macro in USB device stack.
- * Use an unified definiton for the base address of RTxxxx platforms.
- 2.8.3
 - Improvement:
 - * Update the EHCI controller driver to support the address convert for TCM.
 - * Update the USB host EHCI controller driver to make sure the mutual exclusion access under multiple tasks' environment.
- 2.8.2
 - Improvement:
 - * Fix noise issue of UAC 3.1, UAC 5.1, UAC 7.1 on usb audio speaker demo.
 - * Fix the issue that incorrect PC behavior when ejecting USB MSC devices.
 - * Update the EHCI controller driver to support RW610 that does not reply on PHY driver, especially for low power feature.
 - * Update the USB_HostHelperParseAlternateSetting to fix the wrong interface parse.
 - * Update dev_composite_hid_audio_unified_bm demo to support independent mute/unmute and volume control.
- 2.8.1
 - Improvement:
 - * update USB audio demos to use audio component (components/audio).
 - * Add the checking of function call return value.
 - * Add audio multiple channels demo (usb_device_composite_audio_multi_ch_unified) on RT600 audio board.
 - * Fix audio noise on sync mode and improve overflow/underflow checking method.
 - * Support UAC 3.1, 5.1 and 7.1 on audio speaker demo.
 - * Set USB device CDC demo not to depend on DTR setting from host.
 - * Support MCUX toolchain on some RTxxxx platforms.
- 2.8.0
 - Improvement:
 - * Fix the USB device stack vulnerability issues.
 - * Update the audio PLL and FRO adjustment codes for audio examples in RTxxx, LP-C54xxx and LPC55xxx.
 - * Improve the USB PD AMS collision avoidance.
 - * Improve IP3511 controller driver's dedicated ram allocation.
 - * Change the USB_DATA_ALIGN_SIZE to 4 because the controller driver uses the dedicated RAM to do memcpy.
 - New features:
 - * Enable USB host audio recorder demo for mutiple boards.
- 2.7.0
 - Improvement:
 - * Use new feedback solution and low latency playback for usb device speaker demo and unified demos. Add underflow and overflow protection.
 - * Optimize hard code for usb audio demos.
 - * Update Unconstrained Power field in the Sink Capabilities Message according to the

- external power state.
- * Fix CVE-2021-38258 and CVE-2021-38260
- New features:
 - * Enable USB host video demo for mutiple boards.
 - * Enable USB device MTP demo for mutiple boards.
 - * Add PPS message to usb pd stack.
- 2.6.1
 - Improvement:
 - * rename sdcard as disk for all of sdcard demos. For ramdisk demos, they are not changed.
 - * add wrapper for all of disk demos to support emmc.
- 2.6.0
 - Improvement:
 - * Added more ufi event to support dynamic sdcard capacity.
 - * Passed MISRA-2012 mandatory and required rules.
 - Except rule 17.2 in host hub and otg stack.
 - Except rule 5.1, rule 5.4, rule 21.1 and rule 21.2.
 - * Re-implemented USB components and supported NPW.
 - * Improved IP3511 controller driver's cancelling transfer function.
 - * Enabled the audio2.0 defaultly for device audio demos.
 - * Enabled the host audio2.0 function in host audio class driver and host audio speaker demo.
 - New features:
 - * enable two USB controllers in one USB host mouse demo which named as host_hid_-mouse_dual.
 - * enable UAC 5.1 for usb device audio speaker demo.
- 2.5.0
 - Improvement:
 - * Integrated sdk components (OSA, Timer, GPIO and serial_manager) to USB stack and demos.
 - * Improved the ip3511 driver throughput.
 - * Improved audio initialization codes after SDK audio drivers update.
 - * Improved auido to support the audio2.0 in win10.
 - * Add one "enumeration fail" callback event to host stack.
- 2.4.2
 - Improvement:
 - * Put the USB controller data and transfer buffer to noncache section, removed the setting that sets the whole ocram and sdram as noncached.
 - * Separated composite audio examples' channel,sample rate,format parameters from commom macro to in dedicated macro and out dedicated macro.
 - * replaced USB_PrepareData with USB_AudioRecorderGetBuffer.
- 2.4.1
 - New features:
 - * Added enumeration fail callback to host stack when the attached device's enumeration failed.
- 2.4.0
 - Improvement:

- * Device Charger Detection (DCD) software architecture was refactored.
- New features:
 - * Enabled Device Charger Detection (DCD) on RT1060.
 - * Enabled Device Charger Detection on RT600.
 - * Enabled host battery charger function on RT600.
- 2.3.0
 - New features:
 - * Added host video camera support. example: usb_host_video_camera
 - * Added a new device example. example: usb_device_composite_cdc_hid_audio_unified
- 2.2.0
 - New features:
 - * Added device DFU support.
 - * Supported OM13790DOCK on LPCXpresso54018.
 - * Added multiple logical unit support in msc class driver, updated usb_device_lba_information_struct_t to support this.
 - * Supported multiple transfers for host ISO on IP3516HS.
 - Bug fixes:
 - * Fixed device ip3511 prime data length than maxpacket size issue.
 - * Initialized interval attribute in usb_device_endpoint_struct_t/usb_device_endpoint_init_struct_t.
 - * Removed unnecessary header file in device CDC class driver, removed unnecessary usb_echo, and added DEBUG macro for necessary usb_echo in device CDC class driver.
 - * Fixed device IP3511HS unfinished interrupt transfer missing issue.
- 2.1.0
 - New features:
 - * Added host RNDIS support. example: lwip_dhcp_usb
 - * Enabled USB 3.0 support on device stack.
 - * Power Delivery feature: Added OM13790HOST support; Added auto policy feature; Printed e-marked cable information;
- 2.0.1
 - Bug fixes:
 - * Fixed some USB issues: Fixed MSC CV test failed in MSC examples.
 - * Changed audio codec interfaces.
- 2.0.0
 - New features:
 - * PTN5110N support.
 - Bug fix:
 - * Added some comments, fixed some minor USB issues.
- 1.9.0
 - New features:
 - * Examples:
 - usb_pd_alt_mode_dp_host
- 1.8.2
 - Updated license.
- 1.8.1

- Bug fix:
 - * Verified some hardware issues, support aruba_flashless.
- 1.8.0
 - New features:
 - * Examples:
 - usb_device_composite_cdc_vcom_cdc_vcom
 - usb_device_composite_hid_audio_unified
 - usb_pd_sink_battery
 - Changed usb_pd_battery to usb_pd_charger_battery.
 - Bug fix:
 - * Code clean up, removed some irrelevant code.
- 1.7.0
 - New features:
 - * USB PD stack support.
 - Examples:
 - * usb_pd
 - * usb_pd_battery
 - * usb_pd_source_charger
- 1.6.3
 - Bug fix: -IP3511_HS driver control transfer sequence issue, enabled 3511 ip cv test.
- 1.6.2
 - New features:
 - * Multi instance support.
- 1.6.1
 - New features:
 - Changed the struct variable address method for device_video_virtual_camera and host_phdc_manager.
- 1.6.0
 - New features:
 - * Supported Device Charger Detect feature on usb_device_hid_mouse.
- 1.5.0
 - New features:
 - * Supported controllers
 - OHCI (Full Speed, Host mode)
 - IP3516 (High Speed, Host mode)
 - IP3511 (High Speed, Device mode)
 - * Examples:
 - usb_lpm_device_hid_mouse
 - usb_lpm_device_hid_mouse_lite
 - usb_lpm_host_hid_mouse
- 1.4.0
 - New features:
 - * Examples:
 - usb_device_hid_mouse/freertos_static
 - usb_suspend_resume_device_hid_mouse_lite

- 1.3.0
 - New features:
 - * Supported roles
 - OTG
 - * Supported classes
 - CDC RNDIS
 - * Examples
 - usb_otg_hid_mouse
 - usb_device_cdc_vnic
 - usb_suspend_resume_device_hid_mouse
 - usb_suspend_resume_host_hid_mouse
- 1.2.0
 - New features:
 - * Supported controllers
 - LPC IP3511 (Full Speed, Device mode)
- 1.1.0
 - Bug fix:
 - * Fixed some issues in USB certification.
 - * Changed VID and Manufacturer string to NXP.
 - New features:
 - * Supported classes
 - Pinter
 - * Examples:
 - usb_device_composite_cdc_msc_sdcard
 - usb_device_printer_virtual_plain_text
 - usb_host_printer_plain_text
- 1.0.1
 - Bug fix:
 - * Improved the efficiency of device audio speaker by changing the transfer mode from interrupt to DMA, thus providing the ability to eliminate the periodic noise.
- 1.0.0
 - New features:
 - * Supported roles
 - Device
 - Host
 - * Supported controllers:
 - KHCI (Full Speed)
 - EHCI (High Speed)
 - * Supported classes:
 - AUDIO
 - CCID
 - CDC
 - HID
 - MSC
 - PHDC

- VIDEO
- * Examples:
 - usb_device_audio_generator
 - usb_device_audio_speaker
 - usb_device_ccid_smart_card
 - usb_device_cdc_vcom
 - usb_device_cdc_vnic
 - usb_device_composite_cdc_msc
 - usb_device_composite_hid_audio
 - usb_device_composite_hid_mouse_hid_keyboard
 - usb_device_hid_generic
 - usb_device_hid_mouse
 - usb_device_msc_ramdisk
 - usb_device_msc_sdcard
 - usb_device_phdc_weightscale
 - usb_device_video_flexio_ov7670
 - usb_device_video_virtual_camera
 - usb_host_audio_speaker
 - usb_host_cdc
 - usb_host_hid_generic
 - usb_host_hid_mouse
 - usb_host_hid_mouse_keyboard
 - usb_host_msd_command
 - usb_host_msd_fatfs
 - usb_host_phdc_manager
 - usb_keyboard2mouse
 - usb_pin_detect_hid_mouse

NXP WiFi

Version 1.3.r47.p16

- Updates:
 - Updated FC's f/w version to 2.p66.155.
 - Updated CA2 and RB3+ f/w version to 21.p124.
 - Fine tuning of GTK rekey offload feature.

Version 1.3.r47.p15

- Bug Fix:
 - DUT fails to connect to Ex-AP configured with wpa2 Enterprise security (Auth method Fast-mschapv2).
 - Incorrect AKM types PSK(2), PSK(SHA-256) are seen in beacon after configuring APUT in wpa2-psk security mode.
 - Link lost seen after wlan-scan when DUT is connected in wpa2 ft-psk security in 2.4Ghz band.

Version 1.3.r47.p12

- Bug Fix:
 - Throughput numbers for TCP-Rx traffic on APUT are dropping to 0Mbps in HE20/VHT20 mode.

Version 1.3.r47.p11

- Bug Fix:
 - Failed to connect STAUT configured in WPA2+PMF required security to Ex-AP configured in WPA2/WPA3 mixed mode+PMF capable security.
 - STAUT is waking up with Broadcast traffic while running MEF AUTO PING ALLOW and WAKE HOST & MEF AUTO PING DISCARD and WAKE HOST.
 - Cannot connect to AP with security wpa3_sb_192_eap_tls.
 - Cannot out band independent reset successfully.

Version 1.3.r47.p10

- Bug Fix:
 - Auto reconnect - link lost reported when ap shut down.
 - STAUT failed to roam from Ex-AP1 to Ex-AP2 with WPA3 security mode while running UDP-Tx traffic.
 - STAUT is not waking up from suspend state while running Host-sleep test even after running Unicast/Broadcast traffic from Ex-AP to STAUT.
 - STAUT is waking up with Unicast data and without unicast data as well before running Broadcast data traffic from AP backend to STAUT.
 - While running WPS connection with Ex-AP, DUT gets deauthenticated after M4 packet, WP-S_NACK is recvd. M1 packet does not have WPS PBC or keypad specific bit set
 - "Network not found" is seen on the STAUT after disconnection and reconnection of the same profile in UNII4 channels (169,173,177).
 - When DUT reported Link lost, After configured time interval, DUT reconnects to AP but fails to ping to AP_BACKEND after reconnection.
 - Coverity Fixes.

Version 1.3.r47.p9

- Bug Fix:
 - APUT is advertising channel width information for 40Mhz in HE-phy capabilities instead of 20Mhz bandwidth in Assoc response frame.
 - STAUT failed to roam from Ex-AP1 to Ex-AP2 with WPA3 security mode with same channel same band and different channel same band scenario.
 - 15-20% Throughput degradation observed in TCP-TX and UDP-TX of HE[40Mhz/80MHz], VHT[40Mhz/80Mhz] compared to previous release[R45.p12].
 - "WLAN: Network not found" is observed while connecting STAUT configured in wpa2 security PMF capable to Ex-AP in wpa2 security with PMF required.
 - Coverity Fixes.

Version 1.3.r47.p8

- Bug Fix:

- "WLAN: network not found & Warn: Scan temporary failure" is observed after disconnecting from one network and Re-connecting to same network after renaming SSID of Ex-AP.
- "Network not found" is seen on the STAUT after disconnection and reconnection of the same profile in UNII4 channels (169,173,177).
- Beacons not stopping in the older channel, after uAP switching the channel according to the Ex-AP in simultaneous mode.
- Incorrect return value on error.

Version 1.3.r47.p7

- Bug Fix:
 - STA STRESS | Independent Reset, DUT went to hang state after 134 iterations of independent reset with status "ASSERT: wlan_process_hang: 982 Assert failed: Panic"!
 - Error message "'is_mef' undeclared(first use in this function)" is observed while the time of compilation of binary after enabling macro for the host-sleep.
 - DUT not able to roam from AP1 to AP2 in different channel (DFS) and same Band on reducing the RSSI of AP1.
 - STAUT failed to roam from Ex-AP1 to Ex-AP2 with WPA3 security mode while running UDP-Tx traffic.
 - MEF, DUT fails to wakeup with mef conditions ping 1,3 and arp 1 & 3.

Version 1.3.r47.p6

- Features
 - Add GTK rekey offload support.
 - Independent Reset via In-band
 - Independent Reset via Out-of-Band
- Updates:
 - Updated FC's f/w version to 2.p66.14.
- Bug Fix:
 - Power save/host sleep support improvements.
 - Messy log output after in band independent reset(17 in 31)
 - Association_Req content corrupted when Dut config as WPA/WPA2 mix mode and connecting to AP WPA+TKIP
 - Cannot connect to AP with security wpa3_sb_192_eap_tls.
 - Command "wlan-add-packet-filter 1" is not available for setting MEF filter configuration in Host-sleep.
 - WPA3 Enterprise support (Host based - Remaining set of authentication methods)
 - STAUT failed to roam from Ex-AP1 to Ex-AP2 with WPA3 security mode while running UDP-Tx traffic.
 - No Link lost is observed while switching from 11n[2.4G | 20Mhz] to legacy[2.4G] mode.
 - The introduction of new features does not follow the scope limitations of modules and apps
 - STAUT 11r, DUT unable to roam to particular BSSID after 10 to 15 iterations of roam, showing "Roaming already in progress Started FT roaming"
 - STAUT | After DUT wake up from Hostsleep condition 0x10 (WAKE_ON_ARP_BROADCAST) Ping failed from AP_Backend to DUT and Delayed ping from DUT to AP_Backend
 - Error: ignoring scan request in invalid state

- Fix COVERITY Issue for midware_wifi

Version 1.3.r47.p5

- Updates:
 - Updated FC's f/w version to 2.p66.11.
- Bug Fix:
 - AKM checks are wrong for UAP PMF MANDATORY WPA2 PSK STA PMF MANDATORY WPA2 PSK for 11AN[20Mhz] as well as BGN[20Mhz] mode.
 - STAUT is not waking up in any scenarios after giving Host-sleep commands with default power save enabled mode.

Version 1.3.r47.p4

- Features
 - Added support for Doxygen.
- Bug Fix:
 - AKM checks are wrong for UAP PMF MANDATORY WPA2 PSK STA PMF MANDATORY WPA2 PSK for 11AN[20Mhz] as well as BGN[20Mhz] mode.
 - On DUT manually configured parameters of 11axcfg, not getting reset to default after "wlan-reset" command execution.
 - 2G to 5G and 5G to 2G FTOA roaming fails , link lost observed with permanent disconnect
 - Change of bandwidth from 80Mhz to 40 MHz is observed after uAP (APUT) is stopped and started again, in HE/VHT band.
 - STAUT unable to scan Ex-AP in UNII4 channels (169,173,177) with any Band/security.
 - "Command response timed out. command 0x107" observed after 945 iterations in Scan--Connect-Disconnect Test without any traffic [Idle Test].
 - RT Console hangs after enabling WLAN/BT Independent Download.
 - Cannot get full iperf summary.
 - While running UDP Bi-directional traffic, QOS data packets are not going from APUT/STAUT.

Version 1.3.r47.p3

- Updates:
 - Updated FC's f/w version to 2.p66.6.
 - Updated CA2 and RB3+ f/w version to 21.221.
- Features
 - Added FW (Parallel) Download support.
- Bug Fix:
 - WiFi hang up with memory alloc buffer error during perform WiFi-scan along with WiFi independent reset loop test(OT already form the NTW)
 - TCP-Keep-Alive packets are not seen in sniffer after successfully setting configuration commands for cloud keep alive and STAUT is not waking up after putting in suspend state.
 - STAUT is not waking up from suspend state while running Host-sleep test even after running Unicast/Broadcast traffic from Ex-AP to STAUT.
 - AKM checks are wrong for UAP PMF MANDATORY WPA2 PSK STA PMF MANDATORY WPA2 PSK for 11AN[20Mhz] as well as BGN[20Mhz] mode.

- Ex-STA (Kestrel, Firecrest RT1060-EVKC) Failed to associate in WPA2 PMF disabled mode to APUT configured in WPA2/WPA3 mixed security mode.
- STAUT, failed to Associate in WPA2 security with PMF disabled to Ex-AP configured in WPA2/WPA3 mixed security mode.

Version 1.3.r47.p2

- Bug Fix:
 - Fail to disable IEEE Power save mode, while giving command to disable it.
 - NXP devices Kestrel/KF2-RD(STA) unable to connect to DUT as MMH-AP but connecting with Mobile, Four way Handshake Timeout is observed with NXP Devices
 - Ex-STA(Kestrel, KF2) Assoc-reject is observed with uAP configured in OWE security mode.
 - Cannot connect to 5g external AP.

Version 1.3.r47.p1

- Updates:
 - Updated CA2 and RB3+ f/w version to 21.p109
 - Updated 8801 f/w version to 36_186
 - Updated Firecrest firmware version to p66.5.
- Features
 - Added EVKC board support for all SoCs.
 - Added WPA2/3 Enterprise support for EAP-SIM/EAP-AKA/EAP-AKA-PRIME for STA and uAP.
 - Added CSI support.
 - Added Auto Reconnect support.
 - Added CA2 and RB3+ Parallel FW Download support.
 - Added CA2 and RB3+ Independent Reset via In-band support.
 - Added CA2 and RB3+ Independent Reset via Out-of-Band support.
 - Added CA2 and RB3+ Boot sleep patch support.

Version 1.3.r46.p7

- Updates:
 - Updated Falcon to p185, CA2 and RB3+ f/w version to 21.p91.5
- Bug Fix:
 - uAP reassociation not working
 - Traffic stops wen moving from auto rate to mcs0
 - DUT not sending TCP-KEEP-ALIVE in suspend state
 - DUT not able to roam from WPA-R1 enabled AP to WPA3-R3 enabled AP
 - Ex-AP1 to Ex-AP2 roaming not working in wpa3
 - STAUT is nto following BSS transition correctly
 - RF test mode commands crash issue
 - Ping is not working when uAP configured with wpa3-sb-192 bit EAP-TLS/EAP-TTLS/EAP--PEAP enterprise security.
 - DUT shows network not found, even when connected to Ex-AP
 - Stress test fixes

Version 1.3.r46.p5

- Updates:
 - Updated CA2 and RB3+ f/w version to 21.p91.5
- Bug Fix:
 - Wifi random crash issue when DUT set to sleep
 - Enable 11D for uAP by default
 - STA doesnt get IP address when ieee-ps and deep sleep are enabled
 - Coverity fixes
 - BT/BLE fix for PTS test case

Version 1.3.r46.p4

- Updates:
 - Updated CA2 and RB3+ f/w version to 21.p91.4
- Bug Fix:
 - Fixed: pre-cert: 20/40 BSS coexistence management is not supported in association request.
 - Fixed: pre-cert: STAUT is not correctly following the MU EDCA parameters advertised by the AP.
 - Fixed: Fail to connect to external AP with security wpa3.
 - Fixed: uAP with wpa3 security is not connectable.
 - Fixed: DUT not able to connect to Ex-AP in WPA2 security when PMF required is set.
 - Fixed: Remove BAND_B rates from Supported rates for 5G channels during scan.
 - Fixed: pre-cert STAUT is not including operating class 81 and 115 under Alternate Operating Classes in assoc request.
 - Fixed: Cannot connect to external AP successfully.
 - Fixed: 5Ghz channel are configured acceptable in 2.4Ghz band
 - Fixed: wlan_set_rf_tx_power is not consistent with tx-frame power output
 - Fixed: Getting compilation errors while compiling the binary after defining "CONFIG_UNI-I4_BAND_SUPPORT" macro in wifi_config.h file.
 - Fixed: uAP Beacons advertise 3 AKM suites (PSK & PSK SHA256 & SAE SHA256) in RSNIE when WPA2+WPA3 mixed mode is set on AP with mfpc 1 and mfpr 0.
 - Fixed: STAUT configured with WPA3-SAE associates to WPA2 configured AP.
 - Fixed: DUT unable to roam from Ex-AP1 to Ex-AP2 on lowering the RSSI of Ex-AP1.
 - Fixed: pre-cert STAUT is not correctly following the MU EDCA parameters advertised by the AP.
 - Fixed: DUT-STA is unable to associate with WPA/WPA2 Mixed mode AP when DUT-STA is configure WPA security.
 - Fixed: pre-cert STAUT fails to roam from WPA3 configured AP1 to WPA2-PSK configured AP2.
 - Removed IEEE_MGMT_ACTION_CATEGORY_PUBLIC enum.
 - Fixed logic for consecutive connect for both embedded and wpa supplicant.
 - Fixed all compilation errors and warnings seen on dapeng.
 - Resolved MISRA and coverity defects.

Version 1.3.r46.p3

- Updates:
 - Updated CA2 and RB3+ f/w version to 21.p91.2

- Bug Fix:
 - Fixed: TCP and UDP TX traffic stream not working with ex-sta
 - Fixed: pre-cert: STAUT is not following BSS transition correctly
 - Fixed: Extend wlan-get-antcfg to print current antenna
 - Fixed: DUT able to configure MFPR 0 in WPA3 security.
 - Fixed: DUT MCS rates are not updating to default after stopping/removing previous profile with MCS set to fixed number via fixed rate command.
 - Fixed: DUT-uAP unable to start with ACS configuration.
 - Fixed: While running WPS connection with Ex-AP, DUT gets deauthenticated after M4 packet, WPS_NACK is recvd. M1 packet does not have WPS PBC or keypad specific bit set.
 - Fixed: Fail to create ipv6 iperf client.
 - Fixed: uAP beacons contain incorrect bit, UAPSD bit is set in WME QoS Info even though it is not supported.
 - Fixed: uAP beacons contain incorrect bits, SU/MU Beamformer bits are set in HE Phy capabilities and UAPSD bit is set in WME QoS Info even though it is not supported.
 - Fixed: Data path blocks once we initiate wlan-scan on DUT and change Ex-AP channel at the same time.
 - Fixed: MISRA defects.

Version 1.3.r46.p2

- Updates:
 - Updated CA2 and RB3+ f/w version to 21.p91
- Bug Fix:
 - Fixed BT connection issue on RB3+

Version 1.3.r46.p1

- Updates:
 - Updated CA2 and RB3+ f/w version to 21.p90
 - Updated 8801 f/w version to 36_181
 - CA2 Fixed FIPS GCMP support
 - Added all changes as part of r45.p3 as applicable for CA2 and RB3+
 - Added information for BSD3 license.
 - Updated WLCMGR and Wi-Fi driver to support WPA supplicant and hostapd.
 - Deep Sleep feature in BLE peripheral role
 - Deep Sleep and wakeup feature is enabled for CA2 and RB3+
- Features
 - Added WPA supplicant and hostapd support.
 - Added WPS 2.0 support for STA and uAP.
 - Added WPA2 enterprise support for STA and uAP.
 - Added WPA3 enterprise support with suite b and suite b 192 bit mode for STA and uAP.
- Bug Fixes:
 - Fixed: APUT goes into hang state after every disassociation of STA.
 - Fixed: (pre-cert)AMSDU Rx test fails as STAUT returns the throughput received as 0.
 - Fixed: Country Information IE with default country code (WW) is seen in beacon even after

- changing the regions from WW to US/EU/CA/CN.
- Fixed: DUT should show "Network not found" message in cli, when Configure SSID is not present in network area.
 - Fixed: DUT not able to connect to Ex-AP in WPA3R3 security.
 - Fixed: DUT not able to roam from Ex-AP1 to Ex-AP2.
 - Fixed: Hostsleep/MEF test condition failures are seen.
 - Fixed: uAP not coming up in WPA2/WPA3 mixed mode security., getting "WPA initialization failed" on DUT console.
 - Fixed: DUT shows wrong channel number in wlan-scan-opt and wlan-info command results.
 - Fixed: DUT sending QoS data packets with LDPC coding when LDPC coding is disabled on Ex-STA.
 - Fixed: Country Information IE is not seen in uAP Beacon.
 - Fixed: DUT is going into hang state while running Connect/Disconnect Stress Test
 - Fixed: Supported MCS and NSS rates are not properly set under HE capabilities IE in beacon frame when CAPA enabled binary is flashed on DUT.
 - Fixed: Channel Bandwidth is setting to 80Mhz when uAP is configured in 11AX-40Mhz/20-Mhz.
 - Fixed: Beacon frame includes HE IE when uAP is configured in 11AC using Capa command.
 - Fixed: DUT is not connecting in WPA2/WPA3 Enterprise security.
 - Fixed: DUT is not getting ipv4 address after Roaming to Ex-AP2.
 - Fixed: DUT is not roaming from AP1 to Ex-AP2 after running wlan-ft-roam command in 5Ghz.
 - Fixed: DUT is not following BSS transition Correctly.
 - Fixed: DUT is not responding Beacon Report Action frame after sending the Beacon request from Ex-AP to DUT.
 - Fixed: Automatic Channel Selection is not working when DUT configured in MMH mode.
 - Fixed: DUT is going for Reassociation with the same Ex-AP after running "wlan-host-11k-neighbour-req" command.
 - Fixed: uAP Beacons does not include SHA256 in RSNIE when PMF mandatory is set on AP in WPA2-PSK security.
 - Fixed: Low Throughput seen in 11AX, 11AC, 11N (80MHz, 40MHz & 20MHz) TCP-RX/TX, UDP-TX/RX in 2.4Ghz & 5Ghz.
 - Fixed: DUT is not using "FT using PSK" AKM suite when 802.11R is enabled.
 - Fixed: DUT not able to connect to Ex-AP in WPA2 security when PMF required is set.
 - Fixed: Need to reset DUT every time for every new connection to happen.
 - Fixed: Incorrect VHT IE "RX/TX MCS Map" is seen in uAP beacons.
 - Fixed: uAP Beacons is advertising 4 Pairwise Cipher Suites [CCMP (256), GCMP (256), AES (CCM), GCMP (128)] in RSNIE with WPA2/WPA3 security.
 - Fixed: uAP not coming up in 2.4GHz channel 11, getting "uAP start failed, giving up" on DUT console.
 - Fixed: DUT is going in hang state after Stop/Remove/Start network in wpa2 and wpa3 security, when configured in MMH mode.
 - Fixed: HE IE is not reflecting in beacons, when DUT is configured in default mode without capa in MMH.
 - Fixed: DUT is not connecting in WPA3 security, shows "Error: Init of random number

generator failed."

Version 1.3.r45.p12

- Updates:
 - Updated tx pwr limit files for murata 2el module.
 - Added config macro for RU Tx power.
 - Updated WPA2/3 Enterprise support to handle pre-cert tests.
- Bug Fixes:
 - Fixed: uAP not starting up in channel 14 showing start failed when uAP country code is set to JP.

Version 1.3.r45.p11

- Updates:
 - Added integrate tx pwr limit files for murata 2el module.
 - Added support for legacy mode of RU Tx power.
 - Updated WPA2/3 Enterprise support to handle pre-cert tests.
- Bug Fixes:
 - Fixed: Bi-directional traffic converts to uni-directional traffic.
 - Fixed: The support for "wlan-multi-mef" command to configure MEF parameters on cli is not available.
 - Fixed: Country Information IE with default country code (WW) is seen in uAP beacon even after changing the regions from WW to US/EU/CA/CN.
 - Fixed: Failed to wakeup card after turned on IEEE Power Save mode.
 - Fixed: Bi-directional iperf traffic is not running getting dropped to 0.000 bits/sec.
 - Fixed: Macbook not connecting with DUT-AP in channel 48 and BW 80MHz.
 - Fixed: [pre-cert] DUT fails to connect to Ex-AP configured with wpa/wpa2 enterprise security (AES Encryption) and the radius server configured with hostapd, unsupported certificate error.
 - Fixed: Not connecting to AP in WPA-TKIP and WPA-AES, M2 is not being initiated by STA.

Version 1.3.r45.p10

- Updates:
 - Updated firmware version to 2.p7.19 and added TP signed FW(with VDLL) too.
 - Changed the init and command flow for uart_wifi_bridge app.
- Bug Fixes:
 - Fixed: [pre-cert] DUT not correctly receiving AMPDU+AMSDU , AMSDU bit is not set to 1 in QOS data frame.
 - Fixed: uAP not starting up in channel 14 showing start failed when uAP country code is set to JP.
 - Fixed: Data path blocks once we initiate wlan-scan on DUT and change Ex-AP channel at the same time.
 - Fixed: App will block/output error log when trying to set rf tx frame.

Version 1.3.r45.p9

- Updates:
 - Updated firmware version to 2.p7.17 and added TP signed FW(with VDLL) too.

- Added support for channel based RU Tx power.
- Added support of reassociate command on STAUT to test reassociate feature.
- Bug Fixes:
 - Fixed: Ex-STA(Kestrel) not able to connect to uAP with "reassociate" command in first attempt, getting deauthentication and again connection is initiated with Association Request and connection happens, in wpa3 security.
 - Fixed: [pre-cert] STAUT is sending incomplete beacon report response to AP's beacon report request.
 - Fixed: Messy log output when trying to turn off deep sleep mode.
 - Fixed: Unable to set TX-OMI on uAP using command "wlan-set-tx-omi".
 - Fixed: DUT is not re-connecting when bandwidth/mode changed on Ex-AP.

Version 1.3.r45.p7

- Updates:
 - Updated firmware version to 2.p7.15 and added TP signed FW(with VDLL) too.
- Bug Fixes:
 - Fixed: STAUT is not following BSS transition correctly.
 - Fixed: DUT is not sending TCP-KEEP-ALIVE packets in suspend state, when TCP connection is established via Cloud keep alive command.
 - Fixed: DUT not able to roam from Ex-AP1 to Ex-AP2 on lowering the RSSI of Ex-AP1 in wpa3 security.
 - Fixed: STAUT's probe request does not contain MBO-OCE IE
 - Fixed: Ping is not working when uAP configured with wpa3-sb-192 bit EAP-TLS/EAP-TTLS/EAP-PEAP enterprise security.
 - Fixed: STAUT's probe request does not contain MBO-OCE IE.
 - Fixed: Throughput enhancement for STA and uAP mode for various TCP/UDP and Tx-Rx modes for embedded and wpa supplicant.
 - Fixed: "TCP_ABORTED_LOCAL" message seen on DUT console instead of "TCP_DONE" when running iperf traffic
 - Fixed: [pre-cert] STAUT is not dropping TP when AP is increasing MPDU spacing factor from 0 to 3.

Version 1.3.r45.p6

- Updates:
 - Updated firmware version to 2.p7.11 and added TP signed FW(with VDLL) too.
 - uart_wifi_bridge added and errors fixed for RT1170-EVKB.
- Bug Fixes:
 - Fixed: Traffic stops after moving from auto rate to fixed mcs0. Command is triggered on the fly.
 - Fixed: DUT not able to roam from WPA3R1 enabled AP to WPA3R3 enabled AP showing network not found.

Version 1.3.r45.p5

- Updates:
 - Updated firmware version to 2.p7.10 and added TP signed FW(with VDLL) too.

- Bug Fixes:
 - Fixed: Data path blocks once we initiate wlan-scan on DUT and change Ex-AP channel at the same time.
 - Fixed: uAP beacons contain incorrect bits, SU/MU Beamformer bits are set in HE Phy capabilities and UAPSD bit is set in WME QoS Info even though it is not supported.
 - Fixed: uAP beacons contain incorrect bit, UAPSD bit is set in WME QoS Info even though it is not supported.
 - Fixed: [pre-cert] STAUT is not correctly following the MU EDCA parameters advertised by the AP
 - Fixed: [pre-cert] 20/40 BSS coexistence management is not supported in association request.
 - Fixed: Fail to connect to external AP with security wpa3.
 - Fixed: uAP with wpa3 security is not connectable.
 - Fixed: DUT not able to connect to Ex-AP in WPA2 security when PMF required is set.
 - Fixed: Remove BAND_B rates from Supported rates for 5G channels during scan.
 - Fixed: Fix wifi_cli_prov example hang when connect if enable ieee power save and deep sleep.
 - Fixed: RFTM: 5Ghz channel are configured acceptable in 2.4Ghz band.
 - Fixed: RFTM: wlan_set_rf_tx_power is not consistent with tx-frame power output.
 - Fixed: DUT not able to roam from AP1 to AP2 in different channel (DFS) and same Band on reducing the RSSI of AP1.
 - Fixed: uAP Beacons advertise 3 AKM suites (PSK & PSK SHA256 & SAE SHA256) in RSNIE when WPA2+WPA3 mixed mode is set on AP with mfpc 1 and mfpr 0.
 - Fixed: DUT unable to roam from Ex-AP1 to Ex-AP2 on lowering the RSSI of Ex-AP1.
 - Fixed: Fixed build errors and warnings for matter, MISRA, coverity and other compilers.

Version 1.3.r45.p4

- Updates:
 - Updated firmware version to 2.p7.4 and added TP signed FW(with VDLL) too.
- Bug Fixes:
 - Fixed: Fail to create ipv6 iperf client.
 - Fixed: DUT MCS rates are not updating to default after stopping/removing previous profile with MCS set to fixed number via fixed rate command.
 - Fixed: MISRA Defect fixed.
 - Fixed: IEEE PS event not getting triggered.
 - Fixed: While running WPS connection with Ex-AP, DUT gets deauthenticated after M4 packet, WPS_NACK is recvd. M1 packet does not have WPS PBC or keypad specific bit set.
 - Fixed: Fail to create ipv6 iperf client.
 - Fixed: Hang issue seen when assoc reject event received.

Version 1.3.r45.p3

- Updates:
 - Enabled h/w acceleration APIs via mbedtls.
 - VDLL support added.
- Bug Fixes:
 - Fixed: (pre-cert)Need support to set UL MU Disable/Data Disable element through TX-OMI

command.

- Fixed: iTCP and UDP TX traffic stream not working with ex-sta.
- Fixed: STA not able to connect to APUT when APUT is configured in OWE security.
- Fixed: DUT is going on hang state after running "wlan-reset" command.
- Fixed: "TCP_ABORTED_REMOTE" message seen on DUT console when running UDP traffic.
- Fixed: (pre-cert)STAUT is not scanning Non Transmitted SSID in MBSSID test.
- Fixed: 6+ ms of traffic burst is seen which is failing cert criteria of max cot as 6ms.
- Fixed: DUT not getting IP address(DHCP) when "WMM_ENH" Macro is defined in "wifi_config.h" file for TWT feature.
- Fixed: For Tx Frame & tx-continuous, the Measured tx-power value is 2dBm irrespective of configured Tx-power(10/12/15/20dBm)
- Fixed: Data path blocks once we initiate wlan-scan on DUT and change Ex-AP channel at the same time.
- Fixed: DUT not able to scan and connect to DFS channels when Broadcast SSID is disabled.
- Fixed: Aggregation is broken when TCP/UDP TX Data is going with TID 6-7 (Voice), and TID 3 (Best Effort).
- Fixed: uAP not beaconing in 40MHz when uAP configured in channel 12/13-40MHz in MMH mode
- Fixed: STA Ping is not working once it roams to Ex-AP2.
- Fixed: Wi-Fi lwip port does not protect correctly concurrent accesses to lwip stack.
- Fixed: Beacon frame does not include RSN XE IE when uAP configured in WPA3R3 security.
- Fixed: Beacon frame includes HT IE when uAP configured in 11A mode using capa legacy command.
- Fixed: DUT not connecting to Ex-AP when PMF is configured using "wlan-set-pmfcfg" command.
- Resolved IAR, MCUX and MDK build errors.

Version 1.3.r45.p2

- Updates:
 - Updated firmware version to 2.p7.1 and added TP signed FW too.
 - Added RFTM commands missing CLI commands related to OFDMA feature.
 - Added support for EVKB board.
- Bug Fixes:
 - Fixed: uAP not coming up in channel 14 when country code is set to JP.
 - Fixed: DUT not able to connect with 32-character SSID when Ex-AP configured in open/wpa2/wpa3 security.
 - Fixed: QoS Data packets not seen on air when we run "wlan-set-tx-omi 0x48 0xff 16" command on DUT.
 - Fixed: [pre-cert] Not able to set non-preferred channel in MBO through wpa_cli on RTOS.
 - Fixed: Automatic Channel Selection is not working when DUT configured in MMH mode.
 - Fixed: STA not able to connect to APUT when APUT is configured in OWE security.
 - Fixed: Wi-Fi lwip port does not protect correctly concurrent accesses to lwip stack.
 - Fixed: DUT is going on hang state after running "wlan-reset" command.
 - Resolved IAR, MCUX and MDK build errors.

Version 1.3.r43.p9

- Updates:
 - Added support for RF test mode.
- Bug Fixes:
 - Fixed: (pre-cert)Updated help message for TX-OMI command.
 - Fixed: (pre-cert)STAUT is not scanning Non Transmitted SSID in MBSSID test.
 - Fixed: (pre-cert)STAUT is not governing OBSS Narrow Bandwidth RU in UL OFDMA Tolerance Support.
 - Fixed compilation errors and warnings after branch merge.

Version 1.3.r44.p3

- Bug fixes:
 - Added RSNX IE for WPA3-R3 for uAP

Version 1.3.r44.p2

- Updates:
 - Updated CA2, RB3+ firmware to p82
 - Bug fixes:
 - * Fixes for 11KR
 - * Fixes for WPA3-R3 STA
 - * Added extra event for Connected notification in case of uAP

Version 1.3.r44.p1

- Updates:
 - Updated CA2, RB3+ firmware to p79
 - Updated 8801 firmware to p180
- Features:
 - Added 11KVR support for CA2 and RB3+
- Bug fixes:
 - Fixed: Association Request does not include Group Management Cipher Suite IE in RSNIE when PMF mandatory is set on STA in WPA2-PSK/WPA3/OWE security.
 - Fixed: Sending of deauth codes in all scenarios
 - Fixed: Command timeout issues for wlan-eu-crypto
 - Added Coverity fixes
 - Fixed command timeout seen during simultaneous AP + STA mode, with high traffic on 8801
 - Set proper value for Edmac value updated to support Murata 2DS Module
 - Fixed Channel switch announcement is not seen in beacon in AP+STA mode, when AP changes channel on 8801
 - Added few BT Sig qualification fixes

Version 1.3.r43.p8

- Updates:
 - Updated firmware version to p182.1 and added TP signed secure FW too.
- Bug Fixes:
 - Fixed: (pre-cert)APUT beacons does not have RSNXE when configured in H2E mode.

Version 1.3.r43.p7

- Updates:
 - Updated firmware version to p182.
 - Added TP signed secure FW.
 - Added support for 2EL M2 module for Firecrest.
 - Added wifi capability configuration support.
 - Added FIPS validation feature.
 - Added wifi rf test mode example.
 - Unified all wifi examples in wifi_cli.
 - 11R is not supported for FC.
- Bug Fixes:
 - Modified the method to load RU tx power limit with RT.
 - Modified txrate config design for 11AC and 11AX features.
 - Added support for roaming and neighbor request processing.
 - Fixed: "Error in sending Background traffic" messages seen on DUT while running Background TX-Traffic.
 - Fixed: STAUT is not associating in WPA3 Hash-to-Element only mode.
 - Fixed: Not able to change HE MAC Capabilities or HE PHY Capabilities in Association Request using 11axcfg command.
 - Fixed: DUT not waking up from suspend state when we start multicast traffic after 30sec-1 minute.
 - Fixed: STAUT hangs after scanning AP in scenario K of SI-5.2.2 test case.
 - Fixed: STAUT is not including operating class 81 and 115 under Alternate Operating Classes in assoc request.
 - Fixed: STAUT fails to roam from WPA3 configured AP1 to WPA2-PSK configured AP2.
 - Fixed: DUT goes for DHCP DORA Process after roaming to AP2 from AP1.
 - Fixed build errors and warnings.
 - Corrected description related to num_dat_pkts parameter of tx-omi command.

Version 1.3.r43.p6

- Updates:
 - Updated firmware version to p174 and added latest signed FW bins as well.
- Bug Fixes:
 - Fixed: Command timeout issues observed for various offload commands.
 - Fixed: TWT requester bit not set in Extended Capabilities IE (127) in Association Request and Probe Request of STAUT.
 - Fixed: Not able to scan and connect Firecrest-STA to Firecrest-uAP configured in UNII-4 band channels 173 and 177.
 - Fixed: Multiple error messages are seen on DUT while running the stress "RX ping on STA, Start-stop uAP" in loop.
 - Added 11AX macro so that nss settings can be available for 11AC as well as 11AX based configurations.
 - Fixed various warning for IAR, MDK, armgcc and mcuXpresso compilers.

Version 1.3.r43.p5

- Updates:
 - Updated firmware version to p168(however signed firmware is not updated in this release and has version p164).
- Bug Fixes:
 - Fixed: DUT not able to connect to AP configured in WPA2/WPA3 mixed mode security.
 - Fixed: Association Request does not include Group Management Cipher Suite IE in RSNIE when PMF mandatory is set on STA in WPA2-PSK/WPA3/OWE security.
 - Fixed: uAP Beacons does not include SHA256 in RSNIE when PMF mandatory is set on AP in WPA2-PSK security.
 - Fixed: Device getting hanged while setting txpwrlimit by CMD 'wlan-set-txpwrlimit'.
 - Fixed: Not able to load RU tx power limit with RT.
 - Fixed: Not able to change HE parameters in HE MAC Capabilities or HE PHY Capabilities using 11axcfg command.

Version 1.3.r43.p4

- Updates:
 - None.
- Bug Fixes:
 - Updated helper message for owe_only cli.
 - Fixed: Observed bi-directional data traffic on air while running uni-directional TCP-RX traffic.
 - Fixed: UDP-TX traffic going without using aggregation.
 - Throughput values will now be displayed on DUT while running iperf Tx and Rx traffic.

Version 1.3.r43.p3

- Updates:
 - Updated firmware version to p164.
- Bug Fixes:
 - Added ed mac config support for uAP.
 - Modified process_rsn_ie function on driver side to provide support for Group Management Cipher suite.
 - Added conditional logic related to handling of region code.

Version 1.3.r43.p2

- Bug Fixes:
 - Added delay in uart_wifi_bridge application to correctly read calibration data for RB3P board.

Version 1.3.r43.p1

- Updates:
 - Updated firmware version to p162.
 - Added MBO feature support.
 - Added OWE feature support.
- Bug Fixes:
 - uAP Beacons contains MFPC & MFPR bits set to 1 in WPA2-PSK security after removing the previous WPA3 security profile.
 - Added new cli parameters for wlan-set-tx-omi command to send OMI using QoS Null Packet

or QoS Data Packet according to input provided.

- Corrected 11R configuration macro in allMacros_iw61x.txt file and removed the same from ignoreMacros_iw61x.txt file.
- Coverity fix: Changed all WM_FAIL to -WM_FAIL.

Version 1.3.r42.p4 -New Additions:

- CA2 and RB3+ f/w updated to v16_xx.21.p64.1

Version 1.3.r42.p3 -New Additions:

- 8801 f/w update -Bug Fixes:
- Fixed Misra issues.

Version 1.3.r42.p2 -Bug Fixes:

- Fixed Misra issues.

Version 1.3.r42.p1 -New Additions:

- RTOS abstraction improvement
- Added support for FIPS for CA2 and RB3+
- Added uAP 11AC support
- Deprecated support for 88W8977 Wi-Fi SoC
- Added 8978 SoC firmware p64.
- Added 8987 SoC firmware p64.
- Added mlanutl equivalent utility on RT to create wifi fw compatible command arrays.

-Bug Fixes:

- Fixed Misra issues.

Version 1.3.r41.p2 -New Additions:

- Updated license header as a github friendly license.
- Added 8801 SoC firmware p177 with get Coex Statistics support.
- Added 8987 SoC firmware p32.2 with Bluetooth related fixes. Added fix for automatic ble disconnect issue.

-Bug Fixes:

- Fixed Unable to start/stop 11n rx reorder timer(50%).
- Fixed Getting Improper Channel Number in "wlan-get-uap-channel" command.
- Fixed DUT fails to start DHCP intermittently.
- Fixed wrong wlan-list output.
- Fixed station connect issue after host sleep is enabled and disabled.
- Fixed Misra and coverity issues.
- Fixed automatic ble disconnect issue after 30 seconds of connection establishment.

Version 1.3.r41.p1 -Bug Fixes:

- Updated 8801 SoC firmware to toggle GRANT Pin with WLAN/BT time on HIGH Request with Low Priority.
- Added coverity fixes.

Version 1.3.r40.p5 -Bug Fixes:

- Fixed issue of DUT not entering Powersave mode on all SoCs.

Version 1.3.r40.p4 -Bug Fixes:

- Default bandwidth set to 20 MHz for uAP on 8977.
- Fixed an issue in static IPv4 address assignment.
- Fixed warning and errors for IAR, MDK and gn + ninja + armgcc toolchains.

Version 1.3.r40.p3 -Bug Fixes:

- Added wlan_uap_set_httxcf API to set 40 MHz support in 2.4 GHz.
- Fix for build issue seen for wifi_setup 1020

Version 1.3.r40.p2 -Bug Fixes:

- Fixed IPv6 address and state updates as per the networking stack configuration.
- Removed errors shown in channel validation and in setting custom CFP tables
- Added API to support selection of 20 and 40 MHz bandwidth
- Added -b(bandwidth) option in wifi_cli iperf command
- DHCP client doesn't report the failure of obtaining IP address
- Remove dependency for wifi_config_internal.h
- Remove warnings reported for channel list variables
- Fix STA not sending data in 40M bandwidth for BGN 40 in 2.4G

Version 1.3.r38.p2

- New Additions:
 - Added Support for IPv6.
 - Added support for Hostsleep and packet filters.
 - Updated Firecrest Firmware version to p50.5.
 - Updated CA2 Firmware version to p21.22.

-Bug Fixes:

- Fixed an issue where DUT not able to start UAP Network in 2G MMH Mode.
- Fixed an issue where Command 0xb2 timeout is seen when stopped uAP while running DL traffic.

Version 1.3.r37.p4

- New Additions:
 - Added support for new API for HostCmd in RT Platform.
 - Added provision to set tx rates for HE mode in wlan_set_txratecfg API.
 - Added support for new command to set tx OMI.
 - Added WPA3 R3 support for 8801, RB3, RB3+ and CA2.
 - Updated Firecrest firmware version to p50.2.
 - Updated CA2 firmware version to p235.2
 - Updated RB3+ firmware version to p11.3.
 - Updated 8801 firmware version to p191.2
 - Updated RB3 firmware version to p186.2.
- Bug Fixes:

- Fixed an issue where default netif was not set to STA after closing uAP.
- Fixed an issue where 30% low RX throughput was seen with 11AN 40Mhz.
- Fixed wlan_start and wlan_stop API working.
- Fixed High ping latency when DUT is put in IEEEPS mode.
- Fixed an issue where wlan_get_dtim_period API was not returning any value.
- Fixed SVD vulnerability issue on RB3+.
- Fixed MISRA/Coverity issues.
- Fixed SVD vulnerability issue on 8977, 8801 and CA2.
- Fixed an issue in Firecrest where STAUT is disconnecting immediately after ieeeps command is fired.
- Fixed an issue where UDP traffic was not working on uAP mode.
- Country code not being displayed in 11d is being fixed

Version 1.3.r35.p2

- New Additions:
 - Added support for 11ax for Firecrest.
 - Updated Firecrest firmware version to p27.
- Bug Fixes:
 - Fixed an issue for CA2 where tcp/udp Rx traffic was not seen with Linksys AP.
 - Fixed tx power limit issue for Firecrest.
 - Fixed an issue for Firecrest where Ping stops working after TCP traffic is started from DUT.
 - Changed WLAN_PSK_MAX_LENGTH to 64 from 65.
 - Fixed an issue where mfpc was not set properly.

Version 1.3.r34.p2

- New Additions:
 - Updated CA2(8987) SoC firmware to p235.1.
- Bug Fixes:
 - Fixed WPA3 Authentication failure for CA2.
 - Fixed SPP connection issue.

version 1.3.r34.p1

- New Additions:
 - Updated CA2(8987) SoC firmware to p235.
 - Added Separation of fw download from Wi-Fi initialization.
 - Added support for Multicast group creation
 - Updated license content in the wlan src for Murata or Generic customer.
 - Added support for new firmware version display.
 - Added config Macro for disabling Rx SDIO aggregation. This is enabled by default.
- Bug Fixes:
 - Fixed CH 144 connection issue.

version 1.3.r33.p2

- New Additions:
 - Updated RB3+(8978) SoC firmware to p214.

- Updated CA2(8987) SoC firmware to p200.
- Bug Fixes:
 - Fixed an issue where, wifi connection was failing during BLE activity.
 - Fixed an issue where a2dp profile app could not find a2dp sink device.

version 1.3.r32.p5

- New Additions:
 - Updated RB3+(8978) SoC firmware to p198.
 - Updated CA2(8987) SoC firmware to p199.
- Bug Fixes:
 - Fixed an issue where BT pairing auth failure was observed between two 8987 or two IW416 devices due to same DH Public Key being generated.

version 1.3.r32.p4

- New Additions:
 - Updated RB3+(8978) SoC firmware to p197.
- Bug Fixes:
 - Fixed an issue where, in presence of WLAN, BT A2DP SNK scenario can observe continuous glitches because of BT utilizing the WLAN overlapping channels.
 - Fixed an issue where, in presence of BT A2DP SRC scenario and WLAN traffic, both will share air-time. WLAN Rx TP can drop to 15-20% of its baseline because of A2DP occupying more air-time duty cycle.

version 1.3.r32.p3

- New Additions:
 - Updated RB3+(8978) SoC firmware to p196.
 - Updated CA2(8987) SoC firmware to p162.
 - Updated RB3(8977) SoC firmware to p186.
 - Updated Falcon(8801) SoC firmware to p191.
- Bug Fixes:
 - Fixed an issue where STAUT was not advertising extended capabilities in assoc request
 - Fixed an issue where cal data download API is not setting the cal data correctly.
 - Fixed an issue where Ping of 10000 bytes is not working for WMM case with 11n/11ac pre-cert WFA testbed AP's.
 - VU FFD(vulnerability) fixes have been made for CA2(8987)/RB3+(8978)/RB3(8977)/Falcon(8801).

version 1.3.r31.p1

- New Additions:
 - Updated RB3+(8978) SoC firmware to p152.
 - Updated CA2(8987) SoC firmware to p152.
 - Added WMM feature for 8987 SoC -Updates
 - Enabled wifi deep sleep and IEEE802.11b/g/n modes as part of CLI initialisation.
 - Updated module macro for CA2 from WIFI_BOARD_AW_CM358MA to WIFI_BOARD_AW_CM358.

- Bug Fixes:
 - Fixed deepsleep error when called immediately after disconnection
 - Fixed an issue where uAP was not turned on when country is specified using wlan_set_country API.

version 1.3.r30.p2

- New Additions:
 - Updated 8978 SoC firmware to p185.
 - Updated 8987 SoC firmware to p185.
- Bug Fixes:
 - Fixed a regression issue in WiFi FW where Ex-client is not able to associate with uAP on 8978 and 8987 SoC.

version 1.3.r30.p1

- New Additions:
 - Updated RB3+(8978) SoC firmware to p145.
 - Updated CA2(8987) SoC firmware to p145.
- Bug Fixes:
 - Fixed MISRA C-2012 Rule 14.4 issues.
 - Updated TX Power configuration table for uAP.
 - Fixed an issue where in MFG mode measured TX power value for 2.4GHz is 10.97dBm and for 5GHz is 12.03dBm always irrespective of TX-power values configured in CA2.
 - Fixed an issue where in control frames measured TX power is less than configured Tx power by 3dBm for 5Ghz 40Mhz BW and ~7dBm for 5Ghz 80Mhz BW in CA2.

version 1.3.r29.p2

- New Additions:
 - Updated 8987 SoC firmware to p142.
 - Updated 8978 SoC firmware to p142.
- Bug Fixes:
 - Fixed MISRA C-2012 Directive 4.7, Directive 4.10, Rule 15.7 and Rule 14.4 issues
 - Fixed wlan_get_sta_tx_power() API implementation for retrieving station tx power level.
 - Fixed wlan-set/get-txpwrlimit CLI command for setting/getting TX power limit for 11AC modulation groups in 8987.
 - Fixed an issue where in MFG mode(rf_test_mode=1) after disabling wlan_set_rf_cont_mode(CMD18_CW=1) command response timeout is seen in 8987.

version 1.3.r29.p1

- New Additions:
 - Added support for 8987 SoC with p141 firmware version.
 - Updated 8978 SoC firmware to p141.
- Bug Fixes:
 - Fixed Misra C-2012 required category issues for Rule 17.7.
 - Fixes for BCA-TDM in Co-Ex for 8978 SoC.
 - Fixed A2DP glitches in BT when WLAN connected for 8978 SoC.

version 1.3.r27.p2

- New Additions:
 - Updated FW versions to p130 for 8978.

version 1.3.r27.p1

- Bug Fixes:
 - Fixed PMF pre-cert issue where STAUT is not associating to PSK-SHA-256 enabled AP.
 - Fixed 11N pre-cert issue where 11N-5.2.47 STAUT AMPDU TX test case is failing.

version 1.3.r26.p2

- New Additions:
 - Updated FW versions to p184 for 8977 and p122 FW for 8978.
 - Added wifi_cert application under wifi_examples.
- Bug Fixes:
 - Fixed an issue where connection problem is seen with uAP in wifi_webconfig after removing stored credentials.
 - Fixed RF Test Mode issue for setting data rate in uAP mode.
 - Fixed Coverity and MISRA issues in WiFi Driver.
 - Fixed WPA3 SAE pre-cert requirement where there was requirement of Auth confirm to be initiated by either STA or Ex-AP.
 - Removed following API's from WiFi driver as they were not supported:
 - * wifi_auto_reconnect_enable()
 - * wifi_auto_reconnect_disable()
 - * wifi_get_auto_reconnect_config()
 - * wifi_get_tbt_offset()
 - * wifi_set_packet_filters()
 - * wifi_set_auto_arp()
 - * wifi_tcp_keep_alive()
 - * wifi_nat_keep_alive()

version 1.3.r23.p2

- New Additions:
 - Minor update to wifi_test_mode CLIs for better usability.
 - Added bug fixes in WiFi FW and updated FW versions to p155 for 8801, p182 for 8977 and p106 for 8978.
 - Added new wlan_uap_set_htcapinfo() API for setting HT Capability field for uAP.
- Bug Fixes:
 - Fixed RF Test Mode issues for SD8801 and SD8977 reported by QA.
 - Fixed WiFi 802.11n WPA3 SAE pre-cert test failures for SD8978.
 - Fixed stack overflow issue with WLCMGR thread during wlan-connect.
 - Fixed memory corruption issue cause by scan list overflow when using 11D.

version 1.3.r21.p1

- New Additions:
 - Added support for SD8978.

- Added Test Mode support for 8801, 8977 and 8978.
- Added new FW binaries for 8801, 8977 and 8978.
- Added OTP Force Region support in WiFi Driver.
- Added support for DHCP Server CLI to print IP addresses of connected clients to uAP.
- Added support to set HT Capability field for uAP.
- Added wlan_get_chanlist API and CLI.
- Added WiFi Driver task priority configurability option.
- Reduced WiFi Driver SRAM footprint.
- Added support in Wi-Fi driver to print debug events from WLAN FW.
- Added support for FW Dump generation using a micro-USB mass storage device.
- Bug Fixes:
 - Fixed an issue where a redundant Link Loss disconnect timer was defined in the wifi_iperf app.
 - Fixed an issue where sometimes the wlan_disconnect call did not abort an ongoing re-association.
 - Fixed an issue where station connection to an Open security Ext-AP fails after connection to a WPA3-SAE Ext-AP.
 - Fixed an issue where uAP did not start with WPA3-SAE security type.
- Known Issues
 - Wi-Fi sample apps do not work with Rev-C (2018) version of RT685 board

version 1.3.r20.p1

- New Additions:
 - Added new FW for 8801.
- Bug Fixes:
 - Fixed an issue where STAUT went to hang state when doing a disconnect in the wifi_iperf app.
 - Fixed an issue where STAUT failed to reassociate with an Ext-AP after band switch.
 - Fixed an issue where Scan command timeout was observed after changing the ssid of Ext-AP to which STAUT is connected.
- Known Issues
 - Wi-Fi sample apps do not work with Rev-C (2018) version of RT685 board

version 1.3.r19.p1

- New Additions:
 - Added new FW for 8801.
 - Updated WiFi API reference manual.
- Bug Fixes:
 - Fixed an issue where STAUT attempted to connect with Ext-AP continuously even when an Auth Failure occurred due to incorrect passphrase.
 - Fixed an issue where SDIO write error occurred sometimes for some packet sizes during Tx.
 - Fixed an issue where packet leakage was noticed on SD8801 when noise was applied.
 - Fixed an issue to suppress Association Failed warning generated during a connection attempt.
- Known Issues
 - Wi-Fi sample apps do not work with Rev-C (2018) version of RT685 board

version 1.3.r18.p1

- New Additions:
 - Added new FW for 8977.
- Bug Fixes:
 - Fixed an issue where STAUT fails to scan after Channel Switch and STAUT moves to new channel after eCSA IE.
 - Fixed IAR, MDK build compilation warnings.
- Known Issues
 - Wi-Fi sample apps do not work with Rev-C (2018) version of RT685 board

version 1.3.r17.p1

- New Additions:
 - Added new FW for 8977.
 - Added CLI command to get a list of stations connected to uAP.
- Bug Fixes:
 - Fixed an issue where Scan was temporarily aborted due to noise but correct status was not being returned to the host.
 - Fixed an issue where STAUT failed to connect with an AP with hidden SSID.
 - Fixed IAR, MDK build compilation warnings.
 - Fixed an issue where TCP bi-directional throughput stops abruptly for SD8801 and STAUT is disconnected.
 - Fixed an issue where UDP Dual Mode throughput stops abruptly and disconnection is seen.
 - Fixed an issue where Tx got stuck after removal of interference noise.
- Known Issues
 - Wi-Fi sample apps do not work with Rev-C (2018) version of RT685 board

version 1.3.r16.p1

- New Additions:
 - Added PMF Configuration for uAP.
 - Updated WLAN Versioning.
 - Updated WLAN CLIs.
- Bug Fixes:
 - Fixed an issue where the STAUT sometimes did not send aggregated packets during Tx.
 - Fixed an issue External Client could not associate with the UAP in WPA3-SAE mode.
 - Fixed IAR build compilation warnings.
 - Fixed an issue where the STAUT failed to associate with an AP with hidden SSID.
 - Fixed an issue where active scan probe requests were sent for DFS channels.
 - Fixed an issue where 2.4GHz-HT40 power values were exposed via CLI configuration.
 - Fixed an issue where the STAUT will be continuously in re-association mode after link-loss.
- Known Issues
 - Wi-Fi sample apps do not work with Rev-C (2018) version of RT685 board

version 1.3.r15.p1

- New Additions
 - Added support for Tx Power Limit configuration.
 - Added support for Channel List configuration.

- Added support for CW MODE.
- Added support for sysinfo CLI to get threads information, network stats, wlan stats and heap stats.
- Added -d and -r options to iperf CLI for dual and trade-off mode.
- Added support for antenna configuration for 8801.
- Added support for band configuration.
- Added new FW for 8977
- Bug Fixes:
 - Fixed an issue where UDP Rx data rate was low in iperf dual mode operation.
 - Fixed an issue where STAUT traffic was getting halted when there is traffic in overlapping BSS on the extension channel.
 - Fixed an issue where STAUT was not able to transmit above MCS 4.
 - Fixed an issue where association with uAP failed with WPA2 security mode.
 - Fixed an issue where STAUT failed to connect to WPA3 AP due to PMF config mismatch.
 - Fixed an issue where ping loss was observed for packets of size greater than 10000 bytes.
 - Fixed an issue in CLI where CR and LF characters were not handled properly.
 - Fixed an issue where TCP-Tx traffic abruptly stops when parallel traffic is ongoing on another device using the same AP.
 - Fixed an issue where DUT goes into hang state when iperf run is aborted.
 - Fixed an issue where the STATU was not sending aggregated packets to the AP.
 - Fixed an issue where UAP did not start with WPA2 security mode when ACS is configured.
 - Fixed an issue where ED MAC was not enabled by default.
- Known Issues
 - WiFi sample apps do not work with Rev-C (2018) version of RT685 board

version 1.3.r14.p1

- New Additions
 - Added support for Panasonic PAN9026 module.
 - Added -t option in iperf CLI for setting traffic running time.
 - Added -B option for supporting Tx/Rx Multicast packets during iperf runs.
 - Added World Wide Safe Mode configurability to the WiFi Driver.
 - Added ED MAC support for 8977.
 - Added support for PMF APIs and CLIs.
 - Added new FW for 8977 and 8801.
- Bug Fixes:
 - Fixed Coverity and mandatory MISRA issues reported on v1.3.r13.p1 release.
 - Fixed an issue where DUT console was getting stuck after initiating connection with an AP which has different RSN values than expected.
 - Fixed an issue where DUT was not able to roam from SAE-PSK to PSK configured AP.
 - Fixed an issue where the app became unresponsive after Soft AP is stopped.
 - Removed unnecessary files after Blackduck scan.
- Known Issues
 - WiFi sample apps do not work with Rev-C (2018) version of RT685 board

version 1.3.r1r3.p1

- New Additions
 - Updated WiFi Driver to be independent of the LwIP stack. Only the WLAN Connection Manager now uses LwIP.
 - Added WiFi Roaming feature.
 - Added CLI support for iperf in wifi_cli app.
 - Added support for CSA handling from wlan station side.
 - Added WLAN APIs for wlan-set-txratecfg, wlan-get-txratecfg, wlan-get-data-rate, wlan-set-reg and wlan-get-reg.
 - Updated External AP SSID required for wifi_iperf app.
- Bug Fixes:
 - Fixed Coverity issues reported on v1.3.r12.p1 release.
 - Added a fix for ping loss observed during ping test.
 - Added a fix where the console became unresponsive after wlan-start-network+wlan-stop-network commands are executed in loop.
 - Added a fix for HT IE missing from beacon for both 2.4GHz and 5GHz bands.
 - Fixed warnings reported for IAR.
 - Added a fix for increasing DHCP lease time to 24 hrs for long duration stress tests.
- Known Issues
 - WiFi sample apps do not work with Rev-C (2018) version of RT685 board

version 1.3.r12.p1

- New Additions
 - Added support for ping utility.
- Bug Fixes
 - Restart of SoftAP fails once SoftAP is stopped.
 - HT IE is missing from beacon for both 2.4GHz and 5GHz bands.
 - Low TCP-Rx and UDP-Rx throughput observed for 5GHz-HT40 band.
 - uAP couldn't be started after STA is associated to Ex-AP.
 - Stopping SoftAP results in disconnection of In-STA from Ex-AP.
 - DHCP server is not started for BSS created using "wlan-add" command.
- Known Issues
 - N.A

version 1.3.r11.p2

- New Additions
 - Updated SDK version for RT1060 to 2.7.1 RFP RC2.
 - Added FP91 based WiFi FW for SD8801, SD8977, SD8978, and SD8987.
 - SD8801, SD8977, SD8978 and SD8987 WiFi Fw have embedded WPA3 SAE support from version v1.3.r11.p1 onwards.
 - Added WLAN CLI support. Added new wifi_cli for CLI demo.
- Enhancements
 - License and Copyright Updates.
 - Formatted code base with clang-format 10.0.0 standards.
 - Integrated PICK utility for WiFi Driver code scrubbing.
- Bug Fixes

- Removed unnecessary prints LwIP stats display.
- Fixed an issue where uAP did not start on 5GHz band.
- Fixed an issue where loww TCP/UDP-Rx throughput was observed for 5GHz-HT40Mhz.
- Known Issues
 - N.A

version 1.3.r10.p1

- New Additions
 - Moved to using LwIP provided IPerf App.
 - Added SDIO Rx Aggregation support to improve throughput.
 - Added support for 11ac configuration API.
 - Updated License and Copyright information on all WiFi driver files and sample apps.
 - Moved MCUXpresso SDK base for RT1060 platform from 2.6.2 to 2.7.0 rc3.
- Known Issues
 - N.A

version 1.3.r9.p1

- New Additions
 - Achieved additional TP improvements by updating LwIP parameters.
- Bug Fixes
 - Fixed the issue of low throughput for both TCP and UDP (Tx/Rx) by updating LwIP parameters.
 - TCP Throughput was observed to be lower than UDP throughput by 70%-80%. Fixed this issue by updating TCP related LwIP configurations.
- Known Issues
 - N.A

3 RTOS Change Log

kernel for MCUXpresso SDK.

The current version is FreeRTOS kernel 10.4.3-LTS-Patch-2 Original package is available at github.com/FreeRTOS/FreeRTOS-Kernel.

- 10.5.1_rev0
 - update amazon freertos version
- 10.4.3_rev1
 - Apply CM33 security fix from 10.4.3-LTS-Patch-2. See rtos\freertos\freertos_kernel\History.txt
 - Apply CM33 security fix from 10.4.3-LTS-Patch-1. See rtos\freertos\freertos_kernel\History.txt
- 10.4.3_rev0
 - update amazon freertos version.
- 10.4.3_rev0
 - update amazon freertos version.
- 9.0.0_rev3
 - New features:
 - * Tickless idle mode support for Cortex-A7. Add fsl_tickless_epit.c and fsl_tickless_generic.h in portable/IAR/ARM_CA9 folder.
 - * Enabled float context saving in IAR for Cortex-A7. Added configUSE_TASK_FPU_SUPPORT macros. Modified port.c and portmacro.h in portable/IAR/ARM_CA9 folder.
 - Other changes:
 - * Transformed ARM_CM core specific tickless low power support into generic form under freertos/Source/portable/low_power_tickless/.
- 9.0.0_rev2
 - New features:
 - * Enabled MCUXpresso thread aware debugging. Add freertos_tasks_c_additions.h and configINCLUDE_FREERTOS_TASK_C_ADDITIONS_H and configFRTOS_MEMORY_SCHEME macros.
- 9.0.0_rev1
 - New features:
 - * Enabled -fno optimization in GCC by adding **attribute((used))** for vTaskSwitchContext.
 - * Enabled KDS Task Aware Debugger. Apply FreeRTOS patch to enable configRECORD_STACK_HIGH_ADDRESS macro. Modified files are task.c and FreeRTOS.h.
- 9.0.0_rev0
 - New features:
 - * Example freertos_sem_static.
 - * Static allocation support RTOS driver wrappers.
 - Other changes:
 - * Tickless idle rework. Support for different timers is in separated files (fsl_tickless_systick.c, fsl_tickless_lptmr.c).

- * Removed configuration option configSYSTICK_USE_LOW_POWER_TIMER. Low power timer is now selected by linking of appropriate file fsl_tickless_lptmr.c.
 - * Removed configOVERRIDE_DEFAULT_TICK_CONFIGURATION in RVDS port. Use of **attribute((weak))** is the preferred solution. Not same as `_weak`!
- 8.2.3
 - New features:
 - * Tickless idle mode support.
 - * Added template application for Kinetis Expert (KEx) tool (template_application).
 - Other changes:
 - * Folder structure reduction. Keep only Kinetis related parts.

4 Component Change Log

CODEC

The current codec common driver version is 2.3.1.

- 2.3.1
 - Bug Fixes
 - * Fixed violations of MISRA C-2012 rule 16.1,16.3.
- 2.3.0
 - Improvements
 - * Added enum `_codec_volume_capability` for `CODEC_SetVolume/CODEC_SetMute` to cover more volume configurations.
- 2.2.2
 - Bug Fixes
 - * Fixed the typo in codec common driver.
- 2.2.1
 - Bug Fixes
 - * Fixed violations of MISRA C-2012 rule 10.3, 8.3, 10.7, 17.7.
- 2.2.0
 - Improvements
 - * Used `HAL_CODEC_HANDLER_SIZE` which is determined by low level driver instead of use `CODEC_HANDLE_SIZE` for the codec device handle definition.
- 2.1.1
 - Improvements
 - * Supported all of the codec in the codec adapter.
 - * Modified the codec handle definition to improve user experience.
 - * Modified the capability member type from entity to pointer in codec handle.
 - Bug Fixes
 - * Fixed the Coverity issue regrading array compared agaist 0.
- 2.1.0
 - Deprecated APIs
 - * `CODEC_GetMappedFormatBits`
 - * `CODEC_I2C_WriteReg`
 - * `CODEC_I2C_ReadReg`
 - * `CODEC_I2C_ModifyReg`
 - * `CODEC_SetEncoding`
 - new APIs
 - * `CODEC_SetPower`
 - * `CODEC_SetVolume`
 - * `CODEC_SetMute`
 - * `CODEC_SetPlay`
 - * `CODEC_SetRecord`
 - * `CODEC_SetRecordChannel`

- * CODEC_ModuleControl
- new features
 - * Removed duplicate members in codec_handle_t and codec_config_t.
 - * Added codec_config_t pointer in codec_handle_t.
 - * Added codec capability flag in codec_handle_t.
 - * Used codec adapter instead of function pointer in codec common driver.
- 2.0.1
 - Added delayMs function pointer in codec handle.
- 2.0.0
 - Initial version.

WM8904

The current wm8904 driver version is 2.5.1.

- 2.5.1
 - Bug Fixes
 - * Fixed invalid clock divider issue generated from WM8904_SetMasterClock api
 - * Replace ‘__REV16’ with general implementation to swap bytes in a short variable.
- 2.5.0
 - Improvements
 - * Added master clock configuration support in function WM8904_SetAudioFormat.
 - * Align the sysclk parameter definition for the WM8904_SetAudioFormat/WM8904_SetMasterClock.
 - * Added api WM8904_SetDACVolume to support adjust DAC volume.
 - * Fixed the MISRA-2012 violation of 12.2, 10.3.
- 2.4.4
 - Bug Fixes
 - * Added the 11.025kHz/22.05kHz/44.1kHz samplerate support on codec WM8904.
 - * Fixed the MISRA-2012 violation of 4.7.
- 2.4.3
 - Bug Fixes
 - * Fixed the MISRA-2012 violations.
 - Fixed rule 8.6, 9.3, 10.1, 10.3, 10.4, 10.7, 10.8, 11.8, 11.9, 14.4, 16.1, 16.3, 16.4, 17.7, 20.9.
- 2.4.2
 - Bug Fixes
 - * Corrected the volume setting function behavior in wm8904 driver, support range align with its specification range.
 - * Corrected the volume setting function behavior in wm8904 adapter, support range 0 - 100, 0 for mute, 100 for maximum volume.
- 2.4.1
 - Bug Fixes
 - * Fixed the bit width register field overwritten issue.

- 2.4.0
 - New features
 - * Added flt support in wm8904 driver.
- 2.3.0
 - Improvements
 - * Added new API WM8904_SetMasterClock to support BCLK/LRCLK output mode.
- 2.1.0
 - new APIs
 - * WM8904_ReadRegister
 - * WM8904_WriteRegister
 - * WM8904_ModifyRegister
 - * WM8904_SetRecord
 - * WM8904_SetPlay
 - * WM8904_SetRecordChannel
 - * WM8904_SetModulePower
 - * WM8904_SetChannelVolume
 - * WM8904_SetChannelMute

New features

- Removed dependency on codec common driver.
- Added dependency on codec i2c.

Bug Fixes

- Fixed unchecked return value in WM8904_Deinit.
- Fixed the alignment fault issue by adding __NOP between continuous memory access.

2.0.3

- Bug Fixes
 - Fixed issue that wm8904 register access function truncated return value.

2.0.2

- Bug Fixes
 - Fixed using uninitialized value format.fsRatio when calling WM8904_UpdateFormat.

2.0.1

- Added WM8904_CheckAudioFormat API.
- Changed the second parameter's name of WM8904_SetAudioFormat to sysclk.

2.0.0

- Initial version.

.1 WM8960

The current wm8960 driver version is 2.2.4.

- 2.2.4
 - Improvements
 - * Remove CODEC_I2C_Deinit in WM8960_Deinit.
- 2.2.3
 - Improvements
 - * Reinitialise I2C in Deinit function.
- 2.2.2
 - Bug fixes
 - * Fixed violations of MISRA C-2012 rule 10.3.
- 2.2.1
 - Bug fixes
 - * Improved the internal PLL fatctor calculation formula.
- 2.2.0
 - Improvements
 - * Added masterClock member in wm8960_config_t to support wm8960 master mode.
 - Bug Fixes
 - * Fixed violations of MISRA C-2012 rule 4.7, 5.8, 10.3, 10.4, 12.2, 14.4.
 - * Added the bit clock divider configuration when wm8960 act as master.
- 2.1.3
 - Bug Fixes
 - * Fixed the issue that WM8960 had no ack when performing write register by updating the byte count to be written.
 - Bug Fixes
 - * Fixed violations of MISRA C-2012 rule 10.3, 8.3, 10.7, 17.7.
- 2.1.2
 - Improvements
 - * Enabled the class D output in WM8960_Init.
 - Bug Fixes
 - * Corrected the volume setting function behavior in wm8960 driver, support range aligned with its specification range.
 - * Corrected the volume setting function behavior in wm8960 adapter, support range 0 - 100, 0 for mute, 100 for maximum volume.
- 2.1.1
 - Improvements
 - * Removed useless bit clock divider configuration in function WM8960_ConfigDataFormat.
- 2.1.0
 - Improvements
 - * Added new API WM8960_SetPlay.
 - * Fixed error status overwrite issue in WM8960_ConfigDataFormat function.
 - * Removed dependency on codec common driver.
 - * Added dependency on codec i2c.
 - Bug Fixes
 - * Fixed the alignment fault issue by adding __NOP between continuous memory access.
- 2.0.2
 - Removed bit width hard code setting in function WM8960_SetProtocol.

- 2.0.1
 - Corrected the bclk divider calculation.
- 2.0.0
 - Initial version.

SGTL5000

The current sgtl5000 driver version is 2.1.1.

- 2.1.1
 - Improvements
 - * Corrected the volume setting function behavior in SGTL5000 driver, support range align with its specification range.
 - * Corrected the volume setting function behavior in SGTL5000 adapter, support range 0 - 100, 0 for mute, 100 for maximum volume.
 - Bug Fixes
 - * Fixed violations of MISRA C-2012 rule 10.3, 8.3, 10.7, 17.7.
- 2.1.0
 - Improvements
 - * Added API SGTL_SetPlay/SGTL_SetRecord.
 - * Removed dependency on codec common driver.
 - * Added dependency on codec i2c.
 - * Fixed division or modulo by zero issue in SGTL_ConfigDataFormat function.
 - Bug Fixes
 - * Fixed the alignment fault issue by adding __NOP between continuous memory access.
- 2.0.0
 - Initial version.

DA7212

The current da7212 driver version is 2.3.0.

- 2.3.0
 - Improvements
 - * Add input source select in init function.
- 2.2.3
 - Bug Fixes
 - * Fixed violations of MISRA C-2012 rule 10.4, 4.7.
- 2.2.2
 - Bug Fixes
 - * Fixed the MISRA-2012 violations.
 - Fixed rule 8.6, 9.3, 10.1, 10.3, 10.4, 10.7, 10.9, 11.1, 11.8, 14.4, 16.1, 16.3, 17.7, 17.3, 17.7, 20.9.
- 2.2.1

- Improvements
 - * Corrected the volume setting function behavior in DA7212 driver, support range align with its specification range.
 - * Corrected the volume setting function behavior in DA7212 adapter, support range 0 - 100, 0 for mute, 100 for maximum volume.
- 2.2.0
 - Improvements
 - * Added bclk invert parameter in the format structure.
 - * Added API DA7212_SetMasterModeBits/DA7212_SetPLLConfig.
 - * Added pll/sysClkSource parameters in the da7212 configuration structure.
 - * Disabled PLL by default.
- 2.1.0
 - Improvements
 - * Removed dependency on codec common driver.
 - * Added dependency on codec i2c.
 - Bug Fixes
 - * Fixed the alignment fault issue by adding __NOP between continuous memory access.
- 2.0.0
 - Initial version.

CS42888

The current cs42888 driver version is 2.1.3

- 2.1.3
 - Improvements
 - * Removed the assertion for codec reset function pointer.
- 2.1.2
 - Improvements
 - * Corrected the volume setting function behavior in CS42888 adapter, support range 0 - 100, 0 for mute, 100 for maximum volume.
 - Bug Fixes
 - * Fixed violations of MISRA C-2012 rule 4.7, 10.3, 8.3, 10.7, 17.7.
 - * Corrected the channel index during setting AIN volume in CS42888_Init.
- 2.1.1
 - Improvements
 - * Used software delay with delayMs pointer not provided by application.
 - * Fixed error status overwrite issue in CS42888_Init function.
 - * Removed dependency on codec common driver.
 - * Added API CS42888_SelectFunctionalMode/CS42888_SetChannelMute.
 - * Added dependency on codec i2c.
- 2.1.0
 - Improvements
 - * Unified CS42888 codec driver interface.

- * Bug Fixes
 - Corrected the ADC/DAC functional mode macro definitaion.
 - Added TDM and OLM mode support in the function CS42888_SetProtocol.
- 2.0.0
 - Initial version.

SERIAL_MANAGER

The current Serial_Manager component version is 1.0.2.

- 1.0.2
 - Add SerialManager_WriteTimeDelay()/SerialManager_ReadTimeDelay() for serial manager's read/write non-blocking mode.
- 1.0.1
 - Add prefixing fsl_component_xxx/fsl_adapter_xxx.
- 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.

© 2021 NXP B.V.

