

Freescale MQX™ RTOS BSP 移植指南

1 简介

Freescale MQX™ RTOS 为各种飞思卡尔开发板提供源代码和板级支持包 (BSP)。BSP 包含专用于硬件的底层代码，包括内部的微控制器 (MCU) 和电路板上的外部硬件。BSP 提供的底层代码包含了外设驱动程序、时钟配置、存储器映射和存储器管理等。BSP 是您着手定制的绝佳起点。

本文档提供了移植 MQX BSP 的指南，用于定制具有相似 MCU 衍生产品的目标硬件。*Freescale MQX™ RTOS BSP Porting Example User Guide* (文档 MQXBSPEXUG) 中提供了移植流程的示例。包含的 BSP 专为飞思卡尔开发板所编写。如果自定义硬件和开发板之间存在以下任何硬件差异，均需要对 BSP 进行更改：

- 外设和外设驱动程序差异
- 外设引脚和 GPIO 引脚差异
- 时钟源、时钟配置和时钟频率差异
- 自定义驱动程序
- MCU 衍生产品差异

虽然本文档重点讨论飞思卡尔 Kinetis 系列 MCU，但它也适用于 MQX 支持的其他飞思卡尔架构。

1.1 BSP 移植流程简介

以下是移植 MQX BSP 的步骤。其中某些步骤是可选的，包括要不要使用 Processor Expert (PEX)。

内容

| | | |
|----|---------------------------------------|---|
| 1 | 简介..... | 1 |
| 2 | 克隆 Freescale BSP..... | 2 |
| 3 | 在 BSP 中使用 Processor Expert (PEX)..... | 3 |
| 4 | 在 MQX 项目中更改 MCU 衍生产品..... | 3 |
| 5 | 时钟配置..... | 5 |
| 6 | BSP 包含文件..... | 5 |
| 7 | BSP 初始化文件..... | 6 |
| 8 | BSP 驱动程序更改..... | 7 |
| 9 | BSP 存储器映射和链接文件..... | 8 |
| 10 | 链接后批处理文件..... | 8 |
| 11 | CodeWarrior 调试器存储器文件..... | 8 |
| 12 | 移植示例应用程序..... | 9 |
| 13 | 结语..... | 9 |
| 14 | 附录 A: 查找最接近的 MQX BSP..... | 9 |

1. 克隆 BSP——复制所提供的 BSP 并使用新名称（之后可修改）。
2. 将 PEx 添加到 BSP 项目中——只有在结合使用 PEx 工具链的 IAR Embedded Workbench®或 MDK-ARM Keil™时才需要。
3. 更改 PEx CPU 组件/封装——可选步骤（如果在 BSP 中使用 PEx）。
4. 更改 MCU 衍生产品。
5. 修改时钟配置。
6. 修改 BSP 源/包含文件和 user_config.h。
7. 修改衍生产品专用的驱动程序文件。
8. 在 BSP 项目中添加/删除驱动程序源文件。
9. 修改 BSP 存储器映射和链接文件。
10. 修改链接后批处理文件。
11. 更改 CodeWarrior 调试器存储器文件——可选（如果使用 CodeWarrior，取决于衍生产品）。
12. 将示例应用程序移植到新的 BSP 中——可选（用于测试新的 BSP）。

2 克隆 Freescale BSP

创建定制 MQX BSP 时，建议执行以下步骤：

1. 在 MQX 发布包中找到与定制 BSP 所使用的器件最为接近的 BSP。
2. 克隆该 BSP 并以新名称、目录和路径创建副本。
3. 根据需要修改克隆的 BSP。

2.1 选择最接近的 Freescale BSP 进行克隆

当选择了合适的 BSP 进行克隆时，BSP 移植工作将会变得十分容易。当两个衍生产品非常相似时，只需进行少量更改。飞思卡尔不会对每款 Kinetis 衍生产品均提供 MQX BSP，而是对不同的 Kinetis 开发板提供 BSP。之所以选择开发板上的 MCU，是因为此器件具有该 MCU 家族的所有功能。其他衍生产品为类似器件，通常具有精简的功能或较小的存储器。因此，任何 Kinetis 衍生产品 BSP 都可从包含的 MQX BSP 中选择一个进行移植。了解哪块电路板和哪个 MQX BSP 与所需定制硬件上的 MCU 衍生产品最为接近非常重要。本文档的附录 A 可为您做此决定提供帮助。

2.2 BSP Cloning Wizard

在选定 BSP 后，下一步就是克隆 BSP。克隆 BSP 就是复制所有的文件、目录和项目，将原始 BSP 的所有引用重命名并放在 BSP 中。结果形成一个具有名称的 BSP，它是原始 BSP 的克隆。

安装的 Freescale MQX 包含一个自动执行克隆过程的工具，称为 BSP Cloning Wizard (BSP 克隆向导)。使用该工具可以缩短克隆时间并避免可能发生人为错误。BSP Cloning Wizard 是一款 Windows 应用程序，通过 MQX 发布包进行安装，路径为<MQX Installation Directory>\tools\BSPCloningWizard\BSPCloningWizard.exe。有关该工具的更多信息，请参见 *Getting Started with Freescale MQX™ BSP Cloning Wizard*（文档 MQXGSCLW）。

该工具还提供导出新克隆的 BSP 和创建用于选定 BSP 的新 MQX 软件目录结构的功能。由于导出的目录只包含特定 BSP 的文件，因此这对 MQX 软件源代码的共享、归档或进行版本控制很有帮助。

请注意，*MQX Board Support Package Porting Guide*（文档 AN4287）介绍了在 BSP Cloning Wizard 推出之前的克隆流程。即便如此，该文档仍有参考价值，而克隆向导可以自动执行该过程并节省时间和精力。

3 在 BSP 中使用 Processor Expert (PEX)

Processor Expert (PEX) 是 Freescale 的代码生成工具。它可以生成用于 MCU 外设的驱动程序和初始化代码，并且可以简化 MCU 时钟设置。它可以提供快速硬件设置，还可以作为了解外设和寄存器使用情况的工具。PEX 可以作为飞思卡尔的 CodeWarrior 工具链的一部分获取，也可以作为独立应用程序 Processor Expert 驱动程序套件下载，并与其工具链配合使用。要了解有关 PEX 的更多信息、查找相关文档和培训资料或下载该工具，请访问 freescale.com/processorexpert。

PEX 常用来创建 MQX Kinetis BSP。值得一提的是，BSP 中使用的时钟配置就是由 PEX 生成的。关于使用 PEX 修改 BSP 中的时钟配置的更多信息，请参见 *How-to Change Default Clock Settings in Kinetis BSPs* (文档 MQXGSCCLKBSP)。

在 MQX 中使用 PEX 是可选的，可根据您的喜好而定。在 MQX BSP 中使用 PEX 有以下好处：

- 简化时钟配置。PEX 可快速、方便地修改 MCU 的时钟源和频率。即使不在 BSP 中使用 PEX，还是可以使用 PEX 来快速生成时钟配置代码。参见[时钟配置](#)。
- 具有多个时钟配置。一些应用需要动态改变时钟频率和时钟源，尤其是低功耗应用。PEX 允许在运行时修改多个时钟配置和 API。MQX 的低功耗管理器 (LPM) 驱动程序对于不同的低功耗状态使用不同的时钟配置。利用 PEX 可以方便地添加和管理时钟配置。
- 添加或替换 MQX BSP 中的外设驱动程序。PEX 可添加在 MCU 中可用的外设驱动程序，而不是在 MQX 中可用的。更多信息，请参见 *CW for Microcontrollers V10 and Freescale MQX™ RTOS* (文档 MQXCWPP)。

MQX 源代码在使用或不使用 PEX 的情况下均能正常使用。当使用 PEX 时，使用 PEX 生成的硬件初始化和时钟配置。否则，项目默认使用已经存在于 MQX BSP 源代码中的初始化和时钟配置代码。在编译时使用 `PE_LDD_VERSION` 宏来确定是否应该使用 PEX 代码。当使用 PEX 时，宏被定义，并且 PEX 生成的代码会加载到 MQX 源代码中。

4 在 MQX 项目中更改 MCU 衍生产品

如果定制 BSP 中使用的器件与原有 BSP 中的不同，则需要更改 MCU 衍生产品。下面总结了更改 MCU 衍生产品的流程：

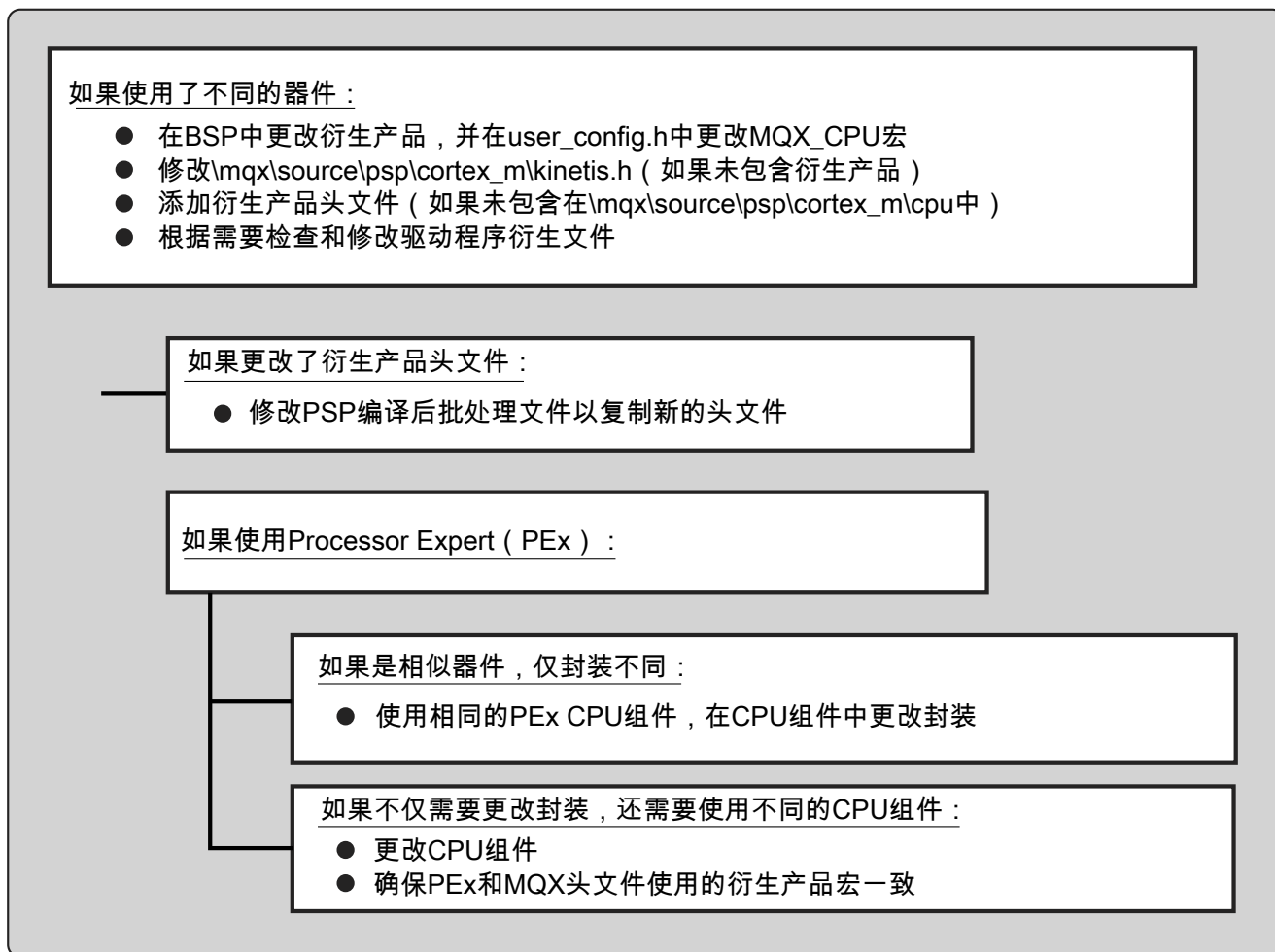


图 1. 更改 MCU 衍生产品流程摘要

MQX 使用针对 MCU 衍生产品、电路板专用和衍生产品头文件的宏。该衍生产品由 user_config.h 中的 MQX_CPU 宏进行定义。文件\mqx\source\psp\cortex_m\kinetis.h 使用该衍生产品，以便将衍生产品头文件包含到 PSP 中。更改 MCU 衍生产品宏同时会导致驱动程序衍生文件发生变化（参见[修改驱动程序衍生文件](#)）。

如果衍生产品头文件更改，也需要修改 PSP 编译后批处理文件。批处理文件在 PSP 项目编译后运行，并将在编译应用程序时需使用的头文件复制到\lib 文件夹中。必须更新批处理文件，以复制不同的头文件。

如果使用 PEX 且 MCU 衍生产品发生改变，则还应更新 PEX 以包含新的衍生产品。如果衍生产品间仅存在封装上的差异，则可以使用相同的 CPU 组件，并在 CPU 组件中更改封装。这种情况下，无需更改 MCU 衍生产品。如果衍生产品间不仅仅是封装差异，则需要更改 PEX CPU 组件。PEX CPU 组件中使用的衍生产品宏必须与 MQX 头文件中使用的衍生产品宏相一致。否则，将包含两个寄存器头文件并导致编译错误。一旦 CPU 组件发生改变，就必须更改 MQX 头文件中的衍生产品宏，使其保持一致。

4.1 修改驱动程序衍生文件

许多 MQX 驱动程序包含具有 MCU 衍生产品特定信息的源文件，如衍生产品上的外设数量、外设基址及其中断向量。MQX 发布包中包含的驱动程序文件未涵盖全部的 MCU 衍生产品；它们支持相应电路板上使用的衍生产品。因此，如果更改 MQX BSP 中使用的 MCU 衍生产品，就需要更改相应的驱动程序源文件。

以下列表显示针对 MCU 衍生产品的驱动程序源文件（用于 Kinetis BSP），适用于 MQX 版本 4.0.2 及更高版本：

- ADC 驱动程序: \mqx\source\io\adc\kadc\ADC_mkxx.c/h
- CAN 驱动程序: \mqx\source\io\can\flexcan\FlexCAN_mkxx.c

- 以太网驱动程序: `\mqx\source\io\enet\macnet\MACNET_mkxx.c`
- SDHC 驱动程序: `\mqx\source\io\sdhc\SDHC_mkxx.c`
- I2C 驱动程序: `\mqx\source\io\i2c\I2C_mkxx.c`
- I2S 驱动程序: `\mqx\source\io\i2s\I2S_mkxx.c`
- LWADC 驱动程序: `\mqx\source\io\lwadc\lwadc_kxx.c`
- NAND 闪存驱动程序: `\mqx\source\io\nandflash\ncf\ncf_kxx.c`
- SAI 驱动程序: `\mqx\source\io\sai\sai_mkxx.c`
- UART 驱动程序: `\mqx\source\io\serial\serl_mkxx.c`
- SPI 驱动程序: `\mqx\source\io\spi\spi_mkxx.c`
- USB DCD 驱动程序: `\mqx\source\io\usb_dcd\usb_dcd_mkxx.c`

MQX 发布包中包含多个驱动程序源文件，用于各种 Kinetis 子系列（K20、K40 和 K60 等）。但是并不包含所有子系列，因为 MQX 只支持相应电路板上使用的衍生产品。请注意，用于具体子系列的驱动程序源文件可能不支持所需的衍生产品。例如，当从 TWRK60D100M 移植到 K40 时，两个衍生产品在芯片中都有一个 SAI 外设。TWRK60D100M BSP 在 SAI 驱动程序（文件 `\mqx\source\io\sai\sai_mk60.c`）中使能一个 SAI。然而，由于其他 K60 BSP 具有两个 SAI，驱动程序文件 `sai_mk60.c` 通过 BSP 中定义的宏有条件地设定支持的 SAI 数量，代码如下：

```
switch (dev_num)
{
    case 0:
        addr = (pointer) I2S0_BASE_PTR;
        break;
#ifdef MCU_MK60D10
    case 1:
        addr = (pointer) I2S1_BASE_PTR;
        break;
#endif
}
#endif
```

当 MCU 衍生产品已更改且不再为 MCU_MK60D10 时，文件 `sai_mk60.c` 仍然包含两个 SAI，而芯片只支持一个 SAI。这会导致编译错误，因为 K40 的头文件不包含用于第二个 SAI 外设的符号。要改正编译错误，可修改 `sai_mk60.c`，使其不包括该衍生产品的第二个 SAI，或者更改 BSP 项目以使用 `sai_mk40.c` 文件。

应根据所需的衍生产品检查其特定的驱动程序源文件，并根据需要做相应修改。查看器件文档，了解衍生产品的支持信息。在器件参考手册的芯片配置章节和系列产品简介中会有关于具体衍生产品的可用外设介绍。

5 时钟配置

在定制 BSP 时，通常必须更改时钟配置。这就需要使用不同的时钟源和频率，并更改内部时钟的频率和分频器。

使用 Processor Expert 可以简化这些更改，因为时钟配置可以通过 PEx GUI 方便地进行更改。即使 BSP 项目中未包含 PEx，仍可通过创建新的 PEx 项目并对 CPU 组件配置所需时钟设置来简化流程。最后，将生成的代码复制到 BSP 源文件中。

关于使用 PEx 更改 BSP 中的时钟配置的更多信息，请参见 *How-to Change Default Clock Settings in Kinetis BSPs*（文档 MQXGSLCKBSP）。

如果在 BSP 项目中不使用 PEx，则 BSP 使用文件 `\mqx\source\bsp\ 来配置时钟。该文件中用来配置时钟的函数实际上来自 PEx。如果 BSP 中不包含 PEx，可以使用新项目中的 PEx 来配置时钟，然后从 \Generated_Code\Cpu.c to bsp_cm.c 中复制函数。或者，也可以手动修改 bsp_cm.c，无需使用 PEx。`

6 BSP 包含文件

BSP 通过多个位于 `\mqx\source\bsp\ 中的头文件进行定制。应检查该目录下的所有文件，并根据电路板和应用程序的需要进行相应修改。以下是一些需要注意的主要头文件。`

6.1 <Board_Name>.h 文件

每个 BSP 均有一个以电路板命名的头文件，如 twrk60d100m.h。该文件包含用于 BSP 的大部分硬件定制，因而在定制 BSP 时必须对其进行检查。一些主要设置包括：

- MCU 存储器映射——除了链接器命令文件，BSP 包含用于具体 MCU 衍生产品的存储器映射的宏。这包括了用于电路板的存储器映射符，可以包含内部和外部存储器，如基址和不同存储器区域的大小（闪存、RAM 和外部存储器等）。关于存储器映射的详细信息，请参见具体器件的器件参考手册。
- 时钟配置修改——除了在[时钟配置](#)中的更改，该文件还可用于更改以下时钟选项：
 - RTOS 节拍计数周期。该文件包含了 BSP_ALARM_FREQUENCY 宏，定义用于调整程序的节拍定时器周期。该宏定义了每秒的节拍数。MQX 默认为 200 (5ms 时钟节拍)。
 - RTOS 节拍定时器。该文件包含一些 BSP_SYSTIMER 宏，定义用于 RTOS 节拍计数的硬件定时器。Kinetis BSP 默认使用 ARM® Cortex®-M4 内核中的 SysTick 定时器。可以更改为其他硬件定时器。
- GPIO 板级规范——该文件包含用于具体电路板的 GPIO 宏，并包含多个示例。可用于定制电路板并使用示例测试 BSP。
- I/O 驱动程序默认选项——许多驱动程序使用该文件中定义的宏作为默认选项。在定制 BSP 时应检查这些选项。
- 默认 MQX 初始化——当初始化 MQX 时，采用该文件中的某些默认设置。

6.2 BSP_prv.h

该文件包含多个对 BSP 中的私有函数和结构的声明。在编译定制 BSP 时无需太多修改。

6.3 BSP.h

该文件包含驱动程序和 BSP 函数所需的头文件，以及应用所使用的公共声明。应根据所支持的不同驱动程序进行修改。

6.4 user_config.h

这是用于 MQX 软件库的主配置文件。应针对电路板和应用程序检查和定制该文件。

7 BSP 初始化文件

BSP 通过多个位于 \mqx\source\bsp**<Board_Name>** 中的源文件进行定制。应检查该目录下的所有文件，并根据电路板和应用程序的需要进行相应修改。以下是一些需要注意的主要源文件。

7.1 init_bsp.c

该文件用于初始化 BSP 和在 BSP 中默认启用的驱动程序。

- `_bsp_pre_init()`和 `bsp_init()`——这些函数初始化了在 BSP 中默认启用的硬件。在内核使用 BSP 的 `mqx.c` 中的 `mqx()`初始化后会调用该文件。由于大多数的驱动程序初始化函数由 `user_config.h` 中定义的宏决定，因而该文件无需太多修改。不过，还是应该对其进行检查。
- 用于计数内核节拍的硬件定时器——MQX 使用硬件定时器驱动程序进行内核节拍计数。该驱动程序也在 `init_bsp.c` 中初始化。如果定制 BSP 需要对用于节拍计数的硬件定时器外设进行更改，请参见 *Freescall MQX™ I/O Drivers User Guide* (文档 MQXIOUG) 中的 HWTIMER 驱动程序章节，并根据需要更新源文件。

7.2 init_HW.c

该文件需要在内核初始化之前进行硬件初始化。该文件中的 `init_hardware()`函数在启动 MQX 之前由 PSP 调用。如果内核需要使用外部存储器，通常在该文件中初始化。

7.3 init_GPIO.c

这是 BSP 中需要定制的主要文件。它控制器件的引脚设置，包括驱动器信号的引脚复用选项。

7.4 MQX_init.c

该文件包含结构 `MQX_init_struct`，该结构包含内核初始化时的默认设置。该结构中使用的参数是在头文件中定义的宏（前面已讨论）。该文件对于定制的 BSP 无需修改。

7.5 vectors.c

该文件用于编程 Kinetis 闪存配置字段和硬件中断向量表。在定制 BSP 时无需更改，不过根据不同的应用可能需要做一定的更改。如果需要更改该文件，请查看具体的器件参考手册并根据需要修改。访问 freescale.com 网站并搜索“Freescall MQX™ RTOS User Guide”，详细了解 MQX 软件管理中断的方式。

7.6 bsp_cm.c

对该文件的更改在[时钟配置](#) 章节中讨论。

8 BSP 驱动程序更改

移植 BSP 包括定制用于 MCU 衍生产品、电路板和应用程序的驱动程序。位于 `\mqx\source\io` 目录下的驱动程序源文件对于不同的 BSP 是通用的，而 BSP 定制的驱动程序使用 `\mqx\source\bsp\<Board_Name>` 目录下的文件。应检查该目录下的所有文件，并根据电路板和应用程序的需要进行相应修改。

根据不同的驱动程序，可能需要修改某些衍生产品专用的源文件。参见[修改驱动程序衍生文件](#)。

如果应用程序所需的驱动程序未包含在 MQX 发布包中，可以另行创建并添加到 BSP 中。关于创建兼容 MQX I/O 子系统的 MQX 驱动程序的详细信息，请参见[应用说明 AN3902](#)。可以使用 PEX 生成外设驱动程序代码并将其包含到 BSP 中。关于在 BSP 中使用 PEX 生成驱动程序的教程，请参见 *CW for Microcontrollers V10 and MQX™* (文档 MQXCWPP)。如果像这样使用 PEX，则需要将 PEX 包含在 BSP 中。更多详情，请参见在 [BSP 中使用 Processor Expert \(PEX\)](#)。

8.1 init_<driver>.c 文件

这些文件包含在 BSP 中的 `\mqx\source\bsp\<Board_Name>` 目录下，用于定制驱动程序。其中包含 MCU 衍生产品、电路板和应用程序的设置。

8.2 用于外部组件的驱动程序

除了用于内部外设的驱动程序，BSP 还可能包含用于外部组件的驱动程序。Freescale MQX RTOS 发布包包含多个外部组件的驱动程序，适用于 Freescale Tower 系统或其他开发板。这些可在定制 BSP 中作为所需驱动程序的参考进行修改或使用。以下为 MQX 发布包中包含的部分外部驱动程序：

- 以太网 PHY 示例，位于 `\mqx\source\io\enet\phy`
- 外部闪存示例，位于 `\mqx\source\io\flashx`
- 段式 LCD 示例，位于 `\mqx\source\io\lcd`
- NAND 闪存示例，位于 `\mqx\source\io\nandflash\nand_devices`

8.3 在 BSP 项目中添加/删除驱动程序源文件

当移植到其他 MCU 衍生产品中时，可能添加或删除一些外设。可以根据需要从 BSP 项目中添加或删除这些驱动程序源文件。例如，从原来的 K60 BSP 移植到 K40 衍生产品，可以删除 ENET 以太网驱动程序，并添加段式 LCD 驱动程序。

9 BSP 存储器映射和链接文件

在更改 MCU 衍生产品时，存储器大小通常会发生变化，有时存储器地址也会相应改变。必须更新 BSP 使用的存储器映射，以反映这些变化。与存储器映射相关的 BSP 宏位于文件 `<Board_Name>.h` (`<Board_Name>.h` 文件) 中。此外，还需更改链接文件中的存储器映射。链接文件位于 `\mqx\source\bsp\<Board_Name>\<toolchain>`。查看工具链文档，了解有关链接文件的详细信息和语法。

10 链接后批处理文件

MQX 使用批处理文件(在软件库编译后运行)，并将应用程序所需的头文件复制到 `\lib` 文件夹中。当使用 BSP Cloning Wizard 时，自定义 BSP 批处理文件也被克隆，而且可能无需修改。但是，如果在 BSP 中新增或删除驱动程序，则需要修改 BSP 批处理文件。如果 MCU 衍生产品头文件更改，也需要修改 PSP 批处理文件。批处理文件位于以下位置：

- BSP - `\mqx\build\bat\bsp_<Board_Name>.bat`
- PSP - `\mqx\build\bat\psp_<Board_Name>.bat`

11 CodeWarrior 调试器存储器文件

如果在 BSP 中使用 CodeWarrior 调试器，则 BSP 中会包含一些配置文件用于配置该调试器。TCL 文件用于初始化该 Kinetis 调试器的连接，不过这些文件无需修改。然而，.MEM 文件用于配置调试器的存储器映射，并且如果定制 BSP 中的 MCU 存储器映射与原先的 MCU 不一致的话，可能需要修改该文件。该文件位于 `\mqx\source\bsp`

\<Board_Name>\<toolchain>\dbg\<Board_Name>.mem。这是一个文本文件，易于手动修改。另一个选择是使用 CodeWarrior 提供的.MEM 文件，并使用位于<CodeWarrior_Installation_Path>\MCU\ARM_EABI_Support\Memory_Config_Files 的文件。

请注意，IAR 或 Keil 无需该步骤，因为它们在项目选项中选择衍生产品时进行配置。

12 移植示例应用程序

MQX 发布包中包含了示例项目，为测试定制 BSP 中的修改内容提供快捷途径。当使用 BSP Cloning Wizard 克隆 BSP 时，该工具还提供克隆示例项目的选项。该功能十分有用，因为该工具可以同时向新 BSP 移植所有的示例项目。通过编译这些示例以确保软件库和应用项目中不存在编译错误，并且这些项目可以在定制目标硬件上运行。

如果 BSP 中的 MCU 衍生产品发生了改变，则应更改应用示例项目中用于 MCU 衍生产品的工具链设置。大多数工具链需要项目指定用于调试和编程程序的具体 MCU。在 CodeWarrior 中，目标类型在用于调试配置的连接中更改。

13 结语

Freescale MQX RTOS 是一个可扩展的 RTOS，是对飞思卡尔的微控制器板系列（包括 Kinetis）的补充。利用本指南所列步骤，可将提供的 MQX BSP 移植到所有飞思卡尔支持的器件中。除了与 MQX 相关的文档，飞思卡尔还提供辅助移植流程的工具，包括 MQX BSP Cloning Wizard、Processor Expert 和 Solution Advisor 工具。有关最新的 MQX 信息和文档，或要下载 MQX 软件，请访问 freescale.com/mqx。

14 附录 A: 查找最接近的 MQX BSP

表格最后更新日期为 2013/5/11

使用该表格根据 Kinetis 部件编号查找最接近的 MQX BSP 进行克隆。更多详情, 请参见“选择最接近的 Freescale BSP 进行克隆”。在以下栏位中输入完整的 Kinetis 部件编号, 并点击查找按钮。请确保输入正确的部件编号, 无其他字符或空格。该表格使用 JavaScript, 可在最新版 Adobe Reader 软件上使用。请确保在 Adobe Reader 软件中启用 JavaScript。

输入完整的 Kinetis 部件编号:

最接近的 MQX BSP 电路板名称

How to Reach Us:

Home Page:
freescale.com

Web Support:
freescale.com/support

本文档中的信息仅供系统和软件实施方使用 Freescale 产品。本文并未明示或者暗示授予利用本文档信息进行设计或者加工集成电路的版权许可。Freescale 保留对此处任何产品进行更改的权利，恕不另行通知。

Freescale 对其产品在任何特定用途方面的适用性不做任何担保、表示或保证，也不承担因为应用程序或者使用产品或电路所产生的任何责任，明确拒绝承担包括但不限于后果性的或附带性的损害在内的所有责任。Freescale 的数据表和/或规格中所提供的“典型”参数在不同应用中可能并且确实不同，实际性能会随时间而有所变化。所有运行参数，包括“经典值”在内，必须经由客户的技术专家对每个客户的应用程序进行验证。Freescale 未转让与其专利权及其他权利相关的许可。Freescale 销售产品时遵循以下网址中包含的标准销售条款和条件：freescale.com/SalesTermsandConditions。

Freescale, the Freescale logo, and Kinetis, are trademarks of Freescale Semiconductor, Inc., Reg. U.S. Pat. & Tm. Off. All other product or service names are the property of their respective owners.

© 2008-2014 Freescale Semiconductor, Inc.

© 2008-2014 飞思卡尔半导体有限公司

