NXP Semiconductors
Release Notes

Document Number: MQX4202RN
Rev. 4.2.0.2, 04/2016

MQX™ RTOS 4.2.0.2 Patch Release Notes

1 Overview

This patch release is based on the MQX™ RTOS 4.2.0 release and provides solutions to software issues identified in the released version.

Contents

2 Release Contents

This table lists the release contents.

Table 1. Release Contents

File Names
\doc\SW-Content-Register-MQX.txt
\ffs\source\wearleveling\buffer_manager\media_buffer_manager.cpp
\mfs\source\generic\mfs_dir_chain.c
\mfs\source\generic\mfs_find.c
\mfs\source\generic\mfs_lfn.c
\mqx\source\bsp\frdmk22f120m\init_bsp.c
\mqx\source\bsp\frdmk64f\init_bsp.c
\mqx\source\bsp\kwikstikk40x256\init_bsp.c
\mqx\source\bsp\twrk20d50m\init_bsp.c
\mqx\source\bsp\twrk20d72m\init_bsp.c
\mqx\source\bsp\twrk21d50m\init_bsp.c
\mqx\source\bsp\twrk21f120m\init_bsp.c
\mqx\source\bsp\twrk22f120m\init_bsp.c
\mqx\source\bsp\twrk24f120m\init_bsp.c
\mqx\source\bsp\twrk40d100m\init_bsp.c
\mqx\source\bsp\twrk40x256\init_bsp.c
\mqx\source\bsp\twrk53n512\init_bsp.c
\mqx\source\bsp\twrk60d100m\init_bsp.c
\mqx\source\bsp\twrk60f120m\init_bsp.c
\mqx\source\bsp\twrk60n512\init_bsp.c
\mqx\source\bsp\twrk64f120m\init_bsp.c
\mqx\source\bsp\twrk65f180m\init_bsp.c
\mqx\source\bsp\twrk65f180m\twrk65f180m.h
\mqx\source\bsp\twrk70f120m\init_bsp.c
\mqx\source\fio\io_fclos.c
\mqx\source\io\serial\dma\serl_dma_kuart.c
\mqx\source\io\serial\serl_kuart.h
\mqx\source\io\serial\serl_mk64.c
\mqx\source\kernel\mutex.c
\mqx\source\psp\cortex_m\cortex.h
\rtcs\source\apps\modem_supp.c
\rtcs\source\if\ipcfg.c
\rtcs\source\include\tftp_prv.h
\rtcs\source\nat\alggen.c
\rtcs\source\nat\natftp.c
\rtcs\source\nat\nattftp.c
\rtcs\source\tcpip\tcp_rcv.c
\mqx\source\bsp\frdmk22f120m\gcc_arm\intflash.ld
\mqx\source\bsp\frdmk22f120m\frdmk22f120m.h
\mqx\source\bsp\frdmk64f\gcc_arm\intflash.ld
\mqx\source\bsp\frdmk64f\gcc_cw\intflash.ld
\mqx\source\bsp\kwikstikk40x256\gcc_arm\intflash.ld
\mqx\source\bsp\kwikstikk40x256\gcc_cw\intflash.ld
\mqx\source\bsp\twrk20d50m\gcc_arm\intflash.ld

File Names
\mqx\source\bsp\twrk20d50m\gcc_cw\intflash.ld
\mqx\source\bsp\twrk20d72m\gcc_arm\intflash.ld
\mqx\source\bsp\twrk20d72m\gcc_cw\intflash.ld
\mqx\source\bsp\twrk21d50m\gcc_arm\intflash.ld
\mqx\source\bsp\twrk21d50m\gcc_cw\intflash.ld
\mqx\source\bsp\twrk21f120m\gcc_arm\intflash.ld
\mqx\source\bsp\twrk21f120m\gcc_cw\intflash.ld
\mqx\source\bsp\twrk22f120m\gcc_arm\intflash.ld
\mqx\source\bsp\twrk22f120m\twrk22f120m.h
\mqx\source\bsp\twrk24f120m\gcc_arm\intflash.ld
\mqx\source\bsp\twrk40d100m\gcc_arm\intflash.ld
\mqx\source\bsp\twrk40d100m\gcc_cw\intflash.ld
\mqx\source\bsp\twrk40x256\gcc_arm\intflash.ld
\mqx\source\bsp\twrk40x256\gcc_cw\intflash.ld
\mqx\source\bsp\twrk53n512\gcc_arm\intflash.ld
\mqx\source\bsp\twrk53n512\gcc_cw\intflash.ld
\mqx\source\bsp\twrk60d100m\gcc_arm\intflash.ld
\mqx\source\bsp\twrk60d100m\gcc_cw\intflash.ld
\mqx\source\bsp\twrk60d100m\twrk60d100m.h
\mqx\source\bsp\twrk60f120m\gcc_arm\intflash.ld
\mqx\source\bsp\twrk60f120m\gcc_cw\intflash.ld
\mqx\source\bsp\twrk60n512\gcc_arm\intflash.ld
\mqx\source\bsp\twrk60n512\gcc_cw\intflash.ld
\mqx\source\bsp\twrk64f120m\gcc_arm\intflash.ld
\mqx\source\bsp\twrk64f120m\gcc_cw\intflash.ld
\mqx\source\bsp\twrk64f120m\twrk64f120m.h
\mqx\source\bsp\twrk65f180m\gcc_arm\intflash.ld
\mqx\source\bsp\twrk65f180m\gcc_cw\intflash.ld
\mqx\source\bsp\twrk70f120m\gcc_arm\intflash_ddrdata.ld
\mqx\source\bsp\twrk70f120m\gcc_arm\intflash_sramdata.ld
\mqx\source\bsp\twrk70f120m\gcc_cw\intflash_ddrdata.ld
\mqx\source\bsp\twrk70f120m\gcc_cw\intflash_sramdata.ld
\mqx\source\io\flashx\flashx.c
\mqx\source\io\spi\spi_dspi.c
\mqx\source\io\spi\spi_dspi_common.c
\mqx\source\io\spi\spi_dspi_common.h
\mqx\source\io\spi\spi_dspi_dma.c
\mqx\source\io\spi\spi_dspi_dma_prv.h
\mqx\source\io\spi\spi_dspi_prv.h
\rtcs\source\apps\httpsrv_supp.c
\rtcs\source\apps\telnetcln_supp.c
\rtcs\source\if\sock_stream.c
\rtcs\source\if\sock_options.c
\rtcs\source\include\ppp.h
\rtcs\source\include\ppp_prv.h
\rtcs\source\include\pppfsm.h
\rtcs\source\include\ppphdlc.h
\rtcs\source\ppp\pppfsm.c
\rtcs\source\ppp\ppphdlc.c
\rtcs\source\snmp\snmp.c
\rtcs\source\tcpip\dhcpclnt.c
\rtcs\source\tcpip\tcp_clos.c
\shell\source\include\sh_rtcs.h
\shell\source\rtcs\sh_ipconfig.c

File Names
\shell\source\rtcs\sh_ppp.c
\usb_v2\usb_core\host\sources\controller\ehci\ehci_host.c

3 Patch Description

This patch release provides the software fixes and workarounds for the following issues identified in the MQX RTOS 4.2.0 release. Note that the issues below are described in more detail in subsequent sections.

- MQX RTOS 4.2.0.2
    - MQX-5686
        - _io_flashx_check_free_space() function in FlashX driver does not check source address alignment and whether the number of bytes to check is 8 bytes or 4 bytes aligned for FTFL.
        - Affects all PSPs.
    - MQX-5684
        - KEEPALIVE does not work properly.
        - Affects RTCS.
    - MQX-5664
        - The FIFO Depth of the SPI is using 4 that share all SPI instances. However, when SPI0 has a 4 depth FIFO, the rest have 2 or 1 depth FIFO.
        - Affects all BSPs.
    - MQX-5662
        - The PPP connection fails after pause and resume.
        - Affects RTCS.
    - MQX-5661
        - DNS incorrect behavior with DHC.
        - Affects RTCS.
    - MQX-5660
        - EHCI USB V2 connection debounce takes too long.
        - Affects USB_V2.
    - MQX-5647
        - DHCPv4 lease renew abnormal when the IPv6 is enabled.
        - Affects RTCS.
    - MQX-5644
        - USB transfer speed on MQX RTOS 4.2.0 is slower than in MQX RTOS 4.0.0 when copying with 1 Mbyte block size.
        - Affects USB.
    - MQX-5618
        - The RTCS telnet client logs in successfully to the server the first time and fails after exiting telnet and logging in again.
        - Affects RTCS.

- MQX-5609
  - Infinite while loop occurs in the httpsrv_read() function; For OPT_RECEIVE_TIMEOUT, the function httpsrv_recv() returns zero bytes.
  - Affects RTCS.
- MQX-5470
  - Missing vector when the flash configuration is commented out.
  - Affects all BSPs.
- MQX-5680
  - SNMP GetBulkRequest returns an error that is too big. Instead, it should return as many errors as it can.
  - Affects RTCS.

- MQX RTOS 4.2.0.1
  - MQX-5540
    - Added checking of the heap low address and the heap high address against the memory address 0x20000000 to see whether the global variables reside on that address and issue warning to the user.
    - Affects all BSPs.
  - MQX-4919
    - When the FFS file system is used and the buffer is destroyed while another task tries to acquire a new buffer, a deadlock occurs.
    - Affects all BSPs
  - MQX-5515
    - When two or more ENET devices try to access the PHY at the same time to get the connection status, the PHY returns an incorrect result.
    - Affected BSPs: TWR-VF65GS10, SVF522REVB, and TWR-MCF54418.
  - MQX-5517
    - The serial driver using DMA does not receive data properly.
    - Affects all BSPs
  - MQX-5598
    - An error occurs when closing a file handler twice.
    - Affects all BSPs
  - MQX-5537
    - MFS dir read does not display correct data.
    - Affects all BSPs and MFS
  - MQX-5612
    - Missing tftfp.h header file in the installer package.
    - Affects all BSPs and RTCS (NAT enabled)
  - MQX-5495
    - Missing protocol in the _mutatr_set_sched_protocol() function.
    - Affects PSPs.
  - MQX-5591
    - RTCS TCP has an issue with the TCP_Process_open function. If more than one are open at the same time, it locks up.
    - Affects all BSPs and RTCS

- MQX-5496
    - MQX RTOS kernel clean up is performed incorrectly.
    - Affects PSPs.
- MQX-5575
    - The Ethernet hardware checksum is disabled in the default configuration for the TWR-K65F180M Tower System module.
    - Affects the TWR-K65F180M Tower System module.

3.1 MQX-5686

- Issue

  This issue relates to the FlashX driver. The K60 writes long words to Flash and the K64/K70 writes phrases (8 bytes). However, the function "_io_flashx_check_free_space" handles only a 4 bytes buffer and isn’t aligned to phrases (8 bytes buffer). In the ftfe_flash_write_sector function, it overwrites a non 0xff area. For this scenario:

  ```
  ret = write(flashx_fp, write_buffer, 12);
  ret = write(flashx_fp, write_buffer, 4);
  ```

  The second write to the same location fails for K64/K70.
- Resolution

  In _io_flashx_check_free_space function, check the align address by 8 bytes that will sync with the ftfe/ftfl_flash_write_sector.

3.2 MQX-5684

- Issue

  MQX RTOS 4.2.0 connection resets after KEEPALIVE is sent. Some configurations set variables that KEEPALIVE function takes as parameters to 0. This causes KEEPALIVE interval time to be too small.
- Resolution

  Check and set variable values before use.

3.3 MQX-5664

- Issue

  The SPI FIFO depth is 4 for all SPI instances. However, when the SPI0 has 4 FIFO depth, the rest have 2 or 1 FIFO depth. This discrepancy can cause transfer data error when using the SPI with depth FIFO which is less than 4.
- Resolution

  Detect the FIFO depth for distinct SPI instance by disabling transmission and PUSHR until FIFO is full after counting the FIFO depth.

3.4 MQX-5662

- Issue

  This issue relates to the PPP connection in the RTCS stack. After calling the ppp_pause() function to pause the PPP connection and then calling the ppp_resume() function to resume the PPP connection, the ping command no longer responds. This issue occurs because the queue buffer size of the serial port is too small and does not handle timeout while waiting for a new HDLC package.

- Resolution

  When the PPP is enabled, increase the queue buffer size for the serial port. Handle the timeout while waiting for a new HDLC package. Moreover, this patch also adds a new feature to check the PPP physical connection. When serial cable is disconnected between the MCU and the PC, a notification message is printed to the terminal.

3.5 MQX-5661

- Issue

  DNS table does not update during network switch. A new DNS module, which replaces the LWDNS, does not update the DNS table automatically.

- Resolution

  Call the DNS update function whenever the network is switched or upon network change in the DNS information. Additionally, add the DNS listing feature in the RTCS shell.

3.6 MQX-5660

- Issue

  When the USB EHCI port change interrupt occurs, the _usb_ehci_process_port_change is called to process the interrupt. The _usb_ehci_process_port_change function calls the _usb_ehci_reset_and_enable_port function a device is attached.

  A delay, which lasts around 100 ms, is provided to de-bounce the device attach in the _usb_ehci_reset_and_enable_port function.

  The _usb_ehci_process_port_change function is called in the USB ISR and starves CPU resources while running inside the ISR.

- Resolution

  1. Set event in the ISR and call the _usb_ehci_process_port_change function in the USB stack task.

  2. Improve the code for clearing the port change status bit in the _usb_ehci_process_port_change function.

  A (2 * 0.125) ms delay is provided to clear the port change status bit. Use the SOF frame value instead of the SOF interrupt.

3.7 MQX-5647

- Issue

  When enabling the IPv6, the DHCPv4 leasing procedure does not work properly and causes threads which have low priority to stop working. The code for IPv6 was erroneously replacing the code for IPv4.
- Resolution

  Correct the mistype.

3.8 MQX-5644

- Issue

  USB transfer speeds on MQX RTOS 4.2.0 are slower than on MQX RTOS 4.0.0. The decrease in speed occurs while transferring data. Each NAK occurrence causes a huge delay before next transfer can start.
- Resolution

  The issue is resolved by setting the USBCFG_DEFAULT_MAX_NAK_COUNT to 30000 and change _time_delay every time NAK occurs to decrease the time delay for transfers.

3.9 MQX-5618

- Issue

  When the telnet client logs in successfully to the server, end of line CR+LF character of serial port is disabled. However, after exiting the telnet client, it does not enable CR+LF. This causes the login account to be incorrect because of the missing CR+LR character.
- Resolution

  This issue is resolved by enabling the CR+LF character for the serial port after exiting the telnet client.

3.10 MQX-5609

- Issue

  When timeout occurs, the function httpsrv_recv() returns zero bytes. Also, when timeout occurs, a “break” command should be called to exit the while loop. However, the break command does not execute because the “received” variable equals 0, which can lead to an infinite while loop.

```
received = httpsrv_recv(session, dst+read, len-read, 0);
if ((uint32_t)RTCS_ERROR != received)
{
    read += received;
}
else
{
    break;
}
```

- Resolution

Check the “received” variable before calling “read += received” command.

```
received = httpsrv_recv(session, dst+read, len-read, 0);
if (received && ((uint32_t)RTCS_ERROR != received))
{
   read += received;
}
else
{
   break;
}
```

3.11 MQX-5470

- Issue

  This issue occurs when the flash configuration in vectors.c is commented out. The result hex and map files don’t contain also vectors (.vectors section get is empty).
- Resolution

  This issue is resolved by adding extern __vector_table to linker file as shown below:

  EXTERN (_cfm __init_hardware __vector_table).

3.12 MQX-5680

- Issue

  RTCS should return as much data as it can rather than calling an error when requested tables have data larger than the response buffer of the RTCS.
- Resolution

  Return the included data table instead of throwing an error when the SNMP_ERROR_tooBig error occurs.

3.13 MQX-5540

- Issue

  This issue relates to the memory issue of Kinetis devices where the memory address 0x20000000 should not be used by an application program. If the memory address is used, an error interrupt is triggered when a variable stored in that memory address is referenced.
- Resolution

  Allocate a memory barrier 0x20000000 protection block only when the system pool (HEAP) is overlapping the barrier. If the system pool is not lying on the barrier and still reports an error to inform the user that global variables are overlapping this barrier. The issue needs to be solved in the linker file. This applies for ARM® Cortex®-M0 and ARM Cortex®-M4 only.

3.14 MQX-4919

- Issue

An issue occurs in the FFS file system when allocating a buffer and destroying a buffer results in a deadlock. The deadlock happens because the system tick timer interrupt activates the timer task to destroy the buffer when there is buffer waiting to be destroyed, while another task is trying to acquire a new buffer.

- Resolution

  To prevent a deadlock, the interrupt is temporarily disabled when a task is trying to acquire a new buffer.

3.15 MQX-5515

- Issue

  An issue occurs when two or more ENET devices try to access the same PHY. As a result, the PHY returns an incorrect result.

- Resolution

  The issue is solved by adding a semaphore to guard the PHY so that only one ENET device can access the PHY at a time.

3.16 MQX-5517

- Issue

  An issue related to shell occurs when an underlying serial driver uses the DMA mechanism internally to transmit and receive data. The serial receiver does not work properly with a shell application when input data from the user through the terminal window is slow.

- Resolution

  The issue is solved by updating the serial driver to properly extract received data out of the receive buffer when the DMA transfer finishes a minor loop transfer but the major loop transfer is not completed and when the DMA transfer finishes the major loop transfer.

3.17 MQX-5598

- Issue

  This issue relates to the MQX RTOS file handler and close mechanism. The Fclose function does not check whether the device pointer is equal to null, which then leads to an invalid handler.

- Resolution

  The issue is solved by updating the fclose function in the PSP FIO component to check the device pointer. It returns an error if the device pointer is null.

3.18 MQX-5537

- Issue

  This issue occurs in the MFS directory listing when the command ‘dir’ read does not display correct data.
- Resolution

  Walking through a directory chain might access an extra sector due to a +-1 bug. In particular, this causes difficulties for long filenames spanning across the cluster boundary.

3.19 MQX-5612

- Issue

  This issue occurs in RTCS when TFTP client application is rewritten in MQX RTOS 4.2.0. The tftp.h is removed but it is not properly updated for all relative components, such as NAT.
- Resolution

  The issue is solved by removing the include tftp header file in alggen.c, natftp.c, and nattftp.c files.

3.20 MQX-5495

- Issue

  The mutex service with _mutatr_set_sched_protocol function is missing the protocol condition (Inherit and Protect).
- Resolution

  The issue is solved by including all supported options (Inherit and Protect) for the mutex protocol validation check.

3.21 MQX-5591

- Issue

  This issue relates to the SOCK_STREAM_listen function that uses a local variable and passes it to the TCP_Process_open function. Each time a socket is opened, the local variable has the same address. Therefore, when the next pointer is stored, it creates a loop in the linked list of pending open requests, which leads to an infinite loop in the TCP_Return_open function.
- Resolution

  The issue is solved by allocating the next open pointer in the link list in TCP_Process_open function and TCP_Process_accept function.

3.22 MQX-5496

- Issue

  This issue relates to the kernel COMPONENT_CLEANUP timer index. When the index is a kernel event, it is incorrect.

- Resolution

  The issue is solved by the correct kernel timer component clean up.

3.23 MQX-5575

- Issue

  This issue relates to the default configuration of the TWR-K65F180M Tower System module where Ethernet hardware checksum is disabled.

- Resolution

  The issue is solved by enabling the hardware check sum calculation. The checksum is calculated by hardware instead of software that accelerates RTCS.

4 Applying Patches

The patch is provided in a **.zip package** and a **patch file** format.

The .zip package should be used for an installation of the MQX RTOS 4.2.0 source files which have not been previously modified because extracting .zip to the MQX RTOS source may overwrite those changes.

The patch file can be used for an installation of the MQX RTOS 4.2.0 source files which have been previously modified because the patch file checks the original file content before it is applied. If the original file content is modified, the patch does not overwrite the changes.

The instructions to apply either the .zip package or the patch file are described below:

These are the steps to apply the **.zip package**:

1. Extract the content of the zip file to the MQX RTOS 4.2.0 installation. The default installation directory is C:\Freescale\Freescale_MQX_4_2.
   To extract by default Windows® unzip tool, open the compressed folder and then drag all files and folders to the MQX RTOS 4.2.0 installation directory.
2. See the *Getting Started with Freescale MQX™ RTOS* (document MQXGSRTOS) to compile MQX RTOS.

These are the steps to apply the **patch file:**

1. Copy the patch file to the MQX RTOS installation folder (by default C:\Freescale\Freescale_MQX_4_2)

2a. Update from 4.2.0; In the MQX RTOS installation folder, execute the command:

```
Windows: patch -p1 --ignore-whitespace -i mqx_patch-4.2.0.2.windows.AllInOne

Linux: patch -p1 --ignore-whitespace -i mqx_patch-4.2.0.2.linux.AllInOne
```

2b. Update from 4.2.0.1; In the MQX RTOS installation folder, execute the command:

```
Windows: patch -p1 --ignore-whitespace -i mqx_patch-4.2.0.2.windows.Incremental

Linux: patch -p1 --ignore-whitespace -i mqx_patch-4.2.0.2.linux.Incremental
```

Note:

Use GIT patch or a similar patch program. To enable the use of the patch command using UnxUtils, download the UnxUtils package, decompress it and add the following path "<file location>\usr\local\wbin" to your Windows "Path" variable. When trying to use an LF file on Windows 32, which uses CRLF, the execution fails with the error “Assertion failed: hunk, file patch.c, line 321". To fix, open it in a WordPad program using the "write" command: cmd:\>write <patch_file> and save the opened file.

3. Because the patch does not support binary, extract and copy “\mqx\examples\hwtimer_README.pdf” file from **.zip package** to “\mqx\examples\hwtimer_README.pdf” file in MQX RTOS installation directory to patch a pdf file.

4. See the *Getting Started with Freescale MQX™ RTOS* (document MQXGSRTOS) to compile MQX RTOS.

5 Known Issues

These sections describe issues that are unresolved in this patch. The workarounds are described for each known issue.

5.1 MQX-5555

- Issue

  This issue relates to the MFS file system when creating a file/directory with the same name as the volume label.

- Workaround

  Windows OS in general allows creating directory records for files/directories with the name matching the volume label. Because the volume label is physically stored as a directory record, it requires special handling to allow creating a file or directory with the same name as the volume name. MFS currently does not have this capability and, as a result, does not allow creating such files/directories. The workaround is to remove or change the volume name.

5.2 MQX-5639

- Issue

  If the MFS_SEARCH_SUBDIR is specified, the internal matching masks are always set to predefined values ignoring other search attributes (hidden/read only/system). As a result, it is not possible to search for subdirectories with particular attributes.

- Workaround

  Change this code in the mfs_find.c file:

  ```
        else if (sp_ptr->ATTRIBUTE & MFS_SEARCH_SUBDIR)
  ```

  `to:`

  ```
        else if ((sp_ptr->ATTRIBUTE & 0x3F) == MFS_SEARCH_SUBDIR).
  ```

5.3 MQX-5644

- Issue

  USB transfer speeds are slower in MQX RTOS 4.2.0 than in MQX RTOS 4.0.0. The USB code in MQX RTOS 4.0.0 and MQX RTOS 4.2.0 is the same. The root cause for speed decrease involves the MQX RTOS kernel changes. The _time_delay results are different.

- Workaround

Note

> This workaround may impact other USB components, such as USB Host HID.

Redefine `USBCFG_DEFAULT_MAX_NAK_COUNT` to 15

Change the `case 0x0a://NAK` in the khci.c file, function `_usb_khci_atom_tr`:

```
    case 0x0a:  // NAK
    res = KHCI_ATOM_TR_NAK;
    if (pipe_desc_ptr->G.PIPETYPE == USB_BULK_PIPE) {
        if (pipe_desc_ptr->G.NAK_COUNT - retry > BSP_ALARM_RESOLUTION) {
           _time_delay(pipe_desc_ptr->G.NAK_COUNT - retry);
        }
    }
    else {
       if (retry) _time_delay(delay_const * ( pipe_desc_ptr->G.NAK_COUNT -
retry));
    }
    break;
```

6 Revision History

This table summarizes revisions to this document.

Table 2. Revision history table

Revision number	Date	Substantive changes
4.2.0.2	04/2016	Initial release

How to Reach Us:

Home Page:
www.nxp.com/

Web Support:
www.nxp.com/support

Information in this document is provided solely to enable system and software implementers to use Freescale 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.

Freescale reserves the right to make changes without further notice to any products herein. Freescale makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale 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 Freescale 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. Freescale does not convey any license under its patent rights nor the rights of others. Freescale sells products pursuant to standard terms and conditions of sale, which can be found at the following address: freescale.com/SalesTermsandConditions.

Document Number: MQX4202RN
Rev. 4.2.0.2
04/2016