AN13970

在 Cadence Tensilica HiFi 4 DSP 上运行 Zephyr RTOS

Rev. 2 — 2023年11月28日 应用笔记

文档信息

信息	内容
关键词	AN13970, i.MX 8M Plus, HiFi 4 DSP, Zephyr, IPC, OpenAMP, remoteproc
摘要	本文介绍了如何在 HiFi 4 DSP 上启动示例应用、HiFi 4 DSP 与主核如何通信，以及如何获取示例应用的输出。

1 介绍

业内广泛讨论如何在Arm Cortex-A或Cortex-M核上运行Zephyr，并且已经产生了许多实例。然而，许多基于Cortex的MCU和MPU也选择集成片上DSP来分流计算密集型任务。

Cadence Tensilica HiFi 4 DSP 就是这类高性能嵌入式数字信号处理器（DSP）的示例之一，它针对音频、语音或神经网络处理进行了优化。本应用笔记重点介绍了：如何在主 Cortex-A 核上运行 Linux 操作系统的同时，在 DSP 上运行 Zephyr 实时操作系统，从而利用 DSP 的强大处理能力。

本文档包含一个简单的示例应用 “hello_world”，但也可以使用其他示例。本文档将解释如下问题：

- 如何在 HiFi 4 DSP 上启动示例应用
- HiFi 4 DSP 和主核如何通信
- 如何获取示例应用的输出

在本文档中，所有示例均使用 Linux 操作系统和 Zephyr 实时操作系统的现有驱动程序和/或框架进行说明。

2 硬件平台

i.MX 8M Plus EVK 板基于NXP i.MX 8M Plus 应用处理器，由以下部分组成：

- 四个 Arm Cortex-A53， 频率高达 1.8 GHz
- 一个 Arm Cortex-M7， 频率高达 800 MHz
- Cadence Tensilica HiFi 4 DSP， 频率高达 800 MHz

图 1 显示了i.MX 8M Plus EVK板的俯视图。

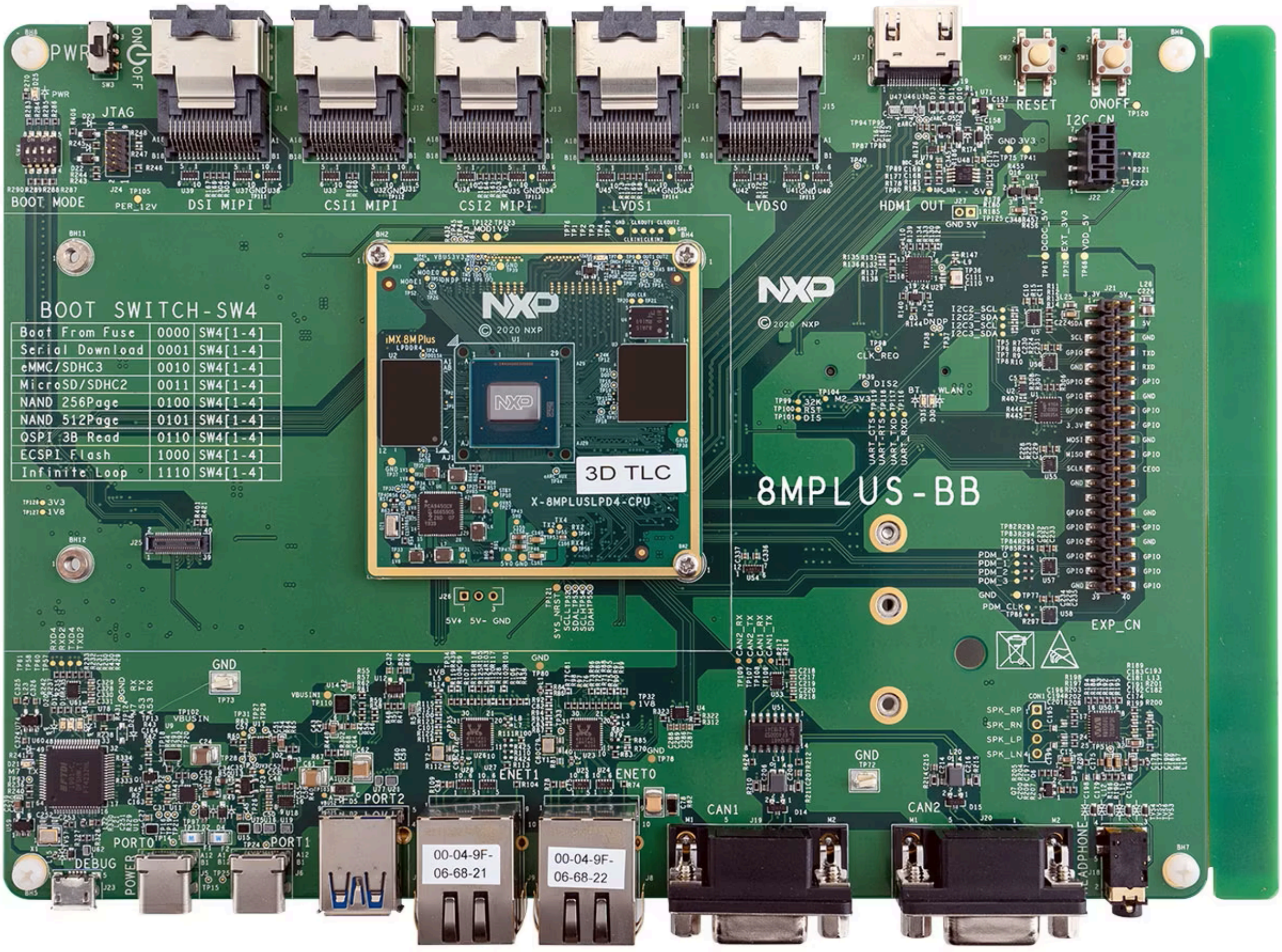


图 **1. i.MX 8M Plus EVK** 板俯视图

如需了解详情，请参阅 i.MX 8M Plus EVK 上的电路板信息。

3 Zephyr 操作系统

Zephyr Project 是一个可扩展的实时操作系统（RTOS），支持多种硬件架构，针对资源受限的器件进行了优化，并在构建时考虑了安全性。它基于小内核，专为在资源受限的系统上使用而设计。

NXP 提供 Zephyr 操作系统支持的各种评估和原型平台。开发人员能够使用包含硬件、开发工具以及传感器和器件驱动程序的真实开源项目，轻松定制符合自身需求的解决方案。Zephyr 操作系统的安全增强功能使设备管理、连接栈和文件系统的实施变得简单。

有关Zephyr RTOS的更多信息，请访问 www.zephyrproject.org/。

4 HiFi 四音频数字信号处理器 DSP

HiFi 四音频引擎是一款高度优化的音频处理器，可实现音频和语音编解码器以及前处理和后处理模块的高效执行。

在 Zephyr 中，支持 i.MX 8M Plus 音频 DSP 的板是 nxp_adsp_imx8m。

4.1 支持的功能

Zephyr nxp_adsp_imx8m 板的配置支持 表 1 中的硬件功能。

表 1. 支持的硬件功能

接口	控制器	驱动程序/组件
SYSTICK	On-chip	systick
CLOCK	On-chip	clock_control
PINMUX	On-chip	pinmux
UART	On-chip	serial port-polling

注：该端口目前不支持其他硬件功能。

默认配置可在 defconfig 文件中找到：boards/xtensa/nxp_adsp_imx8m/nxp_adsp_imx8m_defconfig。

4.2 连接和 I/Os

i.MX 8M Plus EVK 板使用以下pinmux控制器配置进行测试，如 表 2 所示。

表 2. 连接

电路板名称	SoC 名称	用途
UART4 RXD	UART4_TXD	UART 控制台
UART4 TXD	UART4_RXD	UART 控制台

4.3 系统时钟

HiFi 4 DSP 内核配置为以 800 MHz 时钟速度运行。.

4.4 串行端口

i.MX 8M Plus SoC 有四个 UART。仅 UART_4 配置给了 DSP 控制台，其余 UART 没有被使用/测试.

5 在 HiFi 4 DSP 上构建和运行 Zephyr 示例

本节介绍在 HiFi 4 DSP上构建和运行 Zephyr 示例的步骤：

- 章节 5.1
- 章节 5.2

5.1 hello_world 应用

Zephyr's hello_world 应用是一个简单的示例，支持用于多种 Supported boards，并在控制台中打印 “Hello World” 。

5.1.1 在 DSP 上加载 hello_world 应用

要在 DSP 上加载应用，请使用 Linux remoteproc 驱动程序。

在 Linux 中，可以使用 i.MX remoteproc 驱动程序和 DSP 特定驱动程序（imx_dsp_rproc）。

由于应用是在 DSP 上运行，因此要使用 imx_dsp_rproc 驱动程序。为此，请启用 Linux内核中的 `CONFIG_IMX_DSP_REMOTEPROC`。

5.1.2 编译 hello_world application

在 Zephyr 中为 i.MX 8M Plus DSP（即 Zephyr 中的 nxp_adsp_imx8m 板）编译 hello_world 应用。

进入 zephyrproject 中的 zephyr/ 文件夹并运行：

```
~/zephyrproject/zephyr$ west build -p always -b nxp_adsp_imx8m samples/
hello_world/
~/zephyrproject/zephyr$
~/zephyrproject/zephyr$ ls -la build/zephyr
total 4288
drwxr-xr-x 14 user nxp   4096 Oct 17 17:05 .
drwxr-xr-x  7 user nxp   4096 Oct 17 17:05 ..
drwxr-xr-x  5 user nxp   4096 Oct 17 17:05 arch
drwxr-xr-x  3 user nxp   4096 Oct 17 17:05 boards
drwxr-xr-x  5 user nxp   4096 Oct 17 17:05 cmake
-rw-r--r--  1 user nxp     64 Oct 17 17:05 .cmake.dotconfig.checksum
drwxr-xr-x  6 user nxp   4096 Oct 17 17:05 CMakeFiles
-rw-r--r--  1 user nxp  12355 Oct 17 17:05 cmake_install.cmake
-rw-r--r--  1 user nxp  39648 Oct 17 17:05 .config
...
-rw-r--r--  1 user nxp   2275 Oct 17 17:05 zephyr.dts
-rw-r--r--  1 user nxp    619 Oct 17 17:05 zephyr.dts.d
-rw-r--r--  1 user nxp 124460 Oct 17 17:05 zephyr.dts.pre
-rwxr-xr-x  1 user nxp 715896 Oct 17 17:05 zephyr.elf
-rw-r--r--  1 user nxp 408374 Oct 17 17:05 zephyr_final.map
-rw-r--r--  1 user nxp 408374 Oct 17 17:05 zephyr.map
-rwxr-xr-x  1 user nxp 717052 Oct 17 17:05 zephyr_pre0.elf
-rw-r--r--  1 user nxp 408886 Oct 17 17:05 zephyr_pre0.map
-rw-r--r--  1 user nxp   7273 Oct 17 17:05 zephyr.stat
```

`zephyr.elf` 文件将作为固件加载到 DSP 上。

5.1.3 在 DSP 上运行 hello_world 应用的步骤

以下是在 i.MX 8M Plus的HiFi 4 DSP 上运行 hello_world 应用的步骤。

5.1.3.1 启动 i.MX 8M Plus EVK 板

使用特定 DTS 启动 i.MX 8M Plus EVK 板。

使用 `imx8mp-evk-dsp.dtb`， 插入 `imx_dsp_rproc.ko` 内核模块后，您就可以看到如下脚本：

```
root@imx8mpevk:~# ls -la /sys/class/remoteproc/
total 0
drwxr-xr-x  2 root root 0 Mar  3 09:49 .
drwxr-xr-x 90 root root 0 Mar  3 09:49 ..
lrwxrwxrwx  1 root root 0 Mar  3 09:54 remoteproc0 -> ../../devices/
platform/3b6e8000.dsp/remoteproc/remoteproc0
root@imx8mpevk:~# cat /sys/class/remoteproc/remoteproc0/firmware
imx/dsp/hifi4.bin
root@imx8mpevk:~#
```

由于 remoteproc1 用于 DSP，因此使用此模块。

5.1.3.2 检查电路板上的固件

检查电路板上的固件镜像：

```
root@imx8mpevk:~# ls -la /lib/firmware/imx/zephyr/
total 1256
drwxr-xr-x  2 root root   4096 Mar  3 11:03 .
drwxr-xr-x 12 root root   4096 Mar  9  2018 ..
-rwxr-xr-x  1 root root  41520 Mar  9  2018 imx8-hello_world-zephyr.elf
-rwxr-xr-x  1 root root  57100 Mar  9  2018 imx8m-hello_world-zephyr.elf
-rwxr-x---  1 root root 996276 Mar  3 10:38 imx8m-openamp_rsc_table-zephyr.elf
-rwxr-xr-x  1 root root  87876 Mar  9  2018 imx8m-philosophers-zephyr.elf
-rwxr-xr-x  1 root root  58124 Mar  9  2018 imx8m-synchronization-zephyr.elf
-rwxr-xr-x  1 root root  41520 Mar  9  2018 imx8x-hello_world-zephyr.elf
root@imx8mpevk:~#
```

在检测 remoteproc 驱动程序之前，固件必须存在于 /lib/firmware 中，但也可以使用绝对路径。

5.1.3.3 插入 `imx_dsp_rproc.ko` 内核模块

默认情况下，i.MX DSP remoteproc 协议等待远程处理器的 READY 回复。由于不是所有Zephyr示例应用 （特别是不使用mailbox的简单应用）会发送 READY 回复，因此必须使用 remoteproc 模块，无需等待回复。这需要使用 no_maiboxes 内核模块参数来实现：

```
root@imx8mpevk:~# modinfo imx_dsp_rproc
filename:       /lib/modules/6.1.55-02981-g63bd8fa873a2/kernel/drivers/
remoteproc/imx_dsp_rproc.ko
author:         Shengjiu Wang <shengjiu.wang@nxp.com>
description:    i.MX HiFi Core Remote Processor Control Driver
license:        GPL v2
...
depends:
intree:         Y
name:           imx_dsp_rproc
parm:           no_mailboxes:There is no mailbox between cores, so ignore remote
 proc reply after start, default is 0 (off). (int)
root@imx8mpevk:~#
```

默认情况下，`no_maiboxes` 参数是 off，即不忽略来自 rproc 的回复。

因此，首先检查 `imx_dsp_rproc` 参数。如果是 off，请删除模块，然后插入正确参数的模块。

```
root@imx8mpevk:~# grep -H '' /sys/module/imx_dsp_rproc/parameters/* /
*no_mailboxes param is off */
/sys/module/imx_dsp_rproc/parameters/no_mailboxes:0
root@imx8mpevk:~#
root@imx8mpevk:~# rmmod imx_dsp_rproc /* remove kernel module */
[ 797.922929] remoteproc remoteproc0: releasing imx-dsp-rproc
root@imx8mpevk:~#
root@imx8mpevk:~# modprobe imx_dsp_rproc no_mailboxes=1 /* insert kernel module
 with the right parameter */
[ 819.930792] remoteproc remoteproc0: imx-dsp-rproc is available
root@imx8mpevk:~#
root@imx8mpevk:~# ls -la /sys/class/remoteproc/ /* now, we have remoteproc0, for
 DSP */
```

```
total 0
drwxr-xr-x  2 root root 0 Mar  3 09:49 .
drwxr-xr-x 90 root root 0 Mar  3 09:49 ..
lrwxrwxrwx  1 root root 0 Mar  3 10:20 remoteproc0 -> ../../devices/
platform/3b6e8000.dsp/remoteproc/remoteproc0
root@imx8mpevk:~#
```

5.1.3.4 在 DSP 上加载并运行固件

要在 DSP 上加载固件并运行，请执行以下命令：

```
root@imx8mpevk:~# echo -n /lib/firmware/imx/zephyr/imx8m-hello-world-zephyr.elf
 > /sys/class/remoteproc/remoteproc0/firmware
root@imx8mpevk:~# echo start > /sys/class/remoteproc/remoteproc0/state
[  107.320099] remoteproc remoteproc0: powering up imx-dsp-rproc
[  107.326031] remoteproc remoteproc0: Direct firmware load for /lib/firmware/
imx/zephyr/imx8m-hello-world-zephyr.elf failed with error -2
[  107.336696] remoteproc remoteproc0: Falling back to sysfs fallback for: /lib/
firmware/imx/zephyr/imx8m-hello-world-zephyr.elf
[  107.348365] remoteproc remoteproc0: Booting fw image /lib/firmware/imx/
zephyr/imx8m-hello-world-zephyr.elf, size 715896
[  107.360096] remoteproc remoteproc0: no resource table found for this firmware
[  107.367735] remoteproc remoteproc0: remote processor imx-dsp-rproc is now up
root@imx8mpevk:~#
```

5.1.3.5 停止固件

要停止固件，请执行以下命令：

```
root@imx8mpevk:~# echo stop > /sys/class/remoteproc/remoteproc0/state
[  206.148281] remoteproc remoteproc0: stopped remote processor imx-dsp-rproc
root@imx8mpevk:~#
```

5.1.4 获取 hello_world 应用输出

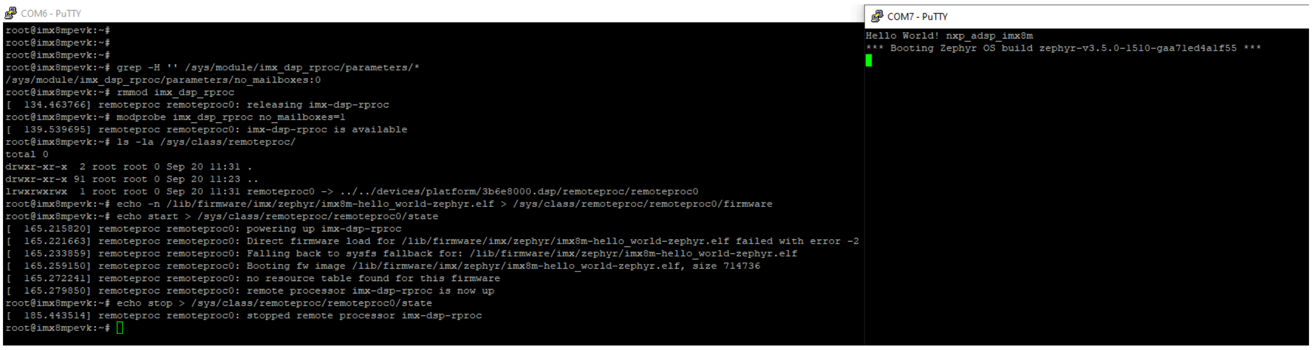
为了获取 hello_world 应用输出，请遵循以下步骤：

1. 通过 UART 获取控制台。
2. 在第四串口上打开串口终端：

   ```
   user@developerpc:~# minicom -D /dev/ttyUSB3
   ```

   您将在终端上看到以下信息（也可在 图 2 中查询）：

   ```
   Hello World! nxp_adsp_imx8m
   *** Booting Zephyr OS build zephyr-v3.5.0-1510-gaa71ed4a1f55 ***
   ```



```
root@imx8mpevk:~#
root@imx8mpevk:~#
root@imx8mpevk:~#
root@imx8mpevk:~# grep -H '' /sys/module/imx_dsp_rproc/parameters/*
/sys/module/imx_dsp_rproc/parameters/no_mailboxes:0
root@imx8mpevk:~# rmmod imx_dsp_rproc
[  134.463766] remoteproc remoteproc0: releasing imx-dsp-rproc
root@imx8mpevk:~# modprobe imx_dsp_rproc no_mailboxes=1
[  139.539695] remoteproc remoteproc0: imx-dsp-rproc is available
root@imx8mpevk:~# ls -la /sys/class/remoteproc/
total 0
drwxr-xr-x  2 root root 0 Sep 20 11:31 .
drwxr-xr-x 91 root root 0 Sep 20 11:23 ..
lrwxrwxrwx  1 root root 0 Sep 20 11:31 remoteproc0 -> ../../devices/platform/3b6e8000.dsp/remoteproc/remoteproc0
root@imx8mpevk:~# echo -n /lib/firmware/imx/zephyr/imx8m-hello_world-zephyr.elf > /sys/class/remoteproc/remoteproc0/firmware
root@imx8mpevk:~# echo start > /sys/class/remoteproc/remoteproc0/state
[  165.215820] remoteproc remoteproc0: powering up imx-dsp-rproc
[  165.221663] remoteproc remoteproc0: Direct firmware load for /lib/firmware/imx/zephyr/imx8m-hello_world-zephyr.elf failed with error -2
[  165.233859] remoteproc remoteproc0: Falling back to sysfs fallback for: /lib/firmware/imx/zephyr/imx8m-hello_world-zephyr.elf
[  165.259150] remoteproc remoteproc0: Booting fw image /lib/firmware/imx/zephyr/imx8m-hello_world-zephyr.elf, size 714736
[  165.272241] remoteproc remoteproc0: no resource table found for this firmware
[  165.279850] remoteproc remoteproc0: remote processor imx-dsp-rproc is now up
root@imx8mpevk:~# echo stop > /sys/class/remoteproc/remoteproc0/state
[  185.443514] remoteproc remoteproc0: stopped remote processor imx-dsp-rproc
root@imx8mpevk:~#
```

```
Hello World! nxp_adsp_imx8m
*** Booting Zephyr OS build zephyr-v3.5.0-1510-gaa71ed4a1f55 ***
```

图 2. **hello_world** 应用输出

您可以使用上述步骤来构建和测试任何其他示例，例如 synchronization 或 philosophers。

您还可以运行更复杂的示例来演示 Linux 和 Zephyr 如何协同工作。下一节举例说明了 openamp_rsc_table 应用程序。

5.2 openamp_rsc_table 应用

Zephyr's openamp_rsc_table 应用演示了如何在 Zephyr 中基于资源表使用开放式非对称多处理（OpenAMP）。它旨在响应：

- Linux rpmsg client sample
- Linux rpmsg tty driver

此示例实现与在主处理器上嵌入 Linux 内核操作系统和在协处理器上嵌入 Zephyr 应用程序的平台兼容。

5.2.1 在 DSP 上加载 openamp_rsc_table 应用

如 章节 5.1.1 中所描述，要在 DSP 上加载 openamp_rsc_table 应用，请在 Linux 内核中启用 `CONFIG_IMX_DSP_REMOTEPROC` 后使用 imx_dsp_rproc 驱动程序。

5.2.2 在 Zephyr 中编译 openamp_rsc_table 应用

在 Zephyr 中为 i.MX 8M Plus DSP 编译 openamp_rsc_table 应用意味着为 nxp_adsp_imx8m 板编译应用。

进入 `zephyrproject` 中的 `zephyr/` 文件夹并运行：

```
~/zephyrproject/zephyr$ west build -p always -b nxp_adsp_imx8m samples/subsys/
ipc/openamp_rsc_table
~/zephyrproject/zephyr$
~/zephyrproject/zephyr$ ls -la build/zephyr total 5284
drwxr-xr-x 14 nxa06898 nxp   4096 Sep 27 17:42 .
drwxr-xr-x  7 nxa06898 nxp   4096 Sep 27 17:42 ..
drwxr-xr-x  5 nxa06898 nxp   4096 Sep 27 17:42 arch
drwxr-xr-x  3 nxa06898 nxp   4096 Sep 27 17:42 boards
drwxr-xr-x  5 nxa06898 nxp   4096 Sep 27 17:42 cmake
-rw-r--r--  1 nxa06898 nxp     96 Sep 27 17:42 .cmake.dotconfig.checksum
drwxr-xr-x  6 nxa06898 nxp   4096 Sep 27 17:42 CMakeFiles
-rw-r--r--  1 nxa06898 nxp  13684 Sep 27 17:42 cmake_install.cmake
-rw-r--r--  1 nxa06898 nxp  41787 Sep 27 17:42 .config
drwxr-xr-x 16 nxa06898 nxp   4096 Sep 27 17:42 drivers
...
drwxr-xr-x  4 nxa06898 nxp   4096 Sep 27 17:42 soc
```

```
drwxr-xr-x 23 nxa06898 nxp   4096 Sep 27 17:42 subsys
-rw-r--r--  1 nxa06898 nxp   2421 Sep 27 17:42 zephyr.dts
-rw-r--r--  1 nxa06898 nxp    730 Sep 27 17:42 zephyr.dts.d
-rw-r--r--  1 nxa06898 nxp 124812 Sep 27 17:42 zephyr.dts.pre
-rw-r--r--  1 nxa06898 nxp 550701 Sep 27 17:42 zephyr_final.map
-rwxr-xr-x  1 nxa06898 nxp 998304 Sep 27 17:42 zephyr_openamp_rsc_table.elf
-rw-r--r--  1 nxa06898 nxp 550701 Sep 27 17:42 zephyr_openamp_rsc_table.map
-rw-r--r--  1 nxa06898 nxp   7463 Sep 27 17:42 zephyr_openamp_rsc_table.stat
-rwxr-xr-x  1 nxa06898 nxp 998476 Sep 27 17:42 zephyr_pre0.elf
-rw-r--r--  1 nxa06898 nxp 551293 Sep 27 17:42 zephyr_pre0.map
...
```

The `zephyr_openamp_rsc_table.elf` file is used as the firmware to be loaded on the DSP.

5.2.3 在 Linux 中的 DSP 上运行 openamp_rsc_table 应用

以下小节介绍了如何在 Linux 中的 i.MX 8M Plus 板上的 HiFi 4 DSP 运行 `openamp_rsc_table` 应用。

`rpmsg_client_sample.ko` 和 `rpmsg_tty.ko` 模块用于与 DSP 上运行的 `openamp_rsc_table` 应用进行通信。这些是在主处理器（Cortex A 内核）上运行的示例模块。

要构建 `rpmsg_client_sample.ko` 和 `rpmsg_tty.ko` 模块，请分别在 Linux 内核中启用 `CONFIG_SAMPLE_RPMSG_CLIENT` and `CONFIG_RPMSG_TTY` 配置。

5.2.3.1 启动 i.MX 8M Plus EVK 板

使用特定的 DTS 启动 i.MX 8M Plus EVK 板。

使用 `imx8mp-evk-dsp.dtb`，插入 `imx_dsp_rproc.ko` 内核模块后得到：

```
root@imx8mpevk:~# insmod imx_dsp_rproc.ko
[  115.172960] remoteproc remoteproc0: imx-dsp-rproc is available
root@imx8mpevk:~# ls -la /sys/class/remoteproc/
total 0
drwxr-xr-x  2 root root 0 Mar  3 09:49 .
drwxr-xr-x 90 root root 0 Mar  3 09:49 ..
lrwxrwxrwx  1 root root 0 Mar  3 20:09 remoteproc0 -> ../../devices/
platform/3b6e8000.dsp/remoteproc/remoteproc0
```

这里使用的是用于DSP的 `remoteproc0`。

5.2.3.2 检查板上固件

如要检查板上的固件镜像，请使用以下命令：

```
root@imx8mpevk:~# ls -la /lib/firmware/imx/zephyr/
total 148
drwxr-xr-x  2 root root   4096 Mar  9  2018 .
drwxr-xr-x 11 root root   4096 Mar  9  2018 ..
-rwxr-xr-x  1 root root  41524 Mar  9  2018 imx8-hello-world-zephyr.elf
-rwxr-xr-x  1 root root  57100 Mar  9  2018 imx8m-hello-world-zephyr.elf
-rwxr-xr-x  1 root root  41524 Mar  9  2018 imx8x-hello-world-zephyr.elf
-rwxr-xr-x  1 root root 998304 Mar  9  2018 imx8m-openamp_rsc_table-zephyr.elf
```

在探测 remoteproc 驱动程序之前，固件必须存在于 `/lib/firmware` 中；然而，它也可以用绝对路径给出。

5.2.3.3 插入 `imx_dsp_rproc.ko` 内核模块

如要插入 `imx_dsp_rproc.ko` 内核模块，请使用以下命令：

```
root@imx8mpevk:~# modprobe imx_dsp_rproc
[  115.172960] remoteproc remoteproc0: imx-dsp-rproc is available
root@imx8mpevk:~# ls -la /sys/class/remoteproc/
total 0
drwxr-xr-x  2 root root 0 Mar  3 09:49 .
drwxr-xr-x 90 root root 0 Mar  3 09:49 ..
lrwxrwxrwx  1 root root 0 Mar  3 20:09 remoteproc0 -> ../../devices/
platform/3b6e8000.dsp/remoteproc/remoteproc0
```

5.2.3.4 插入 rpmsg Linux 客户端示例

如要插入 rpmsg Linux 客户端示例，请使用以下命令：

```
root@imx8mpevk:~# modprobe rpmsg_client_sample.ko     /* rpmsg client sample
 driver used to communicate with remote processor over the rpmsg bus */
root@imx8mpevk:~# modprobe rpmsg_tty.ko               /* export rpmsg endpoints
 as tty devices, usually found as /dev/ttyRPMSGx */
```

5.2.3.5 在 DSP 上加载并运行固件

5.2.3.5.1 rpmsg 客户端示例

从 Linux 向 Zephyr 发送 100 条带有 "hello world" 的消息。

Linux 控制台：

```
root@imx8mpevk:~# echo -n zephyr_openamp_rsc_table.elf > /sys/class/remoteproc/
remoteproc0/firmware
root@imx8mpevk:~# echo start > /sys/class/remoteproc/remoteproc0/state
[  200.630824] remoteproc remoteproc0: powering up imx-dsp-rproc
[  200.637393] remoteproc remoteproc0: Booting fw image
 zephyr_openamp_rsc_table.elf, size 999412
[  200.649895] rproc-virtio rproc-virtio.2.auto: assigned reserved memory node
 vdev0buffer@94300000
[  200.662289] virtio_rpmsg_bus virtio0: rpmsg host is online
[  200.667889] rproc-virtio rproc-virtio.2.auto: registered virtio0 (type 7)
[  200.674715] remoteproc remoteproc0: remote processor imx-dsp-rproc is now up
[  200.681908] virtio_rpmsg_bus virtio0: creating channel rpmsg-client-sample
 addr 0x400
[  200.689959] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: new
 channel: 0x400 -> 0x400!
[  200.700409] virtio_rpmsg_bus virtio0: creating channel rpmsg-tty addr 0x401
[  200.707894] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming
 msg 1 (src: 0x400)
[  200.717703] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming
 msg 2 (src: 0x400)
[  200.726580] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming
 msg 3 (src: 0x400)
[  200.735433] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming
 msg 4 (src: 0x400)
[  200.744289] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming
 msg 5 (src: 0x400)
```

```
[  200.753158] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming
 msg 6 (src: 0x400)
[  200.761988] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming
 msg 7 (src: 0x400)
[  200.770827] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming
 msg 8 (src: 0x400)
[  200.779680] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming
 msg 9 (src: 0x400)
[  200.788511] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming
 msg 10 (src: 0x400)
...
[  201.529273] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming
 msg 93 (src: 0x400)
[  201.538195] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming
 msg 94 (src: 0x400)
[  201.547120] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming
 msg 95 (src: 0x400)
[  201.556048] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming
 msg 96 (src: 0x400)
[  201.564975] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming
 msg 97 (src: 0x400)
[  201.573901] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming
 msg 98 (src: 0x400)
[  201.582816] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming
 msg 99 (src: 0x400)
[  201.591742] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming
 msg 100 (src: 0x400)
[  201.600716] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: goodbye!
[  201.607877] virtio_rpmsg_bus virtio0: destroying channel rpmsg-client-sample
 addr 0x400
[  201.615953] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: rpmsg
 sample client driver is removed
```

Zephyr 控制台：

```
*** Booting Zephyr OS build zephyr-v3.4.0-4490-gd885048637d6 ***
Starting application threads!

OpenAMP[remote]  linux responder demo started

OpenAMP[remote] Linux sample client responder started

[00:00:00.015,000] <dbg> openamp_rsc_table: platform_ipm_callback:
 platform_ipm_callback: msg received from mb 0

[00:00:00.020,000] <dbg> openamp_rsc_table: mailbox_notify: mailbox_notify: msg
 received

[00:00:00.024,000] <dbg> openamp_rsc_table: mailbox_notify: mailbox_notify: msg
 received

[00:00:00.053,000] <dbg> openamp_rsc_table: platform_ipm_callback:
 platform_ipm_callback: msg received from mb 0

[00:00:00.053,000] <dbg> openamp_rsc_table: mailbox_notify: mailbox_notify: msg
 received

[00:00:00.070,000] <dbg> openamp_rsc_table: platform_ipm_callback:
 platform_ipm_callback: msg received from mb 0
```

```
[00:00:00.070,000] <dbg> openamp_rsc_table: mailbox_notify: mailbox_notify: msg
 received

[00:00:00.079,000] <dbg> openamp_rsc_table: platform_ipm_callback:
 platform_ipm_callback: msg received from mb 0

[00:00:00.079,000] <dbg> openamp_rsc_table: mailbox_notify: mailbox_notify: msg
 received

[00:00:00.088,000] <dbg> openamp_rsc_table: platform_ipm_callback:
 platform_ipm_callback: msg received from mb 0

[00:00:00.088,000] <dbg> openamp_rsc_table: mailbox_notify: mailbox_notify: msg
 received

[00:00:00.097,000] <dbg> openamp_rsc_table: platform_ipm_callback:
 platform_ipm_callback: msg received from mb 0

...
[00:00:00.935,000] <dbg> openamp_rsc_table: platform_ipm_callback:
 platform_ipm_callback: msg received from mb 0

[00:00:00.935,000] <dbg> openamp_rsc_table: mailbox_notify: mailbox_notify: msg
 received

[00:00:00.944,000] <dbg> openamp_rsc_table: platform_ipm_callback:
 platform_ipm_callback: msg received from mb 0

[00:00:00.944,000] <dbg> openamp_rsc_table: mailbox_notify: mailbox_notify: msg
 received

[00:00:00.944,000] <dbg> openamp_rsc_table: mailbox_notify: mailbox_notify: msg
 received
```

5.2.3.5.2 rpmsg TTY 演示

在 Linux 控制台上，向 Zephyr 发送信息，其使用 “TTY <add>” 前缀进行回复。<add> 为Zephyr rpmsg-tty 的地址。

Linux 控制台：

```
root@imx8mpevk:~# cat /dev/ttyRPMSG0 &
[1] 1540
root@imx8mpevk:~# echo "Hello Zephyr" >/dev/ttyRPMSG0
TTY 0x0401: Hello Zephyr
root@imx8mpevk:~#
```

Zephyr 控制台：

```
*** Booting Zephyr OS build zephyr-v3.4.0-4490-gd885048637d6 ***
Starting application threads!

OpenAMP[remote]  linux responder demo started

OpenAMP[remote] Linux tty responder started

[00:06:02.049,000] <dbg> openamp_rsc_table: platform_ipm_callback:
 platform_ipm_callback: msg received from mb 0
```

```
[00:06:02.049,000] <dbg> openamp_rsc_table: mailbox_notify: mailbox_notify: msg
 received
```

5.2.3.6 停止固件

要停止固件，请使用以下命令：

```
root@imx8mpevk:~# echo stop > /sys/class/remoteproc/remoteproc0/state
[  495.366531] remoteproc remoteproc0: stopped remote processor imx-dsp-rproc
```

5.2.4 获取 openamp_rsc_table 应用输出

要获取 openamp_rsc_table 应用输出，请执行以下步骤：

1. 通过UART获取控制台。
2. 在第四个串行端口上打开串行终端：

   ```
   user@developerpc:~# minicom -D /dev/ttyUSB3
   ```

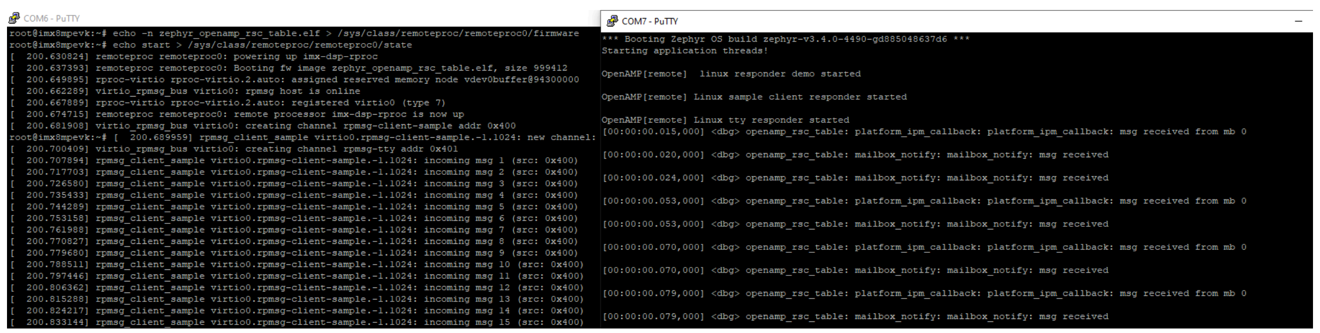
   您在终端中看到以下消息（也显示在 图 3）：

   ```
   *** Booting Zephyr OS build zephyr-v3.4.0-4490-gd885048637d6 ***
   Starting application threads!

   OpenAMP[remote]  linux responder demo started

   OpenAMP[remote] Linux sample client responder started

   OpenAMP[remote] Linux tty responder started
   [00:00:00.015,000] <dbg> openamp_rsc_table: platform_ipm_callback:
    platform_ipm_callback: msg received from mb 0
   ```



```
root@imx8mpevk:~# echo -n zephyr_openamp_rsc_table.elf > /sys/class/remoteproc/remoteproc0/firmware
root@imx8mpevk:~# echo start > /sys/class/remoteproc/remoteproc0/state
[  200.630824] remoteproc remoteproc0: powering up imx-dsp-rproc
[  200.637393] remoteproc remoteproc0: Booting fw image zephyr_openamp_rsc_table.elf, size 999412
[  200.649895] rproc-virtio rproc-virtio.2.auto: assigned reserved memory node vdev0buffer@94300000
[  200.662289] virtio_rpmsg_bus virtio0: rpmsg host is online
[  200.667889] rproc-virtio rproc-virtio.2.auto: registered virtio0 (type 7)
[  200.674715] remoteproc remoteproc0: remote processor imx-dsp-rproc is now up
[  200.681908] virtio_rpmsg_bus virtio0: creating channel rpmsg-client-sample addr 0x400
root@imx8mpevk:~# [  200.689959] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: new channel:
[  200.700409] virtio_rpmsg_bus virtio0: creating channel rpmsg-tty addr 0x401
[  200.707894] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming msg 1 (src: 0x400)
[  200.717703] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming msg 2 (src: 0x400)
[  200.726580] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming msg 3 (src: 0x400)
[  200.735433] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming msg 4 (src: 0x400)
[  200.744289] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming msg 5 (src: 0x400)
[  200.753158] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming msg 6 (src: 0x400)
[  200.761988] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming msg 7 (src: 0x400)
[  200.770827] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming msg 8 (src: 0x400)
[  200.779680] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming msg 9 (src: 0x400)
[  200.788511] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming msg 10 (src: 0x400)
[  200.797446] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming msg 11 (src: 0x400)
[  200.806362] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming msg 12 (src: 0x400)
[  200.815288] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming msg 13 (src: 0x400)
[  200.824217] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming msg 14 (src: 0x400)
[  200.833144] rpmsg_client_sample virtio0.rpmsg-client-sample.-1.1024: incoming msg 15 (src: 0x400)
```

```
*** Booting Zephyr OS build zephyr-v3.4.0-4490-gd885048637d6 ***
Starting application threads!

OpenAMP[remote]  linux responder demo started

OpenAMP[remote] Linux sample client responder started

OpenAMP[remote] Linux tty responder started
[00:00:00.015,000] <dbg> openamp_rsc_table: platform_ipm_callback: platform_ipm_callback: msg received from mb 0

[00:00:00.020,000] <dbg> openamp_rsc_table: mailbox_notify: mailbox_notify: msg received

[00:00:00.024,000] <dbg> openamp_rsc_table: mailbox_notify: mailbox_notify: msg received

[00:00:00.053,000] <dbg> openamp_rsc_table: platform_ipm_callback: platform_ipm_callback: msg received from mb 0

[00:00:00.053,000] <dbg> openamp_rsc_table: mailbox_notify: mailbox_notify: msg received

[00:00:00.070,000] <dbg> openamp_rsc_table: platform_ipm_callback: platform_ipm_callback: msg received from mb 0

[00:00:00.070,000] <dbg> openamp_rsc_table: mailbox_notify: mailbox_notify: msg received

[00:00:00.079,000] <dbg> openamp_rsc_table: platform_ipm_callback: platform_ipm_callback: msg received from mb 0

[00:00:00.079,000] <dbg> openamp_rsc_table: mailbox_notify: mailbox_notify: msg received
```

图 **3. openamp_rsc_table** 应用输出

6 缩略语

Table 3 列出了本文档中使用的缩略语。

Table 3. 缩略语

缩略语	全称
DSP	Digital signal processor

Table 3. 缩略语...continued

缩略语	全称
DTS	Device tree source
IPC	Inter-process communication
MCU	Microcontroller unit
MPU	Microprocessor unit
OpenAMP	Open Asymmetric Multi-Processing
OS	Operating system
rproc	Remote processor
rsc_table	Resource table
RTOS	Real-time operating system
UART	Universal Asynchronous Receiver/Transmitter

7 Note about the source code in the document

Example code shown in this document has the following copyright and BSD-3-Clause license:

Copyright 2023 NXP Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

8 修订记录

表 4 汇总了自初始版以来对本文档所做的更改。

表 4. 修订记录

版本号	日期	说明
2	2023 年 11 月 28 日	• 更新以下章节： – 章节 5.1.2 – 章节 5.1.3.1 – 章节 5.1.3.2

表 **4.** 修订记录...续上页

版本号	日期	说明
		– 章节 5.1.3.3 – 章节 5.1.3.4 – 章节 5.1.3.5 – 章节 5.1.4 • 增加新章节：章节 5.2
1	2023 年 6 月 1 日	初次发布

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at https://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this document expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP B.V. — NXP B.V. is not an operating company and it does not distribute or sell products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, μVision, Versatile — are trademarks and/or registered trademarks of Arm Limited (or its subsidiaries or affiliates) 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.

Cadence — the Cadence logo, and the other Cadence marks found at www.cadence.com/go/trademarks are trademarks or registered trademarks of Cadence Design Systems, Inc. All rights reserved worldwide.

i.MX — is a trademark of NXP B.V.

内容

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.

For more information, please visit: https://www.nxp.com