Change Log for Kinetis SDK MQX™ RTOS from Version 1.0.0 to 1.2.0

1. Introduction

This document contains an overview of feature changes between Kinetis SDK (KSDK) versions 1.2.0 and 1.0.0. The change log maps the history of MQX™ RTOS feature evolution and enables users to select an MQX RTOS version that best fits their needs.

Contents

freescale™

2. MQX RTOS for Kinetis SDK 1.2.0

2.1. MQX RTOS Kernel Components – Version 5.0.2

- Bare metal (boot) stack area is newly re-used as an RTOS Interrupt stack frame during MQX RTOS startup. This saves RAM space, which is used only for configuration during startup.

- Init task was disabled for MQX Lite configuration. All initialization is done as part of a user application. This feature reduces RAM footprint of MQX Lite applications.

- The new TLSF best-fit memory allocators were added.

2.2. RTCS TCP/IPv4 and TCP/IPv6 (optional) stack – Version 4.2.0

- LLMNR – RTCS newly supported Link-Local Multicast Name Resolution (LLMNR) server. This protocol allows resolving simple label names on local subnet without the necessity of having a DNS server.

- Added DHCP Client IPv6 client application protocol support (available in MQX RTOS IPv6 add-on for purchase).

- Added Telnet Client IPv6 protocol support. The API of this component was changed (available in MQX RTOS IPv6 add-on for purchase).

- Added TFTP Client/Server IPv6 protocol support. The API of this component was changed.

- Added support of WebSocket server as part of HTTP server code. The implementation was fully tested by AutoBahn test suite.

- HTTP server was extended by SSL support (available in WolfSSL add-on for evaluation).

- Socket code has been updated by various BSD-compatible options and flags.

- ARP cache handling was modified to protect RTCS against DoS Attack.

- For more details, see the complete change log in the <KSDK>/middleware/rtcs/rtcs_changelog.txt file.

2.3. MFS FAT file system – Version 4.2.0

- Directory read in MFS was reworked to utilize sector cache and FAT chain abstraction.

- Find first/next API was updated to allow extraction of long filenames using single directory chain traversal (performance improvement).

- Path parsing was reworked to avoid allocation of path butters (RAM footprint decreased).

- Operations which do not create or rename directory records now support Unicode characters in the filenames (UTF-8 encoding). In particular, this includes directory search and opening of existing files.

2.4. **CyaSSL evaluation package (available in optional WolfSSL add on for evaluation) – Version 3.3.0**

- The CyaSSL library support was updated to version 3.3.0.

- Resolved problem with memory leak in RTCS SSL wrapper. The memory resources were incorrectly deallocated if the underlying sockets report error conditions.

2.5. **Other**

- Added examples demonstrating all MQX RTOS synchronization objects in the MQX Lite configuration – see <KSDK>/rtos/mqx/mqx/examples/

3. MQX RTOS for Kinetis SDK 1.1.0

- This is the first official release of MQX RTOS for Kinetis SDK. See Release Contents for component version information. Note that most components are based on version 4.1.2 while MQX RTOS kernel was updated by support of MQX Lite configuration and released under version 5.0.1.

- Peripheral I/O drivers are fully based on the Kinetis SDK driver set. MQX RTOS provides POSIX wrappers for the I/O console and filesystem only.

- The file API was changed to comply with POSIX standard (FILE* is used instead of legacy MQX_FILE).

- MQX RTOS provides the POSIX-based I/O for serial console (tty, uart), file system, Ramdisk, and other MQX RTOS-native I/O drivers (TFS, NULL, PIPE, etc).

- MQX RTOS provides its own re-entrant stdlib.h implementation.

- Support of MQX Lite configuration and new example application rtos/mqx/examples/. There are options provided for creating tasks from statically-allocated memory, and the application can define these tasks before MQX RTOS starts (using create_task() API).

- Optional support for RTCS IPv6 (available as a separate package).

- Optional support for CyaSSL stack. Demonstrated by HTTPs example application.

- MFS is integrated with USB stack, demonstrated in MFS example application.

- MQX RTOS Boards Support Package library has been removed and board-related configuration is now directly taken from the Kinetis SDK framework.

4. MQX RTOS for Kinetis SDK 1.0.0 (Beta)

- Released as Beta software only.

Document Number: MQXKSDKCHLGUG
Rev. 0
06/2015