

AN13914

i.MX 8ULP功耗测量

第3版—2024年3月14日

应用笔记

文档信息

信息	内容
关键词	AN13914、i.MX 8ULP、MCIMX8ULP-EVK、i.MX 8ULP功率测量、i.MX 8ULP电源域
摘要	本应用笔记介绍了如何通过不同的用例在恩智浦i.MX 8ULP EVK板上测量恩智浦i.MX 8ULP应用处理器的电流消耗。



1 介绍

本应用笔记旨在帮助系统设计人员创建功率优化的系统。它介绍了如何通过不同的用例在恩智浦i.MX 8ULP EVK板上测量恩智浦i.MX 8ULP应用处理器的电流消耗。用户可以为i.MX 8ULP处理器选择合适的电源域，并了解各种场景下处理器的预期功耗。

注：测量数据基于小批量样本，因此，本文中呈现的结果并不保证适用于所有情况。

2 i.MX 8ULP的架构

图1所示为i.MX 8ULP处理器的高度架构框图。

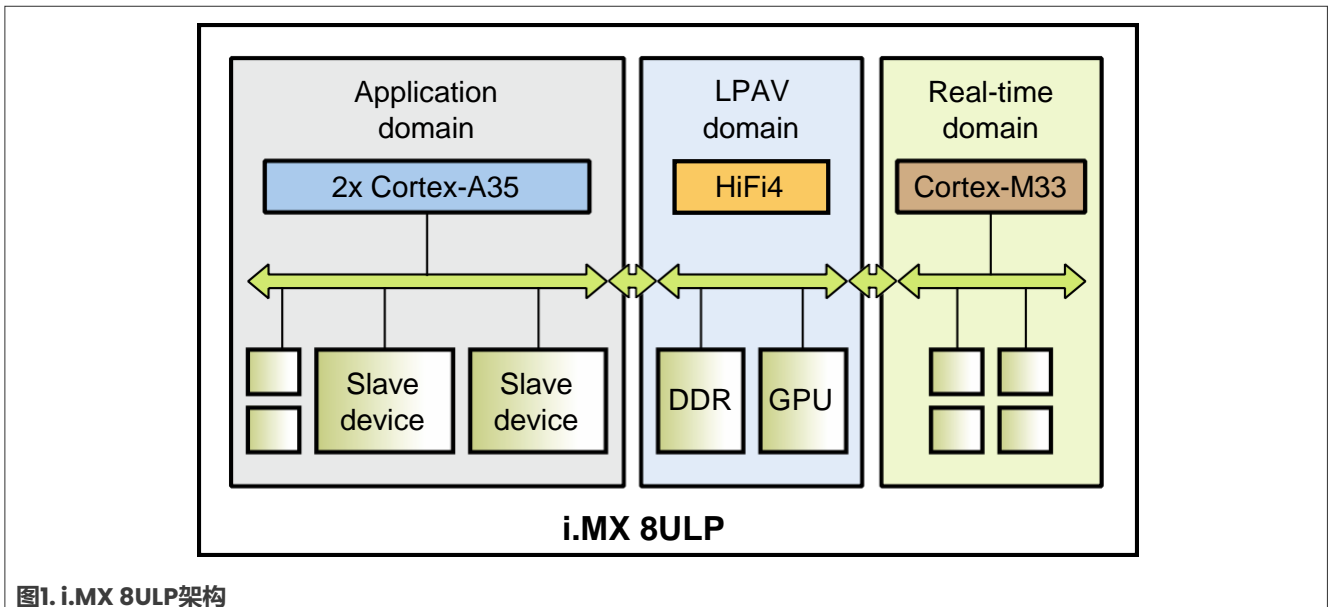


图1. i.MX 8ULP架构

i.MX 8ULP架构包含三个域：

- 实时处理器域（RTD）或Arm Cortex-M33域
- 应用处理器域（APD）或Arm Cortex-A35域
- 低功耗音频/视频（LPAV）域

注：有关i.MX 8ULP处理器的详细架构图，请参阅《i.MX 8ULP处理器参考手册》（iMX8ULPRM）。

i.MX 8ULP架构支持PMIC和非PMIC用例；从而有助于降低系统成本。当系统使用RT域时，可以关闭AP或LPAV域的电，以节省电力。

3 i.MX 8ULP电源概述

i.MX 8ULP处理器有若干电源域，每个电源域包含多个电源。[表1](#)列出了不同的i.MX 8ULP电源域以及每个电源域中的组件。

表1. i.MX 8ULP电源域

电源域	组件
实时处理器域 (RTD)	<ul style="list-style-type: none"> • Arm Cortex-M33平台 • Fusion F1 DSP • 安全子系统 • 电源管理 • 多个外设 • 系统级组件 • 三个GPIO端口：A、B和C
应用处理器域 (APD)	<ul style="list-style-type: none"> • 搭载1个或两个Arm Cortex-A35核的平台 • 多个外设 • 三个GPIO端口：D、E和F
低功耗音频/视频 (LPAV) 域	<ul style="list-style-type: none"> • HiFi4 DSP • 图形处理单元3D (GPU3D) • 图形处理单元2D (GPU2D) • LPDDR3/LPDDR4/LPDDR4x接口 • MIPI-DSI接口 • MIPI-CSI接口
DGO (不间断运行) 域	<ul style="list-style-type: none"> • 复位和系统模式控制逻辑 • 低功耗唤醒单元 (WUU) • 模拟比较器 • 低功耗定时器
VBAT域	<ul style="list-style-type: none"> • 实时时钟 (RTC) • 电池供电安全模块 (BBSM) • 电池支持的非安全模块 (BBNSM)

通常情况下，这些域是相互独立的。在实时处理器域和应用处理器域中，有多种电源模式可用，以根据应用在特定时间的需求优化功耗。这些模式通过降低时钟频率、降低电压、门控时钟和门控电源来优化功耗。

[图2](#)所示为i.MX 8ULP基础电源管理方案的用例，其中包括外部PMIC和旁路的CM33 LDO。此外，还展示了i.MX 8ULP电源的连接和内部电源域的分布。

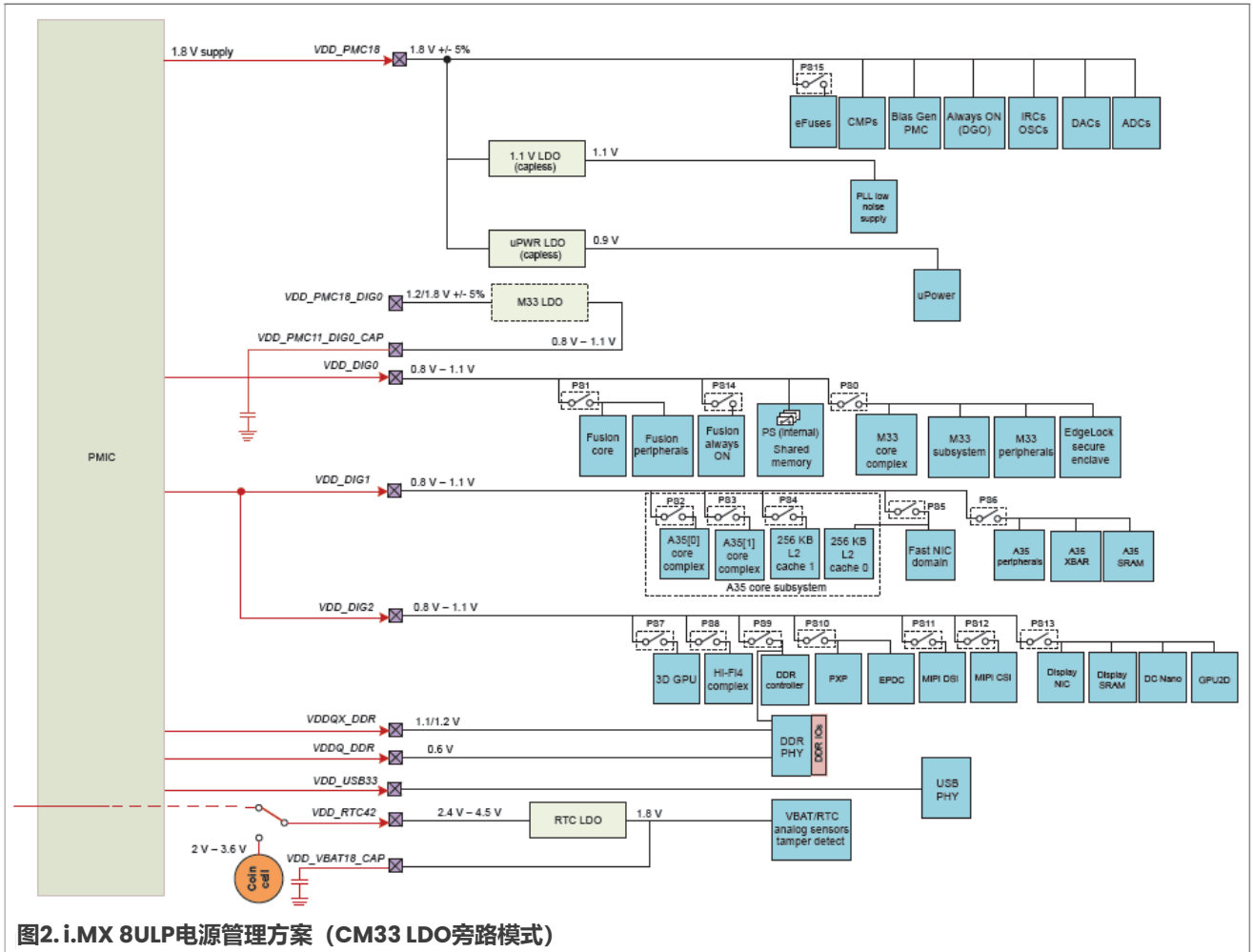


图2. i.MX 8ULP电源管理方案 (CM33 LDO旁路模式)

注:

- 有关i.MX 8ULP基础电源管理方案及其他用例的信息，请参阅《i.MX 8ULP处理器参考手册》(IMX8ULPRM)。
- 有关推荐工作条件和由每个I/O电压电源供电的引脚组的详细信息，请参阅《i.MX 8ULP数据手册》。

图3所示为MCIMX8ULP-EVK板使用的电源管理方案。在该电源管理方案中，外部PMIC为i.MX 8ULP处理器供电，其M33 LDO设置为旁路模式。

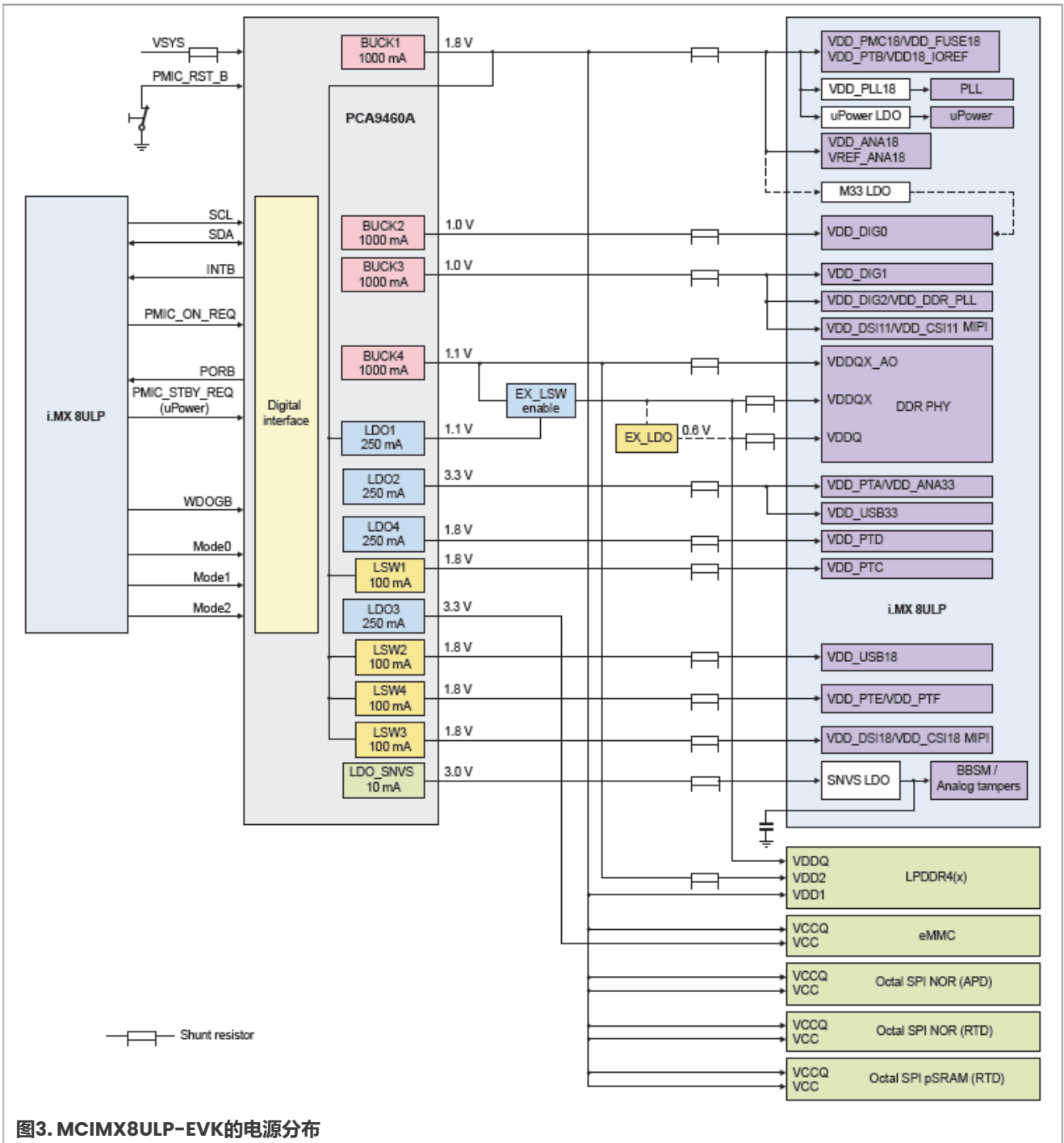


图3. MCIMX8ULP-EVK的电源分布

4 在MCIMX8ULP-EVK板上测量i.MX 8ULP的功率

本文详细介绍了恩智浦在MCIMX8ULP-EVK板上进行i.MX 8ULP功耗测量的几个用例。这些用例在[第5节](#)进行了描述。

测量主要针对[表2](#)中列出的电源域进行。该表还提供了BCU软件的电源轨与i.MX 8ULP处理器的电源域之间的映射关系。如需了解更多信息，请从<https://github.com/nxp-imx/bcu/releases>下载“BCU.pdf”。

表2. 测量的电源域

电源组	BCU中的电源轨	电源域	说明
GROUP_SOC_FULL	BUCK1_CPU_1V8	VDD_PMC18 + VDD_PLL18 + VDD_FUSE18 + VDD18_IOREF + VDD_PTB + VDD_ANA18 + VREF_ANA18	PMC、PLL、熔丝、B端口I/O接口电源和模拟电源
	BUCK1_LSW1_CPU_1V8	VDD_PTC	C端口I/O接口电源
	BUCK1_LSW2_CPU_1V8	VDD_USB_18	USB电源
	BUCK1_LSW3_CPU_1V8	VDD_DSII8 + VDD_CSII8	MIPI DSI/CSI I/O电源
	BUCK1_LSW4_CPU_1V8	VDD_PTE + VDD_PTF	E端口和F端口I/O接口电源
	BUCK2_CPU_1V0	VDD_DIG0	RTD核心电源
	BUCK3_CPU_1V0	VDD_DIG1 + VDD_DIG2 + VDD_DDR_PLL + VDD_DSIII + VDD_CSIII	APD核心电源、LPAV域核心电源、DRAM PHY PLL电源和DSI核心电源
	BUCK4_CPU_1V1	VDDQX_AO_DDR	不间断运行DRAM I/O接口电源
	LDO1_CPU_1V1	VDDQX_DDR	DRAM I/O接口预驱动器电源
	LDO1_CPU_1V1_0V6	VDDQ_DDR	DRAM I/O接口电源
	LDO2_CPU_3V3	VDD_PTA + VDD_ANA33 + VDD_USB_33	A端口I/O接口电源、模拟电源和USB I/O电源
	LDO4_CPU_1V8	VDD_PTD	D端口I/O接口电源
	LDO5_CPU_3V0	VDD_VBAT42	VBAT域电源
	GROUP_PLATFORM	VSYS_5V0_4V2	PMIC上的VSYS

本文中的低功耗模式测量适用于多个电源，主要测量每个有源域中的主要电源：

- 对于实时处理器域（RTD），主要电源为VDD_DIG0。主要电源取决于内部LDO配置：
 - 在CM33 LDO启用模式中，对VDD_PMC18_DIG0施加恒定电压，内部LDO在软件控制下从VDD_PMC11_DIG0_CAP向VDD_DIG0提供较低的电压。在这种情况下，LDO_EN应连接到VDD_PMC18。

- 在CM33 LDO旁路模式中，内部LDO被禁用，外部可变电压通常由外部电源管理IC (PMIC) 提供。在这种情况下，LDO_EN应连接到接地，VDD_PMC18_DIG0和VDD_PMC11_DIG0_CAP都应通过10kΩ电阻连接到接地。默认情况下，MCIMX8ULP-EVK板支持CM33 LDO旁路模式。
- 对于应用处理器域 (APD)，主要电源为VDD_DIG1。
- 对于低功耗音频/视频 (LPAV) 域，主要电源为VDD_DIG2，应在电路板级连接到VDD_DIG1。即使未使用，VDD_DSI11也必须在电路板级连接到VDD_DIG2。如果使用VDD_CSI11，则应在电路板级将其连接到VDD_DIG2。

以下小节介绍了在MCIMX8ULP-EVK板进行i.MX 8ULP功率测量时的多个注意事项。

4.1 DGO (不间断运行) 域电源

DGO域使用VDD_PMC18电源，该电源为以下电路供电：

- CMP0/I
- PMC
- LPTMR
- WUU0/I
- SYSOSC/LPOSC
- uPower
- 多种其他芯片级功能

4.2 实时处理器域 (RTD) 电源

实时处理器域使用以下电源：

- VDD_PMC18_DIG0：为CM33 LDO供电：
 - 在CM33 LDO启用模式下，LDO输出VDD_PMC11_DIG0_CAP连接到外部滤波电容。LDO输出回路到VDD_DIG0，为内部逻辑供电。有关退耦和大容量电容的要求，请参阅《i.MX 8ULP硬件开发指南》(IMX8ULPHDG)。
 - 在CM33 LDO旁路模式下，VDD_PMC18_DIG0和VDD_PMC11_DIG0_CAP都应通过10kΩ电阻连接到接地。
- VDD_PMC18：为以下电路供电：
 - uPower LDO
 - DGO (不间断运行) 域逻辑和电源管理控制器 (PMC)
 - PLL的低噪声LDO
- VDD_PTA：为A端口的I/O供电 (信号名为PTAn)。
- VDD_PTB：为B端口的I/O供电 (信号名为PTBn)。
- VDD_PTC：为C端口的I/O供电 (信号名为PTCn)。

VDD_PTA、VDD_PTB和VDD_PTC的功耗完全取决于应用，因此，本文中与此些电源相关的测量结果仅供参考。

有关可用于根据单个I/O信号的活动估计GPIO段功率的等式，请参阅i.MX 8ULP数据手册中的“绝对最大额定值”表。

4.3 应用处理器域 (APD) 电源

应用处理器域使用以下电源：

- VDD_DIG1：为CA35核应用处理器域逻辑供电。

- VDD_USB33和VDD_USB18：为USB PHY供电。即使不使用，VDD_USB33和VDD_USB18也必须通电。
- VDD_PTD：为D端口的I/O供电（名为PTD_n的信号）。
- VDD_PTE：为E端口的I/O供电（名为PTE_n的信号）。
- VDD_PTF：为F端口的I/O供电（名为PTF_n的信号）。

有关可用于根据单个I/O信号的活动估计GPIO段功率的等式，请参阅i.MX 8ULP数据手册中的“最大供电电流”表。

4.4 低功耗音频/视频域 (LPAV) 电源

LPAV域使用以下电源：

- VDD_DIG2：为低功耗音频/视频域逻辑和DRAM控制器供电。
- VDD_DSI18和VDD_DSI11：为MIPI-DSI接口供电。即使未使用，VDD_DSI11也必须在电路板级连接到VDD_DIG2。
- VDD_CSI18和VDD_CSI11：为MIPI-CSI接口供电。如果使用VDD_CSI11，则应在电路板级将其连接到VDD_DIG2。
- VDD_DDR_PLL、VDDQX_DDR、VDDQ_DDR和VDDQX_AO_DDR：为LPDDR3/LPDDR4/LPDDR4x PHY供电：
 - VDD_DDR_PLL在电路板级必须始终连接到VDD_DIG2。
 - VDDQ_DDR是LPDDR3/LPDDR4/LPDDR4x接口I/O的1.2V / 1.1V / 0.6V电源。
 - VDDQX_AO_DDR是LPDDR3/LPDDR4/LPDDR4x接口不间断运行I/O (DDR_CKE/DDR_RAM_RST_B) 的1.2V / 1.1V / 1.1V电源。

4.5 VBAT域电源

VDD_VBAT42向VBAT域供电。在大多数应用中，此电源由电池提供。内部LDO将输出调节为VBAT域的内部逻辑所使用的1.8V。VDD_VBAT18_CAP连接到外部电容。有关退耦和大容量电容的要求，请参阅《i.MX 8ULP硬件开发指南》(IMX8ULPHDG)。

4.6 模拟和其他电源

以下电源用于模拟和芯片级功能：

- VDD_ANA33：它是用于模拟功能的1.8V或3.3V电源。此电源的电压电平应与VDD_PTA的电压电平相匹配。
- VDD_ANA18：它是用于模拟功能的1.8V电源。VDD_ANA18应在电路板级连接到VDD_PMC18。
- VDD_PLL18：它是PLL模拟部分的1.8V电源。
- VDD_fuse18：它是熔丝IP的1.8V电源。VDD_fuse18应在电路板级连接到VDD_PMC18。
- VREFH_ANA18：它是ADC范围的高端1.8V参考电压。
- VDD18_IOREF_1和VDD18_IOREF_2：这两个电源均为I/O使用的1.8V参考电源。

4.7 测量过程中如何使用电压电平和VFS

除非另有规定，否则每个电源的电压电平均设置为《i.MX 8ULP硬件开发指南》(IMX8ULPHDG)中定义的典型电压电平。

为了最大限度地减少每种电源模式下的功耗，可以在运行用例时通过修改VDD_DIG0、VDD_DIG1和VDD_DIG2来执行电压和频率缩放（VFS）。有关每种电源模式的电压规格，请参阅i.MX 8ULP数据手册中的“建议工作条件”表。

4.8 温度

除非另有规定，否则本文中描述的功率测量在室温（约25°C）下进行。

4.9 硬件和软件要求

表3提供了功率测量过程中使用的硬件和软件的详细信息。

表3. 使用的硬件和软件

类别	说明
硬件	恩智浦MCIMX8ULP-EVK, 76882_A6 + 76883_B3带A2芯片, 部件编号为PIMX8UD5 CVP08SC
系统软件	Linux BSP 6.1.55_2.2.0
应用程序软件	SDK 2.14.1
	BCU 1.1.68。有关BCU的详细信息，请访问 https://github.com/nxp-imx/bcu 。

4.10 构建i.MX Yocto Project

要构建i.MX Yocto Project，请执行以下步骤：

1. 下载并构建i.MX Yocto Project社区BSP配方层：

注：有关i.MX Yocto Project的更多信息，请参阅《[i.MX Yocto Project用户指南](#)》。

```
repo init -u https://github.com/nxp-imx/imx-manifest -b imx-linux-mickledore
-m imx-6.1.55-2.2.0.xml
repo sync
DISTRO=fsl-imx-xwayland MACHINE=imx8ulp-lpddr4-evk source imx-setup-
release.sh -b build-imx8ulp-evk
```

2. 对于某些音频或视频用例，需要gstreamer1.0-libav。将以下命令放在build-imx8ulp-evk/conf/local.conf文件的末尾：

```
LICENSE_FLAGS_ACCEPTED += "commercial"
IMAGE_INSTALL:append = "gstreamer1.0-libav"
PACKAGECONFIG:append_pn-gstreamer1.0-libav = "x264"
```

3. 构建镜像：

```
bitbake imx-image-full
```

构建的镜像可以在build-imx8ulp-evk/tmp/deploy/images中找到。

4.11 功耗测量步骤

以下是在MCIMX8ULP-EVK板上测量i.MX 8ULP功耗所需的步骤：

1. 用一根微型USB电缆连接主机和MCIMX8ULP-EVK板上的USB micro-B端口J17。
2. 在BCU路径中启动监测器：

```
bcu.exe monitor -board=imx8ulpevkb2
```

3. 按照第5节所述，运行相关的用例。
4. 启动用例后，按“3”重置该值。
5. 可选择按“4”更改测量精度：mA/auto/ μ A。
6. 等待1分钟，并记录BCU中的测量数据。

4.12 软件包

与i.MX 8ULP功率测量相关的一些用例二进制文件可以在以下软件包中找到：

<https://www.nxp.com/docs/en/application-note-software/AN13914SW.zip>

5 用例和测量结果

以下小节介绍了在i.MX 8ULP平台上进行i.MX 8ULP内部功率测量基准测试的主要用例和子用例。

注：

- 在运行用例之前，必须运行所需的配置脚本来配置环境。有关更多详细信息，请参阅第8节。
- 所有用例的启动模式均为Single Boot (eMMC) 模式。

表4汇总了在MCIMX8ULP-EVK板上执行的多种用例的功率测量结果。

表4. MCIMX8ULP-EVK功率总结报告

用例类别	用例	总功率 (平均功率之和)	
		值	Unit
低功耗模式用例	挂起模式——RTD活动	52.91	mW
	挂起模式——RTD睡眠	46.72	mW
	挂起模式——RTD深度睡眠	0.77	mW
	挂起模式——RTD断电	60.68	μ W
	系统空闲，显示器开启	405.38	mW
	系统空闲，显示器关闭	178.55	mW
音频/视频播放用例	全频率音频播放 (gplay)	425.24	mW
	96MHz DDR时钟的音频低总线播放 (gplay)	223.77	mW
	通过蓝牙播放音频 (gplay)	435.66	mW
	本地视频播放 (gplay)	531.07	mW
	视频播放流 (gplay)	550.96	mW
核心基准用例	Dhrystone	512.45	mW
	CoreMark	479.01	mW
GPU用例	MM07	577.61	mW
	MM06	612.78	mW
	GLMark	624.10	mW

表4. MCIMX8ULP-EVK功率总结报告 (续)

用例类别	用例	总功率 (平均功率之和)	
		值	单位
重载用例	双核Dhrystone + MM07 + HiFi4 + CM33 CoreMark	720.23	mW
	双核Stream + MM07 + HiFi4 + CM33 CoreMark	663.65	mW
	GPU2D + GPU3D + CSI/DSI	660.91	mW
内存用例	memset	642.61	mW
	memcpy	543.09	mW
	Stream	646.94	mW
存储器 (eMMC) 用例	DD_RD_eMMC	490.70	mW
	DD_WRT_eMMC	415.32	mW
产品用例	并行电子墨水屏翻页	453.36	mW
	并行电子墨水屏部分更新	463.04	mW
	机器视觉	598.47	mW
	eIQ基准测试	452.30	mW
	UAC音频播放	381.09	mW
	不间断运行显示器, 1fps刷新率	90.329	mW
系统级功率估计用例	电池	8.16	μW
	256KB L2缓存	488.87	mW

5.1 低功耗模式用例

本节介绍以下低功耗模式用例场景：

- 挂起模式 (四个子用例)
- 系统空闲, 显示器开启
- 系统空闲, 显示器关闭

5.1.1 挂起模式——RTD活动

此用例需要以下设置：

- Linux内核处于挂起模式。
- APD处于断电模式。
- LPAV处于断电模式, DDR处于自刷新模式。
- RTD处于活动模式, 运行电源模式开关 (PMS) 演示。
- APD和LPAV中的所有时钟和PLL均关闭。

要配置和运行该用例, 请执行以下步骤：

1. 打开ON/OFF开关 (SW10), 给MCIMX8ULP-EVK板上电。
2. 确保以下几点：
 - 在RTD控制台上, 启动PMS演示。
 - 在APD控制台上, 使用默认DTB配置 (imx8ulp-evk.dtb) 启动Linux镜像。

3. 使Linux内核进入挂起（断电）模式：

```
echo mem > /sys/power/state
```

- 在RTD控制台的菜单上，按下“A”，使RTD进入活动模式（默认情况下，RTD处于活动模式）。
- 测量功率并记录结果。

表5所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表5. “挂起模式——RTD活动”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	9.00	16.3	52.91	未测量芯片温度，因为CA35核被挂起。
	BUCK1_LSW1_CPU_IV8	1.80	0.05	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW3_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW4_CPU_IV8	1.80	-0.05	-0.1		
	BUCK2_CPU_IV0	1.00	32.50	32.6		
	BUCK3_CPU_IV0	0.74	0.00	0.0		
	BUCK4_CPU_IV1	1.10	0.09	0.1		
	LDO1_CPU_IV1	0.00	0.00	0.0		
	LDO1_CPU_IV1_0V6	0.00	0.00	0.0		
	LDO2_CPU_3V3	3.31	1.20	3.9		
	LDO4_CPU_IV8	1.80	0.00	0.0		
LDO5_CPU_3V0	2.92	0.00	0.0			

5.1.2 挂起模式——RTD睡眠

此用例需要以下设置：

- Linux内核处于挂起模式。
- APD处于断电模式。
- LPAV处于断电模式，DDR处于自刷新模式。
- RTD处于睡眠模式。
- APD和LPAV中的所有时钟和PLL均关闭。

要配置和运行该用例，请执行以下步骤：

- 打开ON/OFF开关（SW10），给MCIMX8ULP-EVK板上电。
- 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用默认DTB配置（imx8ulp-evk.dtb）启动Linux镜像。
- 使Linux内核进入挂起（断电）模式：

```
echo mem > /sys/power/state
```

- 在RTD控制台的菜单上，先按下“D”，再按下“S”，使RTD进入睡眠模式。
- 测量功率并记录结果。

表6所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表6. “挂起模式——RTD睡眠”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	9.10	16.3	46.72	未测量芯片温度，因为CA35核被挂起。
	BUCK1_LSW1_CPU_IV8	1.80	0.05	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW3_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW4_CPU_IV8	1.80	0.00	0.0		
	BUCK2_CPU_IV0	1.00	26.20	26.3		
	BUCK3_CPU_IV0	0.74	0.00	0.0		
	BUCK4_CPU_IV1	1.10	0.09	0.1		
	LDO1_CPU_IV1	0.00	0.00	0.0		
	LDO1_CPU_IV1_0V6	0.00	0.00	0.0		
	LDO2_CPU_3V3	3.31	1.20	3.9		
	LDO4_CPU_IV8	1.80	0.00	0.0		
	LDO5_CPU_3V0	2.92	0.00	0.0		

5.1.3 挂起模式——RTD深度睡眠

此用例需要以下设置：

- Linux内核处于挂起模式。
- APD处于断电模式。
- LPAV处于断电模式，DDR处于自刷新模式。
- RTD处于深度睡眠模式。
- APD和LPAV中的所有时钟和PLL均关闭。

要配置和运行该用例，请执行以下步骤：

1. 打开ON/OFF开关 (SW10)，给MCIMX8ULP-EVK板上电。
2. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用默认DTB配置 (imx8ulp-evk.dtb) 启动Linux镜像。
3. 使Linux内核进入挂起（断电）模式：


```
echo mem > /sys/power/state
```
4. 在RTD控制台的菜单上，先按下“E”，再按下“S”，使RTD进入深度睡眠模式。
5. 测量功率并记录结果。

表7所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表7. “挂起模式——RTD深度睡眠”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	0.05	0.1	0.77	未测量芯片温度，因为CA35核被挂起。
	BUCK1_LSW1_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW2_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW3_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW4_CPU_IV8	1.80	0.00	0.0		
	BUCK2_CPU_IV0	0.75	0.80	0.6		
	BUCK3_CPU_IV0	0.74	0.00	0.0		
	BUCK4_CPU_IV1	1.10	0.09	0.1		
	LDO1_CPU_IV1	0.00	0.00	0.0		
	LDO1_CPU_IV1_0V6	0.00	0.00	0.0		
	LDO2_CPU_3V3	3.30	0.00	0.0		
	LDO4_CPU_IV8	1.80	0.00	0.0		
	LDO5_CPU_3V0	2.95	0.00	0.0		

5.1.4 挂起模式——RTD断电

此用例需要以下设置：

- Linux内核处于挂起模式。
- APD处于断电模式。
- LPAV处于断电模式，DDR处于自刷新模式。
- RTD处于断电模式（使用MCIMX8ULP-EVK板上的SW8按钮（唤醒源）给CM33核断电）。
- APD和LPAV中的所有时钟和PLL均关闭。

要配置和运行该用例，请执行以下步骤：

1. 打开ON/OFF开关（SW10），给MCIMX8ULP-EVK板上电。
2. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用默认DTB配置（imx8ulp-evk.dtb）启动Linux镜像。
3. 使Linux内核进入挂起（断电）模式：


```
echo mem > /sys/power/state
```
4. 在RTD控制台的菜单上，先按下“F”，再按下“S”，使RTD进入断电模式。
5. 测量功率并记录结果。

表8所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表8. “挂起模式——RTD断电”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (μ W)	0区芯片温度 ($^{\circ}$ C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	0.05	0.1	60.68	未测量芯片温度，因为CA35核被挂起。
	BUCK1_LSW1_CPU_IV8	0.13	0.00	0.0		
	BUCK1_LSW2_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW3_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW4_CPU_IV8	1.80	0.00	0.0		
	BUCK2_CPU_IV0	0.71	0.00	0.0		
	BUCK3_CPU_IV0	0.74	-0.13	-0.1		
	BUCK4_CPU_IV1	1.10	0.09	0.1		
	LDO1_CPU_IV1	0.00	0.00	0.0		
	LDO1_CPU_IV1_0V6	0.00	0.00	0.0		
	LDO2_CPU_3V3	3.30	0.00	0.0		
	LDO4_CPU_IV8	1.80	0.00	0.0		
	LDO5_CPU_3V0	2.95	0.00	0.0		

注：对于这些用例，功率单位为 μ W：

- 挂起模式——RTD断电
- 电池

5.1.5 系统空闲，显示器开启

此用例需要以下设置：

- CPU频率设置为800MHz（默认值）。
- DDR频率设置为528MHz (1056MT/s)。
- Arm Cortex-M33核（RTD）处于睡眠模式。
- 所有未使用的PLL均关闭，未使用的时钟进行了门控。
- 720p电视显示器通过HDMI接口连接到MCIMX8ULP-EVK板。

要配置和运行该用例，请执行以下步骤：

1. 打开ON/OFF开关（SW10），给MCIMX8ULP-EVK板上电。
2. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用默认DTB配置（imx8ulp-evk.dtb）启动Linux镜像。
3. 要使Linux内核进入空闲模式，请运行setup_default.sh（有关详细信息，请参阅[第8节](#)）。
4. 在RTD控制台的菜单上，先按下“D”，再按下“S”使RTD进入睡眠模式。
5. 测量功率并记录结果。

表9所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表9. “系统空闲，显示器开启”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	16.60	30.0	405.38	32
	BUCK1_LSW1_CPU_IV8	1.80	0.05	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	1.80	3.2		
	BUCK1_LSW4_CPU_IV8	1.80	0.72	1.3		
	BUCK2_CPU_IV0	1.00	26.20	26.3		
	BUCK3_CPU_IV0	1.10	237.60	261.4		
	BUCK4_CPU_IV1	1.10	0.63	0.7		
	LDO1_CPU_IV1	1.10	3.40	3.8		
	LDO1_CPU_IV1_0V6	1.09	60.10	65.7		
	LDO2_CPU_3V3	3.31	1.20	3.9		
	LDO4_CPU_IV8	1.80	0.38	0.7		
	LDO5_CPU_3V0	2.92	0.00	0.0		

5.1.6 系统空闲，显示器关闭

此用例需要以下设置：

- CPU频率设置为650MHz。
- DDR频率设置为96MHz（低频率）。
- 如果Linux内核处于最低空闲级别，则两个CA35核均进行电源门控。
- CM33核（RTD）处于睡眠模式。
- 所有未使用的PLL均关闭，未使用的时钟进行了门控。
- 720p电视显示器通过HDMI接口连接到MCIMX8ULP-EVK板。

要配置和运行该用例，请执行以下步骤：

1. 打开ON/OFF开关（SW10），给MCIMX8ULP-EVK板上电。
2. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用DTB配置imx8ulp-evk-nd.dtb启动Linux镜像。
3. 要使Linux内核进入空闲模式，请运行DDRC_96MHz_setup.sh（请参阅第8节）。
4. 在RTD控制台的菜单上，先按下“D”，再按下“S”，使RTD进入睡眠模式。
5. 测量功率并记录结果。

表10所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表10. “系统空闲，显示器关闭”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	13.50	24.3	178.55	28
	BUCK1_LSW1_CPU_IV8	1.80	0.05	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW4_CPU_IV8	1.80	0.83	1.5		
	BUCK2_CPU_IV0	1.00	26.40	26.5		
	BUCK3_CPU_IV0	1.00	85.20	85.4		
	BUCK4_CPU_IV1	1.10	0.18	0.2		
	LDO1_CPU_IV1	1.10	0.54	0.6		
	LDO1_CPU_IV1_0V6	1.10	24.60	27.0		
	LDO2_CPU_3V3	3.31	1.20	3.9		
	LDO4_CPU_IV8	1.80	0.38	0.7		
	LDO5_CPU_3V0	2.92	0.00	0.0		

5.2 音频/视频播放用例

本节介绍以下与音频/视频播放相关的用例场景：

- 全频率音频播放 (gplay)
- 96MHz DDR时钟的音频低总线播放 (gplay)
- 通过蓝牙播放音频 (gplay)
- 本地视频播放 (gplay)
- 视频播放流 (gplay)

5.2.1 全频率音频播放 (gplay)

此用例需要一个具有以下规格的音频文件：

- 文件类型：MP3
- 比特率：128kbit/s
- 采样率：每秒44kHz

使用以下命令播放音频文件：

```
gplay-1.0 $audio_file
```

此用例需要以下设置：

- CPU频率设置为800MHz（默认值）。
- DDR频率设置为528MHz (1056MT/s)。
- CA35核处于活动模式，负责处理MP3音频解码。
- CM33核处于活动模式，通过I2S接口处理音频PCM播放。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

注：在循环中运行该用例，并以所需的时间间隔（建议时间间隔为1分钟）记录功率测量结果。

1. 打开ON/OFF开关（SW10），给MCIMX8ULP-EVK板上电。
2. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用默认DTB配置（imx8ulp-evk.dtb）启动Linux镜像。
3. 运行setup.sh（有关详细信息，请参阅[第8节](#)）。
4. 运行gplay_audio.sh：

```
audio_file='Mpeg1L3_44kHz_128kbps_s_Ed_Rush_Sabotage_mplayer.mp3'
gplay-1.0 $audio_file
```

注：要运行上述命令，请准备MP3文件。为了获得与[表1](#)中相似的结果，音频比特率应该为约28kbit/s。

5. 测量功率并记录结果。

[表1](#)所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表11. “全频率音频播放（gplay）”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	16.90	30.4	425.24	32
	BUCK1_LSW1_CPU_IV8	1.80	0.05	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW4_CPU_IV8	1.80	0.88	1.6		
	BUCK2_CPU_IV0	1.00	33.80	33.9		
	BUCK3_CPU_IV0	1.10	263.70	290.0		
	BUCK4_CPU_IV1	1.10	0.54	0.6		
	LDO1_CPU_IV1	1.10	3.60	3.9		
	LDO1_CPU_IV1_0V6	1.10	46.80	51.2		
	LDO2_CPU_3V3	3.31	1.40	4.5		
	LDO4_CPU_IV8	1.80	0.38	0.7		
	LDO5_CPU_3V0	2.92	0.00	0.0		

5.2.2 96MHz DDR时钟的音频低总线播放（gplay）

此用例需要一个具有以下规格的音频文件：

- 文件类型：MP3
- 比特率：128kbit/s
- 采样率：每秒44kHz

使用以下命令播放音频文件：

```
gplay-1.0 $audio_file
```

此用例需要以下设置：

- CA35核频率设置为650MHz。
- DDR频率设置为96MHz。
- 如果Linux内核处于最低空闲级别，则CA35核进行电源门控，并负责处理MP3音频解码。
- CM33核处于活动模式，通过I2S接口处理音频PCM播放。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

1. 打开ON/OFF开关 (SW10)，给MCIMX8ULP-EVK板上电。

2. 确保以下几点：

- 在RTD控制台上，启动PMS演示。
- 在APD控制台上，使用默认DTB配置imx8ulp-evk-nd.dtb启动Linux镜像。

注： imx8ulp-evk-nd.dtb配置使GPU3D/2D、HiFi4和USDHC0/1/2在标称驱动(nd)模式下处于低功耗状态。因此，系统被设置为ND模式，并且BUCK3在较低的电压(1.0V)下工作，以降低功耗。

3. 运行DDRC_96MHz_setup.sh (有关详细信息，请参阅[第8节](#))。

4. 运行gplay_audio.sh：

```
audio_file='Mpeg1L3_44kHz_128kbps_s_Ed_Rush_Sabotage_mplayer.mp3'
gplay-1.0 $audio_file
```

注： 要运行上述命令，请准备MP3文件。为了获得与[表2](#)中相似的结果，音频比特率应该为约128kbit/s。

5. 测量功率并记录结果。

[表12](#)所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表12. “96 MHz DDR时钟的音频低总线播放 (gplay)” 用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	13.50	24.3	223.77	31
	BUCK1_LSW1_CPU_IV8	1.80	0.05	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW4_CPU_IV8	1.80	1.10	2.0		
	BUCK2_CPU_IV0	1.00	33.10	33.2		
	BUCK3_CPU_IV0	1.00	107.50	107.6		
	BUCK4_CPU_IV1	1.10	0.18	0.2		
	LDO1_CPU_IV1	1.10	0.54	0.6		
	LDO1_CPU_IV1_0V6	1.10	38.50	42.2		
	LDO2_CPU_3V3	3.31	1.40	4.5		
	LDO4_CPU_IV8	1.80	0.38	0.7		
	LDO5_CPU_3V0	2.92	0.00	0.0		

5.2.3 通过蓝牙播放音频 (gplay)

此用例需要一个具有以下规格的音频文件：

- 文件类型：MP3
- 比特率：128kbit/s
- 采样率：每秒44kHz

使用以下命令播放音频文件：

```
gplay-1.0 $audio_file
```

此用例需要以下设置：

- CPU频率设置为800MHz（默认值）。
- CM33核处于活动模式，通过I2S接口处理音频PCM播放。
- HiFi4 DSP处于活动模式，负责处理MP3音频解码。
- 蓝牙扬声器用于播放音频流。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

注：在循环中运行该用例，并以所需的时间间隔（建议时间间隔为1分钟）记录功率测量结果。

1. 打开ON/OFF开关（SW10），给MCIMX8ULP-EVK板上电。
2. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用默认DTB配置（imx8ulp-evk.dtb）启动Linux镜像。
3. 在RTD控制台的菜单上，在PMS演示过程中，按下“A”，使RTD进入活动模式。
4. 运行setup.sh（有关详细信息，请参阅[第8节](#)）。
5. 运行setup_bt_88w8987.sh或setup_bt_iw416.sh（取决于与电路板连接的蓝牙设备）。有关更多信息，请参阅[第8节](#)。
6. 准备一个蓝牙音箱。
7. 要搜索音箱的MAC地址，请运行setup_pulseaudio.sh（有关更多详细信息，请参阅[第8节](#)）。
8. 当蓝牙音箱出现在终端上时，复制其MAC地址（使用CTRL+C组合键）。在当前的示例中，蓝牙音箱命名为SoundCore 2，其MAC地址为08:EB:ED:57:EE:69（见[图4](#)）。

```

[NEW] Device 6C:F0:90:EA:4A:7E 6C-F0-90-EA-4A-7E
[CHG] Device D0:62:2C:1C:A7:12 RSSI: -80
[CHG] Device CC:14:B0:8B:5C:EB RSSI: -76
[NEW] Device 46:9B:4C:75:E6:C9 46-9B-4C-75-E6-C9
[CHG] Device CC:14:B0:8B:5C:EB RSSI: -64
[CHG] Device 55:F2:B2:C0:2E:5F RSSI: -86
[CHG] Device 55:51:1E:AC:C3:39 RSSI: -80
[CHG] Device 55:F2:B2:C0:2E:5F RSSI: -65
[CHG] Device 40:F0:C6:FD:B3:4F RSSI: -92
[NEW] Device 59:12:48:54:54:AE 59-12-48-54-54-AE
[NEW] Device 67:06:F5:DB:2C:28 67-06-F5-DB-2C-28
[CHG] Device 14:0A:29:CE:41:57 ManufacturerData Key: 0x038f
[CHG] Device 14:0A:29:CE:41:57 ManufacturerData Value:
16 01 16 03 6e 1e 1e ff be ce a3 0a 14 29 57 41 ....n.....)WA
ce ec 0a 14 29 57 41 ce ....)WA.
[CHG] Device 14:0A:29:CE:41:57 ManufacturerData Key: 0x1727
[CHG] Device 14:0A:29:CE:41:57 ManufacturerData Value:
08 03 02 50 69 29 ce 41 57 28 ... Pi).AW(
[CHG] Device 40:F0:C6:FD:B3:4F RSSI: -77
[CHG] Device 74:DB:90:B3:C3:C4 ManufacturerData Key: 0x004c
[CHG] Device 74:DB:90:B3:C3:C4 ManufacturerData Value:
10 06 7d 1e 06 db b7 3c ..}....<
[NEW] Device 08:EB:ED:57:EE:69 SoundCore 2
[NEW] Device E4:6A:35:96:65:76 真我 GI Neos
^Croot@imx8ulpevk:~# bluetoothctl pair 08:EB:ED:57:EE:69
Attempting to pair with 08:EB:ED:57:EE:69
[CHG] Device 08:EB:ED:57:EE:69 Connected: yes
[CHG] Device 08:EB:ED:57:EE:69 Modalias: usb:v099Ap0500d011B
[CHG] Device 08:EB:ED:57:EE:69 UUIDs: 0000110b-0000-1000-8000-00805f9b34fb
[CHG] Device 08:EB:ED:57:EE:69 UUIDs: 0000110c-0000-1000-8000-00805f9b34fb
[CHG] Device 08:EB:ED:57:EE:69 UUIDs: 0000110e-0000-1000-8000-00805f9b34fb
[CHG] Device 08:EB:ED:57:EE:69 UUIDs: 0000111e-0000-1000-8000-00805f9b34fb
[CHG] Device 08:EB:ED:57:EE:69 UUIDs: 00001200-0000-1000-8000-00805f9b34fb
[CHG] Device 08:EB:ED:57:EE:69 ServicesResolved: yes
[CHG] Device 08:EB:ED:57:EE:69 Paired: yes
Pairing successful
root@imx8ulpevk:~# bluetoothctl trust 08:EB:ED:57:EE:69
[CHG] Device 08:EB:ED:57:EE:69 Trusted: yes
Changing 08:EB:ED:57:EE:69 trust succeeded
root@imx8ulpevk:~# connect 08:EB:ED:57:EE:69
-sh: connect: command not found
root@imx8ulpevk:~# bluetoothctl connect 08:EB:ED:57:EE:69
Attempting to connect to 08:EB:ED:57:EE:69
[CHG] Device 08:EB:ED:57:EE:69 Connected: yes
[CHG] Device 08:EB:ED:57:EE:69 Paired: yes
[NEW] Endpoint /org/bluez/hci0/dev_08_EB_ED_57_EE_69/sep1
[NEW] Endpoint /org/bluez/hci0/dev_08_EB_ED_57_EE_69/sep2
[NEW] Transport /org/bluez/hci0/dev_08_EB_ED_57_EE_69/sep1/fd0
Connection successful

```

图4. 复制蓝牙MAC地址

9. 通过运行以下命令与蓝牙音箱建立连接 (将<BT_MAC_ADDR>替换为实际MAC地址) :

```

bluetoothctl pair <BT_MAC_ADDR>
bluetoothctl trust <BT_MAC_ADDR>
bluetoothctl connect <BT_MAC_ADDR>

```

10. 运行gplay_audio.sh:

```

audio_file='Mpeg1L3_44kHz_128kbps_s_Ed_Rush_Sabotage_mplayer.mp3'
gplay-1.0 $audio_file

```

11. 测量功率并记录结果。

表13所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表13. “通过蓝牙播放音频 (gplay)” 用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	17.20	31.0	435.66	36
	BUCK1_LSW1_CPU_IV8	1.80	0.05	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW4_CPU_IV8	1.80	4.00	7.2		
	BUCK2_CPU_IV0	1.00	33.00	33.0		
	BUCK3_CPU_IV0	1.10	271.50	298.6		
	BUCK4_CPU_IV1	1.10	0.63	0.7		
	LDO1_CPU_IV1	1.10	3.60	3.9		
	LDO1_CPU_IV1_0V6	1.10	43.90	48.1		
	LDO2_CPU_3V3	3.31	1.20	4.0		
	LDO4_CPU_IV8	1.80	0.38	0.7		
	LDO5_CPU_3V0	2.93	0.00	0.0		

5.2.4 本地视频播放 (gplay)

此用例需要一个具有以下规格的视频文件：

- 文件类型：MKV
- 视频压缩标准：HEVC
- 显示分辨率：全高清分辨率，每秒29.97帧
- 音频编码：AAC，采用双通道配置
- 音频采样率：每秒44.1kHz

使用以下命令在本地播放视频文件：

注： i.MX 8ULP处理器没有硬件解码器；因此，需要搭配使用软件解码器。

```
gplay-1.0 $path/$FILE
```

此用例需要以下设置：

- CPU频率设置为800MHz（默认值）。
- CA35核处于活动模式，负责处理音频解码和视频解码。
- CM33核处于活动模式，通过I2S接口处理音频PCM播放。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。
- 720p电视显示器通过HDMI接口连接到MCIMX8ULP-EVK板。

要配置和运行该用例，请执行以下步骤：

注： 在循环中运行该用例，并以所需的时间间隔（建议时间间隔为1分钟）记录功率测量结果。

1. 通过HDMI接口将显示器连接到MCIMX8ULP-EVK板。

2. 打开ON/OFF开关 (SW10) , 给电路板上电。
3. 确保以下几点:
 - 在RTD控制台上, 启动PMS演示。
 - 在APD控制台上, 使用默认DTB配置 (imx8ulp-evk.dtb) 启动Linux镜像。
4. 运行setup_video.sh (有关详细信息, 请参阅[第8节](#))。
5. 在RTD控制台的菜单上, 在PMS演示过程中, 按下“A”, 使RTD进入活动模式。
6. 运行gplay_videoplayback.sh:

```
gplay-1.0 ./480p24.mp4
```

注: 要运行上述命令, 请准备MP4文件。为了获得与[表4](#)中相似的结果, 请使用以下设置:

- 分辨率: 480p, 24帧速率
- 视频比特率: 约1200kbit/s
- 编码: H.264格式

7. 测量功率并记录结果。

[表14](#)所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表14. “本地视频播放 (gplay)” 用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	17.10	30.7	531.07	34
	BUCK1_LSW1_CPU_IV8	1.80	0.11	0.2		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	1.80	3.2		
	BUCK1_LSW4_CPU_IV8	1.80	1.10	2.0		
	BUCK2_CPU_IV0	1.00	32.80	32.9		
	BUCK3_CPU_IV0	1.10	332.70	365.7		
	BUCK4_CPU_IV1	1.10	0.63	0.7		
	LDO1_CPU_IV1	1.09	3.80	4.1		
	LDO1_CPU_IV1_0V6	1.09	71.30	77.9		
	LDO2_CPU_3V3	3.31	1.40	4.5		
	LDO4_CPU_IV8	1.80	0.44	0.8		
	LDO5_CPU_3V0	2.93	0.00	0.0		

5.2.5 视频播放流 (gplay)

此用例需要一个具有以下规格的视频文件:

- 文件类型: MKV
- 视频压缩标准: HEVC
- 显示分辨率: 全高清分辨率, 每秒29.97帧
- 音频编码: AAC, 采用双通道配置
- 音频采样率: 每秒44.1kHz

使用以下命令通过以太网适配器进行视频流式传输:

```
gplay-1.0 $network_path/$FILE
```

此用例需要以下设置：

- CPU频率设置为800MHz（默认值）。
- CA35核处于活动模式，负责处理音频解码和视频解码。
- CM33核处于活动模式，通过I2S接口处理音频PCM播放。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。
- 安装了一个服务器来托管要流式传输的MKV视频文件。
- 720p电视显示器通过HDMI接口连接到MCIMX8ULP-EVK板。

要配置和运行该用例，请执行以下步骤：

注：在循环中运行该用例，并以所需的时间间隔（建议时间间隔为1分钟）记录功率测量结果。

1. 将PC和MCIMX8ULP-EVK板连接到同一个本地网络。

2. 在服务器PC上：

a. 从<https://nodejs.org>下载node.js。

b. 使用以下命令安装http服务器：

```
npm install http-server -g
```

c. 在终端中输入包含目标视频（480p24.mp4）的目标文件夹。

d. 运行以下命令：

```
http-server -c-1
```

现在，可以使用<ip_server>。

3. 在电路上：

a. 打开ON/OFF开关（SW10），给电路板上电。

b. 确保以下几点：

- 在RTD控制台上，启动PMS演示。
- 在APD控制台上，使用默认DTB配置（imx8ulp-evk.dtb）启动Linux镜像。

c. 连接HDMI显示器并运行setup_video_stream.sh。

d. 在RTD控制台的菜单上，在PMS演示过程中，按下“A”，使RTD进入活动模式。

e. 在电路板上运行以下命令：

```
gplay-1.0 http://<ip_server>/480p24.mp4
```

注：要运行上述命令，请准备MP4文件。为了获得与表15中相似的结果，请使用以下设置：

- 分辨率：480p，24帧速率
 - 视频比特率：约1200kbit/s
 - 编码：H.264格式
- f. 测量电路板的功率并记录结果。

表15所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表15. “视频播放流 (gplay)” 用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	17.20	31.0	550.96	35
	BUCK1_LSW1_CPU_IV8	1.80	0.05	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	1.80	3.2		
	BUCK1_LSW4_CPU_IV8	1.80	0.88	1.6		
	BUCK2_CPU_IV0	1.00	32.50	32.6		
	BUCK3_CPU_IV0	1.10	350.20	384.8		
	BUCK4_CPU_IV1	1.10	0.63	0.7		
	LDO1_CPU_IV1	1.09	3.80	4.2		
	LDO1_CPU_IV1_0V6	1.09	72.50	79.1		
	LDO2_CPU_3V3	3.31	1.40	4.5		
	LDO4_CPU_IV8	1.80	0.38	0.7		
	LDO5_CPU_3V0	2.93	0.00	0.0		

5.3 核心基准测试用例

本节介绍了如何使用CA35核测试以下用例场景：

- Dhrystone
- CoreMark

5.3.1 Dhrystone

Dhrystone是一个综合基准测试，用于衡量处理器和编译器的整数计算性能。Dhrystone基准测试的小尺寸使其能够适应L1缓存，并最大限度地减少对L2缓存和DDR的访问。

在这个用例中，在两个CA35核上执行Dhrystone测试。因为Dhrystone是一个单线程基准测试，所以已经启动了两个实例。两个CA35核都在800MHz的频率下循环运行测试。

此用例需要以下设置：

- CPU频率设置为800MHz（默认值）。
- DDR频率设置为528MHz（1056MT/s）。
- CM33核处于睡眠模式。
- 显示器关闭。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

注：在循环中运行该用例，并以所需的时间间隔（建议时间间隔为1分钟）记录功率测量结果。

1. 打开ON/OFF开关（SW10），给MCIMX8ULP-EVK板上电。
2. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。

- 在APD控制台上，使用默认DTB配置 (imx8ulp-evk.dtb) 启动Linux镜像。
- 在RTD控制台的菜单上，先按下“D”，再按“S”，使RTD进入睡眠模式。
 - 运行setup.sh (有关详细信息，请参阅[第8节](#))。
 - 运行dhrystone_loop.sh:

```
while [ "1" == "1" ]; do
    taskset -c 0 ./dhry2 &
    taskset -c 1 ./dhry2
done
```

- 测量功率并记录结果。

[表16](#)所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表16. “Dhrystone” 用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	16.40	29.6	512.45	35
	BUCK1_LSW1_CPU_IV8	1.80	0.05	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW4_CPU_IV8	1.80	0.88	1.6		
	BUCK2_CPU_IV0	1.00	26.60	26.6		
	BUCK3_CPU_IV0	1.10	368.70	405.0		
	BUCK4_CPU_IV1	1.10	0.54	0.6		
	LDO1_CPU_IV1	1.10	3.40	3.8		
	LDO1_CPU_IV1_OV6	1.10	29.30	32.1		
	LDO2_CPU_3V3	3.31	1.20	4.0		
	LDO4_CPU_IV8	1.80	0.44	0.8		
	LDO5_CPU_3V0	2.93	0.00	0.0		

5.3.2 CoreMark

CoreMark是一种现代化且复杂的基准测试，可以准确地测量处理器的性能，旨在取代旧的Dhrystone基准测试。Arm建议使用CoreMark而不是Dhrystone。

注：平台未连接显示器。

此用例需要以下设置：

- CPU频率设置为800MHz (默认值)。
- DDR频率设置为528MHz (1056MT/s)。
- CM33核处于睡眠模式。
- 显示器关闭。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

注：在循环中运行该用例，并以所需的时间间隔 (建议时间间隔为1分钟) 记录功率测量结果。

1. 打开ON/OFF开关 (SW10) , 给MCIMX8ULP-EVK板上电。
2. 确保以下几点:
 - 在RTD控制台上, 启动PMS演示。
 - 在APD控制台上, 使用默认DTB配置 (imx8ulp-evk.dtb) 启动Linux镜像。
3. 在RTD控制台的菜单上, 先按下“D”, 再按“S”, 使RTD进入睡眠模式。
4. 运行setup.sh (有关详细信息, 请参阅[第8节](#))。
5. 为了获得更好的性能, 编译coremark应用程序 (在i.MX 8ULP中, coremark应用为双核而构建) :

```
make XCFLAGS="-DMULTITHREAD=2 -DUSE_PTHREAD -pthread"
```

6. 运行coremark_loop.sh:

```
while true; do
  ./coremark
done
```

7. 测量功率并记录结果。

[表17](#)所示为当此用例应用于i.MX 8ULP处理器时的测量结果, 在四个CPU核上进行CoreMark基准测试。

表17. “CoreMark” 用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	16.50	29.8	479.01	34
	BUCK1_LSW1_CPU_IV8	1.80	0.05	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW4_CPU_IV8	1.80	0.88	1.6		
	BUCK2_CPU_IV0	1.00	26.40	26.4		
	BUCK3_CPU_IV0	1.10	338.30	371.7		
	BUCK4_CPU_IV1	1.10	0.54	0.6		
	LDO1_CPU_IV1	1.10	3.40	3.8		
	LDO1_CPU_IV1_0V6	1.10	29.10	31.9		
	LDO2_CPU_3V3	3.31	1.20	4.0		
	LDO4_CPU_IV8	1.80	0.38	0.7		
	LDO5_CPU_3V0	2.93	0.00	0.0		

5.4 GPU用例

本节介绍GPU用例场景:

- MM07
- MM06
- GLMark

5.4.1 MM07

MM07是一个3D游戏基准测试，包括将图形从eMMC存储器加载到DDR存储器中，使用GPU3D进行处理，并将其复制到DDR存储器的显示缓冲区中。图形通过HDMI接口在720p电视显示器上显示。

此用例需要以下设置：

- CPU频率设置为800MHz（默认值）。
- DDR频率设置为528MHz（1056MT/s）。
- CM33核处于睡眠模式。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

1. 通过HDMI接口将显示器连接到MCIMX8ULP-EVK板。
2. 打开ON/OFF开关（SW10），给电路板上电。
3. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用默认DTB配置（imx8ulp-evk.dtb）启动Linux镜像。
4. 在RTD控制台的菜单上，先按下“D”，再按“S”，使RTD进入睡眠模式。
5. 运行setup_video.sh（有关详细信息，请参阅[第8节](#)）。
6. 运行gpu_mm07.sh：

```
export WL_EGL_SWAP_INTERVAL=0
cd mm07/
while true; do
    ./fm_oes2_mobile_player
done
```

注：MM06和MM07用例并非该软件版本的组成部分，而是来自第三方供应商。要对这些用例进行评估，需要购买。

7. 开始测量功率并记录结果。

表18所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表18. “MM07” 用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	17.10	30.7	577.61	35
	BUCK1_LSW1_CPU_IV8	1.80	0.05	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	1.80	3.2		
	BUCK1_LSW4_CPU_IV8	1.80	0.83	1.5		
	BUCK2_CPU_IV0	1.00	26.30	26.4		
	BUCK3_CPU_IV0	1.10	366.80	402.7		
	BUCK4_CPU_IV1	1.10	0.54	0.6		
	LDO1_CPU_IV1	1.09	4.00	4.4		
	LDO1_CPU_IV1_0V6	1.09	87.00	94.7		
	LDO2_CPU_3V3	3.31	1.20	3.9		
	LDO4_CPU_IV8	1.80	0.55	1.0		
	LDO5_CPU_3V0	2.93	0.00	0.0		

5.4.2 MM06

MM06是一个3D游戏基准测试，包括将图形从eMMC存储器加载到DDR存储器中，使用GPU3D进行处理，并将其复制到DDR存储器的显示缓冲区中。图形通过HDMI接口在720p电视显示器上显示。

此用例需要以下设置：

- CPU频率设置为800MHz（默认值）。
- DDR频率设置为528MHz（1056MT/s）。
- CM33核处于睡眠模式。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

1. 通过HDMI接口将显示器连接到MCIMX8ULP-EVK板。
2. 打开ON/OFF开关（SW10），给电路板上电。
3. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用默认DTB配置（imx8ulp-evk.dtb）启动Linux镜像。
4. 在RTD控制台的菜单上，先按下“D”，再按“S”，使RTD进入睡眠模式。
5. 运行setup_video.sh（有关详细信息，请参阅第8节）。
6. 运行gpu_mm06.sh：

```
export WL_EGL_SWAP_INTERVAL=0
cd mm06/
while true; do
  ./fm_oes_player
done
```

注：MM06和MM07用例并非该软件版本的组成部分，而是来自第三方供应商。要对这些用例进行评估，需要购买。

7. 开始测量功率并记录结果。

表19所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表19. “MM06”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	17.10	30.8	612.78	35
	BUCK1_LSW1_CPU_IV8	1.80	0.05	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	1.80	3.2		
	BUCK1_LSW4_CPU_IV8	1.80	0.88	1.6		
	BUCK2_CPU_IV0	1.00	26.50	26.6		
	BUCK3_CPU_IV0	1.10	387.40	425.3		
	BUCK4_CPU_IV1	1.10	0.63	0.7		
	LDO1_CPU_IV1	1.09	4.10	4.5		
	LDO1_CPU_IV1_0V6	1.09	98.50	107.1		
	LDO2_CPU_3V3	3.31	1.20	3.9		
	LDO4_CPU_IV8	1.80	0.38	0.7		
	LDO5_CPU_3V0	2.93	0.00	0.0		

5.4.3 GLMark

GLmark用例需要以下设置：

- CPU频率设置为800MHz（默认值）。
- DDR频率设置为528MHz（1056MT/s）。
- CM33核处于睡眠模式。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

1. 通过HDMI接口将显示器连接到MCIMX8ULP-EVK板。
2. 打开ON/OFF开关（SW10），给电路板上电。
3. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用默认DTB配置（imx8ulp-evk.dtb）启动Linux镜像。
4. 在RTD控制台的菜单上，先按下“D”，再按“S”，使RTD进入睡眠模式。
5. 运行setup_video.sh（有关详细信息，请参阅[第8节](#)）。
6. 运行gpu_glmark.sh：

```
while true; do
    glmark2-es2-wayland -fullscreen
done
```

7. 开始测量功率并记录结果。

表20所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表20. “GLMark” 用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	16.90	30.5	624.10	35
	BUCK1_LSW1_CPU_IV8	1.80	0.11	0.2		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.4		
	BUCK1_LSW3_CPU_IV8	1.80	1.80	3.2		
	BUCK1_LSW4_CPU_IV8	1.80	0.83	1.5		
	BUCK2_CPU_IV0	1.00	26.30	26.4		
	BUCK3_CPU_IV0	1.10	405.10	444.6		
	BUCK4_CPU_IV1	1.10	0.63	0.7		
	LDO1_CPU_IV1	1.09	4.20	4.6		
	LDO1_CPU_IV1_0V6	1.09	91.30	99.3		
	LDO2_CPU_3V3	3.31	1.20	4.0		
	LDO4_CPU_IV8	1.80	0.44	0.8		
	LDO5_CPU_3V0	2.93	0.00	0.0		

5.5 重载用例

本节介绍以下重载用例场景：

- 双核Dhrystone + MM07 + HiFi4 + CM33 CoreMark
- 双核Stream + MM07 + HiFi4 + CM33 CoreMark
- GPU2D + GPU3D + CSI/DSI

这些用例场景旨在展示极端条件下的功耗。对于这些用例，720p电视显示器通过HDMI接口连接到MCIMX8ULP-EVK板。

5.5.1 双核Dhrystone + MM07 + HiFi4 + CM33 CoreMark

此用例并行运行双核Dhrystone、GPU MM07、HiFi4和CM33 CoreMark。720p电视显示器通过HDMI接口连接到MCIMX8ULP-EVK板。

此用例需要以下设置：

- CPU频率设置为800MHz（默认值）。
- CM33核频率设置为160MHz。
- DDR频率设置为528MHz (1056MT/s)。
- HiFi4正在进行Math Power繁忙环路压力测试。
- GPU正在进行MM07压力测试。
- 两个CA35核都在进行Dhrystone基准测试。
- CM33核正在进行CoreMark基准测试。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

1. 通过HDMI接口将显示器连接到MCIMX8ULP-EVK板。
2. 将电路板上的SW5[8:1]设置更改为01xx_xxxx（串行下载模式）。

- 将flash_pms_coremark.bin二进制镜像下载到eMMC:

注: flash_pms_coremark.bin镜像与默认二进制镜像 (flash.bin) 不同。因此, 对于其他需要默认二进制镜像的用例, 必须下载并使用flash_singleboot_default.bin镜像。

```
uuu -b emmc flash_pms_coremark.bin
```

- 确保将电路板配置为默认启动设置 (SW5[8:1] = 1000_xx00 (单启动-eMMC模式))。
- 打开ON/OFF开关 (SW10), 给电路板上电。
- 确保以下几点:
 - 在RTD控制台上, 使用CoreMark启动PMS演示。
 - 在APD控制台上, 使用默认DTB配置 (imx8ulp-evk.dtb) 启动Linux镜像。
- 运行setup_video.sh (有关更多详细信息, 请参阅[第8节](#))。
- 启动两个Dhrystone实例, 每个实例绑定在单独的CPU上:

```
while [ "1" == "1" ]; do
    taskset -c 0 ./dhry2 &
    taskset -c 1 ./dhry2
done
```

- 将hifi4_imx8ulp-MathPower_busy_loop.bin复制到/lib/firmware/imx/dsp/, 并将其重命名为hifi4.bin。
- 使用此命令加载并运行固件:

```
echo start > /sys/class/remoteproc/remoteproc<n>/state
```

其中<n>是HiFi4 (DSP) 远程处理器索引。

- 使用此命令查看HiFi4 (DSP) 远程处理器的名称:

```
cat /sys/class/remoteproc/remoteproc<n>/name
```

- 在RTD控制台的菜单上, 按下“K”, 运行CoreMark基准测试。
- 运行gpu_mm07.sh:

```
export WL_EGL_SWAP_INTERVAL=0
cd mm07/
while true; do
    ./fm_oes2_mobile_player
done
```

注: MM06和MM07用例并非该软件版本的组成部分, 而是来自第三方供应商。要对这些用例进行评估, 需要购买。

- 开始测量功率并记录结果。

表21所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表21. “双核Dhrystone + MM07 + HiFi4 + CM33 CoreMark” 用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	17.30	31.2	720.23	37
	BUCK1_LSW1_CPU_IV8	1.80	0.11	0.2		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.4		
	BUCK1_LSW3_CPU_IV8	1.80	1.80	3.2		
	BUCK1_LSW4_CPU_IV8	1.80	0.88	1.6		
	BUCK2_CPU_IV0	1.00	34.80	34.9		
	BUCK3_CPU_IV0	1.10	490.00	536.8		
	BUCK4_CPU_IV1	1.10	0.54	0.6		
	LDO1_CPU_IV1	1.09	4.20	4.6		
	LDO1_CPU_IV1_0V6	1.09	86.00	93.6		
	LDO2_CPU_3V3	3.31	1.20	3.9		
	LDO4_CPU_IV8	1.80	0.66	1.2		
	LDO5_CPU_3V0	2.93	0.00	0.0		

5.5.2 双核Stream + MM07 + HiFi4 + CM33 CoreMark

此用例并行运行双核Stream、GPU MM07、HiFi4和CM33 CoreMark。720p电视显示器通过HDMI接口连接到MCIMX8ULP-EVK板。

此用例需要以下设置：

- CPU频率设置为800MHz（默认值）。
- CM33核频率设置为160MHz。
- DDR频率设置为528MHz (1056MT/s)。
- HiFi4正在进行Math Power繁忙环路压力测试。
- GPU正在进行MM07压力测试。
- 两个CA35核都在进行Stream基准测试。
- CM33核正在进行CoreMark基准测试。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

1. 通过HDMI接口将显示器连接到MCIMX8ULP-EVK板。
2. 将电路板上的SW5[8:1]设置更改为01xx_xxxx（串行下载模式）。
3. 将flash_pms_coremark.bin二进制镜像下载到eMMC：

注： flash_pms_coremark.bin镜像与默认二进制镜像 (flash.bin) 不同。因此，对于其他需要默认二进制镜像的用例，必须下载并使用flash_singleboot_default.bin镜像。

```
uuu -b emmc flash_pms_coremark.bin
```

4. 确保将电路板配置为默认启动设置 (SW5[8:1] = 1000_xx00 (单启动-eMMC模式))。

5. 打开ON/OFF开关 (SW10) , 给电路板上电。
6. 确保以下几点:
 - 在RTD控制台上, 使用CoreMark启动PMS演示。
 - 在APD控制台上, 使用默认DTB配置 (imx8ulp-evk.dtb) 启动Linux镜像。
7. 运行setup_video.sh (有关更多详细信息, 请参阅[第8节](#)) 。
8. 启动两个Stream, 每个绑定在单独的CPU上:

```
while [ "1" == "1" ]; do
    taskset -c 0 stream -M 200M -N 1000 &
    taskset -c 1 stream -M 200M -N 1000
done
```

9. 将hifi4_imx8ulp-MathPower_busy_loop.bin复制到/lib/firmware/imx/dsp/, 并将其重命名为hifi4.bin。
10. 使用此命令加载并运行固件:

```
echo start > /sys/class/remoteproc/remoteproc<n>/state
```

其中<n>是HiFi4 (DSP) 远程处理器索引。

11. 使用此命令查看HiFi4 (DSP) 远程处理器的名称:

```
cat /sys/class/remoteproc/remoteproc<n>/name
```

12. 在RTD控制台的菜单上, 按下“K”, 运行CoreMark基准测试。
13. 运行gpu_mm07.sh:

```
export WL_EGL_SWAP_INTERVAL=0
cd mm07/
while true; do
    ./fm_oes2_mobile_player
done
```

注: MM06和MM07用例并非该软件版本的组成部分, 而是来自第三方供应商。要对这些用例进行评估, 需要购买。

14. 开始测量功率并记录结果。

表22所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表22. “双核Stream + MM07 + HiFi4 + CM33 CoreMark” 用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	17.20	30.9	663.65	37
	BUCK1_LSW1_CPU_IV8	1.80	0.11	0.2		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	1.80	3.2		
	BUCK1_LSW4_CPU_IV8	1.80	0.83	1.5		
	BUCK2_CPU_IV0	1.00	35.10	35.2		
	BUCK3_CPU_IV0	1.10	436.10	478.3		
	BUCK4_CPU_IV1	1.10	0.54	0.6		
	LDO1_CPU_IV1	1.09	4.30	4.7		
	LDO1_CPU_IV1_0V6	1.09	87.90	95.7		
	LDO2_CPU_3V3	3.31	1.20	4.0		
	LDO4_CPU_IV8	1.80	0.61	1.1		
	LDO5_CPU_3V0	2.93	0.00	0.0		

5.5.3 GPU2D + GPU3D + CSI/DSI

此用例并行运行GPU2D、GPU3D和CSI/DSI。720p电视显示器通过HDMI接口连接到MCIMX8ULP-EVK板。

此用例需要以下设置：

- CPU频率设置为800MHz（默认值）。
- DDR频率设置为528MHz（1056MT/s）。
- CM33核处于睡眠模式。
- GPU正在进行GLMark基准测试。
- G2D正在运行G2D测试程序。
- MIPI-CSI/DSI正在运行摄像头图片预览。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

1. 通过HDMI接口将显示器连接到MCIMX8ULP-EVK板。
2. 通过CSI接口将OV5640摄像头连接到电路板。
3. 打开ON/OFF开关（SW10），给电路板上电。
4. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用默认DTB配置（imx8ulp-evk.dtb）启动Linux镜像。
5. 在RTD控制台的菜单上，先按下“D”，再按“S”，使RTD进入睡眠模式。
6. 运行setup_video.sh（有关更多详细信息，请参阅第8节）。
7. 使用以下命令运行1080p、30fps的摄像头捕获：

```
gst-launch-1.0 -v v4l2src device=/dev/video0 ! \
"video/x-raw,format=YUY2,width=1920,height=1080,framerate=30/1" ! \
queue ! waylandsink
```

8. 使用以下命令运行G2D测试程序:

```
while true; do
  /opt/g2d_samples/g2d_basic_test
done
```

9. 使用以下命令运行G3D测试程序:

```
while true; do
  glmark2-es2-wayland -s 350x350
done
```

10. 开始测量功率并记录结果。

表23所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表23. “GPU2D + GPU3D + CSI/DSI” 用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	17.00	30.6	660.91	38
	BUCK1_LSW1_CPU_IV8	1.80	0.11	0.2		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.4		
	BUCK1_LSW3_CPU_IV8	1.80	3.10	5.5		
	BUCK1_LSW4_CPU_IV8	1.80	0.88	1.6		
	BUCK2_CPU_IV0	1.00	26.40	26.5		
	BUCK3_CPU_IV0	1.10	438.40	480.7		
	BUCK4_CPU_IV1	1.10	0.54	0.6		
	LDO1_CPU_IV1	1.09	4.30	4.7		
	LDO1_CPU_IV1_0V6	1.09	89.40	97.3		
	LDO2_CPU_3V3	3.31	1.20	3.9		
	LDO4_CPU_IV8	1.80	0.50	0.9		
	LDO5_CPU_3V0	2.93	0.00	0.0		

5.6 内存用例

本节介绍以下以存储器为中心的用例场景:

- memset
- memcpy
- Stream

memset和memcpy是perf-bench (基准套件的通用框架) 的一部分。

5.6.1 memset

此用例通过多种方式评估简单内存集的性能。

此用例需要以下设置:

- CPU频率设置为800MHz (默认值)。
- DDR频率设置为528MHz (1056MT/s)。
- 内存缓冲区的大小设置为1024MB。

- CM33核处于睡眠模式。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

1. 打开ON/OFF开关 (SW10)，给MCIMX8ULP-EVK板上电。
2. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用默认DTB配置 (imx8ulp-evk.dtb) 启动Linux镜像。
3. 在RTD控制台的菜单上，先按下“D”，再按下“S”，使RTD进入睡眠模式。
4. 运行setup.sh (有关详细信息，请参阅[第8节](#))。
5. 运行memset_loop.sh：

```
while true; do
    buff_size=`cat /proc/meminfo | grep CmaFree | awk '{print$2}'`
    perf bench -f simple mem memset -l 20000 -s ${buff_size}KB
done
```

6. 开始测量功率并记录结果。

[表24](#)所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表24. “memset”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	16.40	29.6	642.61	34
	BUCK1_LSW1_CPU_IV8	1.80	0.11	0.2		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW4_CPU_IV8	1.80	0.72	1.3		
	BUCK2_CPU_IV0	1.00	26.40	26.5		
	BUCK3_CPU_IV0	1.10	330.80	363.5		
	BUCK4_CPU_IV1	1.10	0.54	0.6		
	LDO1_CPU_IV1	1.09	3.10	3.3		
	LDO1_CPU_IV1_0V6	1.08	190.20	204.5		
	LDO2_CPU_3V3	3.31	1.20	3.9		
	LDO4_CPU_IV8	1.80	0.38	0.7		
	LDO5_CPU_3V0	2.93	0.00	0.0		

5.6.2 memcpy

此用例通过多种方式评估简单内存拷贝的性能。

此用例需要以下设置：

- CPU频率设置为800MHz (默认值)。
- DDR频率设置为528MHz (1056MT/s)。
- 内存缓冲区的大小设置为1024MB。
- CM33核处于睡眠模式。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

1. 打开ON/OFF开关 (SW10)，给MCIMX8ULP-EVK板上电。
2. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用默认DTB配置 (imx8ulp-evk.dtb) 启动Linux镜像。
3. 在RTD控制台的菜单上，先按下“D”，再按下“S”，使RTD进入睡眠模式。
4. 运行setup.sh (有关详细信息，请参阅[第8节](#))。
5. 运行memcpy_loop.sh：

```
while true; do
    buff_size=`cat /proc/meminfo | grep CmaFree | awk '{print$2}'`
    perf bench -f simple mem memcpy -l 20000 -s ${buff_size}KB
done
```

6. 开始测量功率并记录结果。

[表25](#)所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表25. “memcpy”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	16.50	29.8	543.09	34
	BUCK1_LSW1_CPU_IV8	1.80	0.05	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW4_CPU_IV8	1.80	0.88	1.6		
	BUCK2_CPU_IV0	1.00	26.20	26.3		
	BUCK3_CPU_IV0	1.10	342.90	376.8		
	BUCK4_CPU_IV1	1.10	0.63	0.7		
	LDO1_CPU_IV1	1.09	4.30	4.7		
	LDO1_CPU_IV1_0V6	1.09	82.70	90.2		
	LDO2_CPU_3V3	3.31	1.20	3.9		
	LDO4_CPU_IV8	1.80	0.38	0.7		
	LDO5_CPU_3V0	2.93	0.00	0.0		

5.6.3 Stream

Stream基准测试是一个简单的综合基准测试程序，用于测量简单向量内核的可持续内存带宽 (MB/s) 和相应的计算速率。

此用例需要以下设置：

- CPU频率设置为800MHz (默认值)。
- DDR频率设置为528MHz (1056MT/s)。
- 包括所有阶段 (“复制”、“缩放”、“添加”和“三元组”)。
- CM33核处于睡眠模式。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

1. 打开ON/OFF开关 (SW10) , 给MCIMX8ULP-EVK板上电。
2. 确保以下几点:
 - 在RTD控制台上, 启动PMS演示。
 - 在APD控制台上, 使用默认DTB配置 (imx8ulp-evk.dtb) 启动Linux镜像。
3. 在RTD控制台的菜单上, 先按下“D”, 再按下“S”, 使RTD进入睡眠模式。
4. 运行setup.sh (有关详细信息, 请参阅[第8节](#))。
5. 运行streamcpy_loop.sh:

```
while [ "1" == "1" ]; do
  taskset -c 0 stream -M 200M -N 1000 &
  taskset -c 1 stream -M 200M -N 1000
done
```

6. 开始测量功率并记录结果。

[表26](#)所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表26. “Stream” 用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	16.50	29.8	646.94	35
	BUCK1_LSW1_CPU_IV8	1.80	0.05	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW4_CPU_IV8	1.80	0.83	1.5		
	BUCK2_CPU_IV0	1.00	33.10	33.2		
	BUCK3_CPU_IV0	1.10	420.30	461.1		
	BUCK4_CPU_IV1	1.10	0.63	0.7		
	LDO1_CPU_IV1	1.09	4.30	4.7		
	LDO1_CPU_IV1_0V6	1.09	94.40	102.8		
	LDO2_CPU_3V3	3.31	1.20	3.9		
	LDO4_CPU_IV8	1.80	0.38	0.7		
	LDO5_CPU_3V0	2.93	0.00	0.0		

5.7 存储器 (eMMC) 用例

本节介绍了以下与存储器相关的用例场景:

- DD_RD_eMMC
- DD_WRT_eMMC

注: 在eMMC存储器上创建一个分区, 并在上面运行基准测试。

5.7.1 DD_RD_eMMC

此用例使用dd命令读取eMMC分区上的数据。

此用例需要以下设置:

- CPU频率设置为800MHz (默认值)。
- DDR频率设置为528MHz (1056MT/s)。

- CM33核处于睡眠模式。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

注：在循环中运行该用例，并以所需的时间间隔（建议时间间隔为1分钟）记录功率测量结果。

1. 打开ON/OFF开关（SW10），给MCIMX8ULP-EVK板上电。
2. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用默认DTB配置（imx8ulp-evk.dtb）启动Linux镜像。
3. 在RTD控制台的菜单上，先按下“D”，再按“S”，使RTD进入睡眠模式。
4. 运行setup.sh（有关详细信息，请参阅第8节）。
5. 把dd_read_emmc.sh复制到eMMC分区并运行。
6. 开始测量功率并记录结果。

表27所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表27. “DD_RD_eMMC”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	16.50	29.7	490.70	35
	BUCK1_LSW1_CPU_IV8	1.80	0.05	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW4_CPU_IV8	1.80	0.88	1.6		
	BUCK2_CPU_IV0	1.00	26.30	26.4		
	BUCK3_CPU_IV0	1.10	323.40	355.5		
	BUCK4_CPU_IV1	1.10	0.54	0.6		
	LDO1_CPU_IV1	1.10	3.80	4.2		
	LDO1_CPU_IV1_0V6	1.09	54.50	59.6		
	LDO2_CPU_3V3	3.31	1.20	3.9		
	LDO4_CPU_IV8	1.80	0.38	0.7		
	LDO5_CPU_3V0	2.93	0.00	0.0		

5.7.2 DD_WRT_eMMC

此用例使用dd命令将数据写入eMMC分区。

此用例需要以下设置：

- CPU频率设置为800MHz（默认值）。
- DDR频率设置为528MHz（1056MT/s）。
- CM33核处于睡眠模式。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

注：此命令覆盖用户空间区域偏移量为10 GB的数据。请谨慎使用此命令以避免对rootfs造成任何损坏。

要配置和运行该用例，请执行以下步骤：

注：在循环中运行该用例，并以所需的时间间隔（建议时间间隔为1分钟）记录功率测量结果。

1. 打开ON/OFF开关（SW10），给MCIMX8ULP-EVK板上电。
2. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用默认DTB配置（imx8ulp-evk.dtb）启动Linux镜像。
3. 在RTD控制台的菜单上，先按下“D”，再按“S”，使RTD进入睡眠模式。
4. 运行setup.sh（有关详细信息，请参阅[第8节](#)）。
5. 把dd_write_emmc.sh复制到eMMC分区并运行。
6. 开始测量功率并记录结果。

[表28](#)所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表28. “DD_WRT_eMMC”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	16.60	29.9	415.32	35
	BUCK1_LSW1_CPU_IV8	1.80	0.05	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW4_CPU_IV8	1.80	0.83	1.5		
	BUCK2_CPU_IV0	1.00	26.40	26.4		
	BUCK3_CPU_IV0	1.10	261.90	287.9		
	BUCK4_CPU_IV1	1.10	0.54	0.6		
	LDO1_CPU_IV1	1.10	3.50	3.8		
	LDO1_CPU_IV1_0V6	1.10	38.80	42.5		
	LDO2_CPU_3V3	3.31	1.20	4.0		
	LDO4_CPU_IV8	1.80	5.70	10.3		
	LDO5_CPU_3V0	2.93	0.00	0.0		

5.8 产品用例

本节介绍了以下用例场景：

- 并行电子墨水屏翻页
- 并行电子墨水屏部分更新
- 机器视觉
- eIQ基准测试
- UAC音频播放
- 不间断运行显示器，1fps刷新率

5.8.1 并行电子墨水屏翻页

对于此用例，EPDC扩展端口连接到并行VB3300-FOC电子墨水面板。在本用例中，两页内容缓冲区已准备就绪（无渲染）。使用FBIOPAN_DISPLAY_IOCTL翻转缓冲区。

此用例需要以下设置：

- CPU频率设置为800MHz（默认值）。
- CM33核处于睡眠模式。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

注：在循环中运行该用例，并以所需的时间间隔（建议时间间隔为1分钟）记录功率测量结果。

1. 通过并行接口将EPDC面板连接到电路板。
2. 打开ON/OFF开关（SW10），给MCIMX8ULP-EVK板上电。
3. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用DTB配置imx8ulp-evk-epdc.dtb启动Linux镜像。
4. 在RTD控制台的菜单上，先按下“D”，再按“S”，使RTD进入睡眠模式。
5. 运行setup.sh（有关详细信息，请参阅[第8节](#)）。
6. 运行以下命令：

```
/unit_tests/Display/mxc_epdc_v2_fb_test.out -n 20
```

7. 开始测量功率并记录结果。

[表29](#)所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表29. “并行电子墨水屏翻页”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	16.90	30.4	453.36	32
	BUCK1_LSW1_CPU_IV8	1.80	0.11	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	1.70	3.1		
	BUCK1_LSW4_CPU_IV8	1.80	2.70	4.9		
	BUCK2_CPU_IV0	1.00	26.60	26.7		
	BUCK3_CPU_IV0	1.10	266.30	292.9		
	BUCK4_CPU_IV1	1.10	0.63	0.7		
	LDO1_CPU_IV1	1.09	3.70	4.1		
	LDO1_CPU_IV1_0V6	1.09	71.40	77.9		
	LDO2_CPU_3V3	3.31	1.10	3.6		
	LDO4_CPU_IV8	1.80	0.38	0.7		
	LDO5_CPU_3V0	2.92	0.00	0.0		

5.8.2 并行电子墨水屏部分更新

对于此用例，EPDC扩展端口连接到并行VB3300-FOC电子墨水面板。在这个用例中，仅在屏幕上的一个小矩形部分（10%-20%）显示更新内容。

此用例需要以下设置：

- CPU频率设置为800MHz（默认值）。
- CM33核处于睡眠模式。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

注：在循环中运行该用例，并以所需的时间间隔（建议时间间隔为1分钟）记录功率测量结果。

1. 通过并行接口将EPDC面板连接到电路板。
2. 打开ON/OFF开关（SW10），给MCIMX8ULP-EVK板上电。
3. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用DTB配置imx8ulp-evk-epdc.dtb启动Linux镜像。
4. 在RTD控制台的菜单上，先按下“D”，再按“S”，使RTD进入睡眠模式。
5. 运行setup.sh（有关详细信息，请参阅[第8节](#)）。
6. 运行以下命令：

```
/unit_tests/Display/mxc_epdc_v2_fb_test.out -n 19
```

7. 开始测量功率并记录结果。

[表30](#)所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表30.“并行电子墨水屏部分更新”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	16.80	30.3	463.04	32
	BUCK1_LSW1_CPU_IV8	1.80	0.11	0.2		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	1.70	3.1		
	BUCK1_LSW4_CPU_IV8	1.80	2.20	4.0		
	BUCK2_CPU_IV0	1.00	32.90	33.0		
	BUCK3_CPU_IV0	1.10	272.40	299.5		
	BUCK4_CPU_IV1	1.10	0.63	0.7		
	LDO1_CPU_IV1	1.09	3.70	4.0		
	LDO1_CPU_IV1_OV6	1.09	69.30	75.6		
	LDO2_CPU_3V3	3.31	1.10	3.6		
	LDO4_CPU_IV8	1.80	0.38	0.7		
	LDO5_CPU_3V0	2.92	0.00	0.0		

5.8.3 机器视觉

此用例检测视频文件中通过OV5640摄像头捕获的对象。

要获得“机器视觉”用例所需的文件，请在/usr/bin/ tensorflow-lite-2.12.1/examples中使用以下命令（对于不同的BSP版本，路径可能会不同）：

```
wget https://github.com/google-coral/test_data/raw/master/
ssd_mobilenet_v2_coco_quant_postprocess.tflite
wget https://github.com/google-coral/test_data/raw/master/coco_labels.txt
```

此用例需要以下设置：

- CPU频率设置为800MHz（默认值）。

- 使用OV5640以1080p的分辨率和每秒30帧的速度捕获YUY2数据。
- 使用G2D执行色彩空间转换并缩小到480p。
- 使用基于文件的输入进行对象检测。
- 为了模拟应用压力，对每个CA35核施加100%压力。
- CM33核处于睡眠模式。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

注：在循环中运行该用例，并以所需的时间间隔（建议时间间隔为1分钟）记录功率测量结果。

1. 通过HDMI接口将显示器连接到MCIMX8ULP-EVK板。
2. 打开ON/OFF开关（SW10），给电路板上电。
3. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用默认DTB配置（imx8ulp-evk.dtb）启动Linux镜像。
4. 在RTD控制台的菜单上，先按下“D”，再按“S”，使RTD进入睡眠模式。
5. 运行setup.sh（有关详细信息，请参阅第8节）。
6. 使用基于文件的输入进行对象检测。
7. 将MV.sh复制到rootfs中的/usr/bin/tensorflow-lite-2.12.1/examples，并运行/usr/bin/tensorflow-lite-2.12.1/examples/MV.sh。有关详细信息，请参阅第8节。

注：要将测试视频记录到此shell文件，请使用以下命令：

```
gst-launch-1.0 v4l2src device=/dev/video0 ! videoconvert ! 'video/x-raw,format=YUY2,width=640,height=480,framerate=(fraction)30/1' ! filesink location=yuv.raw
```

8. 开始测量功率并记录结果。

表31所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表31. “机器视觉”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	17.00	30.7	598.47	35
	BUCK1_LSW1_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.4		
	BUCK1_LSW3_CPU_IV8	1.80	2.60	4.6		
	BUCK1_LSW4_CPU_IV8	1.80	0.90	1.7		
	BUCK2_CPU_IV0	1.00	26.60	26.7		
	BUCK3_CPU_IV0	1.10	384.90	422.3		
	BUCK4_CPU_IV1	1.10	0.63	0.7		
	LDO1_CPU_IV1	1.09	4.10	4.4		
	LDO1_CPU_IV1_0V6	1.09	76.60	83.5		
	LDO2_CPU_3V3	3.31	1.10	3.7		
	LDO4_CPU_IV8	1.80	6.50	11.7		
LDO5_CPU_3V0	2.93	0.00	0.0			

5.8.4 eIQ基准测试

此用例在循环中运行benchmark_model应用程序，以进行eIQ基准测试。

此用例需要以下设置：

- CPU频率设置为800MHz（默认值）。
- 运行eIQ基准测试（标准型号）。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

注：在循环中运行该用例，并以所需的时间间隔（建议时间间隔为1分钟）记录功率测量结果。

1. 打开ON/OFF开关（SW10），给MCIMX8ULP-EVK板上电。
2. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用默认DTB配置（imx8ulp-evk.dtb）启动Linux镜像。
3. 在RTD控制台的菜单上，先按下“D”，再按“S”，使RTD进入睡眠模式。
4. 运行setup.sh（有关详细信息，请参阅[第8节](#)）。
5. 将ML.sh复制到rootfs中的/usr/bin/tensorflow-lite-2.12.1/examples，并运行/usr/bin/tensorflow-lite-2.12.1/examples/ML.sh 1。
6. 开始测量功率并记录结果。

[表32](#)所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表32. “eIQ基准测试”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	16.40	29.6	452.30	35
	BUCK1_LSW1_CPU_IV8	1.80	0.05	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW4_CPU_IV8	1.80	0.88	1.6		
	BUCK2_CPU_IV0	1.00	26.60	26.7		
	BUCK3_CPU_IV0	1.10	305.10	335.5		
	BUCK4_CPU_IV1	1.10	0.54	0.6		
	LDO1_CPU_IV1	1.10	3.40	3.8		
	LDO1_CPU_IV1_0V6	1.10	37.90	41.6		
	LDO2_CPU_3V3	3.31	1.20	3.9		
	LDO4_CPU_IV8	1.80	0.38	0.7		
	LDO5_CPU_3V0	2.92	0.00	0.0		

5.8.5 UAC音频播放

此用例需要在i.MX 8ULP处理器上创建一个Linux UAC小工具设备，通过USB接口将该设备连接到Ubuntu PC（主机），通过UAC接收主机音频，并使用板载编解码器解码/播放音频。

此用例需要以下设置：

- CPU频率设置为800MHz（默认值）。
- DDR频率设置为528MHz（1056MT/s）。
- CM33核处于睡眠模式。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

注：在循环中运行该用例，并以所需的时间间隔（建议时间间隔为1分钟）记录功率测量结果。

1. 准备一台安装了最新Ubuntu的PC主机。
2. 使用Type-C USB电缆将MCIMX8ULP-EVK板的USB0连接到Ubuntu PC。
3. 打开ON/OFF开关（SW10），给电路板上电。
4. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用默认DTB配置（imx8ulp-evk.dtb）启动Linux镜像。
5. 在RTD控制台的菜单上，先按下“D”，再按“S”，使RTD进入睡眠模式。
6. 在电路板上运行setup.sh（有关详细信息，请参阅第8节）。
7. 在电路板上运行uac2_gadget.sh（有关详细信息，请参阅第8节）。
8. 为了确保Ubuntu PC可以使用板载编解码器（WM8960）播放音频，请通过运行[aplay -l](#)命令查看是否在Ubuntu PC上检测到WM8960设备，如图5所示。

注：如果在运行[aplay -l](#)命令后未列出UAC小工具设备，请查看uac2_gadget.sh脚本文件，以确保最后一个命令（`echo ci_hdrc.0 > UDC #8mm 8mn 8ulp`）中的UDC与当前路径上可用的UDC小工具的名称相一致。

```
root@imx8ulpevk:~# aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: imxspdif [imx-spdif], device 0: S/PDIF PCM snd-soc-dummy-dai-0 [S/PDIF PCM snd-soc-dummy-dai-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 1: wm8960audio [wm8960-audio], device 0: rpmsg hifi rpmsg-codec-wm8960.0-001a-0 []
  Subdevices: 0/1
  Subdevice #0: subdevice #0
card 2: btscoscoaudio [bt-sco-audio], device 0: 29890000.sai-bt-sco-pcm-wb bt-sco-pcm-wb-0 [29890000.sai-bt-sco-pcm-wb bt-sco-pcm-wb-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 3: UAC2Gadget [UAC2_Gadget], device 0: UAC2 PCM [UAC2 PCM]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

图5. MCIMX8ULP-EVK板上的“aplay -l”命令输出

9. 为了确保MCIMX8ULP-EVK板能够记录来自UAC小工具设备的音频，请通过运行[arecord -l](#)命令查看在该电路板上是否检测到UAC小工具设备，如图6所示。

注：如果运行[arecord -l](#)命令后未列出WM8960音频设备，请查看uac2_gadget.sh脚本文件，以确保最后一个命令（`echo ci_hdrc.0 > UDC #8mm 8mn 8ulp`）中的UDC与当前路径上可用的UDC小工具的名称相一致。

```
root@imx8ulpvdk:~# arecord -l
**** List of CAPTURE Hardware Devices ****
card 1: wm8960audio [wm8960-audio], device 0: rpmsg hifi rpmsg-codec-wm8960.0-001a-0 []
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 2: btscoaudio [bt-sco-audio], device 0: 29890000.sai-bt-sco-pcm-wb bt-sco-pcm-wb-0 [29890000.sai-bt-sco-pcm-wb bt-sco-pcm-wb-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 3: UAC2Gadget [UAC2_Gadget], device 0: UAC2 PCM [UAC2 PCM]
  Subdevices: 0/1
  Subdevice #0: subdevice #0
```

图6. MCIMX8ULP-EVK板上的“arecord -l”命令输出

10. 为了确保Ubuntu PC可以向UAC小工具设备播放音频，请查看Ubuntu PC上是否检测到UAC小工具设备，如图7所示。

```
root@imx8mnevdk:~# aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: imxspdif [imx-spdif], device 0: S/PDIF PCM snd-soc-dummy-dai-0 [S/PDIF PCM snd-soc-dummy-dai-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 2: btscoaudio [bt-sco-audio], device 0: 30020000.sai-bt-sco-pcm-wb bt-sco-pcm-wb-0 [30020000.sai-bt-sco-pcm-wb bt-sco-pcm-wb-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 3: wm8524audio [wm8524-audio], device 0: HiFi wm8524-hifi-0 [HiFi wm8524-hifi-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 3: wm8524audio [wm8524-audio], device 1: HiFi-ASRC-FE (*) []
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 4: Gadget [Linux USB Audio Gadget], device 0: USB Audio [USB Audio]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

图7. 在Ubuntu PC上检测UAC小工具设备

11. 使用此命令在Ubuntu PC上播放音乐：

```
aplay -Dhw:4,0 audio48k16b2c.wav -vv
```

注：用在Ubuntu PC上运行`aplay -l`命令后得到的UAC小工具设备索引替换上面命令中“-Dhw:4,0”中的4’。

12. 使用此命令接收音乐数据并在MCIMX8ULP-EVK板上播放：

```
arecord -Dhw:3,0 -c2 -r48000 -fS16_LE |
aplay -Dhw:1,0
```

注：用在MCIMX8ULP-EVK板上运行`arecord -l`命令后得到的UAC小工具设备索引替换上面命令中“-Dhw:3,0”中的3’。

注：用在MCIMX8ULP-EVK板上运行`aplay -l`命令后得到的WM8960音频卡索引替换上面命令中“-Dhw:1,0”中的“1”。

13. 查看通过MCIMX8ULP-EVK板上的3.5mm音频插孔能否听到音乐。

14. 开始测量功率并记录结果。

表33所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表33. “UAC音频播放”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	16.50	29.7	381.09	36
	BUCK1_LSW1_CPU_IV8	1.80	0.05	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	7.80	14.1		
	BUCK1_LSW3_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW4_CPU_IV8	1.80	0.76	1.4		
	BUCK2_CPU_IV0	1.00	26.60	26.7		
	BUCK3_CPU_IV0	1.10	239.00	262.9		
	BUCK4_CPU_IV1	1.10	0.63	0.7		
	LDO1_CPU_IV1	1.10	3.40	3.8		
	LDO1_CPU_IV1_OV6	1.10	29.70	32.6		
	LDO2_CPU_3V3	3.31	2.60	8.5		
	LDO4_CPU_IV8	1.80	0.38	0.7		
	LDO5_CPU_3V0	2.93	0.00	0.0		

5.8.6 不间断运行显示器，1fps刷新率

此用例展示了CM33核每秒从睡眠模式唤醒，更新缓冲区以进行显示。

此用例需要以下设置：

- APD域处于挂起（断电）模式。
- DDR处于保留模式。
- CM33核重新初始化LPAV域，以控制MIPI-DSI和DCNano。
- 通过FlexSPI在pSRAM中分配一个帧缓冲区。
- DCNano从pSRAM获取显示内容。
- CM33核每秒从睡眠模式唤醒，更新缓冲区以进行显示。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

1. 通过MIPI接口将RK055HDMIPI4M面板连接到MCIMX8ULP-EVK板。
2. 从<https://mcuxpresso.nxp.com/>下载最新的CM33 SDK（本文提供的功率测量结果使用了SDK 2.14.1）。
3. 构建“low_power_display”发布工程，然后使用m33_image镜像构建flash.bin镜像，m33_image镜像可以在{CURRENT_REPO}\armgcc\release\sdk20-app.bin或{CURRENT_REPO}\iar\Release\sdk20-app.bin中找到。

注：有关如何构建flash.bin镜像的详细信息，请参阅SDK_2_xx_x_EVK-MIMX8ULP/docs中《面向EVK-MIMX8ULP和EVK9-MIMX8ULP的MCUXpresso SDK快速入门》中的第6节第5步。

4. 将电路板上的SW5[8:1]设置更改为01xx_xxxx（串行下载模式）。

5. 将flash.bin二进制镜像下载到eMMC：

注：新的flash.bin镜像与默认flash.bin镜像不同。因此，对于其他需要默认二进制镜像的用例，必须下载并使用flash_singleboot_default.bin镜像。

```
uuu -b emmc flash.bin
```

注： RK055AHD091和RK055MHD091均属于RK055HDMIPI4M。如果镜像无法显示，请在lcdif_support.h文件中将USE_MIPI_PANEL值从MIPI_PANEL_RK055MHD091更改为MIPI_PANEL_RK055AHD091：

```
/*
 * Copyright 2023 NXP
 *
 * SPDX-License-Identifier: BSD-3-Clause
 */
#ifndef _LCDIF_SUPPORT_H_
#define _LCDIF_SUPPORT_H_

/*****
 * Definitions
 *****/

#define DEMO_LCDIF LCDIF
#define DEMO_LCDIF_IRQn DCNano_IRQn
#define DEMO_LCDIF_IRQHandler DCNano_IRQHandler

#define MIPI_PANEL_RK055AHD091 0 /* 720 * 1280 */
#define MIPI_PANEL_RK055MHD091 1 /* 720 * 1280 */

#ifndef USE_MIPI_PANEL
#define USE_MIPI_PANEL MIPI_PANEL_RK055MHD091
#endif
#endif
```

6. 确保将电路板配置为默认启动设置（SW5[8:1]=1000_xx00（单启动-eMMC模式））。

7. 打开ON/OFF开关（SW10），给电路板上电。

8. 确保以下几点：

- 在RTD控制台上，启动Always On Display (AOD) 演示。
- 在APD控制台上，使用默认DTB配置（imx8ulp-evk.dtb）启动Linux镜像。

9. 使Linux内核进入挂起（断电）模式：

```
echo mem > /sys/power/state
```

10. 在RTD控制台上，按下“Z”，运行低功耗显示器。

11. 查看整个屏幕是否被颜色填满，其中红色、绿色和蓝色每秒切换。

12. 开始测量功率并记录结果。

表34所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表34. “不间断运行显示器，1fps刷新率”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.8	6.00	10.8	90.329	未测量芯片温度，因为CA35核被挂起。
	BUCK1_LSW1_CPU_IV8	1.8	1.20	2.2		
	BUCK1_LSW2_CPU_IV8	1.8	0.00	0.0		
	BUCK1_LSW3_CPU_IV8	1.8	1.30	2.3		
	BUCK1_LSW4_CPU_IV8	1.8	0.00	0.0		
	BUCK2_CPU_IV0	1.0	31.50	31.6		
	BUCK3_CPU_IV0	1.0	43.20	43.4		
	BUCK4_CPU_IV1	1.1	0.09	0.1		
	LDO1_CPU_IV1	0.0	0.00	0.0		
	LDO1_CPU_IV1_0V6	0.0	0.00	0.0		
	LDO2_CPU_3V3	3.3	0.00	0.0		
	LDO4_CPU_IV8	1.8	0.00	0.0		
	LDO5_CPU_3V0	2.9	0.00	0.0		

5.9 系统级功率估计用例

本节介绍了以下用例场景：

- 电池
- 256KB L2缓存

5.9.1 电池

i.MX 8ULP处理器的VBAT模式是一种低功耗模式，其中只有VBAT域的电源保持开启。VBAT模式是芯片级的状态，具有以下条件：

- 除VDD_VBAT42外，所有电源均处于外部关闭状态。
- VDD_VBAT42开启，且在i.MX 8ULP数据手册中规定的电压范围内。
- 安全实时时钟（SRTC）保持并正常运行。
- 保留防篡改逻辑电路。
- SNVS处于1.8V DGO（VBAT输入：3V）。
- 关闭APD和LPAV中的所有时钟和PLL。

在VBAT模式下，应用程序关闭，电池保留SRTC和防篡改逻辑电路。

要配置和运行该用例，请执行以下步骤：

1. 打开ON/OFF开关（SW10），给MCIMX8ULP-EVK板上电。
2. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用默认DTB配置（imx8ulp-evk.dtb）启动Linux镜像。
3. 按下ON/OFF按钮（SW6）3秒钟，关闭i.MX 8ULP处理器。
4. 测量功率并记录结果。

表35所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表35. “电池”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (μ A)	平均功率 (μ W)	平均功率之和 (μ W)	0区芯片温度 ($^{\circ}$ C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	0.00	0.00	0.000	8.16	未测量芯片温度，因为CA35核被挂起。
	BUCK1_LSW1_CPU_IV8	0.00	0.00	0.000		
	BUCK1_LSW2_CPU_IV8	0.00	0.00	0.000		
	BUCK1_LSW3_CPU_IV8	0.00	0.00	0.000		
	BUCK1_LSW4_CPU_IV8	0.00	0.00	0.000		
	BUCK2_CPU_IV0	0.00	0.00	0.000		
	BUCK3_CPU_IV0	0.00	0.00	0.000		
	BUCK4_CPU_IV1	0.00	0.00	0.000		
	LDO1_CPU_IV1	0.00	0.00	0.000		
	LDO1_CPU_IV1_0V6	0.00	0.00	0.000		
	LDO2_CPU_3V3	0.00	0.00	0.000		
	LDO4_CPU_IV8	0.00	0.00	0.000		
LDO5_CPU_3V0	2.95	2.77	8.158			

注：对于这些用例，功率单位为 μ W：

- 挂起模式——RTD断电
- 电池

5.9.2 256KB L2缓存

此用例要求在CA35核上运行CoreMark基准测试，并将L2缓存大小减半。

此用例需要以下设置：

- CPU频率设置为800MHz（默认值）。
- DDR频率设置为528MHz (1056MT/s)。
- Arm L2缓存的大小设置为256KB（默认大小为512KB）。
- RTD处于睡眠模式。
- 所有未使用的PLL均关闭，未使用的时钟进行门控。

要配置和运行该用例，请执行以下步骤：

注：在循环中运行该用例，并以所需的时间间隔（建议时间间隔为1分钟）记录功率测量结果。

1. 将MCIMX8ULP-EVK板上的SW5[8:1]设置更改为0lxx_xxxx（串行下载模式）。
2. 将flash_l2_256k_cache.bin二进制镜像下载到eMMC：

注：flash_l2_256k_cache.bin镜像与默认二进制镜像（flash.bin）不同。因此，对于其他需要默认二进制镜像的用例，必须下载并使用flash_singleboot_default.bin镜像。

```
uuu -b emmc flash_l2_256k_cache.bin
```

3. 打开ON/OFF开关（SW10），给电路板上电。
4. 确保以下几点：
 - 在RTD控制台上，启动PMS演示。
 - 在APD控制台上，使用默认DTB配置（imx8ulp-evk.dtb）启动Linux镜像。
5. 运行setup.sh。

6. 运行coremark_loop.sh:

```
while [ "1" == "1" ]
do
./coremark > /dev/null 2>&1
done
```

7. 在RTD控制台的菜单上，先按下“D”，再按下“S”，使RTD进入睡眠模式。

8. 测量功率并记录结果。

表36所示为当此用例应用于i.MX 8ULP处理器时的测量结果。

表36. “256KB L2缓存”用例的功率测量结果

电源组	电源轨	平均电压 (V)	平均电流 (mA)	平均功率 (mW)	平均功率之和 (mW)	0区芯片温度 (°C)
GROUP_SOC_FULL	BUCK1_CPU_IV8	1.80	16.40	29.6	488.87	34
	BUCK1_LSW1_CPU_IV8	1.80	0.05	0.1		
	BUCK1_LSW2_CPU_IV8	1.80	4.60	8.3		
	BUCK1_LSW3_CPU_IV8	1.80	0.00	0.0		
	BUCK1_LSW4_CPU_IV8	1.80	0.88	1.6		
	BUCK2_CPU_IV0	1.00	33.00	33.1		
	BUCK3_CPU_IV0	1.10	341.00	374.7		
	BUCK4_CPU_IV1	1.10	0.63	0.7		
	LDO1_CPU_IV1	1.10	3.40	3.7		
	LDO1_CPU_IV1_0V6	1.10	29.50	32.5		
	LDO2_CPU_3V3	3.31	1.20	3.9		
	LDO4_CPU_IV8	1.80	0.38	0.7		
	LDO5_CPU_3V0	2.92	0.00	0.0		

6 降低功耗的建议

系统总功耗取决于软件优化和系统硬件实现。

以下建议有助于降低系统功耗：

- 当不使用时钟或模块时，通过在外设时钟控制器（PCC）模块中配置寄存器来应用时钟门控。
- 对于活动模式，请使用能够满足应用程序要求的最慢频率。
- 尽可能减少活动PLL的数量。启用的PLL会消耗几毫安的电流。
- 实施核心VFS和系统总线缩放：对Arm核实施VFS，并缩放AXI、AHB和IPG总线时钟的频率，可以显著降低功耗。然而，降低操作频率可能会延长访问DDR的时间，进而增加DDR I/O和存储器的功耗。因此，在每种模式下，都应仔细权衡这些因素，以量化它们对系统功率的总体影响。
- 尽可能使恩智浦i.MX 8ULP处于低功耗模式，并使其进入支持当前应用程序要求的最低功耗模式。

- 对于每种操作模式，使用能够满足i.MX 8ULP数据手册中电压规格要求的最低电压（考虑电源容差）。
- DDR接口优化：
 - 精心布局DDR存储器的电路板布线，使PCB走线长度尽可能短。
 - 使用适当的输出驱动阻抗，为DDR接口引脚提供良好的阻抗匹配。为了节省DDR I/O引脚的电流，选择尽可能低的驱动强度来提供所需的性能。
 - 当DDR存储器处于自刷新模式时，将i.MX 8ULP DDR接口引脚设置为high-Z。当DDR存储器处于自刷新模式时，关闭VDDQ_DDR、VDDQX_DDR以及LPDDR3/LPDDR4/LPDDR4x (VDDQ)的I/O电源。
 - LPDDR3/LPDDR4/LPDDR4x存储器产品使用最新工艺技术，可以显著降低DDR设备和DDR I/O的功耗。

其中一些建议已经在Linux BSP和/或SDK中实现。每个单独用户的系统均可以进行进一步的优化。

注：未来的软件版本计划进一步优化功率。要获取最新的软件版本，请访问<https://www.nxp.com.cn/imxsw>。

注：有关如何通过软件优化i.MX 8ULP的详细信息，请参阅《i.MX 8 ULP功耗优化》(AN13951)。

以下各小节介绍了一些可以降低系统功耗的常见系统优化技术。

- [第6.1节 “快速运行和空闲”](#)
- [第6.2节 “时钟门控”](#)
- [第6.3节 “减少PLL”](#)
- [第6.4节 “核心VFS和系统总线缩放”](#)
- [第6.5节 “使用LPDDR4x而非LPDDR4”](#)
- [第6.6节 “降低DDR频率”](#)
- [第6.7节 “DDR接口优化”](#)
- [第6.8节 “PHY的电源门控”](#)
- [第6.9节 “工作负载分布”](#)
- [第6.10节 “使用SSRAM减少DDR访问时间”](#)
- [第6.11节 “通过热管理减少漏电电流”](#)
- [第6.12节 “标称驱动模式”](#)

6.1 快速运行和空闲

恩智浦测试和各项研究表明，对于大多数客户用例，最佳的功率/能耗管理协议是以最大速度运行内核，以应对工作负载，然后尽快降至最低功耗模式。对于需要不断处理数据的数据用例（例如，低延迟音频播放），这种策略无法提供最佳的节能。但此策略适用于其他标准工作负载。对每个应用程序都应仔细权衡速度和功耗，以量化对系统功耗/能耗的总体影响。

建议尽可能使i.MX 8ULP处理器处于低功耗模式。

6.2 时钟门控

i.MX 8ULP的内部时钟生成与控制（CGC）和外设时钟控制器（PCC）模块提供了一种可编程方法，用于在不使用某些模块时禁用它们的时钟源。要实现正确的时钟门控，需要始终配置CGC和PCC寄存器。

这是减少能源浪费的最简单方法之一。无论在SoC还是PCB上，驱动任何未使用的信号，都只是对线路和该信号的负载电容进行充电和放电。在默认情况下，恩智浦BSP发布的软件实施时钟门控。

6.3 减少PLL

每个PLL块在活动状态都会消耗大量电力。每个应用都有独特的要求，如果可能，应该减少活动PLL的数量。i.MX 8ULP中的时钟生成和控制（CGC）模块为时钟根选择提供可编程控制。它支持应用程序中的公共PLL根时钟，并在工作时减少活动PLL的数量。

当进入低功耗状态（部分活动/睡眠/深度睡眠）或低功耗操作状态（音频播放）时，减少活动PLL的数量，这样可以降低这些状态的电源需求。确保应用在转回全操作状态时考虑PLL重新锁定时间。

6.4 核心VFS和系统总线缩放

对Arm核应用电压和频率缩放（VFS），并缩放（非动态）NOC、AXI、AHB和IPG系统总线时钟的频率，可以显著降低VDD_DIG1和VDD_DIG2域的功耗。然而，降低系统频率的操作会导致访问DDR的时间变长，进而增加特定用例的能耗。对每个模式都要权衡速度和功耗，以量化对系统功耗的总体影响。

6.5 使用LPDDR4x而非LPDDR4

要在运行时实现更低的I/O功耗，请使用LPDDR4x而不是LPDDR4。LPDDR4x有助于降低活动模式下的功耗，例如在空闲和音频用例中。

MCIMX8ULP-EVK板支持一个512 Meg x 32（两通道x 16 I/O）LPDDR4/LPDDR4x DRAM芯片（MT53E512M32D1ZW-046 WT:B），提供2 GB RAM内存。此DRAM存储器同时支持LPDDR4和LPDDR4x模式。默认情况下，电路板使用LPDDR4模式。

在MCIMX8ULP-EVK板中，默认DRAM I/O电压为1.1V。但是，为了支持LPDDR4x模式，该电路板有一个LDO稳压器，可产生0.6V电源轨。要启用LPDDR4x模式，需要进行硬件改造。

表37所示为LPDDR4和LPDDR4x模式的电阻配置，其中√表示“安装”，X表示“未安装”。

表37. LPDDR4和LPDDR4x模式的电阻配置

模式	R41	R37	R43	R34	R22	R20	R26	R33	R38
LPDDR4	X	√	X	X	√	√	X	√	X
LPDDR4x	√	X	√	√	X	X	√	X	√

要在电路板设计中支持LPDDR4x模式，可以使用恩智浦PCA9460B生成0.6 V DDR I/O电源轨。

与LPDDR4相比，LPDDR4x的主要变化为使I/O电压（VDDQ）降低了近50%（1.1V至0.6V）。总线的存储器侧和系统侧的功耗都降低了。

6.6 降低DDR频率

如前所述，DDR I/O总线频率也会影响DDR I/O电流。在第6.5节中的公式中，该频率用“F”表示，必须平衡它与系统操作要求的关系。对于低功耗工作状态，降低DDR总线频率可以提供额外的节能效果。

6.7 DDR接口优化

可通过以下方式优化DDR接口：

- 精心布局DDR存储器的电路板布线，使PCB走线长度尽可能短。走线长度越长、过孔越多，会为信号增加更多的PCB电容，进而导致信号路径上浪费的能源增加。
- 保持片上端接（ODT）值尽可能低。所使用的端接对DDR接口引脚的功耗有很大影响。为了确保ODT变化不会降低总线信号的完整性，需要对DDR接口进行仿真。
- 使用适当的输出驱动阻抗为DDR接口引脚提供良好的阻抗匹配。选择能提供所需性能的最低驱动强度，以减少流经DDR I/O引脚的电流。请记住，必须进行仿真以确保信号的完整性。
- 使用最新工艺技术的DDR存储器产品可以显著减少DDR设备和DDR I/O的功耗。
- 确定DDR存储器的大小非常重要。如果您选择了4GB的存储器，却只使用2GB，则会浪费未使用的2GB DDR的刷新电流。
- ECC DDR区域的大小选择非常重要，因为区域越大使用的能源越多。

6.8 PHY的电源门控

在寻求节能时，未使用模块的PHY经常被忽视。多个PHY包含本地PLL（或时钟电路）和电压参考，这些组件即使在不使用时也会消耗电力。以太网、MIPI、HDMI、PCIe和USB等高速PHY也是如此。

i.MX 8ULP处理器使用uPower控制器来管理SoC中的电源门控。有关更多详细信息，请参阅《i.MX 8ULP处理器参考手册》（IMX8ULPRM）中的“微功率控制器子系统（uPower控制器）”一章。

6.9 工作负载分布

分布式工作负载概念是指审查系统需求，并确定哪个SoC块最适合哪项任务。理想情况下，在分散了工作负载后，系统可以很快返回到“等待/停止”（WAIT/STOP）状态。它适用于多核分布以及适合HiFi4 DSP、ML引擎或图形核心的功能。

系统设计人员应确保设计使用i.MX 8ULP上合适的核来处理特定工作负载或任务，以实现最大效率。如果系统能够更快地恢复到低功耗状态（参见[第6.1节](#)），可以显著节省电力。

6.10使用SSRAM减少DDR访问时间

将高频访问的代码加载到同步静态随机存取存储器（SSRAM）中，可以大幅节省电力。这样，i.MX 8ULP处理器和DDR存储器的电流消耗都会减少。使用SSRAM的另一个优势是性能提升，因为DDR存储器访问时间延迟了该代码。

i.MX BSP确保所有常用的低功耗例程都位于内部SSRAM的Arm Trusted Firmware（ATF）中。

6.11 通过热管理减少漏电电流

热管理也是降低功率的关键因素。随着温度的升高，芯片内每个栅极的SoC栅极漏电电流会相应地增加。

数百万个高栅极漏电电流的累积，对降低总功耗有重大影响。如前所述，任何节能措施都会降低SoC的温度，并提高芯片的生命周期可靠性。

由于每个系统都是独一无二的，系统设计人员必须确保SoC的工作温度尽可能低，以减少漏电电流损失。如果无法通过软件控制达到这个温度，则设计人员必须通过散热片或其他热管理方法来去除SoC的热量。

有关散热指南，请参阅《i.MX 8ULP硬件开发指南》(IMX8ULPHDG)。

6.12 标称驱动模式

默认情况下，恩智浦提供的Linux BSP GA版本将系统配置为以“超驱动”（Overdrive）模式（ODM）运行。某些用户应用不需要此“超驱动”模式及其相关性能。在这种情况下，用户可以转换到i.MX 8ULP数据手册中定义的标称驱动模式（NDM）。

7 在最低功耗模式下控制i.MX 8ULP电源

在待机状态下，许多应用程序预计将使用CM33-PD/CA35-PD功率模式组合，以最大限度地减少功耗。这些组合是i.MX 8ULP上的最低功率组合。

为了最大限度地减少功耗，I/O电压电源VDD_PTA、VDD_PT B、VDD_PTC、VDD_PTD、VDD_PTE和VDD_PTF必须保持通电状态。为了避免漏电电流，只有VDD_PTC和VDD_PTD可以选择性关闭（参见表38）。

表38所示为在PD模式下使功耗降至最低的电源配置。

表38. 在PD模式下使功耗降至最低的电源配置

电源域	CM33-PD/CA35-PD
VDD_VBAT42	ON
VDD_FUSE18	ON (内部电源门控)
VDDQ_DDR	OFF
VDDQX_DDR	OFF
VDDQX_AO_DDR	ON
VDD_DIG0	ON
VDD_DIG1	ON
VDD_DIG2	ON
VDD_PMC18_DIG0	<ul style="list-style-type: none"> • ON (CM33 LDO启用模式) • OFF (CM33 LDO旁路模式)
VDD_DDR_PLL	ON
VDD_PMC18	ON
VDD_ANA18	ON
VDD_PTA	ON
VDD_PT B	ON
VDD_PTC	ON (可选OFF)
VDD_PTD	ON (可选OFF)
VDD_PTE	ON

表38. 在PD模式下使功耗降至最低的电源配置 (续)

电源域	CM33-PD/CA35-PD
VDD_PTF	ON
VDD18_IOREF_1/VDD18_IOREF_2	ON
VDD_DSII8	ON (如使用)
VDD_CSII8	ON (如使用)
VDD_DSIII	ON (内部电源门控)
VDD_CSIII	ON (如使用) (内部电源门控)
VDD_USB0_33/VDD_USB1_33	ON
VDD_USB0_18/VDD_USB1_18	ON
VDD_ANA33	ON
VREFH_ANA18	ON

7.1 使用PCA9460 PMIC的电源分布

恩智浦PCA9460处理器是一款专为i.MX 8ULP处理器而设计的电源管理集成电路 (PMIC)。有关PCA9460的更多详细信息, 请参阅恩智浦官网上的[PCA9460产品摘要页面](#)。

PCA9460有以下三种型号:

- PCA9460A: 面向恩智浦 i.MX 8ULP + LPDDR4
- PCA9460B: 面向恩智浦 i.MX 8ULP + LPDDR4x
- PCA9460C: 面向恩智浦 i.MX 8ULP + LPDDR3

对于每个PCA9460型号, BUCK4和LDO1的默认输出都不同。有关更多详细信息, 请参阅《PCA9460数据手册》。

[图8](#)显示所示为使用面向恩智浦 i.MX 8ULP + LPDDR4的PCA9460A PMIC进行电源分布的示例。

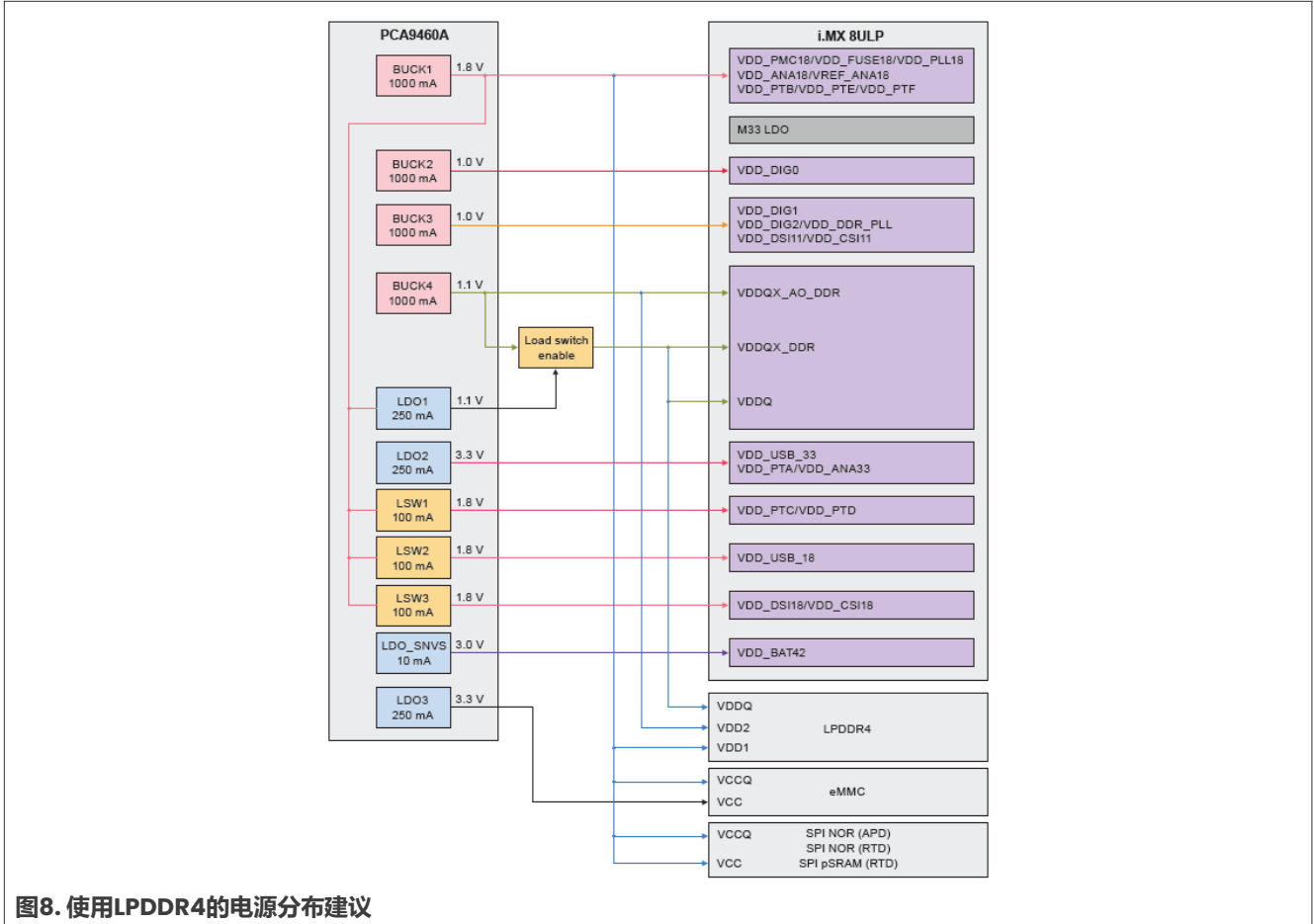
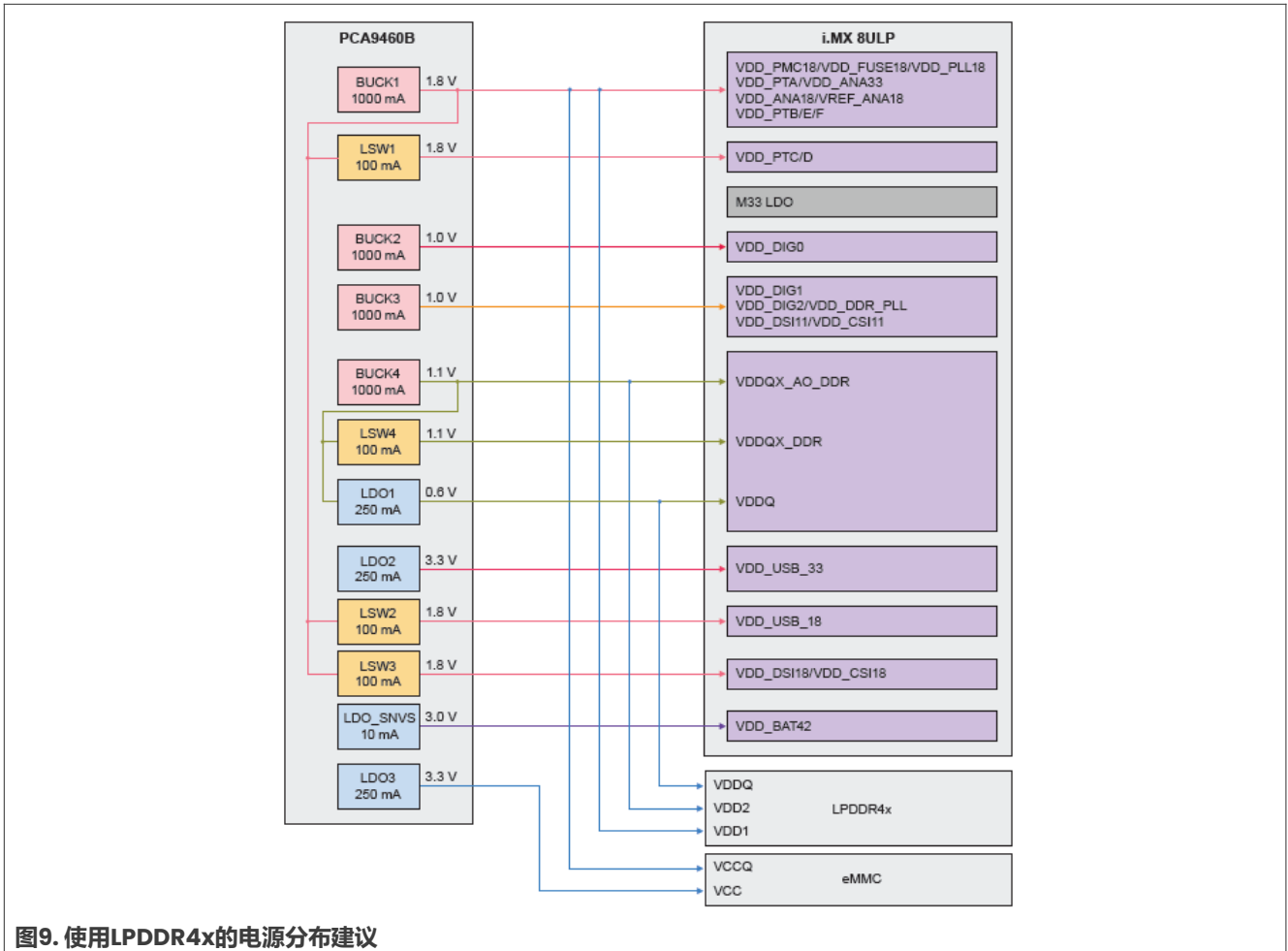


图8. 使用LPDDR4的电源分布建议

图9所示为使用面向恩智浦i.MX 8ULP + LPDDR4x的PCA9460B PMIC进行电源分布的示例。



8 重要命令

下面介绍了一些重要的脚本和命令：

- 配置环境的脚本：在运行用例之前，必须通过运行所需的配置脚本来配置环境，如 `setup.sh`、`setup_default.sh`、`setup_video.sh`、`setup_video_stream.sh` 和 `DDRC_96MHz_setup.sh`。下面介绍了一些重要的配置脚本：
 - `setup.sh`：此脚本禁用以太网，停止Weston服务，并关闭显示器。将内核为单个文件提前读取的最大数据量设置为512kB。该脚本如下所示：

```
#!/bin/bash
systemctl stop weston.service
for i in {0..9}; do
    device="/sys/class/graphics/fb${i}"
    if [ -e "$device" ]; then
        echo 1 > /sys/class/graphics/fb${i}/blank
    fi
done
partitions=`lsblk |awk '$1 !~/-/{print $1}' |grep 'blk\|sd'`
for partition in $partitions; do
    echo 512 > /sys/block/$partition/queue/read_ahead_kb
done
```

```
done
eth_int=`ifconfig -a | grep 'eth[0-9]|awk {'print substr($1, 0, 4)}`
for eth in $eth_int; do
    ifconfig $eth down
done
```

- setup_default.sh: 此脚本禁用以太网并关闭显示器。它将内核为单个文件提前读取的最大数据量设置为512KB。脚本如下所示:

```
#!/bin/bash
for i in {0..9}; do
    device="/sys/class/graphics/fb${i}"
    if [ -e "$device" ]; then
        echo 1 > /sys/class/graphics/fb${i}/blank
    fi
done
partitions=`lsblk |awk '$1 !~/-/{print $1}' |grep 'blk\|sd`
for partition in $partitions; do
    echo 512 > /sys/block/$partition/queue/read_ahead_kb
done
eth_int=`ifconfig -a | grep 'eth[0-9]|awk {'print substr($1, 0, 4)}`
for eth in $eth_int; do
    ifconfig $eth down
done
```

- setup_video.sh: 此脚本禁用以太网并唤醒显示器。它将内核为单个文件提前读取的最大数据量设置为512KB。该脚本如下所示:

```
#!/bin/bash
export WL_EGL_SWAP_INTERVAL=0
partitions=`lsblk |awk '$1 !~/-/{print $1}' |grep 'blk\|sd`
for partition in $partitions; do
    echo 512 > /sys/block/$partition/queue/read_ahead_kb
done
eth_int=`ifconfig -a | grep 'eth[0-9]|awk {'print substr($1, 0, 4)}`
for eth in $eth_int; do
    ifconfig $eth down
done
device="/sys/class/graphics/fb0"
if [ -e "$device" ]; then
    echo 1 > /sys/class/graphics/fb0/blank
    echo 0 > /sys/class/graphics/fb0/blank
fi
```

- setup_video_stream.sh: 此脚本打开以太网, 在线播放视频。它唤醒显示器, 将内核为单个文件提前读取的最大数据量设置为512KB。该脚本如下所示:

```
#!/bin/bash
export WL_EGL_SWAP_INTERVAL=0
partitions=`lsblk |awk '$1 !~/-/{print $1}' |grep 'blk\|sd`
for partition in $partitions; do
    echo 512 > /sys/block/$partition/queue/read_ahead_kb
done
eth_int=`ifconfig -a | grep 'eth[0-9]|awk {'print substr($1, 0, 4)}`
for eth in $eth_int; do
    ifconfig $eth up
done
device="/sys/class/graphics/fb0"
if [ -e "$device" ]; then
    echo 1 > /sys/class/graphics/fb0/blank
```

```
echo 0 > /sys/class/graphics/fb0/blank
fi
```

- DDRC_96MHz_setup.sh: 此脚本禁用以太网, 停止Weston服务, 并关闭显示器。该脚本如下所示:

```
#!/bin/bash
systemctl stop weston.service
for i in {0..9}; do
    device="/sys/class/graphics/fb${i}"
    if [ -e "$device" ]; then
        echo 1 > /sys/class/graphics/fb${i}/blank
    fi
done
#echo 8 > /proc/sys/kernel/printk;
eth_int=`ifconfig -a | grep 'eth[0-9]'|awk {'print substr($1, 0, 4)'} `
for eth in $eth_int; do
    ifconfig $eth down
done
echo 1 > /sys/devices/platform/imx8ulp-lpm/enable
```

运行该脚本时, 您会看到指示DDR频率在528MHz (高总线模式) 和96MHz (低总线模式) 之间切换的日志。这是由于DDR的电压和频率缩放 (VFS) 引起的。DDR VFS是为了省电。

- dd_write_emmc.sh: 此脚本用于在eMMC上运行dd write命令。该脚本如下所示:

```
#!/bin/bash
emmc_index=$(ls /dev/mmc*rpmb | head -1 | \
awk -F '/dev/mmcblk' '{print $2}' | \
awk -F 'rpmb' '{print $1}')

RC=1
while true; do
    time -p dd if=/dev/zero of=/dev/mmcblk${emmc_index} \
    bs=4096 seek=2621440 count=1024000 conv=fsync || \
    { echo "TFAIL:dd write emmc fail!";exit $RC; }
done
```

- dd_read_emmc.sh: 此脚本用于在emmc上运行dd read命令。该脚本如下所示:

```
#!/bin/bash
emmc_index=$(ls /dev/mmc*rpmb | head -1 | \
awk -F '/dev/mmcblk' '{print $2}' | \
awk -F 'rpmb' '{print $1}')

RC=1
while true; do
    time -p dd if=/dev/mmcblk${emmc_index} \
    of=/dev/null bs=4096 count=102400 || \
    { echo "TFAIL:dd read emmc fail!";exit $RC; }
done
```

- setup_bt_88w8987.sh (适用于88w8987蓝牙设备): 此脚本用于初始化内核中的主控制器接口 (HCI)。该脚本如下所示:

```
#!/bin/bash
modprobe moal mod_para=nxp/wifi_mod_para.conf
sleep 5
hciattach /dev/ttyLP2 any 115200 flow
```

```
sleep 1
hciconfig hci0 up
hcitool -i hci0 cmd 0x3f 0x0009 0xc0 0xc6 0x2d 0x00
killall hciattach
hciattach /dev/ttyLP2 any -s 3000000 3000000 flow
sleep 1 hciconfig hci0 up
hciconfig hci0 piscan
hciconfig hci0 noencrypt
hciconfig -a
sleep 1
```

- setup_bt_iw416.sh (适用于IW416蓝牙设备)：此脚本用于初始化Linux内核中的主控制器接口 (HCI)。该脚本如下所示：

```
#!/bin/bash
modprobe moal mod_para=nxp/wifi_mod_para.conf
modprobe btnxpuart
hciconfig hci0 up
hciconfig hci0 piscan
hciconfig hci0 noencrypt
hciconfig -a
sleep 1
```

- setup_pulseaudio.sh: 此脚本用于设置PulseAudio:

```
#!/bin/sh
pulseaudio --start --log-target=syslog
bluetoothctl power on
bluetoothctl agent on
bluetoothctl default-agent
bluetoothctl scan on # Got MAC address of remote BT device.
```

- uac2_gadget.sh: 此脚本用于在MCIMX8ULP-EVK板上创建用户帐户控制 (UAC) 小工具设备:

```
#!/bin/bash
modprobe libcomposite

# Create gadget
mkdir /sys/kernel/config/usb_gadget/g1
cd /sys/kernel/config/usb_gadget/g1
echo 0x0200 > bcdUSB
echo 0x0101 > idProduct
echo 0x1d6b > idVendor
mkdir strings/0x409
echo "1234567890" > strings/0x409/serialnumber
echo "Microsoft Applied Sciences" > strings/0x409/manufacturer
echo "Test UAC2 Gadget" > strings/0x409/product

# Create config
mkdir configs/c.1
echo 120 > configs/c.1/MaxPower
mkdir configs/c.1/strings/0x409
echo "Conf 1" > configs/c.1/strings/0x409/configuration

# Create function
mkdir functions/uac2.0
echo 48000 > functions/uac2.0/c_srate
echo 48000 > functions/uac2.0/p_srate
echo 2 > functions/uac2.0/c_ssize
echo 2 > functions/uac2.0/p_ssize
```

```
echo 0x3 > functions/uac2.0/c_chmask
echo 0x3 > functions/uac2.0/p_chmask

### config UAC2 parameters here
ln -s functions/uac2.0 configs/c.1

# UDC should correspond to udc name
# Activate
echo ci_hdrc.0 > UDC #8mm 8mn 8ulp
```

- MV.sh: 此脚本启动一个机器视觉示例:

```
#!/bin/bash
gst-launch-1.0 filesrc location=yuv.raw num-buffers=-1 ! \
clocksync ! queue max-size-buffers=2 max-size-time=0 ! \
rawvideoparse format=yuy2 width=640 height=480 framerate=30/1 ! \
tee name=t t. ! queue name=thread-nn max-size-buffers=2 leaky=2 ! \
imxvideoconvert_g2d ! video/x-raw,width=300,height=300,format=RGBA ! \
videoconvert ! video/x-raw,format=RGB ! tensor_converter ! \
tensor_filter framework=tensorflow-lite \
model=ssd_mobilenet_v2_coco_quant_postprocess.tflite \
custom=NumThreads:2 ! tensor_decoder mode=bounding_boxes \
option1=tf-ssd option2=coco_labels.txt option3=0:1:2:3,50 option4=640:480 \
option5=300:300 ! mix. t. ! queue name=thread-img max-size-buffers=2 ! \
imxcompositor_g2d name=mix sink_0::zorder=2 sink_1::zorder=1 ! \
queue name=thread-display max-size-buffers=2 ! waylandsink
```

- ML.sh: 此脚本用于运行[第5.8节](#)中介绍的产品用例。该脚本启动一个机器学习示例:

```
#!/bin/bash
TENSORFLOW_EXAMPLES_DIR=$(find /usr/bin/tensorflow-lite-*/examples | head -n1)
if [ ! -d "$TENSORFLOW_EXAMPLES_DIR" ]; then
    echo "TFAIL: not found or no directory of tensorflow-lite example!"
fi
cd $TENSORFLOW_EXAMPLES_DIR
while true; do
    ./benchmark_model --graph=mobilenet_v1_1.0_224_quant.tflite || break;
done
```

• U-Boot控制台命令: 以下是一些常用的U-Boot控制台命令:

- printenv: 显示环境变量。
- setenv: 更新环境变量:
- setenv <name> <value> ...: 将环境变量 "name" 设置为 "value..."。
- setenv <name>: 删除环境变量 "name"。
- saveenv: 保存对环境变量的更新。
- bootargs: 将内核命令行传递给内核。

• Linux操作系统控制台命令: 以下是一些常用的Linux操作系统控制台命令:

- cat /proc/cmdline: 显示命令行。
- cat /sys/devices/virtual/thermal/thermal_zone0/temp: 将温度打印到屏幕上 (应校准SoC)。
注: 芯片温度值从外部 (不在SD卡上) 记录 (写入), 以避免影响功耗。
- cat /sys/kernel/debug/clk/clk_summary: 将所有时钟打印到屏幕上。

- 获取当前CPU使用情况的脚本：以下是获取当前CPU使用情况的脚本：

```
#!/bin/bash

# by Paul Colby (http://colby.id.au), no rights reserved

PREV_TOTAL=0
PREV_IDLE=0

While true; do:
# Get the total CPU statistics, discarding the 'cpu ' prefix
CPU=(`sed -n 's/^cpu\s//p' /proc/stat`)
# Get the idle CPU time
IDLE=${CPU[3]}

# Calculate the total CPU time
TOTAL=0 for VALUE in "${CPU[@]}"; do
    let "TOTAL=$TOTAL+$VALUE"
done

# Calculate the CPU usage since we last checked
let "DIFF_IDLE=$IDLE-$PREV_IDLE"
let "DIFF_TOTAL=$TOTAL-$PREV_TOTAL"
let "DIFF_USAGE=(1000*($DIFF_TOTAL-$DIFF_IDLE)/$DIFF_TOTAL)/10"
echo -en "\rCPU: $DIFF_USAGE% \b\b"

# Remember the total and idle CPU times for the next check
PREV_TOTAL="$TOTAL"
PREV_IDLE="$IDLE"

# Wait before checking again
sleep 1 done
```

- 用于构建cpulimit应用程序的命令：运行以下命令来构建cpulimit应用程序：

1. 克隆cpulimit repo:

```
git clone https://github.com/opsengine/cpulimit.git
```

2. 从<https://developer.arm.com/downloads/-/arm-gnu-toolchain-downloads>下载aarch64工具链。

3. 解压工具链包，置aarch64工具链：

```
$ export CROSS_COMPILE=<path to toolchain>/bin/aarch64-linux-gnu-
$ export CC=${CROSS_COMPILE}gcc
$ export LD=${CROSS_COMPILE}ld
```

4. 构建应用程序：

```
make clean
make
```

9 相关文档

表39列出并介绍了其他文档和资源，您可以参考这些文档和资源，获得有关该电路板的更多信息。以下列出的部分文档需要签署保密协议（NDA）。要请求访问这些文档，请联系当地的现场应用工程师（FAE）或销售代表。

表39. 相关文档

文档	说明	链接/如何获取
《i.MX 8ULP处理器参考手册》 (IMX8ULPRM)	提供有关i.MX 8ULP处理器及其功能的详细说明，包括存储器映射、电源和时钟。	IMX8ULPRM.pdf
《i.MX 8ULP应用处理器——工业产品数据手册》 (IMX8 ULPIEC)	提供有关电气特性、硬件设计注意事项和订购信息的信息。	IMX8ULPIEC.pdf
《i.MX 8ULP应用处理器——商业产品数据手册》 (IMX8ULPCEC)		IMX8ULPCEC.pdf
《i.MX 8ULP硬件开发指南》 (IMX8ULPHDG)	提供有关电路板布板建议和设计检查清单的信息，以确保一次成功，避免电路板出现问题。它旨在帮助硬件工程师设计和测试基于i.MX 8ULP处理器的设计。	IMX8ULPHDG.pdf
《i.MX 8ULP功耗优化》 (AN13951)	介绍如何使用i.MX 8ULP在不同场景中优化系统级功耗。	AN13951.pdf
《PCA9460数据手册》	提供有关恩智浦PCA9460的详细信息，PCA9460是一款支持超低功耗应用的电源管理集成电路 (PMIC) 。PCA9460旨在与i.MX 8ULP处理器配套使用，从而实现此类关键应用。	PCA9460.pdf
《i.MX Yocto Project用户指南》	介绍如何使用Yocto Project构建环境为i.MX电路板构建镜像，并描述了i.MX发布层和i.MX专用的使用方法。	IMX_YOCTO_PROJECT_USERS_GUIDE.pdf
《电路板控制程序版本笔记》 (BCU.pdf)	提供有关BCU软件版本信息。	GitHub

10 缩略语

表40列出了本文中使用的缩略语。

表40. 缩略语

缩略语	说明
CA35	Arm Cortex-A35核
CM33	Arm Cortex-M33核
ADC	模数转换器
AHB	Arm AMBA高性能总线
AOD	不间断运行显示器
APD	应用处理器域
ATF	Arm可信固件
AXI	Arm高级可扩展接口
BBNSM	电池供电非安全模块

表40. 缩略语 (续)

缩略语	说明
BBSM	电池供电安全模块
BSP	板级支持包
CGC	时钟生成和控制
CMP	i.MX 8ULP模拟比较器模块
DAC	数模转换器
DDR	双数据速率DRAM
DGO	常开电源域的指示符
DRAM	动态随机存取存储器
EPDC	电泳显示控制器
EVK	评估套件
fps	每秒帧数
GND	接地
GPIO	通用输入/输出
GPU	图形处理单元
GPU2D	图形处理单元2D
GPU3D	图形处理单元3D
HCI	主控制器接口
HEVC	高效视频编码标准
High-Z	高阻抗
I2S	IC间声音总线
I/O	输入/输出
LDO	低压降稳压器
LPAV	低功耗音频/视频域
LPDDR3	低功耗DDR3 SDRAM
LPDDR4	低功耗DDR4 SDRAM, 带1.1 V I/O电源
LPDDR4x	低功耗DDR4 SDRAM, 带0.6 V I/O电源
LPTMR	i.MX 8ULP低功耗定时器
MAC	媒体访问控制
MIPI-CSI	MIPI-摄像头串行接口控制器
MIPI-DSI	MIPI-显示器串行接口控制器
MT/s	每秒兆次传输
ND	标称驱动
ODM	超驱动模式
ODT	片上终止
PCB	印刷电路板
PCC	外设时钟控制器

表40. 缩略语 (续)

缩略语	说明
PCM	脉冲编码调制
PLL	锁相环时钟发生器
PMC	电源管理控制器
PMIC	电源管理集成电路
PMS	电源模式开关
PTA	与处理器A端口相关的信号
PTB	与处理器B端口相关的信号
PTC	与处理器C端口相关的信号
PTD	与处理器D端口相关的信号
PTE	与处理器E端口相关的信号
PTF	与处理器F端口相关的信号
RAM	随机存取存储器
RTC	实时时钟
RTD	实时处理器域
SDK	软件开发工具包
SoC	片上系统
SRAM	片上静态随机存取存储器
SRTC	安全实时时钟
SSRAM	同步静态随机存取存储器
UAC	用户帐户控制
USB	通用串行总线
VFS	电压和频率缩放
WUU	唤醒单元

11 关于本文中源代码的说明

本文中所示的示例代码具有以下版权和BSD-3-Clause许可：

2024年恩智浦版权所有；在满足以下条件的情况下，可以源代码和二进制文件的形式重新分发和使用本源代码（无论是否经过修改）：

- 重新分发源代码必须保留上述版权声明、这些条件和以下免责声明。
- 以二进制文件形式重新分发时，必须在文档和/或随分发提供的其他材料中复制上述版权声明、这些条件和以下免责声明。
- 未经事先书面许可，不得使用版权所有者的姓名或参与者的姓名为本软件的衍生产品进行背书或推广。

本软件由版权所有者和参与者“按原样”提供，不承担任何明示或暗示的担保责任，包括但不限于对适销性和特定用途适用性的暗示保证。在任何情况下，无论因何种原因或根据何种法律条例，版权所有或参与者均不对因使用本软件而导致的任何直接、间接、偶然、特殊、惩戒性或后果性损害（包括但不限于采购替代商品或服务；

使用损失、数据损失或利润损失或业务中断) 承担责任, 无论是因合同、严格责任还是侵权行为 (包括疏忽或其他原因) 造成的, 即使事先被告知有此类损害的可能性也不例外。

12 修订历史

[表41](#)汇总了对本文的修订。

表41. 修订历史

文档ID	发布日期	说明
AN13914 v.3	2024年3月14日	更新了 表3
		更新了 第4.10节
		添加了 第4.12节
		更新了 第5节 及其小节
		更新了 第8节
		更新了 表39
		添加了AN13951作为参考文件
AN13914 v.2	2023年9月21日	更新了 表39 中的《PCA9460数据手册》的链接
		从文件中删除了AN13951参考
AN13914 v.1	2023年8月11日	首次公开发布

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.cn/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, μ Vision, 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.

目录

1	介绍	2	5.7	存储器 (eMMC) 用例.....	39
2	i.MX 8ULP的架构	2	5.7.1	DD_RD_eMMC.....	39
3	i.MX 8ULP电源概述	3	5.7.2	DD_WRT_eMMC.....	40
4	在MCIMX8ULP-EVK板上测量i.MX 8ULP的功率 ... 6		5.8	产品用例.....	41
4.1	DGO (不间断运行) 域电源.....	7	5.8.1	并行电子墨水屏翻页.....	41
4.2	实时处理器域 (RTD) 电源.....	7	5.8.2	并行电子墨水屏部分更新.....	42
4.3	应用处理器域 (APD) 电源.....	7	5.8.3	机器视觉.....	43
4.4	低功耗音频/视频域 (LPAV) 电源.....	8	5.8.4	eIQ基准测试.....	45
4.5	VBAT域电源.....	8	5.8.5	UAC音频播放.....	45
4.6	模拟和其他电源.....	8	5.8.6	不间断运行显示器, 1fps刷新率.....	48
4.7	测量过程中如何使用电压电平和VFS.....	8	5.9	系统级功率估计用例.....	50
4.8	温度.....	9	5.9.1	电池.....	50
4.9	硬件和软件要求.....	9	5.9.2	256KB L2缓存.....	51
4.10	构建i.MX Yocto Project.....	9	6	降低功耗的建议	52
4.11	功耗测量步骤.....	9	6.1	快速运行和空闲.....	53
4.12	软件包.....	10	6.2	时钟门控.....	53
5	用例和测量结果	10	6.3	减少PLL.....	54
5.1	低功耗模式用例.....	11	6.4	核心VFS和系统总线缩放.....	54
5.1.1	挂起模式——RTD活动.....	11	6.5	使用LPDDR4x而非LPDDR4.....	54
5.1.2	挂起模式——RTD睡眠.....	12	6.6	降低DDR频率.....	54
5.1.3	挂起模式——RTD深度睡眠.....	13	6.7	DDR接口优化.....	55
5.1.4	挂起模式——RTD断电.....	14	6.8	PHY的电源门控.....	55
5.1.5	系统空闲, 显示器开启.....	15	6.9	工作负载分布.....	55
5.1.6	系统空闲, 显示器关闭.....	16	6.10	使用SSRAM减少DDR访问时间.....	55
5.2	音频/视频播放用例.....	17	6.11	通过热管理减少漏电电流.....	55
5.2.1	全频率音频播放 (gplay).....	17	6.12	标称驱动模式.....	56
5.2.2	96MHz DDR时钟的音频低总线播放 (gplay).....	18	7	在最低功耗模式下控制i.MX 8ULP电源	56
5.2.3	通过蓝牙播放音频 (gplay).....	19	7.1	使用PCA9460 PMIC的电源分布.....	57
5.2.4	本地视频播放 (gplay).....	22	8	重要命令	59
5.2.5	视频播放流 (gplay).....	23	9	相关文档	64
5.3	核心基准测试用例.....	25	10	缩略语	65
5.3.1	Dhrystone.....	25	11	关于本文中源代码的说明	67
5.3.2	CoreMark.....	26	12	修订历史	68
5.4	GPU用例.....	27		法律声明	69
5.4.1	MM07.....	28			
5.4.2	MM06.....	29			
5.4.3	GLMark.....	30			
5.5	重载用例.....	31			
5.5.1	双核Dhrystone + MM07 + HiFi4 + CM33 CoreMark.....	31			
5.5.2	双核Stream + MM07 + HiFi4 + CM33 CoreMark.....	33			
5.5.3	GPU2D + GPU3D + CSI/DSI.....	35			
5.6	内存用例.....	36			
5.6.1	memset.....	36			
5.6.2	memcpy.....	37			
5.6.3	Stream.....	38			

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