AN13054

i.MX 8M Plus Power Consumption Measurement

Rev. 0 — 04/2021 Application Note

Contents

1 Introduction

This application note helps system designers to create power optimized systems. It illustrates how to measure the current drain of the i.MX 8M Plus application processor taken on the NXP PWR EVK platform (equipped with PWR CPU board with consumer product) through several different use cases. Users may choose the appropriate power supply domains for the i.MX 8M Plus processor and become familiar with the expected processor power consumption in these various scenarios.

Since the data presented in this application note is based on empirical measurements taken on a small sample size, the presented results are not guaranteed.

2 Overview of the voltage supplies

The i.MX 8M Plus processor has several power supply domains (voltage supply rails) and several internal power domains. Figure 1 shows the connectivity of these supply rails and the distribution of the internal power domains.

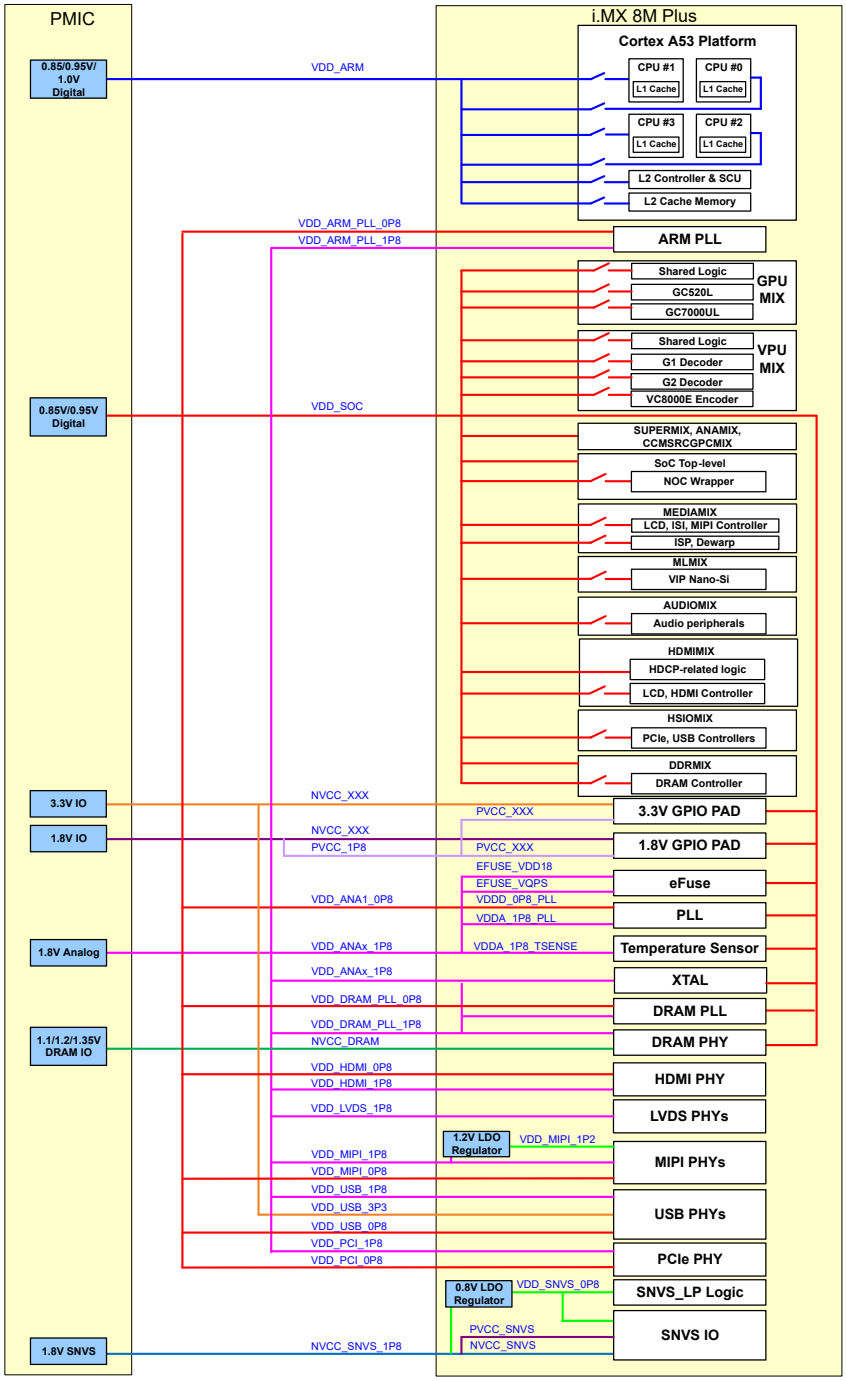

Figure 1. i.MX 8M Plus power rails

NOTE

For the recommended operating conditions of each supply rail and for a detailed description of the groups of pins that are powered by each I/O voltage supply, see *i.MX 8M Plus Applications Processor Datasheet for Consumer Products* (document IMX8MPCEC).

The i.MX 8M Plus LPDDR4 PWR CPU board (8MPLUSLPD4-PWR) supply design is identical to the standard i.MX 8M Plus LPDDR4 CPU board (8MPLUSLPD4-CPU) but with access points inserted inline of the power supply nets to measure power. They are both software compatible and use the same baseboard (8MPLUS-BB).

3 Internal power measurements of the i.MX 8M Plus

Several use cases (described in Use-Case Configuration and Usage Guidelines) are run on the PWR EVK platform. The measurements are taken mainly for the power supply domains given in Table 1.

Table 1. Measured power supply domains

Power groups	Power supply domains	Description
GROUP_SOC	VDD_ARM	Arm® Cortex®-A53 cores supply
	VDD_SOC	SoC logic, DRAM controller, GPU, and VPU controllers
	NVCC_DRAM	i.MX 8M Plus DRAM IO power supply
	NVCC_SNVS_1P8	I/O supply and I/O Pre-driver supply for GPIO in SNVS bank
	VDD_xxx_1P8[1] + VDD_xxx _0P8[2] + VDD_USB_3P3	PLLs[3], eFUSE, Temperature sensors, 24M XTAL, USB PHY, PCIe PHY, HDMI PHY, LVDS PHY, MIPI PHY 0.8 V/1.8 V/3.3 V power supply
GROUP_DRAM	VDD1 + VDDQ + VDD2	DRAM device power supply. Excluded from i.MX 8M Plus internal power

1. VDD_24M_XTAL_1P8 + VDD_ARM_PLL_1P8 + VDD_DRAM_PLL_1P8 + VDD_AVPLL_1P8 + VDD_SAI_PLL_1P8 + VDD_ANA0_1P8 + VDD_ANA1_1P8 + VDD_ANA2_1P8 + VDD_HDMI_1P8 + VDD_EARC_1P8 + VDD_LVDS_1P8 + VDD_MIPI_1P8 + VDD_PCI_1P8 + VDD_USB_1P8
2. VDD_ARM_PLL_0P8 + VDD_DRAM_PLL_0P8 + VDD_SAI_PLL_0P8 + VDD_ANA1_0P8 + VDD_HDMI_0P8 + VDD_MIPI_0P8 + VDD_PCI_0P8 + VDD_USB_0P8
3. All PLLs rails apart from HDMI and EARC's AV_PLLs 0.8V rails supplied from VDD_SOC

These supply domains consume the majority of the processor's and DRAM device's power. The power consumption of the SNVS is comparatively negligible (except for the suspend mode).

The NVCC_* IO power consumption depends primarily on the board-level configuration and the components. Therefore, it is not included in the i.MX 8M Plus internal power analysis.

The power consumption of these supplies (in different use cases) is provided in Table 4 through Table 42.

3.1 DDR I/O power

The DDR I/O is supplied from the NVCC_DRAM, which provides the power for the DDR I/O pads. The target voltage for this supply depends on the DDR interface used. The target voltages for the different DDR interfaces are:

- 1.1 V for LPDDR4 (used in current EVK).
- 1.2 V for DDR4.

The power consumption of the NVCC_DRAM supply is affected by various factors, including:

- The amount of activity on the DDR interface.
- On-Die Termination (ODT): enabled/disabled, termination value, which is used for the DDR controller and the DDR memories.
- Board termination for the DDR control and the address bus.
- Configuration of the DDR pads (such as the drive strength).

- PCB board layout.
- Load of the DDR memory devices.

NOTE

Due to the factors specified in the previous paragraph, the measurements provided in the following tables vary from one system to another. The provided peak data and average data are for guidance only and should not be treated as a specification.

The measured current on the PWR EVK Platform also includes the current of the onboard LPDDR4 memory device. The total power consumption of the onboard LPDDR4 memory device is reported separately via GROUP_DRAM, see Measuring channels on the PWR EVK platform.

3.2 Voltage levels in the measurement process

The voltage levels of all the supplies (except for VDD_ARM, VDD_SOC, VDD_xxx_PLL_0P8, VDD_ANA1_0P8, and VDD_PHYs_0P8) are set to the typical voltage levels, as defined in the *i.MX 8M Plus Data Sheet for Consumer Products*.

The VDD_ARM, VDD_SOC, VDD_xxx_PLL_0P8, VDD_ANA1_0P8, and VDD_ PHYs_0P8 supply voltages change during normal operation. To save power, these power voltages are changed during the runtime of the use cases. The voltage levels of these supplies can be lowered to standby voltage levels in low-power modes, see Low-power mode use cases, Suspend mode, and System idle mode for more detail.

3.2.1 VDD voltage levels

The target voltage levels of the VDD_ARM, VDD_SOC, VDD_xxx_PLL_0P8, VDD_ANA1_0P8, and VDD_ PHYs_0P8 may vary for different modes according to the use cases. The modes are the nominal mode, the overdrive mode, and the super overdrive mode. Several factors contribute to the decision of which mode is used, with the power mode (Run/ Idle/ DSM mode) and module load being the most important. The other factors are module latency requirements, thermal restrictions, and peripheral I/O performance requirements. The voltage levels used for the measurements are listed in Table 2.

Table 2. VDD_ARM/ VDD_SOC/ VDD_xxx_PLL_0P8/ VDD_ANA1_0P8/ VDD_ PHYs_0P8 voltage levels

Power rail	Vmin (V)	Vtyp (V)	Vmax (V)	Description
VDD_ARM	0.805	0.850	0.950	Nominal mode, 1.2 GHz
	0.900	0.950	1.000	Overdrive mode, 1.6 GHz
	0.950	1.000	1.050	Super overdrive mode, 1.8 GHz[3]
VDD_SOC	0.805	0.850	0.900	Nominal mode
	0.900	0.950	1.000	Overdrive mode
VDD_xxx_PLL_0P8[1]	0.805	0.850	0.900	Nominal mode
	0.900	0.950	1.000	Overdrive mode
VDD_ ANA1_0P8	0.805	0.850	0.900	Nominal mode
	0.900	0.950	1.000	Overdrive mode
VDD_ PHYs_0P8[2]	0.805	0.850	0.900	Nominal mode
	0.900	0.950	1.000	Overdrive mode

1. VDD_ARM_PLL_0P8 + VDD_DRAM_PLL_0P8 + VDD_SAI_PLL_0P8.
2. VDD_HDMI_0P8 + VDD_MIPI_0P8 + VDD_PCI_0P8 + VDD_USB_0P8.
3. Supported only on consumer products.

NOTE

For the official operating points, see the operating ranges table in the *i.MX 8M Plus Applications Processor Datasheet for Consumer Products* (document IMX8MPCEC).

VDD_SOC, VDD_xxx_PLL_0P8, VDD_ANA1_0P8, and VDD_PHYs_0P8 are separated on PWR EVK but combined on the standard EVK board as a demonstration of simplified board design.

Most of the measurements are performed using these voltage levels and the power data that appears in this document is in accordance with these values. If the measurement is done at different voltage levels, the power consumption change with the voltage. In real applications, the software (in conjunction with the hardware) automatically adjusts the voltage and frequency values based on the use case requirements. DVFS is only supported on VDD_ARM.

3.3 Hardware and software used

The software versions used for the measurements are:

- Yocto rootfs, Linux® Kernel version: *L 5.4.70_2.3.0 i.MX 8M Plus*.
- The boards used for the measurements are i.MX 8M Plus Rev.A1 LPDDR4 PWR EVK platform (8MPLUSLPD4-PEVK) populated with consumer part.
- The measurements were performed using the on-board measurement circuitry and BCU software tool (The BCU tool can be found at https://github.com/NXPmicro/bcu).
- These measurements have been taken at room temperature without thermal forcing equipment.

3.4 Measuring channels on the PWR EVK platform

To measure the power consumption, the PWR CPU board is designed with current sense resistors of suitable value inserted between the PMIC and CPU for each key power rail. The power data is obtained by sampling the average voltage drop across the sense resistors using power monitor chip PAC1934. The voltage drop for each power supply is divided by its sense resistor value within the BCU PC software tool to calculate the current.

Some rails have a second current sense resistor that enables a lower current measurement range. The low current shunts are kept shorted out until low current measurements are enabled by the BCU software. While in this application note, all the data was collected with the default high current measurement range (Rs1).

The measurement channels for the various supply domains are listed in Table 3:

Table 3. Measuring channels on PWR EVK

Power Group	Supply Domain	Power Rail	Net Name	Rs1 (OHM)	Rs2 (OHM)
Group SOC	VDD_ARM	VDD_ARM	VDD_ARM	0.02 (1 %)	4.99 (0.1 %) +0.02 (1 %)[1]
	VDD_SOC	VDD_SOC	VDD_SOC	0.01 (1 %)	1.0 (1 %)+0.01 (1 %)[1]
	NVCC_DRAM	NVCC_DRAM	NVCC_DRAM_1V1	0.05 (1 %)	2.0 (1 %)+0.05 (1 %)[1]
	NVCC_SNVS_1P8	NVCC_SNVS_1P8	NVCC_SNVS_1V8	10 (0.1 %)	499 (0.1 %)+10 (0.1 %)[1]
	VDD_xxx_1P8	VDD_24M_XTAL_1P8 VDD_ARM_PLL_1P8 VDD_DRAM_PLL_1P8	VDD_PLL_ANA_1V8	1.0 (1 %) +0.008[2]	10.0 (0.1 %)+1.0 (1 %)[1]

Table continues on the next page...

Table 3. Measuring channels on PWR EVK (continued)

Power Group	Supply Domain	Power Rail	Net Name	Rs1 (OHM)	Rs2 (OHM)
		VDD_ANA0_1P8 VDD_ANA1_1P8 VDD_ANA2_1P8 VDD_AVPLL_1P8 VDD_SAI_PLL_1P8			
		VDD_EARC_1P8	VDD_EARC_1V8	2.0 (1 %)	-
		VDD_HDMI_1P8	VDD_HDMI_1V8	2.0 (1 %)	-
		VDD_LVDS_1P8	VDD_LVDS_1V8	1.0 (1 %)	-
		VDD_MIPI_1P8	VDD_MIPI_1V8	2.0 (1 %)	-
		VDD_PCI_1P8	VDD_PCI_1V8	1.0 (1 %)	-
		VDD_USB_1P8	VDD_USB_1V8	2.0 (1 %)	
	VDD_xxx _0P8	VDD_ARM_PLL_0P8 VDD_DRAM_PLL_0P8 VDD_SAI_PLL_0P8 VDD_ANA1_0P8	VDD_PLL_ANA_0V8	1.0 (1 %)	
		VDD_HDMI_0P8	VDD_HDMI_0V8	1.0 (1 %)	
		VDD_MIPI_0P8	VDD_MIPI_0V8	1.0 (1 %)	
		VDD_PCI_0P8	VDD_PCI_0V8	0.4 (1 %)	
		VDD_USB_0P8	VDD_USB_0V8	0.4 (1 %)	
	VDD_USB_3P3	VDD_USB_3P3	VDD_USB_3V3	1.0 (1 %)	-
Group_DRAM	VDD1+VDDQ+VDD2	VDD1	LPD4_VDD1	4.99 (0.1 %)	-
		VDDQ	LPD4_VDDQ	0.05 (1 %)	2.0 (1 %)+0.05 (1 %)[1]
		VDD2	LPD4_VDD2	0.05 (1 %)	-

1. The actual sense resistor value for low measurement range is Rs2=Rs2'+Rs1, when the load switch is open.
2. For the most dual range rails, the Rs1=Rs1' when the load switch is closed, since the sampling point for high measurement range is P2 and P3, except for VDD_PLL_ANA_1V8 and NVCC_SNVS_1V8 that use P1 and P3, and Rs1=Rs1'+Ron, see Figure 2.

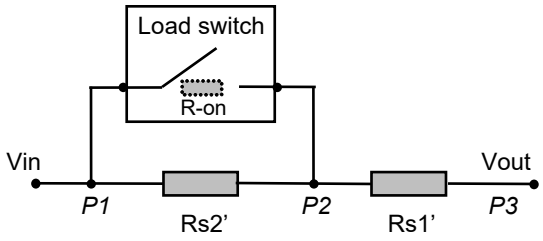

Figure 2. Dual range rails

4 Use cases and measurement results

The main use cases and subtypes that form the benchmarks for the i.MX 8M Plus internal power measurements on the PWR EVK platform are described in the following sections.

For all use cases, the platform is booted from an SD card with the default dtb configuration in the U-Boot stage. In the U-Boot stage, use the following U-Boot commands to configure the default dtb file to be `imx8mp-evk.dtb`:

- setenv fdt_file `imx8mp-evk.dtb`
- saveenv
- printenv

4.1 Low-power mode use cases

The following use case scenarios were tested:

- Suspend mode
- IDLE_DEFAULT
- IDLE_LOW_BUS

NOTE

In IDLE_DEFAULT mode, the DRAM PLL is enabled and the DDR clock is 2000 MHz.

In IDLE_LOW_BUS mode, the DRAM PLL is disabled and 50 MHz PLL Bypass clock is used as DDR clock.

DDR data rate is 2 times of DDR clock. For example, the DDR clock at 50 MHz means 100 MTS data rate.

4.1.1 Suspend mode

This mode is called either “Dormant mode” or “Deep sleep mode” in the Linux BSP. This is the lowest possible power state where the external supplies are still on.

The use case is as follows:

- The Arm® platform is power gated.
- The L2 cache peripherals are power gated.
- The Arm® Cortex®-M7 is in the reset status.
- All PLL (Phase-Locked Loop) and CCM (Clock Controller Module) generated clocks are off.
- The CKIL (32 kHz) input is on.

- All modules are disabled.
- The external high-frequency crystal and the on-chip oscillator are powered down (by asserting the SBYOS bit in the CCM).

Table 4 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 4. Suspend mode

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	0.01	0.00	2.70	0.00	0.00	0.00
	VDD_SOC	0.86	0.85	83.90	12.90	70.80	10.90
	NVCC_DRAM	1.11	1.10	39.60	0.00	43.50	0.30
	NVCC_SNVS_1P8	1.79	1.79	0.20	0.00	0.30	0.20
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	3.74
	Total	-	-	-	-	-	15.14
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	3.97
Details for [VDD_xxx_1P8 + VDD_xxx _0P8 + VDD_USB_3P3] and Group DRAM							
VDD_xxx_1P8	VDD_PLL_ANA_1V8	1.81	1.8	0.9	0	1.7	1.3
	VDD_EARC_1V8	1.81	1.8	0	0	0.1	0
	VDD_HDMI_1V8	1.81	1.8	0	0	0.1	0
	VDD_LVDS_1V8	1.81	1.8	0.1	0	0.1	0
	VDD_MIPI_1V8	1.81	1.8	0.1	0	0.2	0.1
	VDD_PCI_1V8	1.81	1.8	0.1	0	0.1	0
	VDD_USB_1V8	1.81	1.8	0.3	0	0.5	0.4
VDD_xxx_0P8	VDD_PLL_ANA_0V8	0.86	0.85	1.8	0	1.5	0
	VDD_HDMI_0V8	0.86	1.8	1.5	0	1.3	0
	VDD_MIPI_0V8	0.86	0.85	1.8	0	1.5	0.1
	VDD_PCI_0V8	0.85	0.85	2.9	1.5	2.5	1.3
	VDD_USB_0V8	0.86	0.85	2.3	0	2	0.1
VDD_USB_3P3	VDD_USB_3V3	3.33	3.31	1.7	0	5.5	0.4
Group DRAM	VDD1	1.81	1.79	2.4	0	4.3	1.1
	VDDQ	1.11	1.1	34.4	0	37.7	-0.1
	VDD2	1.12	1.1	49.6	2.7	54.2	3

Table continues on the next page...

Table 4. Suspend mode (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Measurement duration: 60 s Die temperature: Zone0 NA / Zone1 NA Utilization: CPU - NA, DDR - NA							

For more details about this use case and settings, see Use case configuration and usage guidelines.

4.1.2 IDLE_DEFAULT

For this use case, a MIPI-DSI-to-HDMI card adapter was connected to the EVK DSI port and no display was attached to the MIPI-DSI card port.

The use case is as follows:

- The CPU frequency is the default value of 1200 MHz.
- The Arm® Cortex®-A53 core is power-gated if the kernel is in the lowest level of idle.
- The Arm L2 cache and PLAT are powered on.
- The Arm Cortex-M7 is in the reset status.
- All the unused PLLs are off and the unused clocks are gated.
- The GPU and DISPMIX are in low-power mode.
- The operating system is on.
- The DDR frequency is set to 2000 MHz (default).

Table 5 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 5. IDLE_DEFAULT

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	0.86	0.85	135.00	7.90	114.40	6.70
	VDD_SOC	0.94	0.93	878.00	829.10	819.70	773.10
	NVCC_DRAM	1.10	1.10	114.60	67.00	125.20	73.40
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3						202.66
	Total						1056.26
Group DRAM	VDD1 + VDDQ + VDD2						3.77

Table continues on the next page...

Table 5. IDLE_DEFAULT (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Details for [VDD_xxx_1P8 + VDD_xxx _0P8 + VDD_USB_3P3] and Group DRAM							
VDD_xxx_1P8	VDD_PLL_ANA_1V8	1.82	1.81	14	13.7	25.3	24.7
	VDD_EARC_1V8	1.82	1.81	0	0	0.1	0
	VDD_HDMI_1V8	1.82	1.81	0.1	0	0.1	0
	VDD_LVDS_1V8	1.82	1.81	0.1	0	0.2	0
	VDD_MIPI_1V8	1.82	1.81	0.1	0	0.2	0.2
	VDD_PCI_1V8	1.77	1.76	50.1	50	88.5	88
	VDD_USB_1V8	1.82	1.81	0.2	0	0.4	0.3
VDD_xxx_0P8	VDD_PLL_ANA_0V8	0.95	0.94	3.8	3.3	3.6	3.1
	VDD_HDMI_0V8	0.95	1.81	1.5	1.2	1.4	1.1
	VDD_MIPI_0V8	0.96	0.95	1.1	0	1	0.6
	VDD_PCI_0V8	0.88	0.87	79.8	79.4	69.8	68.8
	VDD_USB_0V8	0.96	0.95	5.4	2.5	5.1	2.4
VDD_USB_3P3	VDD_USB_3V3	3.32	3.3	6	4.1	19.8	13.4
Group DRAM	VDD1	1.81	1.79	4.6	0	8.1	1.1
	VDDQ	1.12	1.1	29.8	0	33	-0.1
	VDD2	1.11	1.1	108.3	2.5	118.4	2.7
Measurement duration: 60 s Die temperature: Zone0 37 °C / Zone1 38 °C Utilization: CPU - 0%, DDR - 0.08%							

For more details about this use case and settings, see Use-Case Configuration and Usage Guidelines.

4.1.3 IDLE_LOW_BUS

For this use case, a MIPI-DSI-to-HDMI card adapter was connected to the EVK DSI port and no display was attached to the MIPI-DSI card port.

The use case is as follows:

- The CPU frequency governor is set to `powersave` (The CPU frequency is set to the minimum value 1200 MHz).
- The Arm® Cortex®-A53 core is power-gated if the kernel is in the lowest level of idle.
- The Arm L2 cache and PLAT are powered on.
- The Arm Cortex-M7 is in the reset status.
- All the unused PLLs are off and the unused clocks are gated.

- The operating system is on.
- The DDR frequency is set to 50 MHz.

Table 6 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 6. IDLE_LOW_BUS

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	0.86	0.85	171.50	8.40	144.00	7.10
	VDD_SOC	0.95	0.94	852.70	243.40	796.30	228.80
	NVCC_DRAM	1.11	1.10	97.20	34.30	106.50	37.70
	NVCC_SNVS_1P8	1.80	1.78	0.30	0.00	0.40	0.20
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	194.15
	Total	-	-	-	-	-	467.95
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	4.53
Details for [VDD_xxx_1P8 + VDD_xxx _0P8 + VDD_USB_3P3] and Group DRAM							
VDD_xxx_1P8	VDD_PLL_ANA_1V8	1.82	1.81	13.9	9.8	25.2	17.7
	VDD_EARC_1V8	1.82	1.81	0	0	0.1	0
	VDD_HDMI_1V8	1.82	1.81	0.1	0	0.1	0
	VDD_LVDS_1V8	1.82	1.81	0.1	0	0.2	0
	VDD_MIPI_1V8	1.82	1.81	0.1	0	0.2	0.1
	VDD_PCI_1V8	1.77	1.76	50.1	50	88.6	88
	VDD_USB_1V8	1.82	1.81	0.2	0	0.4	0.3
VDD_xxx_0P8	VDD_PLL_ANA_0V8	0.95	0.94	3.7	2.6	3.5	2.4
	VDD_HDMI_0V8	0.95	1.81	2	1.2	1.9	1.1
	VDD_MIPI_0V8	0.96	0.95	1.2	0	1.1	0.4
	VDD_PCI_0V8	0.88	0.87	79.8	78.8	69.5	68.4
	VDD_USB_0V8	0.96	0.95	5.6	2.6	5.2	2.4
VDD_USB_3P3	VDD_USB_3V3	3.32	3.3	5.8	4	19.2	13.3
Group DRAM	VDD1	1.81	1.79	3.8	0	6.8	1.2
	VDDQ	1.11	1.1	33.6	0	36.8	0
	VDD2	1.11	1.1	105.9	3.1	116.5	3.4
Measurement duration: 60s Die temperature: Zone0 39 °C / Zone1 40 °C Utilization : CPU - 0%, DDR - 0.08%							

For more details about this use case and settings, see Use-Case Configuration and Usage Guidelines.

4.2 Audio_Playback, M7 idle

The following use case scenarios were tested:

- Audio_Playback(gplay)
- Audio_Playback(gplay)_ DDRC_25MHz
- Audio+Video_Playback(gplay)

4.2.1 Audio_Playback(gplay)

The audio file used was an mp3 file with a 128 kbit/s bit rate and a 44 kHz sample rate/s, played using the following options:`gplay-1.0 $audio_file`

The use case is as follows:

- The CPU frequency governor is set to *performance* (The CPU frequency is set to the maximum value 1800 MHz).
- The Arm® L2 cache and PLAT are powered on.
- The Arm® Cortex®-M7 is in the reset status.
- The GPU, and DISPMIX are in the low-power mode.
- All the unused PLLs are off and the unused clocks are gated.
- The operating system is on.
- The DDR frequency is set to 2000 MHz.

Shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 7. Audio_Playback(gplay)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.01	1.00	595.40	34.00	582.80	33.90
	VDD_SOC	0.94	0.93	936.90	864.20	876.00	805.30
	NVCC_DRAM	1.10	1.10	148.40	70.60	162.00	77.40
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	208.57
	Total	-	-	-	-	-	1125.57
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	12.82
Details for [VDD_xxx_1P8 + VDD_xxx _0P8 + VDD_USB_3P3] and Group DRAM							
VDD_xxx_1P8	VDD_PLL_ANA_1V8	1.82	1.81	15.1	14.7	27.3	26.6

Table continues on the next page...

Table 7. Audio_Playback(gplay) (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
	VDD_EARC_1V8	1.82	1.81	2	1.9	3.6	3.5
	VDD_HDMI_1V8	1.82	1.81	0.1	0	0.1	0
	VDD_LVDS_1V8	1.82	1.81	0.1	0	0.2	0
	VDD_MIPI_1V8	1.82	1.81	0.1	0	0.2	0.1
	VDD_PCI_1V8	1.77	1.76	50.2	50	88.6	88
	VDD_USB_1V8	1.82	1.81	0.2	0	0.4	0.3
VDD_xxx_0P8	VDD_PLL_ANA_0V8	0.95	0.94	4.3	3.9	4.1	3.7
	VDD_HDMI_0V8	0.96	1.81	1.6	1.2	1.5	1.1
	VDD_MIPI_0V8	0.96	0.95	1.1	0	1	0.6
	VDD_PCI_0V8	0.88	0.87	79.9	79.4	69.7	68.8
	VDD_USB_0V8	0.96	0.95	5.1	2.5	4.9	2.4
VDD_USB_3P3	VDD_USB_3V3	3.32	3.3	5.9	4	19.5	13.3
	VDD1	1.8	1.79	4.3	1.1	7.6	1.9
	VDDQ	1.11	1.1	30	0	33.1	0.4
Group DRAM	VDD2	1.11	1.1	127	9.5	137.8	10.5
Measurement duration: 60 s Die temperature: Zone0 39 °C / Zone1 40 °C Utilization : CPU - 1%, DDR - 0.32%							

For more details about this use case and settings, see Use-Case Configuration and Usage Guidelines.

4.2.2 Audio_Playback(gplay)_LOW_BUS

For this use case, the audio file used was an mp3 file with a 128 kbit/s bit rate and a 44 kHz sample rate/s, played using the following command:

```
gplay-1.0 $audio_file
```

The use case is as follows:

- The CPU frequency governor is set to *powersave* (CPU frequency is set to minimum value).
- The Arm® Cortex®-A53 core is power-gated if the kernel is in the lowest level of idle.

- The Arm L2 cache and PLAT are powered on.
- The Arm Cortex-M7 is in the reset status.
- The GPU and DISPMIX are in the low-power mode.
- All the unused PLLs are OFF and the unused clocks are gated.
- The operating system is on.
- The DDR frequency is set to 50 MHz.

Table 8 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 8. Audio_Playback(gplay)_LOW_BUS

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	0.86	0.85	341.90	48.00	287.20	40.60
	VDD_SOC	0.95	0.94	903.00	322.00	844.70	302.10
	NVCC_DRAM	1.11	1.10	201.80	64.40	219.40	70.50
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.20
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	201.34
	Total	-	-	-	-	-	614.74
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	37.36
Details for [VDD_xxx_1P8 + VDD_xxx _0P8 + VDD_USB_3P3] and Group DRAM							
VDD_xxx_1P8	VDD_PLL_ANA_1V8	1.82	1.81	15.3	11.3	27.6	20.5
	VDD_EARC_1V8	1.82	1.81	2	1.9	3.6	3.5
	VDD_HDMI_1V8	1.82	1.81	0.1	0	0.1	0
	VDD_LVDS_1V8	1.82	1.81	0.1	0	0.2	0
	VDD_MIPI_1V8	1.82	1.81	0.1	0	0.3	0.1
	VDD_PCI_1V8	1.77	1.76	50.1	50	88.6	88
	VDD_USB_1V8	1.82	1.81	0.2	0	0.4	0.3
VDD_xxx_0P8	VDD_PLL_ANA_0V8	0.95	0.94	4.4	3.3	4.1	3.1
	VDD_HDMI_0V8	0.95	1.81	1.9	1.2	1.8	1.1

Table continues on the next page...

Table 8. Audio_Playback(gplay)_LOW_BUS (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
	VDD_MIPI_0V8	0.95	0.95	1.1	0	1	0.4
	VDD_PCI_0V8	0.88	0.87	79.7	78.9	69.6	68.5
	VDD_USB_0V8	0.96	0.95	5.9	2.6	5.6	2.4
VDD_USB_3P3	VDD_USB_3V3	3.32	3.3	5.7	4	18.9	13.3
Group DRAM	VDD1	1.8	1.79	4.1	1.3	7.3	2.3
	VDDQ	1.11	1.1	126.3	9.8	138.4	10.7
	VDD2	1.11	1.1	105.5	22.1	115.9	24.3
Measurement duration: 60 s Die temperature: Zone0 39 °C / Zone1 40 °C Utilization: CPU - 1 %, DDR - 0.32 %							

4.2.3 Audio+Video_Playback(gplay)

For this use case, the MIPI-DSI port was connected to a 1080p TV display.

The video file used for playback was an mkv file format compressed using the HEVC standard with full HD resolution at 29.97 fps and the audio encoding was AACL with a 44.1 kHz sample rate in a 2-channel configuration.

The video file was locally played using gplay, with the following options:

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

The use case is as follows:

- The CPU frequency governor is set to *performance* (The CPU frequency is set to the maximum value 1800 MHz).
- The Arm® L2 cache and PLAT are powered on.
- The Arm® Cortex®-M7 is in the reset status.
- All the unused PLLs are OFF and the unused clocks are gated.
- The operating system is on.

Table 9 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 9. Audio+Video_Playback(gplay)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.01	1.00	572.20	65.90	564.30	65.50
	VDD_SOC	0.94	0.93	1525.90	1037.40	1403.50	963.10
	NVCC_DRAM	1.10	1.09	341.70	129.20	369.20	141.10
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	240.85
	Total	-	-	-	-	-	1410.95
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	140.94
Details for [VDD_xxx_1P8 + VDD_xxx _0P8 + VDD_USB_3P3] and Group DRAM							
VDD_xxx_1P8	VDD_PLL_ANA_1V8	1.82	1.81	17.8	17.2	32.2	31.1
	VDD_EARC_1V8	1.82	1.81	2	2	3.6	3.5
	VDD_HDMI_1V8	1.82	1.81	0.1	0	0.1	0
	VDD_LVDS_1V8	1.82	1.81	0.1	0	0.2	0
	VDD_MIPI_1V8	1.81	1.8	3.7	3.6	6.8	6.6
	VDD_PCI_1V8	1.77	1.76	50.2	50	88.6	88.1
	VDD_USB_1V8	1.82	1.81	0.2	0	0.4	0.3
VDD_xxx_0P8	VDD_PLL_ANA_0V8	0.95	0.94	6.1	5.6	5.7	5.3
	VDD_HDMI_0V8	0.96	1.81	1.6	1.2	1.5	1.1
	VDD_MIPI_0V8	0.94	0.92	24.7	22	22.8	20.3
	VDD_PCI_0V8	0.88	0.87	79.9	79.4	69.7	68.8
	VDD_USB_0V8	0.96	0.95	5.4	2.5	5.1	2.4
VDD_USB_3P3	VDD_USB_3V3	3.32	3.3	6.1	4	20	13.3
Group DRAM	VDD1	1.8	1.78	9.1	4	16	7.1

Table continues on the next page...

Table 9. Audio+Video_Playback(gplay) (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
	VDDQ	1.11	1.1	97.3	11.8	107.2	13
	VDD2	1.11	1.09	421.1	110.7	454.6	120.8
Measurement duration: 60 s Die temperature: Zone0 42 °C / Zone1 44 °C Utilization: CPU - 4 %, DDR - 8.6 %							

4.2.4 Audio+Video_Stream(gplay)

For this use case, MIPI-DSI port was connected to a 1080p TV display.

The video file used for playback was mkv file format compressed using the HEVC standard with full HD resolution at 29.97 fps and the audio encoding was AACL in a 2-channel configuration with 44.1 kHz samples/s.

A server was setup to host the mkv video file for streaming.

The video streaming was done using an Ethernet adapter and the player was gplay:

```
gplay-1.0 $FILE
```

The use case is as follows:

- The CPU frequency governor is set to *performance* (The CPU frequency is set to the maximum value 1800 MHz).
- The Arm® Cortex®-A53 core is power-gated if the kernel is in the lowest level of idle.
- The Arm L2 cache and PLAT are powered on.
- The Arm Cortex-M7 is in the reset status.
- All the unused PLLs are off and the unused clocks are gated.
- The operating system is on.

Table 10 shows the measurement results when this use case is applied on the i.MX 8M Plus processor.

Table 10. Audio_+Video_Stream(gplay)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.01	1.00	666.40	65.80	653.60	65.50
	VDD_SOC	0.94	0.93	1532.90	1051.20	1410.60	975.40
	NVCC_DRAM	1.10	1.09	343.20	129.20	371.20	141.20

Table continues on the next page...

Table 10. Audio_+Video_Stream(gplay) (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	240.96
	Total	-	-	-	-	-	1423.46
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	137.01
Details for [VDD_xxx_1P8 + VDD_xxx _0P8 + VDD_USB_3P3] and Group DRAM							
VDD_xxx_1P8	VDD_PLL_ANA_1V8	1.81	1.81	17.7	17.2	32.1	31.1
	VDD_EARC_1V8	1.82	1.81	2	2	3.6	3.5
	VDD_HDMI_1V8	1.82	1.81	0.1	0	0.1	0
	VDD_LVDS_1V8	1.82	1.81	0.1	0	0.3	0
	VDD_MIPI_1V8	1.81	1.8	3.7	3.7	6.8	6.6
	VDD_PCI_1V8	1.77	1.76	50.2	50	88.7	88.1
	VDD_USB_1V8	1.82	1.81	0.2	0	0.4	0.3
VDD_xxx_0P8	VDD_PLL_ANA_0V8	0.95	0.94	6.2	5.6	5.8	5.3
	VDD_HDMI_0V8	0.96	1.81	1.6	1.2	1.5	1.1
	VDD_MIPI_0V8	0.94	0.92	24.8	22	22.9	20.4
	VDD_PCI_0V8	0.88	0.87	79.9	79.5	69.8	68.8
	VDD_USB_0V8	0.96	0.95	5.3	2.6	5	2.4
VDD_USB_3P3	VDD_USB_3V3	3.32	3.31	5.9	4	19.4	13.4
Group DRAM	VDD1	1.8	1.78	8.9	3.8	15.7	6.8
	VDDQ	1.11	1.1	104.6	11.4	114.3	12.5
	VDD2	1.11	1.09	424.4	107.7	455.4	117.6
Measurement duration: 60 s Die temperature: Zone0 43 °C / Zone1 44 °C							

Table continues on the next page...

Table 10. Audio_+Video_Stream(gplay) (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Utilization : CPU - 3%, DDR - 8.6%							

4.3 Core benchmark

The following use case scenarios were tested:

- 4-core Dhrystone
- 4-core Whetstone
- Coremark

4.3.1 4-core Dhrystone

Dhrystone is a synthetic benchmark used to measure the integer computational performance of processors and compilers. The small size of the Dhrystone benchmark enables it to fit into the L1 cache and minimizes accesses to the L2 cache and DDR.

In this use case, the Dhrystone test is performed by 4 Cortex®-A53 cores (because Dhrystone is a single thread benchmark, 4 instances were started). All Cortex-A53 cores run the test in a loop at a frequency of 1800 MHz.

- The DDR clock is 2000 MHz.
- The NOC clock is 1000 MHz.
- The AXI clock is 400 MHz.
- The AHB clock is 133 MHz.
- The IPG clock is 67 MHz.
- CPU frequency governor is set to `performance` (CPU frequency is set to the maximum value 1800 MHz).

Table 11 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 11. Dhrystone

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	0.98	0.97	1168.50	1144.20	1133.70	1105.80
	VDD_SOC	0.94	0.93	880.40	842.30	819.80	785.30
	NVCC_DRAM	1.11	1.10	119.80	67.50	130.60	74.00
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	202.78

Table continues on the next page...

Table 11. Dhrystone (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
	Total	-	-	-	-	-	2168.28
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	4.81
Measurement duration: 60 s Die temperature: Zone0 48 °C / Zone1 53 °C Utilization : CPU - 99%, DDR - 0.09%							

4.3.2 4-core Whetstone

Whetstone is a similar benchmark for integer and string operations like Dhrystone. The Whetstone is also a synthetic benchmark which primarily measures the floating-point arithmetic performance.

In this use case, the Whetstone test is performed by 4 Cortex®-A53 cores (because Whetstone is a single thread benchmark too, 4 instances were started). All Cortex-A53 cores run the test in a loop at a frequency of 1800 MHz.

- CPU frequency governor is set to `performance` (CPU frequency is set to maximum value 1800 MHz)
- The DDR clock is 2000 MHz

Table 12 shows the measurement results when this use case is applied on the i.MX 8M Plus processor.

Table 12. Whetstone

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	0.99	0.98	1244.40	822.40	1204.30	801.50
	VDD_SOC	0.94	0.93	888.10	839.40	824.20	782.50
	NVCC_DRAM	1.10	1.10	127.50	66.60	139.10	73.00
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	202.58
	Total	-	-	-	-	-	1859.98
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	3.39
Measurement duration: 60 s Die temperature: Zone0 46 °C / Zone1 49 °C Utilization : CPU - 99%, DDR - 0.08%							

4.3.3 Coremark

Coremark is a modern, sophisticated benchmark that lets you accurately measure the processor performance and is intended to replace the older Dhrystone benchmark. Arm® recommends using Coremark over Dhrystone.

For the best performance, compile as follows:

```
-O2 -DMULTITHREAD=4 -DUSE_PTHREAD -lpthread -O3 -funroll-all-loops
--param max-inline-insns-auto=550 -ftracer -falign-jumps=16 -ftree-loop-im -fivopts
-ftree-loop-ivcanon -fvect-cost-model -fvariable-expansion-in-unroller
--param max-unrolled-insns=999999 --param max-average-unrolled-insns=99999999
--param iv-max-considered-uses=9999999 --param iv-consider-all-candidates-bound=99999
--param iv-always-prune-cand-set-bound=999999   -fmodulo-sched
-fmodulo-sched-allow-regmoves -fgcse-lm -fgcse-sm -fgcse-las -funsafe-loop-optimizations
-freschedule-modulo-scheduled-loops  -ftree-vectorize -DPERFORMANCE_RUN=1   -lrt
```

- The CPU frequency governor is set to *performance* (The CPU frequency is set to the maximum value 1800 MHz).
- The DDR clock is 2000 MHz.

Table 13 shows the measurement results when this use case is applied to the i.MX 8M Plus processor with Coremark running on the four CPU cores.

Table 13. Coremark

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.00	0.97	1089.30	952.80	1056.00	924.90
	VDD_SOC	0.94	0.93	880.10	840.90	820.40	783.90
	NVCC_DRAM	1.11	1.10	100.20	66.70	110.20	73.20
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	202.65
	Total	-	-	-	-	-	1985.05
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	3.92
Measurement duration: 60 s Die temperature: Zone0 45 °C / Zone1 49 °C Utilization: CPU - 92%, DDR - 0.08%							

4.4 GPU

For GPU, the following use case scenarios were tested:

- MM07
- MM06
- GPU_Kanzi

- GPU_GLmark

MM07 and MM06 are 3D-gaming benchmarks. The graphics are loaded from the SD card into the DDR memory, processed by GPU3D, and copied to a display buffer in the DDR memory. It is displayed on the1080p display (through MIPI-DSI).

- The CPU frequency governor is set to `performance` (The CPU frequency is set to the maximum value 1800 MHz)
- The DDR frequency is 2000 MHz

4.4.1 GPU_MM07

Table 14 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 14. GPU_MM07

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.01	1.00	717.30	74.20	705.60	73.70
	VDD_SOC	0.94	0.92	1880.50	1504.30	1721.20	1383.60
	NVCC_DRAM	1.10	1.09	373.30	206.20	403.50	224.50
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	231.79
	Total	-	-	-	-	-	1913.99
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	286.26
Measurement duration: 60 s Die temperature: Zone0 47 °C / Zone1 49 °C Utilization: CPU - 4%, DDR - 19%							

4.4.2 GPU_MM06

Table 15 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 15. GPU_MM06

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.01	1.00	621.20	133.20	610.40	132.40
	VDD_SOC	0.93	0.92	1943.70	1704.30	1769.00	1562.20

Table continues on the next page...

Table 15. GPU_MM06 (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
	NVCC_DRAM	1.10	1.09	372.30	263.30	401.70	285.80
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	232.65
	Total	-	-	-	-	-	2213.45
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	363.95
Measurement duration: 60 s Die temperature: Zone0 46 °C / Zone1 49 °C Utilization: CPU - 5%, DDR - 34%							

4.4.3 GPU_Kanzi

Table 16 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 16. GPU_Kanzi

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.01	0.99	1057.70	189.70	1021.40	188.00
	VDD_SOC	0.93	0.92	2003.20	1517.40	1821.20	1395.20
	NVCC_DRAM	1.10	1.09	431.20	197.80	465.10	215.20
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	231.19
	Total	-	-		-	-	2029.99
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	258.45
Measurement duration: 60 s							

Table continues on the next page...

Table 16. GPU_Kanzi (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Die temperature: Zone0 45 °C / Zone1 47 °C Utilization: CPU - 0%, DDR - 14%							

4.4.4 GPU_GLmark

Table 17 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 17. GPU_GLmark

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.01	1.00	620.00	150.30	609.00	149.10
	VDD_SOC	0.93	0.92	1976.00	1594.40	1807.70	1463.40
	NVCC_DRAM	1.10	1.09	318.50	243.40	344.70	264.40
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3						229.26
	Total						2106.56
Group DRAM	VDD1 + VDDQ + VDD2						310.00
Measurement duration: 60 s Die temperature: Zone0 47 °C / Zone1 49 °C Utilization: CPU - 5 %, DDR - 25 %							

4.5 Heavy load use cases

The following heavy load use case scenarios were tested:

- VPU
- 4-core Dhryst + MM07 + VPU
- 4-core Memtest + MM07 + VPU
- 4-core Stream + MM07 + VPU
- Coremark + Kanzi

The purpose of these use cases is to provide power data for heavy load use cases to show the power consumption in extreme conditions.

4.5.1 VPU

This use case has the following features:

- 1080p display ON, 1920 x 1080, 60 fps.
- HEVC 1080p30 file decoding via the Hantro decoder (no video playback). `g2dec -P -Ers -ibs -b -N200 -X HEVC_1920x1080_29.97fps_AACLC_44.1Khz_2ch.mkv`
- The CPU frequency governor is set to *performance* (The CPU frequency is set to the maximum value 1800 MHz).
- The DDR frequency is set to 2000 MHz.

Table 18 shows the measurement results when this use case is applied to the i.MX 8M Mini processor.

Table 18. VPU

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	0.99	0.99	672.10	472.10	660.80	465.40
	VDD_SOC	0.94	0.93	1066.60	954.30	991.70	887.50
	NVCC_DRAM	1.10	1.09	159.40	125.20	173.60	136.90
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	234.78
	Total	-	-	-	-	-	1724.98
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	131.78
Measurement duration: 60 s Die temperature: Zone0 45 °C / Zone1 48 °C Utilization: CPU - 50 %, DDR - 4 %							

4.5.2 4-core Dhryst + MM07 + VPU

This use case runs 4-core Dhrystone, GPU_MM07, and VPU in parallel.

- The CPU frequency governor is set to `performance` (The CPU frequency is set to the maximum value 1800 MHz).
- The DDR frequency is set to 2000 MHz.

Table 19 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 19. Dhryst + MM07 + VPU

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	0.98	0.97	1192.20	1070.90	1156.20	1037.30
	VDD_SOC	0.94	0.92	1838.70	1539.80	1694.10	1415.40
	NVCC_DRAM	1.10	1.09	353.90	206.70	382.40	225.10
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	235.20
	Total	-	-	-	-	-	2913.40
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	307.49
Measurement duration: 60 s Die temperature: Zone0 55 °C / Zone1 60 °C Utilization: CPU - 99 %, DDR - 21 %							

4.5.3 4-core Memtest + MM07 + VPU

This use case runs memtester (an effective userspace tester for stress-test the memory subsystem) and Suspend mode in parallel.

- The CPU frequency governor is set to `performance` (The CPU frequency is set to the maximum value 1800 MHz).
- The DDR frequency is set to 2000 MHz.

Table 20 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 20. 4-core Memtest + MM07 + VPU

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	0.99	0.97	1207.30	883.40	1166.40	860.30
	VDD_SOC	0.93	0.92	1883.50	1566.00	1700.90	1438.70
	NVCC_DRAM	1.10	1.09	440.10	265.60	473.80	288.20
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40

Table continues on the next page...

Table 20. 4-core Memtest + MM07 + VPU (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	235.22
	Total	-	-	-	-	-	2822.82
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	425.75
Measurement duration: 60 s Die temperature: Zone0 56 °C / Zone1 62 °C Utilization: CPU - 100 %, DDR - 56 %							

4.5.4 4-core Stream + MM07 + VPU

This use case run Stream and Suspend mode in parallel.

- The CPU frequency governor is set to *performance* (The CPU frequency is set to the maximum value 1800 MHz).
- The DDR frequency is set to 2000 MHz.

Table 21 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 21. 4-core Stream + MM07 + VPU

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	0.99	0.98	1162.60	854.90	1121.40	833.20
	VDD_SOC	0.93	0.92	1856.40	1576.20	1687.80	1447.90
	NVCC_DRAM	1.10	1.09	319.80	241.80	344.90	263.00
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	235.25
	Total	-	-	-	-	-	2779.75
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	413.15
Measurement duration: 60 s							

Table continues on the next page...

Table 21. 4-core Stream + MM07 + VPU (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Die temperature: Zone0 56 °C / Zone1 61 °C Utilization: CPU - 100 %, DDR - 56 %							

4.5.5 Coremark + Kanzi

This use case runs Coremark and GPU_Kanzi in parallel.

- The CPU frequency governor is set to `performance` (The CPU frequency is set to the maximum value 1800 MHz).
- The DDR frequency is set to 2000 MHz.

Table 22 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 22. Coremark + Kanzi

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.00	0.97	1123.00	921.50	1082.50	895.30
	VDD_SOC	0.94	0.92	2026.10	1473.90	1845.00	1356.10
	NVCC_DRAM	1.10	1.09	432.50	192.20	465.80	209.20
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	231.70
	Total	-	-	-	-	-	2692.70
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	246.65
Measurement duration: 60 s Die temperature: Zone0 54 °C / Zone1 58 °C Utilization: CPU - 91 %, DDR - 14 %							

4.6 Memory

The following memory-centric use case scenarios were tested:

- Memset
- Memcpy

- Stream

Memset and Memcpy are part of a perf-bench (a general framework for benchmark suites).

4.6.1 Memset

Suite for evaluating the performance of a simple memory set in various ways.

- The size of the memory buffers is set to 1024 MB.
- The CPU frequency governor is set to `performance` (The CPU frequency is set to the maximum value 1800 MHz).
- The DDR frequency is set to 2000 MHz.

Table 23 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 23. Memset

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.00	0.99	466.20	324.70	460.20	321.60
	VDD_SOC	0.94	0.93	1195.70	1160.00	1110.20	1075.00
	NVCC_DRAM	1.09	1.08	593.90	423.00	634.90	455.00
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	202.74
	Total	-	-	-	-	-	2054.74
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	241.40
Measurement duration: 60 s Die temperature: Zone0 48 °C / Zone1 51 °C Utilization: CPU - 26 %, DDR - 65 %							

4.6.2 Memcpy

Memcpy is a suite for evaluating the performance of a simple memory copy in various ways.

- The size of the memory buffers is set to 1024 MB.
- The CPU frequency governor is set to `performance` (The CPU frequency is set to the maximum value 1800 MHz).
- The DDR frequency is set to 2000 MHz.

Table 24 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 24. Memcpy

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.00	0.99	525.10	306.90	515.40	304.00
	VDD_SOC	0.94	0.93	1001.90	977.60	932.20	908.90
	NVCC_DRAM	1.10	1.09	188.20	171.50	206.20	187.00
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	202.67
	Total	-	-	-	-	-	1602.97
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	219.44
Measurement duration: 60 s Die temperature: Zone0 43 °C / Zone1 46 °C Utilization: CPU - 25 %, DDR - 34 %							

4.6.3 Stream

The Stream benchmark is a simple synthetic benchmark program that measures the sustainable memory bandwidth (in MB/s) and the corresponding computation rate for simple vector kernels.

- The stream array size is set to 102400000 elements.
- All phases are included (Copy, Scale, Add, and Triad).
- The CPU frequency governor is set to `performance` (The CPU frequency is set to the maximum value 1800 MHz).
- The DDR frequency is set to 2000 MHz.

Table 25 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 25. Stream

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.00	0.98	1143.20	724.60	1108.10	708.10
	VDD_SOC	0.94	0.93	1143.50	1015.20	1067.30	943.20
	NVCC_DRAM	1.10	1.09	318.70	216.50	346.20	235.60

Table continues on the next page...

Table 25. Stream (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	202.81
	Total	-	-	-	-	-	2090.11
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	319.98
Measurement duration: 60 s Die temperature: Zone0 48 °C / Zone1 51 °C Utilization: CPU - 87 %, DDR - 46 %							

4.7 Storage – SD3.0 card

The following use case scenarios were tested:

- DD_RD_SDCARD
- DD_WRT_SDCARD

4.7.1 DD_RD_SDCARD

- The CPU frequency governor is set to `performance` (The CPU frequency is set to the maximum value 1800 MHz).
- The DDR frequency is set to 2000 MHz.

Table 26 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 26. DD_RD_SDCARD

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.01	1.00	515.40	148.10	504.10	146.80
	VDD_SOC	0.94	0.93	1086.10	870.70	1009.60	811.10
	NVCC_DRAM	1.10	1.09	333.70	117.30	360.30	128.20
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	202.29

Table continues on the next page...

Table 26. DD_RD_SDCARD (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
	Total	-	-	-	-	-	1288.79
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	76.37
Measurement duration: 60 s Die temperature: Zone0 43 °C / Zone1 45 °C Utilization: CPU - 25 %, DDR - 6.8 %							

4.7.2 DD_WRT_SDCARD

- Set the maximum amount of data that the kernel reads ahead for a single file to 512 KB. `echo 512 > /sys/block/<bdev>/queue/read_ahead_kb`
- The CPU frequency governor is set to *`performance`* (The CPU frequency is set to the maximum value 1800 MHz).
- The DDR frequency is set to 2000 MHz .

Table 27 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 27. DD_WRT_SDCARD

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.01	0.99	824.90	232.40	803.20	230.00
	VDD_SOC	0.94	0.93	1000.70	865.40	927.60	806.30
	NVCC_DRAM	1.10	1.09	177.20	94.30	194.00	103.20
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	202.25
	Total	-	-	-	-	-	1342.15
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	74.36
Measurement duration: 60 s Die temperature: Zone0 39 °C / Zone1 41 °C Utilization: CPU - 65 %, DDR - 0.77 %							

4.8 Storage – eMMC

The following use case scenarios were tested:

- DD_RD_eMMC
- DD_WRT_eMMC

4.8.1 DD_RD_eMMC

- Set the maximum amount of data that the kernel reads ahead for a single file to 512 KB. `echo 512 > /sys/block/<bdev>/queue/read_ahead_kb`
- The CPU frequency governor is set to *performance* (The CPU frequency is set to the maximum value 1800 MHz).
- The DDR frequency is set to 2000 MHz.

Table 28 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 28. DD_RD_eMMC

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.01	0.99	660.40	279.50	646.90	276.90
	VDD_SOC	0.94	0.93	1097.40	927.20	1020.00	863.00
	NVCC_DRAM	1.11	1.09	331.10	150.60	358.10	164.30
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	202.48
	Total	-	-	-	-	-	1507.08
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	144.23
Measurement duration: 60 s Die temperature: Zone0 43 °C / Zone1 45 °C Utilization: CPU - 21 %, DDR - 8.9 %							

4.8.2 DD_WRT_eMMC

- Set the maximum amount of data that the kernel reads ahead for a single file to 512 KB.

  ```
  echo 512 > /sys/block/<bdev>/queue/read_ahead_kb
  ```

- The CPU frequency governor is set to *performance* (The CPU frequency is set to the maximum value 1800 MHz).
- The DDR frequency is set to 2000 MHz.

Table 29 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 29. DD_WRT_eMMC

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.00	0.99	699.80	328.50	685.90	325.00
	VDD_SOC	0.94	0.93	1000.40	869.80	933.70	810.50
	NVCC_DRAM	1.10	1.09	177.50	97.10	193.40	106.30
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	202.44
	Total	-	-	-	-	-	1444.64
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	92.77
Measurement duration: 60 s Die temperature: Zone0 43 °C / Zone1 44 °C Utilization: CPU - 34 %, DDR - 4.1 %							

4.9 Storage – USB 3.0

The following use case scenarios were tested:

- DD_RD_USB3.0
- DD_WRT_ USB3.0

4.9.1 DD_RD_USB3.0

- Set the maximum amount of data that the kernel reads ahead for a single file to 512 kbit.

  ```
  echo 512 > /sys/block/<bdev>/queue/read_ahead_kb
  ```

- The CPU frequency governor is set to *performance* (The CPU frequency is set to the maximum value 1800 MHz).
- The DDR frequency is set to 2000 MHz.

Table 30 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 30. DD_RD_ USB3.0

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.01	0.99	663.10	192.90	649.20	191.10
	VDD_SOC	0.94	0.93	1091.90	895.30	1015.00	833.90

Table continues on the next page...

Table 30. DD_RD_ USB3.0 (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
	NVCC_DRAM	1.11	1.09	336.30	107.30	363.50	117.40
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	375.90
	Total	-	-	-	-	-	1518.70
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	81.78
Measurement duration: 60 s Die temperature: Zone0 43 °C / Zone1 45 °C Utilization: CPU - 25 %, DDR - 6.8 %							

4.9.2 DD_WRT_ USB3.0

- Set the maximum amount of data that the kernel reads ahead for a single file to 512 kbit.

  ```
  echo 512 > /sys/block/<bdev>/queue/read_ahead_kb
  ```

- The CPU frequency governor is set to *performance* (The CPU frequency is set to the maximum value 1800 MHz).
- The DDR frequency is set to 2000 MHz.

Table 31 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 31. DD_WRT_USB3.0

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.01	0.99	668.00	300.70	654.30	297.50
	VDD_SOC	0.94	0.93	994.90	888.30	923.40	827.40
	NVCC_DRAM	1.11	1.09	171.80	97.40	187.70	106.60
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.00	0.50	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-		364.51
	Total	-	-	-	-	-	1596.41

Table continues on the next page...

Table 31. DD_WRT_USB3.0 (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	88.54
Measurement duration: 60 s Die temperature: Zone0 41 °C / Zone1 42 °C Utilization: CPU - 35 %, DDR - 0.52 %							

4.10 Product use cases

The following use case scenarios were tested:

- Fleet management, driver monitoring systems, dual camera, stereo vision, and vacuum robot
- Video conferencing
- Surveillance - headless (2xISP)
- HMI
- Machine vision (1xISP)
- eIQ - Benchmarking tests
- M7 Coremark
- Display stress test
- ISP camera test

4.10.1 Fleet management, driver monitoring systems, dual camera, stereo vision, and vacuum robot

- Dual camera 1080p60 (one for preview, one for encode).
- HEVC encodes one camera feeds and write to the USB disk.
- 2 instances of Object detection from the file-based input.
- MIPI-DSI Display 1080p60 - camera preview.
- GPU - run simple UI examples from GPU tutorials.
- Stress 1x Cortex®-A53 core to simulate application stress - 50 %.

Table 32 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 32. Fleet management, driver monitoring systems, dual camera, stereo vision, and vacuum robot

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.01	0.96	878.75	323.82	852.17	312.87

Table continues on the next page...

Table 32. Fleet management, driver monitoring systems, dual camera, stereo vision, and vacuum robot (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
	VDD_SOC	0.92	0.91	3394.47	2327.23	3005.81	2104.08
	NVCC_DRAM	1.10	1.08	357.30	228.01	383.37	247.21
	NVCC_SNVS_1P8	1.79	1.77	0.30	0.24	0.53	0.42
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	184.14
	Total	-	-	-	-	-	2848.72
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	404.13
Measurement Duration: 60 s Die temperature: Zone0 58 °C / Zone1 53 °C Utilization: CPU - 38 %, DDR - 43 %							

4.10.2 Video conferencing

- Camera 1080p60 w/ISP preview.
- H264 encodes of camera feed and stream over Ethernet.
- Receive H264 encoded stream over Ethernet, decode and display.
- Stress 1xA53 core to simulate application stress - 75 %.

Table 33 and Table 34 show the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 33. Video conferencing (with camera OV5640)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.01	0.96	902.86	263.93	886.13	255.73
	VDD_SOC	0.94	0.93	1940.00	1126.48	1774.51	1043.95
	NVCC_DRAM	1.10	1.09	243.96	145.68	264.54	158.52
	NVCC_SNVS_1P8	1.79	1.78	0.28	0.22	0.50	0.39
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	62.50

Table continues on the next page...

Table 33. Video conferencing (with camera OV5640) (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
	Total	-	-	-	-	-	1521.09
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	172.86
Measurement duration: 60 s Die temperature: Zone0 44 °C / Zone1 46 °C Utilization: CPU - 23 %, DDR - 12 %							

Table 34. Video conferencing (with Basler 4K add-on camera kit)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.01	0.96	871.12	263.36	855.11	255.23
	VDD_SOC	0.94	0.92	2106.02	1334.45	1913.62	1231.76
	NVCC_DRAM	1.10	1.09	249.39	159.58	271.11	173.55
	NVCC_SNVS_1P8	1.79	1.78	0.29	0.22	0.51	0.40
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	63.55
	Total	-	-	-	-	-	1724.49
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	183.43
Measurement duration: 60 s Die temperature: Zone0 47 °C / Zone1 48 °C Utilization: CPU - 22 %, DDR - 14 %							

4.10.3 Surveillance - headless (2xISP)

- Dual camera 1080p60 w/ISP capture.
- HEVC encodes one of the camera feed and stream over network.
- Stress 1xA53 core to simulate application stress - 50 %.
- 2 instances of Object detection from the file-based input - ML stress.

Table 35 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 35. Surveillance - headless (2xISP)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.01	0.95	800.78	281.58	779.76	271.15
	VDD_SOC	0.93	0.91	3638.31	2242.27	3214.64	2027.35
	NVCC_DRAM	1.10	1.08	367.68	218.11	396.11	236.44
	NVCC_SNVS_1P8	1.79	1.78	0.30	0.23	0.53	0.41
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	26.82
	Total	-	-	-	-	-	2562.17
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	383.50
Measurement duration: 60 s Die temperature: Zone0 54 °C / Zone1 53 °C Utilization: CPU - 31 %, DDR - 40 %							

4.10.4 HMI

- 1x1080p60 (MIPI-DSI) - H264 decode and display stream received ENET (TSN).
- 1x1080p60 (LVDS) - UI (using GPU examples).
- Stress 1xA53 core to simulate application stress - 50 % load.
- Stress M7 - run CoreMark, running at 800 MHz.

Table 36 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 36. HMI

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.01	0.92	862.12	198.04	838.85	188.01
	VDD_SOC	0.94	0.92	1602.17	1292.67	1474.04	1194.17
	NVCC_DRAM	1.10	1.09	254.64	133.39	276.02	145.30
	NVCC_SNVS_1P8	1.79	1.78	0.29	0.22	0.51	0.40

Table continues on the next page...

Table 36. HMI (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	135.62
	Total	-	-	-	-	-	1663.50
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	161.70
Measurement duration: 60 s Die temperature: Zone0 48 °C / Zone1 47 °C Utilization: CPU - 17 %, DDR - 9 %							

4.10.5 Machine vision (1xISP)

- 4Kp30 ISP capture + dewarp downscale to 1080p30.
- HEVC encodes camera feed and stream over ENET (TSN).
- Object detection from the file-based input.
- Stress 1xA53 core to simulate application stress - 50 %.

Table 37 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 37. Machine vision (1xISP)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.01	0.92	889.28	235.87	870.01	224.11
	VDD_SOC	0.93	0.91	3725.89	2090.79	3268.75	1898.07
	NVCC_DRAM	1.10	1.09	355.47	211.22	383.87	229.31
	NVCC_SNVS_1P8	1.79	1.78	0.29	0.23	0.52	0.41
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	22.98
	Total	-	-	-	-	-	2374.87
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	356.19
Measurement duration: 60 s							

Table continues on the next page...

Table 37. Machine vision (1xISP) (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Die temperature: Zone0 53 °C / Zone1 52 °C Utilization: CPU - 23 %, DDR - 37 %							

4.10.6 eIQ - Benchmarking tests

- Run eIQ benchmark test (standard models).

Table 38 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 38. eIQ - Benchmarking tests

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.00	0.85	1040.04	76.91	1004.20	65.37
	VDD_SOC	0.94	0.92	2876.89	1814.42	2576.67	1652.31
	NVCC_DRAM	1.11	1.09	349.00	160.56	376.83	174.54
	NVCC_SNVS_1P8	1.79	1.78	0.29	0.22	0.51	0.39
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	21.95
	Total	-	-	-	-	-	1914.57
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	259.26
Measurement duration: 60 s Die temperature: Zone0 46 °C / Zone1 46 °C Utilization: CPU - 5 %, DDR - 28 %							

4.10.7 M7 Coremark

- Run CoreMark on M7 core, running at 800 MHz.

Table 39 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 39. M7 Coremark

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.00	0.85	374.91	14.34	317.87	12.23
	VDD_SOC	0.94	0.93	1142.88	1057.08	1057.85	981.31
	NVCC_DRAM	1.10	1.09	135.62	67.48	147.81	73.70
	NVCC_SNVS_1P8	1.79	1.78	0.28	0.21	0.50	0.38
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	20.75
	Total	-	-	-	-	-	1088.37
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	6.28
Measurement duration: 60 s Die temperature: Zone0 38 °C / Zone1 39 °C Utilization: CPU - 0 %, DDR - 0.1 %							

4.10.8 Display stress test

- 1x1080p60 (MIPI-DSI) + 2x720p60 (HDMI and LVDS).
- HDMI - 720p60 HEVC video decode and display.
- MIPI-DSI - 1080p60 ISP camera capture and display.
- LVDS - 720p60 run GPU tutorial examples.

Table 40 and Table 41 show the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 40. Display stress test (with camera OV5640)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.00	0.85	484.92	64.80	453.95	55.14
	VDD_SOC	0.94	0.93	1482.54	1179.56	1370.10	1091.76
	NVCC_DRAM	1.10	1.09	223.82	145.57	242.19	158.47
	NVCC_SNVS_1P8	1.79	1.78	0.29	0.22	0.52	0.40

Table continues on the next page...

Table 40. Display stress test (with camera OV5640) (continued)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	215.71
	Total	-	-	-	-	-	1521.47
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	195.15
Measurement duration: 60 s Die temperature: Zone0 48 °C / Zone1 49 °C Utilization: CPU - 5 %, DDR - 14 %							

Table 41. Display stress test (with Basler 4K Add-on camera kit)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.00	0.85	654.60	77.54	608.96	66.23
	VDD_SOC	0.93	0.92	1676.33	1491.51	1535.30	1372.41
	NVCC_DRAM	1.10	1.09	242.25	193.72	263.09	210.41
	NVCC_SNVS_1P8	1.79	1.78	0.29	0.23	0.51	0.41
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	216.53
	Total	-	-	-	-	-	1865.99
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	306.13
Measurement duration: 60 s Die temperature: Zone0 52 °C / Zone1 53 °C Utilization: CPU - 7 %, DDR - 26 %							

4.10.9 ISP Camera test

- ISP Camera test – Process 4kp40 Bayer image and display on 1080p60 monitor.

Table 42 shows the measurement results when this use case is applied to the i.MX 8M Plus processor.

Table 42. ISP test (with Basler 4K add-on camera kit)

Power Group	Supply Domain	L5.4.70-2.3.0					
		Voltage(V)		I(mA)		P(mW)	
		peak	avg	peak	avg	peak	avg
Group SOC	VDD_ARM	1.00	0.85	493.77	40.09	464.31	34.22
	VDD_SOC	0.94	0.92	1463.32	1270.78	1349.75	1174.70
	NVCC_DRAM	1.10	1.09	240.30	170.68	261.58	185.55
	NVCC_SNVS_1P8	1.79	1.78	0.29	0.22	0.51	0.39
	VDD_xxx_1P81 + VDD_xxx _0P82 + VDD_USB_3P3	-	-	-	-	-	29.91
	Total	-	-	-	-	-	1424.76
Group DRAM	VDD1 + VDDQ + VDD2	-	-	-	-	-	214.14
Measurement duration: 60 s Die temperature: Zone0 42 °C / Zone1 43 °C Utilization: CPU - 2 %, DDR - 17 %							

5 Reducing power consumption

The overall system power consumption depends on system optimization. This includes both software optimization and how the system hardware is implemented. The following sections provides suggestions for system optimization. These are not an exhaustive list of every feature that could provide power reductions, but they are the easiest and most common ones.

- Run fast and idle
- Clock gating
- PLL reduction
- Core DVFS and System Bus scaling
- Use of LPDDR4 instead of DDR4
- Lower DDR frequencies
- DDR interface optimization
- Power gating of PHYs
- Distribution of workloads
- OCRAM use to minimize DDR accesses
- Thermal management to reduce leakage
- Future low-power modes from NXP

5.1 Run fast and idle

NXP testing and various research have shown that for most customer use cases, the best power/energy management protocol is to run the cores at maximum speeds for the workload and then drop to the lowest power mode possible. Although this strategy may not provide optimal energy savings for some use cases where constant data is being processed – (For example, low latency audio playback), this does work for other normal workloads. This trade-off must be considered for each application to quantify the overall effect on the system power/energy consumption.

Users should place the i.MX 8M Plus into the low-power modes (STOP) whenever possible.

5.2 Clock gating

The Clock Controller Module (CCM) within the i.MX 8M Plus provides a programmable method to disable the clock sources to modules when the modules are not used. Users should always configure the CCGR registers to apply proper clock gating. This is the first and simplest method to reduce wasted energy. Driving any unused signal whether on the SoC or the PCB is simply charging and discharging the line and load capacitance of this signal. The NXP BSP software releases implement clock gating by default.

5.3 PLL reduction

Each PLL block consumes significant energy when active. Each application has unique requirements, but when possible reduce the number of operating PLLs. The Clock Controller Module (CCM) within the i.MX 8M Plus provides programmable control for Clock Root Selection. This may allow common PLL Root Clocks within the application and reduce the number of active PLLs when operating.

When entering low-power states (WAIT/STOP), or low-power operating states (Audio Playback), reduce the number of active PLLs. This will lower the power requirements for these states. Ensure that the application takes into account for the PLL re-lock time when transitioning back to full operation.

5.4 Core DVFS and system bus scaling

Applying Dynamic Voltage and Frequency Scaling (DVFS) for the Arm® cores and scaling (not dynamic) the frequencies of the NOC, AXI, AHB, and IPG system bus clocks can significantly reduce the power consumption of the VDD_ARM and VDD_SOC domains. However, due to reduced operation of system frequencies, the accesses to the DDR or external memories may take longer, which may increase the energy consumption for specific use cases. This trade-off must be considered for each mode to quantify the overall effect on the system power consumption.

The power consumption on the VDD_ARM supply varies linearly with the CPU frequency and the number of Arm cores enabled. Table 43 presents VDD_ARM power consumption measured depending on the CPU Operating frequency and number of CPU cores enabled, with the enabled cores running Coremark benchmark.

Further power reduction on the VDD_ARM supply is therefore possible by lowering the CPU frequency or reducing the number of enabled Arm cores. However the VDD_ARM supply voltage cannot be set lower than the minimum voltage set point, see respective operating range tables in the *i.MX 8M Plus Applications Processor Datasheet for Consumer Products* (document IMX8MPCEC).

Table 43. VDD_ARM power consumption in function of CPU frequency and number of CPU cores active and running Coremark benchmark

VDD_ARM Power Consumption (mW)			
Number of enabled Arm Cortex-A53 CPU core running Coremark	Arm Cortex-A53 CPU Running Frequency		
	1.2 GHz	1.6 GHz	1.8 GHz
1 Core enabled	138	234	300
2 Cores enabled	242	425	535
4 Cores enabled	436	743	938

NOTE

Unused CPU cores are disabled using Linux CPU hotplug feature.

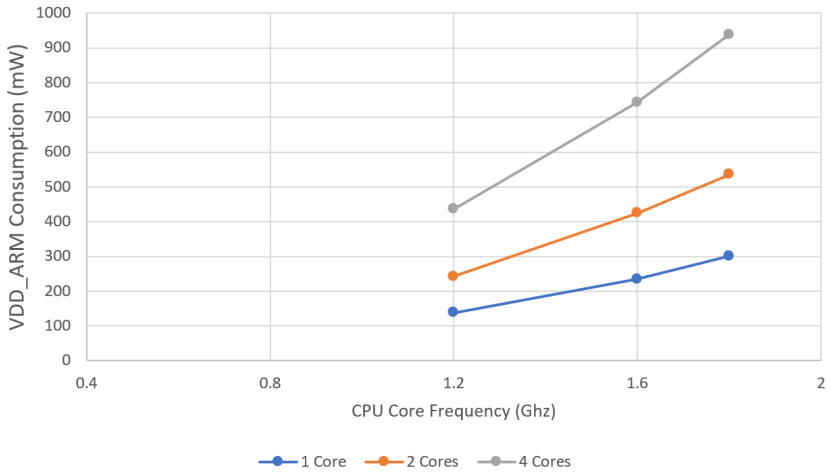

Figure 3. VDD_ARM power consumption vs #cores and CPU frequency

5.5 Use of LPDDR4 instead of DDR4

The memory architecture of LPDDR4 has been modified to achieve higher bandwidth and lower power consumption. It has reduced power consumption by lowering the supply voltage. The IO voltage of DDR4 is 1.2 V where the LPDDR4 IO voltage is 1.1 V. The DDR IO power can be calculated from the following formula:

Imax = N x C x V x (0.5 x F)

Where:

N - Number of DDR IO pins supplied by the power line

C - Equivalent external capacitive load (PCB trace and via capacitance + DDR pin capacitance)

V - DDR IO voltage (1.1 V or 1.2 V)

(0.5 x F)—Data change rate (DDR bus frequency)

In this equation, Imax is in Amps, C (in Farads), V (in Volts), and F (in Hertz).

NOTE

This is a worst-case value as DDR address and data signals are not toggling every cycle. Users can create their own estimates by adjusting the “0.5” value to correspond to their usage profile.

This equation shows the direct relational increase of current when the IO pin voltage is increased. An additional thought here is this additional power is consumed by both the DDR memory and the i.MX 8M Plus as they drive the DDR bus. The additional power consumption adds to the increased thermal dissipation of each device. Thus, it has a double effect on energy consumption. Thermal effects and management are explained below.

5.6 Lower DDR frequencies

As explained previously, the DDR IO bus frequency also contributes to the DDR IO current. This was represented as “**F**” in the above equation. This must be balanced with the system operating requirements. For lower power operating states, reduction of the DDR bus frequency can provide additional power savings.

5.7 DDR interface optimization

- Employ careful board routing of the DDR memories, maintaining the PCB trace lengths as short as possible. Longer trace lengths and additional vias create additional PCB capacitance for the signal. Thus, more wasted energy transitioning the signal path.
- Use as reduced an ODT (On-Die Termination) setting as possible. The termination used greatly influences the power consumption of the DDR interface pins. The DDR interface should be simulated to ensure the ODT variance does not reduce the bus signal integrity.
- Use a proper output driver impedance for the DDR interface pins that provide good impedance matching. Select the lowest possible drive strength that provides the required performance to reduce the current flowing through the DDR I/O pins. Again, simulation is recommended to ensure signal integrity.
- The use of the DDR memory offerings in the latest process technology can significantly reduce the power consumption of the DDR devices and the DDR I/O.
- Sizing DDR memory is important. If you select a 4 GB memory when only 2 GB is used, then you will be wasting the refresh current for the unused 2 GB of DDR.
- Sizing of ECC DDR regions is important as they use additional energy for this feature.

5.8 Power gating of PHYs

The PHYs of unused modules are often overlooked when searching for power savings. But many PHYs contain local PLLs or clocking circuits, and voltage references which consume power even when not in use. This relates to high-speed PHYs like Ethernet, MIPI, HDMI, PCIe, and USB.

The i.MX 8M Plus contains the General Power Controller (GPC) module for managing the power gating within the SoC. The GPC controls the following functions:

- Provide low-power mode control for the Cortex®-A53 and Cortex-M7 CPU platforms.
- Provide Power domain management all Arm® and SoC power domain.
- Provide domain control mechanism based on the Cortex-A53 and Cortex-M7 CPU platforms.
- Provide handshake with CCM for clock management in low-power mode.
- Provide handshake with SRC for power down and power up sequence.
- Provide handshake with analog for deep sleep mode control.

The i.MX 8M Plus contains 20 Power Gating Controllers (PGC) within the SoC. PGC is a power management component that controls the power-down and power-up sequencing of individual subsystems.

Utilize the i.MX 8M Plus power gating controls to reduce the power of hardware blocks not required within the SoC. If a PHY cannot be power gated, then ensure the PHY is clock gated as previously explained.

5.9 Distribution of workloads

The concept of distributed workloads is to review the system requirements and determine which SoC block is best suited for each task. By spreading the workload, then ideally the system can return to WAIT/STOP state much sooner. This applies to multicore distributions and to functions that might be suited for HiFi4 DSP, ML Engine, or graphics cores. System designers should ensure the design utilizes the optimal cores for the specific workloads or tasks on the i.MX 8M Plus for maximum efficiency. This is often easier said than done but does provide significant power savings if the system can return to the low-power state faster (run fast and idle).

5.10 OCRAM use to minimize DDR accesses

Significant power savings can be achieved by loading common highly accessed code into the OCRAM (On-Chip RAM). This reduces both the i.MX 8M Plus and the DDR memory current consumptions. Another advantage of utilizing the OCRAM is a performance increase since this code is delayed by DDR memory access time.

The i.MX BSP ensures that all commonly accessed low-power routines are located in the Arm® Trusted Firmware (ATF) from the internal OCRAM.

5.11 Thermal management to reduce leakage

Thermal management is also a key element of power reduction. As shown in Figure 4, as the temperature increases so does the SoC gate leakage current for each gate within the device. Millions of high gate leakage add-up when looking for the lowest power consumption. Figure 4 is not intended to be an exact leakage current measurement, but an illustration of the effects of temperature. As explained earlier, any power savings that can be achieved also reduce the temperature of the SoC and improve the lifetime reliability of the device.

As each system is unique, the system design should ensure the operating temperature of the SoC is as low as possible to reduce the leakage current loss. If this cannot be achieved from software controls, then the design should include a heat sink or other thermal management methods to remove the heat from the SoC.

Refer to the respective processor hardware developers guide for thermal guidelines.

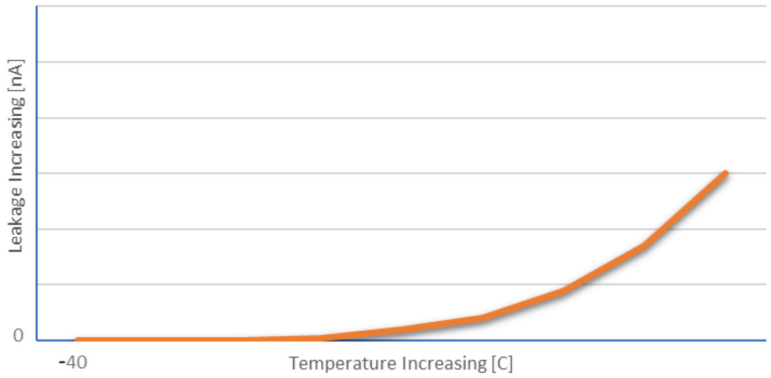

Figure 4. General Leakage Curve for 14-nm FF gate

5.12 Nominal drive mode

The NXP supplied Linux BSP GA release will configure the system to run in Over Drive Mode (ODM) by default. For certain user applications, this overdrive mode and associated performance may not be necessary. Users can instead transition to nominal drive mode as defined in the i.MX 8MPLus data sheet. Please note that dynamically changing from Over Drive Mode(ODM) to Nominal Drive Mode (NDM) is not supported.

5.13 Future low-power modes from NXP

It is recommended to routinely check for new and updated low-power software solutions. NXP understands that system power is a key element for most embedded designs. The NXP team is actively working to provide new ideas and software solutions to reduce SoC and system power.

6 Use case configuration and usage guidelines

NOTE

Before running a use case, `<configuration_script>.sh` must be run to configure the environment. These are: *setup.sh, setup_default.sh, setup_video.sh, DDRC_25MHz_setup.sh* , see Important commands.

6.1 Suspend mode

6.1.1 Steps to be performed before entering suspend (deep-sleep) mode

NOTE

All the programming steps below are performed in the Arm® trusted firmware from the internal RAM.

1. Read the DBGCAM register in DDRC to make sure that the explicit transaction command queue is empty. Wait until the AXI port is idle.
2. Do the following:
   - a) Put the DDR into self-refresh.
   - b) Transition the DDR PHY into LP3/IO retention state by using the DFI frequency operation.
   - c) Set the PwrOkIn signal in SRC to 0. This enables the data retention feature on the CKE and MEMRESET.
   - d) Gate the DDRC's CORE clock and APB clock.
   - e) Enable DDRMIX ISO to power-gate the DDRC and PHY.
3. Enter suspend mode.

6.1.2 Steps to be performed after exiting suspend mode

1. Restore all the settings for the DDRC and PHY to the required values.
2. The system proceeds to the run mode.

6.1.3 Suspend measurement use case

In this use case, all clocks and PLLs are turned off, except for the 32 kHz clock which is used to wake up the system:

1. Boot up the Linux image.
2. Run this command to put the system into the suspend (deep-sleep) mode:

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

3. Measure the power and record the result.

6.2 System idle mode

NOTE

No display was connected to the platform.

6.2.1 IDLE_DEFAULT

6.2.1.1 Clock configuration

The clock configuration in Table 44 is aligned with release L5.4.24.

Table 44. IDLE_DEFAULT clock configuration

Clock name	Frequency (MHz)
NOC	1000
AXI	400
AHB	133

Table continues on the next page...

Table 44. IDLE_DEFAULT clock configuration (continued)

Clock name	Frequency (MHz)
CPU	1200
DDRC	1000

NOTE

DDRC clock is used for DRAM controller, and it is 1/2 of DDR frequency. For example, DDRC clock at 1000 MHz means DDR frequency at 2000 MHz, and data rate at 4000 MTS.

6.2.1.2 PLL configuration

The PLL configuration in Table 45 is aligned with release L5.4.24.

Table 45. IDLE_DEFAULT PLL configuration

Clock root	Source selected	Frequency (MHz)
ARM_A53_CLK	PLL_ARM_MAIN_CLK	1200
NOC_CLK	SYS_PLL2_1000M	1000
MAIN_AXI_CLK	SYS_PLL1_800M	400
MEDIA_AXI_CLK	SYS_PLL2_1000M	off
ENET_AXI_CLK	SYS_PLL1_266M	off
NAND_USDHC_BUS_CLK	SYS_PLL1_266M	off
AHB_CLK_ROOT	SYS_PLL1_133M	133
IPG_CLK	AHB_CLK_ROOT	67

6.2.1.3 System setup

Disconnect everything except for the SD card.

Make sure that there are no displays connected to the platform.

1. Boot up the Linux OS.
2. Run `setup_default.sh` to put the system to idle mode, see Important commands.
3. Measure the power and record the result.

6.2.2 IDLE_LOW_BUS

6.2.2.1 Clock configuration

The clock configuration in Table 46 is aligned with release L5.4.24.

Table 46. IDLE_DDRC_266MHz clock configuration

Clock name	Frequency (MHz)
NOC	200
AXI	24
AHB	22.2
CPU	1200
DDRC	25

6.2.2.2 PLL configuration

The PLL configuration in Table 47 is aligned with release L5.4.24.

Table 47. IDLE_DDRC_266MHz PLL configuration

Clock root	Source selected	Frequency (MHz)
ARM_A53_CLK	PLL_ARM_MAIN_CLK	1200
NOC_CLK	SYS_PLL2_1000M	200
MAIN_AXI_CLK	OSC_24M	24
MEDIA_AXI_CLK	SYS_PLL2_1000M	off
ENET_AXI_CLK	SYS_PLL1_266M	off
NAND_USDHC_BUS_CLK	SYS_PLL1_266M	off
AHB_CLK_ROOT	SYS1_PLL_133M	22.2
IPG_CLK	AHB_CLK_ROOT	11.1

6.2.2.3 System setup

Disconnect everything except for the SD card.

Make sure that there are no displays connected to the platform.

1. Boot up the Linux OS.
2. Run `DDRC_25MHz_setup.sh` to put the system to idle mode, see Important commands.
3. Measure the power and record the result.

6.3 Audio_Playback

6.3.1 Clock configuration

The clock configuration in the following table is aligned with release L5.4.24.

Table 48. Audio_Playback(gplay) clock configuration

Clock name	Frequency (MHz)
NOC	200
AXI	24
AHB	22.2
CPU	1200
DDRC	25

6.3.2 PLL configuration

The PLL configuration in the following table is aligned with release L5.4.24.

Table 49. Audio_Playback(gplay) PLL configuration

Clock root	Source selected	Frequency (MHz)
ARM_A53_CLK	PLL_ARM_MAIN_CLK	1200
NOC_CLK	SYS_PLL2_1000M	200
MAIN_AXI_CLK	24M OSC	24
MEDIA_AXI_CLK	SYS_PLL1_1000M	off
ENET_AXI_CLK	SYS_PLL1_266M	266
NAND_USDHC_BUS_CLK	SYS_PLL1_266M	266
AHB_CLK_ROOT	SYS_PLL1_133M	22.2
IPG_CLK	AHB_ROOT_CLK	11.1
DRAM_CLK	SYS_PLL1_100M	25
DISP_PIXEL_CLK	VIDEO_PLL_CLK	--
SAIx_CLK	24M OSC	24
ENETx_REF_CLK	SYS2_PLL_125M	125
ENETx_TIME_CLK	SYS2_PLL_100M	100
ENET_PHY_REF_CLK	24M OSC	24
NAND_CLK	SYS1_PLL_400M	off
QSPI_CLK	SYS1_PLL_800M	off

Table continues on the next page...

Table 49. Audio_Playback(gplay) PLL configuration (continued)

Clock root	Source selected	Frequency (MHz)
USHDCx_CLK	SYS1_PLL_400M	400
I2Cx_CLK	24M OSC	24
UARTx_CLK	24M OSC	24
ECSPIx_CLK	24M OSC	off
PWMx_CLK	24M OSC	24
GPTx_CLK	24M OSC	24
TRACE_CLK	24M OSC	off
WDOG_CLK	24M OSC	off

6.3.3 Audio_Playback(gplay)

The audio file used was an mp3 file with a 128 kbps bitrate and a 44 kHz sample rate/s, played using the following options:

1. Boot up the Linux OS.
2. Run `setup.sh` , see Important commands.
3. Run `gplay_audio.sh` :

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

4. Measure the power and record the result.

6.3.4 Audio_Playback(gplay)_LOW_BUS

The audio file used was an mp3 file with a 128 kbps bitrate and a 44-kHz sample rate/s, played using the following options:

1. Boot up the Linux OS.
2. Run `DDRC_25MHz_setup.sh` , see Important commands.
3. Run `gplay_audio.sh` :

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

4. Measure the power and record the result.

6.3.5 Audio+Video_Playback(gplay)

For this use case, the MIPI-DSI port was connected to a 1080p TV display.

The video file used for playback is .mkv file format compressed with HEVC standard with full HD resolution at 29.97 fps and the audio encoding is AACL with a 44.1 kHz samples/s and 2-channel configuration.

The video file was locally played using gplay-1.0, with the following options:

1. Boot up the Linux OS.

2. Run `setup_video.sh` to put the system into idle mode, see Important commands.
3. Run `gplay_videoplayback.sh` :

```
path=`pwd`
FILE=HEVC_1920x1080_29.97fps_AACLC_44.1Khz_2ch.mkv
gplay-1.0 $path/$FILE
```

4. Measure the power and record the result.

6.3.6 Audio+Video_Playback_Stream(gplay)

For this use case, the MIPI-DSI port was connected to a 1080p TV display.

The video file used for playback is .mkv file format compressed with HEVC standard with full HD resolution at 29.97 fps and the audio encoding is AACL with a 44.1 kHz samples/s and 2-channel configuration.

The Video file was locally played using gplay-1.0, with the following options:

1. Boot up the Linux OS.
2. Run `setup_video_stream.sh` , see Important commands.
3. Run `gplay_video_stream.sh` :

```
video=HEVC_1920x1080_29.97fps_AACLC_44.1Khz_2ch.mkv server=ip or <server_hostname>
FILE=http://$server/$video
gplay-1.0 $FILE
```

4. Measure the power and record the result.

6.4 Core Dhrystone

NOTE

No display was connected to the platform.

6.4.1 Clock configuration

The clock configuration in Table 50 is aligned with release L5.4.24.

Table 50. 4 core Dhrystone clock configuration

Clock name	Frequency (MHz)
NOC	1000
AXI	400
AHB	133
CPU	1800
DDRC	1000

6.4.2 PLL configuration

The PLL configuration in Table 51 is aligned with release L5.4.24.

Table 51. 4 core Dhrystone PLL configuration

Clock root	Source selected	Frequency (MHz)
ARM_A53_CLK	PLL_ARM_MAIN_CLK	1800
NOC_CLK	SYS_PLL2_1000M	1000
MAIN_AXI_CLK	SYS_PLL1_800M	400
MEDIA_AXI_CLK	SYS_PLL2_1000M	off
ENET_AXI_CLK	SYS_PLL1_266M	266
NAND_USDHC_BUS_CLK	SYS_PLL1_266M	266
AHB_CLK_ROOT	SYS_PLL1_133M	133
IPG_CLK	AHB_ROOT_CLK	67
DRAM_CLK	PLL_DRAM_MAIN_CLK	1000
DISP_PIXEL_CLK	VIDEO_PLL_CLK	--
SAIx_CLK	24M OSC	24
ENETx_REF_CLK	SYS2_PLL_125M	125
ENETx_TIME_CLK	SYS2_PLL_100M	100
ENET_PHY_REF_CLK	24M OSC	24
NAND_CLK	SYS1_PLL_400M	off
QSPI_CLK	SYS1_PLL_800M	off
USHDCx_CLK	SYS1_PLL_400M	400
I2Cx_CLK	24M OSC	24
UARTx_CLK	24M OSC	24
ECSPIx_CLK	24M OSC	off
PWMx_CLK	24M OSC	24
GPTx_CLK	24M OSC	24
TRACE_CLK	24M OSC	off
WDOG_CLK	24M OSC	off

6.4.3 Steps

1. Boot up the Linux image and boot the board to the SD rootfs.

2. Run `setup.sh` , see Important commands.
3. Run `dhrystone_loop.sh` :

```
while [ "1" == "1" ]
do
sudo taskset -c 0 ./dhrystone/gcc_dry2 &
sudo taskset -c 1 ./dhrystone/gcc_dry2 &
sudo taskset -c 2 ./dhrystone/gcc_dry2 &
sudo taskset -c 3 ./dhrystone/gcc_dry2
done
```

4. Measure the power and record the result.

6.5 Core Whetstone

NOTE

No display was connected to the platform.

6.5.1 Clock configuration

The clock configuration in Table 52 is aligned with release L5.4.24.

Table 52. 4 core Whetstone clock configuration

Clock name	Frequency (MHz)
NOC	1000
AXI	400
AHB	133
CPU	1800
DDRC	1000

6.5.2 PLL configuration

The PLL configuration in Table 53 is aligned with release L5.4.24.

Table 53. 4 core Whetstone PLL configuration

Clock root	Source selected	Frequency (MHz)
ARM_A53_CLK	PLL_ARM_MAIN_CLK	1800
NOC_CLK	SYS_PLL2_1000M	1000
MAIN_AXI_CLK	SYS_PLL1_800M	400
MEDIA_AXI_CLK	SYS_PLL2_1000M	off
ENET_AXI_CLK	SYS_PLL1_266M	266

Table continues on the next page...

Table 53. 4 core Whetstone PLL configuration (continued)

Clock root	Source selected	Frequency (MHz)
NAND_USDHC_BUS_CLK	SYS_PLL1_266M	266
AHB_CLK_ROOT	SYS_PLL1_133M	133
IPG_CLK	AHB_ROOT_CLK	67
DRAM_CLK	PLL_DRAM_MAIN_CLK	1000
DISP_PIXEL_CLK	VIDEO_PLL_CLK	--
SAIx_CLK	24M OSC	24
ENETx_REF_CLK	SYS2_PLL_125M	125
ENETx_TIME_CLK	SYS2_PLL_100M	100
ENET_PHY_REF_CLK	24M OSC	24
NAND_CLK	SYS1_PLL_400M	off
QSPI_CLK	SYS1_PLL_800M	off
USHDCx_CLK	SYS1_PLL_400M	400
I2Cx_CLK	24M OSC	24
UARTx_CLK	24M OSC	24
ECSPIx_CLK	24M OSC	off
PWMx_CLK	24M OSC	24
GPTx_CLK	24M OSC	24
TRACE_CLK	24M OSC	off
WDOG_CLK	24M OSC	off

6.5.3 Steps

1. Boot up the Linux image and boot the board to the SD rootfs.
2. Run `setup.sh` , see Important commands.
3. Run `whetstone_loop.sh` :

```
while [ "1" == "1" ]
do
sudo taskset -c 0 ./whetsSP &
sudo taskset -c 1 ./whetsSP &
sudo taskset -c 2 ./whetsSP &
sudo taskset -c 3 ./whetsSP
done
```

4. Measure the power and record the result.

6.6 Coremark

NOTE

No display was connected to the platform.

6.6.1 Clock configuration

The clock configuration in Table 54 is aligned with release L5.4.24.

Table 54. Coremark clock configuration

Clock name	Frequency (MHz)
NOC	1000
AXI	400
AHB	133
CPU	1800
DDRC	1000

6.6.2 PLL configuration

The PLL configuration in Table 55 is aligned with release L5.4.24.

Table 55. Coremark PLL configuration

Clock root	Source selected	Frequency (MHz)
ARM_A53_CLK	PLL_ARM_MAIN_CLK	1800
NOC_CLK	SYS_PLL2_1000M	1000
MAIN_AXI_CLK	SYS_PLL1_800M	400
MEDIA_AXI_CLK	SYS_PLL2_1000M	off
ENET_AXI_CLK	SYS1_PLL_266M	266
NAND_USDHC_BUS_CLK	SYS1_PLL_266M	266
AHB_CLK_ROOT	SYS1_PLL_133M	133
IPG_CLK	AHB_ROOT_CLK	67
DRAM_CLK	PLL_DRAM_MAIN_CLK	1000
DISP_PIXEL_CLK	VIDEO_PLL_CLK	--
SAIx_CLK	24M OSC	24

Table continues on the next page...

Table 55. Coremark PLL configuration (continued)

Clock root	Source selected	Frequency (MHz)
ENETx_REF_CLK	SYS2_PLL_125M	125
ENETx_TIME_CLK	SYS2_PLL_100M	100
ENET_PHY_REF_CLK	24M OSC	24
NAND_CLK	SYS1_PLL_400M	off
QSPI_CLK	SYS1_PLL_800M	off
USHDCx_CLK	SYS1_PLL_400M	400
I2Cx_CLK	24M OSC	24
UARTx_CLK	24M OSC	24
ECSPIx_CLK	24M OSC	off
PWMx_CLK	24M OSC	24
GPTx_CLK	24M OSC	24
TRACE_CLK	24M OSC	off
WDOG_CLK	24M OSC	off

6.6.3 Steps

1. Boot up the Linux image and boot the board to the SD rootfs.
2. Run `setup.sh` , see Important commands.
3. Run `Coremark_loop.sh` :

```
while true; do
./Coremark.exe > /dev/null 2>&1
done
```

4. Measure the power and record the result.

6.7 GPU

Two benchmarks were used for GPU power measurements. A 1080p TV display was connected to the MIPI-DSI port. Before running any benchmark, the governor must be set to **performance** (see Important commands):

```
cpufreq-set -g performance
```

After setting the governor, run the respective GPU benchmark in a loop and start the power measurements logging at the desired time interval.

6.7.1 GPU_MM07

1. Run *setup_video.sh* , see Important commands.

2. Run `gpu_mm07.sh`:

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

3. Start power measurement and record the result.

6.7.1.1 Clock configuration

The clock configuration in Table 56 is aligned with release L5.4.24.

Table 56. MM07 clock configuration

Clock name	Frequency (MHz)
NOC	1000
AXI	400
AHB	133
CPU	1800
DDRC	1000

6.7.1.2 PLL configuration

The PLL configuration in Table 57 is aligned with release L5.4.24.

Table 57. MM07 PLL configuration

Clock root	Source selected	Frequency (MHz)
ARM_A53_CLK	PLL_ARM_MAIN_CLK	1800
NOC_CLK	SYS_PLL2_1000M	1000
MAIN_AXI_CLK	SYS_PLL1_800M	400
MEDIA_AXI_CLK	SYS_PLL2_1000M	500
ENET_AXI_CLK	SYS_PLL1_266M	266
NAND_USDHC_BUS_CLK	SYS_PLL1_266M	266
AHB_CLK_ROOT	SYS_PLL1_133M	133
IPG_CLK	AHB_ROOT_CLK	67
DRAM_CLK	PLL_DRAM_MAIN_CLK	1000
DISP_PIXEL_CLK	VIDEO_PLL_CLK	--

Table continues on the next page...

Table 57. MM07 PLL configuration (continued)

Clock root	Source selected	Frequency (MHz)
SAIx_CLK	24M OSC	24
ENETx_REF_CLK	SYS2_PLL_125M	125
ENETx_TIME_CLK	SYS2_PLL_100M	100
ENET_PHY_REF_CLK	24M OSC	24
NAND_CLK	SYS1_PLL_400M	off
QSPI_CLK	SYS1_PLL_800M	off
USHDCx_CLK	SYS1_PLL_400M	400
I2Cx_CLK	24M OSC	24
UARTx_CLK	24M OSC	24
ECSPIx_CLK	24M OSC	off
PWMx_CLK	24M OSC	24
GPTx_CLK	24M OSC	24
TRACE_CLK	24M OSC	off
WDOG_CLK	24M OSC	off

6.7.2 GPU_MM06

1. Run `setup_video.sh` , see Important commands.
2. Run `gpu_mm06.sh` :

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

3. Start power measurement and record the result.

6.7.2.1 Clock configuration

The clock configuration in Table 58 is aligned with release L5.4.24.

Table 58. MM06 clock configuration

Clock name	Frequency (MHz)
NOC	1000

Table continues on the next page...

Table 58. MM06 clock configuration (continued)

Clock name	Frequency (MHz)
AXI	400
AHB	133
CPU	1800
DDRC	1000

6.7.2.2 PLL configuration

The PLL configuration in Table 59 is aligned with release L5.4.24.

Table 59. MM06 PLL configuration

Clock root	Source selected	Frequency (MHz)
ARM_A53_CLK	PLL_ARM_MAIN_CLK	1800
NOC_CLK	SYS_PLL2_1000M	1000
MAIN_AXI_CLK	SYS_PLL1_800M	400
MEDIA_AXI_CLK	SYS_PLL2_1000M	500
ENET_AXI_CLK	SYS_PLL1_266M	266
NAND_USDHC_BUS_CLK	SYS_PLL1_266M	266
AHB_CLK_ROOT	SYS_PLL1_133M	133
IPG_CLK	AHB_ROOT_CLK	67
DRAM_CLK	PLL_DRAM_MAIN_CLK	1000
DISP_PIXEL_CLK	VIDEO_PLL_CLK	--
SAIx_CLK	24M OSC	24
ENETx_REF_CLK	SYS2_PLL_125M	125
ENETx_TIME_CLK	SYS2_PLL_100M	100
ENET_PHY_REF_CLK	24M OSC	24
NAND_CLK	SYS1_PLL_400M	off
QSPI_CLK	SYS1_PLL_800M	off
USHDCx_CLK	SYS1_PLL_400M	400

Table continues on the next page...

Table 59. MM06 PLL configuration (continued)

Clock root	Source selected	Frequency (MHz)
I2Cx_CLK	24M OSC	24
UARTx_CLK	24M OSC	24
ECSPIx_CLK	24M OSC	off
PWMx_CLK	24M OSC	24
GPTx_CLK	24M OSC	24
TRACE_CLK	24M OSC	off
WDOG_CLK	24M OSC	off

6.7.3 GPU_Kanzi

1. Run `setup_video.sh` , see Important commands.
2. Run `gpu_kanzi.sh` :

```
kanzi_dir=`pwd`"/Kanzi/KPA_1_0_1_137/linux-aarch64"
cd $kanzi_dir
./kanzi.sh
Where kanzi.sh:
export LD_LIBRARY_PATH="$PWD"
while true;do
./kpa.exe
done
```

3. Start power measurement and record the result.

6.7.3.1 Clock configuration

The clock configuration in Table 60 is aligned with release L5.4.24.

Table 60. GPU_Kanzi clock configuration

Clock name	Frequency (MHz)
NOC	1000
AXI	400
AHB	133
CPU	1800
DDRC	1000

6.7.3.2 PLL configuration

The PLL configuration in Table 61 is aligned with release L5.4.24.

Table 61. GPU_Kanzi PLL configuration

Clock root	Source selected	Frequency (MHz)
ARM_A53_CLK	PLL_ARM_MAIN_CLK	1800
NOC_CLK	SYS_PLL2_1000M	1000
MAIN_AXI_CLK	SYS_PLL1_800M	400
MEDIA_AXI_CLK	SYS2_PLL2_1000M	500
ENET_AXI_CLK	SYS_PLL1_266M	266
NAND_USDHC_BUS_CLK	SYS_PLL1_266M	266
AHB_CLK_ROOT	SYS_PLL1_133M	133
IPG_CLK	AHB_ROOT_CLK	67
DRAM_CLK	PLL_DRAM_MAIN_CLK	1000
DISP_PIXEL_CLK	VIDEO_PLL_CLK	--
SAIx_CLK	24M OSC	24
ENETx_REF_CLK	SYS2_PLL_125M	125
ENETx_TIME_CLK	SYS2_PLL_100M	100
ENET_PHY_REF_CLK	24M OSC	24
NAND_CLK	SYS1_PLL_400M	off
QSPI_CLK	SYS1_PLL_800M	off
USHDCx_CLK	SYS1_PLL_400M	400
I2Cx_CLK	24M OSC	24
UARTx_CLK	24M OSC	24
ECSPIx_CLK	24M OSC	off
PWMx_CLK	24M OSC	24
GPTx_CLK	24M OSC	24
TRACE_CLK	24M OSC	off
WDOG_CLK	24M OSC	off

6.7.4 GPU_GLmark

1. Run `setup_video.sh` , see Important commands.

2. Run `gpu_glmark.sh`:

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

3. Start power measurement and record the result.

6.7.4.1 Clock configuration

The clock configuration in Table 62 is aligned with release L5.4.24.

Table 62. GPU_GLmark clock configuration

Clock name	Frequency (MHz)
NOC	1000
AXI	400
AHB	133
CPU	1800
DDRC	1000

6.7.4.2 PLL configuration

The PLL configuration in Table 63 is aligned with release L5.4.24.

Table 63. GPU_GLmark PLL configuration

Clock root	Source selected	Frequency (MHz)
ARM_A53_CLK	PLL_ARM_MAIN_CLK	1800
NOC_CLK	SYS_PLL2_1000M	1000
MAIN_AXI_CLK	SYS_PLL1_800M	400
MEDIA_AXI_CLK	SYS_PLL2_1000M	off
ENET_AXI_CLK	SYS_PLL1_266M	266
NAND_USDHC_BUS_CLK	SYS_PLL1_266M	266
AHB_CLK_ROOT	SYS_PLL1_133M	133
IPG_CLK	AHB_ROOT_CLK	67
DRAM_CLK	PLL_DRAM_MAIN_CLK	1000
DISP_PIXEL_CLK	VIDEO_PLL_CLK	--
SAIx_CLK	24M OSC	24

Table continues on the next page...

Table 63. GPU_GLmark PLL configuration (continued)

Clock root	Source selected	Frequency (MHz)
ENETx_REF_CLK	SYS2_PLL_125M	125
ENETx_TIME_CLK	SYS2_PLL_100M	100
ENET_PHY_REF_CLK	24M OSC	24
NAND_CLK	SYS1_PLL_400M	off
QSPI_CLK	SYS1_PLL_800M	off
USHDCx_CLK	SYS1_PLL_400M	400
I2Cx_CLK	24M OSC	24
UARTx_CLK	24M OSC	24
ECSPIx_CLK	24M OSC	off
PWMx_CLK	24M OSC	24
GPTx_CLK	24M OSC	24
TRACE_CLK	24M OSC	off
WDOG_CLK	24M OSC	off

6.8 Heavy load use cases

Four use cases were used for power measurements. A 1080p TV display was connected to the MIPI-DSI port. Before running any benchmark, the governor must be set to **performance** (see Important commands):

```
cpufreq-set -g performance
```

After setting the governor, run the respective use case in a loop and start power measurements at the desired time interval (recommended is 1 minute) according to section Important commands.

6.8.1 VPU

A 1080p TV display was connected to the MIPI-DSI port. Before running any benchmark, the governor must be set to **performance** (see Important commands):

```
cpufreq-set -g performance
```

After setting the governor, run the respective use case in a loop and start power measurements logging at the desired time interval (recommended is 1 minute) according to section Important commands.

1. Run `setup_video.sh` , see Important commands.
2. Run `vpu_g2dec.sh` :

```
EXT_LOOP=10000
FILE=HEVC_1920x1080_29.97fps_AACLC_44.1Khz_2ch.mkv
export LD_LIBRARY_PATH=/unit_tests/VPU/hantro/:$LD_LIBRARY_PATH
while true;do
cnt=1
```

```
while [ $cnt -le $EXT_LOOP ]
do
/unit_tests/VPU/hantro/g2dec -P -Ers -ibs -N200 -X -b $FILE
cnt=$(($cnt+1));
done
done
```

3. Start power measurement and record data.

6.8.2 4-core Dhryst + MM07 + VPU

A 1080p TV display was connected to the MIPI-DSI port. Before running any benchmark, the governor must be set to **performance** (see Important commands):

```
cpufreq-set -g performance
```

After setting the governor, run the respective use case in a loop and start power measurements logging at the desired time interval (recommended is 1 minute) according to section Important commands.

1. Run *setup_video.sh* , see Important commands.
2. Start 4 Dhrystone, each bind on separate CPU:

```
while [ "1" == "1" ]
do
sudo taskset -c 0 ./Dhrystone/gcc_dry2 &
sudo taskset -c 1 ./Dhrystone/gcc_dry2 &
sudo taskset -c 2 ./Dhrystone/gcc_dry2 &
sudo taskset -c 3 ./Dhrystone/gcc_dry2
done
```

3. Start Taiji use case in a loop, see GPU_MM07.
4. Start power measurement and record data.

6.8.3 4-core Memtest + MM07 + VPU

A 1080p TV display was connected to the MIPI-DSI port. Before running any benchmark, the governor must be set to **performance** (see Important commands):

```
cpufreq-set -g performance
```

After setting the governor, run the respective use case in a loop and start power measurements logging at the desired time interval (recommended is 1 minute) according to section Important commands.

1. Run `setup_video.sh` (see Important commands) .
2. Start 4 memtesters, each bind on separate CPU:

```
while [ "1" == "1" ]
do
sudo taskset -c 0 memtester 200M &
sudo taskset -c 1 memtester 200M &
sudo taskset -c 2 memtester 200M &
sudo taskset -c 3 memtester 200M
done
```

3. Start Taiji use case in a loop, see GPU_MM07.
4. Start power measurement and record data.

6.8.4 4-core Stream + MM07 + VPU

A 1080p TV display was connected to the MIPI-DSI port. Before running any benchmark, the governor must be set to **performance** (see Important commands):

```
cpufreq-set -g performance
```

After setting the governor, run the respective use case in a loop and start power measurements logging at the desired time interval (recommended is 1 minute) according to section see Important commands.

1. Run `setup_video.sh` , see Important commands.
2. Start 4 streams, each bind on separate CPU:

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

3. Start Taiji use case in a loop, see GPU_MM07.
4. Start power measurement and record data.

6.8.5 Coremark + Kanzi

A 1080p TV display was connected to the MIPI-DSI port. Before running any benchmark, the governor must be set to **performance**, see Important commands:

```
cpufreq-set -g performance
```

After setting the governor, run the respective use case in a loop and start power measurements logging at the desired time interval (recommended is 1 minute) according to section Important commands.

1. Run `setup_video.sh` , see Important commands.
2. Start Coremark use case in a loop, see Coremark.
3. Start GPU_Kanzi use case, see GPU_Kanzi.
4. Start power measurement and record data.

6.9 Memory

Three use cases were used for power measurements. Before running any benchmark, the governor must be set to **performance** and the display must be turned off (see Important commands):

```
cpufreq-set -g performance
echo 1 > /sys/class/graphics/fb0/blank
```

NOTE

No display was connected to the platform.

After setting the governor, run the respective use case in a loop and start power measurements logging at the desired time interval (recommended is 1 minute) according to section see Important commands.

6.9.1 Memset

1. Run `setup.sh` , see Important commands.

2. Run `memset_loop.sh` :

```
while true; do
perf bench -f simple mem memset -l10000 -s 1024MB
done
```

3. Start power measurement and record data.

6.9.2 Memcpy

1. Run `setup.sh` , see Important commands.
2. Run `memcpy_loop.sh` :

```
while true; do
perf bench -f simple mem memcpy -l10000 -s 1024MB
done
```

3. Start power measurement and record data.

6.9.3 Stream

Make sure stream libraries are added to `LD_LIBRARY_PATH`

1. Run `setup.sh` , see Important commands.
2. Run `streamcpy_loop.sh` :

```
export LD_LIBRARY_PATH=`pwd`:$LD_LIBRARY_PATH
while true; do
./stream
done
```

3. Start power measurement and record data.

6.10 Storage – SD3.0 card

An SD card was used to run the benchmarks.

6.10.1 DD_RD_SDCARD

1. Run `setup.sh` , see Important commands.
2. Copy `dd_read_SD10.sh` on the SD card partition and run it as shown below.
3. Start power measurement and record data.

```
#!/bin/bash
# Since we're dealing with dd, abort if any errors occur
set -e
TEST_FILE=${1:-dd_ibs_testfile}
#if [ -e "$TEST_FILE" ]; then TEST_FILE_EXISTS=$?; fi
#TEST_FILE_SIZE=3221225472
#134217728
# Exit if file exists
#if [ -e $TEST_FILE ]; then
#  echo "Test file $TEST_FILE exists, aborting."
#  exit 1
#fi
#TEST_FILE_EXISTS=1
if [ $EUID -ne 0 ]; then
```

```
echo "NOTE: Kernel cache will not be cleared between tests without sudo. This will likely cause
inaccurate results." 1>&2
fi
# Create a test file
#echo 'Generating test file...'
#BLOCK_SIZE=65536
#COUNT=$(($TEST_FILE_SIZE / $BLOCK_SIZE))
#dd if=/dev/urandom of=$TEST_FILE bs=$BLOCK_SIZE count=$COUNT conv=fsync > /dev/null 2>&1
# Header
PRINTF_FORMAT="%8s : %s\n"
printf "$PRINTF_FORMAT" 'block size' 'transfer rate'
# Block sizes of 512b 1K 2K 4K 8K 16K 32K 64K 128K 256K 512K 1M 2M 4M 8M 16M 32M 64M
for BLOCK_SIZE in 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152
4194304 8388608 16777216 33554432 67108864
do
# Clear kernel cache to ensure more accurate test
[ $EUID -eq 0 ] && [ -e /proc/sys/vm/drop_caches ] && echo 3 > /proc/sys/vm/drop_caches
# Read test file out to /dev/null with specified block size
DD_RESULT=$(dd if=$TEST_FILE of=/dev/null bs=$BLOCK_SIZE 2>&1 1>/dev/null)
# Extract transfer rate
TRANSFER_RATE=$(echo $DD_RESULT | \grep --only-matching -E '[0-9.]+ ([MGk]?B|bytes)/s(ec)?')
printf "$PRINTF_FORMAT" "$BLOCK_SIZE" "$TRANSFER_RATE"
done
```

6.10.2 DD_WRT_SDCARD

1. Run `setup.sh` , see Important commands.
2. Copy `dd_write_SD10.sh` on the SD card partition and run it as shown below.
3. Start power measurement and record data.

```
#!/bin/bash
# Since we're dealing with dd, abort if any errors occur
set -e
TEST_FILE=${1:-dd_obs_testfile}
TEST_FILE_EXISTS=0
if [ -e "$TEST_FILE" ]; then TEST_FILE_EXISTS=1; fi
TEST_FILE_SIZE=3221225472
#134217728
if [ $EUID -ne 0 ]; then
echo "NOTE: Kernel cache will not be cleared between tests without sudo. This will likely cause
inaccurate results." 1>&2
fi
# Header
PRINTF_FORMAT="%8s : %s\n"
printf "$PRINTF_FORMAT" 'block size' 'transfer rate'
# Block sizes of 512b 1K 2K 4K 8K 16K 32K 64K 128K 256K 512K 1M 2M 4M 8M 16M 32M 64M
for BLOCK_SIZE in 512 1024 2048 4096 8192 16384 32768 65536 131072 262144 524288 1048576 2097152
4194304 8388608 16777216 33554432 67108864
do
# Calculate number of segments required to copy
COUNT=$(($TEST_FILE_SIZE / $BLOCK_SIZE))
if [ $COUNT -le 0 ]; then
echo "Block size of $BLOCK_SIZE estimated to require $COUNT blocks, aborting further tests."
break
fi
# Clear kernel cache to ensure more accurate test
[ $EUID -eq 0 ] && [ -e /proc/sys/vm/drop_caches ] && echo 3 > /proc/sys/vm/drop_caches
# Create a test file with the specified block size
```

```
DD_RESULT=$(dd if=/dev/zero of=$TEST_FILE bs=$BLOCK_SIZE count=$COUNT conv=fsync 2>&1 1>/dev/
null)
# Extract the transfer rate from dd's STDERR output
TRANSFER_RATE=$(echo $DD_RESULT | \grep --only-matching -E '[0-9.]+ ([MGk]?B|bytes)/s(ec)?')
## Clean up the test file if we created one
#if [ $TEST_FILE_EXISTS -ne 0 ]; then rm $TEST_FILE; fi
# Output the result
printf "$PRINTF_FORMAT" "$BLOCK_SIZE" "$TRANSFER_RATE"
done
```

6.11 Storage – eMMC

Created a partition on eMMC and run the benchmarks on it.

6.11.1 DD_RD_eMMC

1. Run `setup.sh` , see Important commands.
2. Copy `dd_read_SD10.sh` on eMMC partition and run.
3. Start power measurement and record data.

6.11.2 DD_WRT_eMMC

1. Run `setup.sh` , see Important commands.
2. Copy `dd_write_SD10.sh` on eMMC partition and run.
3. Start power measurement and record data.

6.12 Storage – USB3.0

A USB 3.0 was used to run the benchmarks.

6.12.1 DD_RD_USB3.0

1. Run `setup.sh` , see Important commands.
2. Copy `dd_read_SD10.sh` on USB 3.0 partition and run.
3. Start power measurement and record data.

6.12.2 DD_WRT_USB3.0

1. Run `setup.sh` , see Important commands.
2. Copy `dd_write_SD10.sh` on USB 3.0 partition and run.
3. Start power measurement and record data.

6.13 Product use cases

Following is the test setup:

1. Linux host machine connected to local network
2. i.MX 8M Plus connected to the local network
3. MIPI CSI camera
4. LVDS display
5. MIPI DSI display

6. HDMI display

NOTE

Change the Host and target IP addresses into scripts as required. Also, change the media file name/locations into the script before running the script. Target and host are only mentioned for gstreamer commands, rest all commands are only for target. The following sub-sections provides an example for copying the caps filter, which can be used for many use cases.

6.13.1 Fleet management, driver monitoring systems, dual camera, stereo vision, and, vacuum robot

- Stress A53 core to simulate application stress with Coremark:

  ```
  /run/media/mmcblk1p3/coremark_cpu_limit.sh 50
  ```

  For detail, see Important commands.

- Run UI example from GPU tutorials:

  ```
  /usr/bin/weston-simple-egl
  ```

- Run 2 instances of object detection from the file based input:

  ```
  /usr/bin/tensorflow-lite-2.3.1/examples/ML.sh
  ```

  For detail, see Important commands.

- Run dual camera, one is being previewed and other camera feed is encoded to HEVC and written to USB disk:

  ```
  gst-launch-1.0 -v v4l2src device=<device 0> ! waylandsink &
  gst-launch-1.0 -v v4l2src device=<device 1> do-
  timestamp=true ! "video/x-raw,width=3840,height=2160" ! imxvideoconvert_g2d ! 'video/x-
  raw,width=1920,height=1080,framerate=30/1' ! vpuenc_hevc qos=false ! h265parse ! qtmux ! filesink
  location=<save_to_file_on_USB> -e
  ```

6.13.2 Video conferencing

- Stress A53 core to simulate application stress with Coremark:

  ```
  /run/media/mmcblk1p3/coremark_cpu_limit.sh 50
  ```

  For detail, see Important commands.

- On host, send H264 stream to target:

  ```
  gst-launch-1.0 -v filesrc location=<h264_stream_file> ! qtdemux ! video/x-h264 ! rtph264pay
  pt=96 ! udpsink host=<target_ip_addr> port=<portno>
  ```

- Camera 1080p60 w/ ISP preview, H264 encode of camera feed and stream over Ethernet, Receive H264 encoded stream over Ethernet, decode and display:

  ```
  gst-launch-1.0 -v imxcompositor_g2d name=c sink_1::xpos=0 sink_1::ypos=0 sink_1::width=1920
  sink_1::height=1080 sink_1::keep-ratio=true sink_0::xpos=1600 sink_0::ypos=740 sink_0::width=320
  sink_0::height=240 sink_0::keep-ratio=true ! waylandsink udpsrc port=<port_no> caps="" !
  rtpjitterbuffer latency=1000 ! rtph264depay ! vpudec ! c.sink_1 v4l2src device=<Device
  1> do-timestamp=true ! "video/x-raw,width=3840,height=2160" ! imxvideoconvert_g2d ! 'video/x-
  raw,width=1920,height=1080,framerate=30/1' ! vpuenc_h264 qos=false ! rtph264pay ! udpsink host=<>
  port=<>
  ```

- On host receive H264 stream and display:

```
gst-launch-1.0 -v udpsrc port=<portno> caps="" ! rtpjitterbuffer latency=1000 ! rtph264depay !
avdec_h264 ! autovideosink
```

6.13.3 Surveillance - headless (2xISP)

- Stress A53 core to simulate application stress with Coremark:

  `/run/media/mmcblk1p3/coremark_cpu_limit.sh 50`

  For detail, see Important commands.

- Run 2 instances of object detection from the file based input:

  `/usr/bin/tensorflow-lite-2.3.1/examples/ML.sh`

  For detail, see Important commands.

- On target, run Dual camera 1080p60 w/ISP capture, HEVC encode one of the camera feed and stream over network:

```
gst-launch-1.0 -v v4l2src device=<device0> ! fakesink &
gst-launch-1.0 -v v4l2src device=<device1> do-
timestamp=true ! "video/x-raw,width=3840,height=2160" ! imxvideoconvert_g2d ! 'video/x-
raw,width=1920,height=1080,framerate=30/1' ! vpuenc_hevc qos=false ! rtph265pay ! udpsink
host=<ipaddr> port=<portno>
```

- On host, receive camera feed and display:

  `gst-launch-1.0 -v udpsrc port=<portno> caps="" ! rtpjitterbuffer latency=1000 ! rtph265depay ! avdec_h265 ! autovideosink`

6.13.4 HMI

- Stress A53 core to simulate application stress with Coremark:

  `/run/media/mmcblk1p3/coremark_cpu_limit.sh 50`

  For detail, see Important commands.

- On host, send h264 stream to target:

  `gst-launch-1.0 -v filesrc location=<h264stream> ! qtdemux ! video/x-h264 ! rtph264pay pt=96 ! udpsink host=<target_ip_addr> port=<portno>`

- Move the cursor to MIPI screen.
- On target, receive 1x1080p60 H264 stream, decode and display:

  `gst-launch-1.0 udpsrc port=<portno> caps="" ! rtpjitterbuffer latency=1000 ! rtph264depay ! vpudec ! waylandsink`

- Move the cursor to LVDS screen.
- Run UI example from GPU tutorials:

  `/usr/bin/weston-simple-egl`

6.13.5 Machine vision (1xISP)

- Stress A53 core to simulate application stress with Coremark:

  `/run/media/mmcblk1p3/coremark_cpu_limit.sh 50`

  For detail, see Important commands.

- Run object detection from the file based input:

```
/usr/bin/tensorflow-lite-2.3.1/examples/ML.sh 1
```

- Run camera capture 4Kp30, downscale to 1080p30, HEVC encode camera feed and stream:

```
gst-launch-1.0 -v v4l2src device=<device1> do-timestamp=true ! "video/x-raw,width=3840,height=2160" !
imxvideoconvert_g2d ! 'video/x-raw,width=1920,height=1080,framerate=30/1' ! vpuenc_hevc qos=false !
rtph265pay ! udpsink host=<host_ip_addr> port=<portno>
```

- On host, receive stream and display:

```
gst-launch-1.0 -v udpsrc port=<portno> caps="" ! rtpjitterbuffer latency=1000 ! rtph265depay !
avdec_h265 ! autovideosink
```

6.13.6 eIQ - Benchmarking tests

- Run object detection from the file based input:

```
/usr/bin/tensorflow-lite-2.3.1/examples/ML.sh 1
```

6.13.7 6.13.7. M7 Coremark

NOTE

Flash an image built with Coremark

- Setup Coremark on Uboot with following commands:

```
setenv m7_boot 'fatload mmc 1:1 0x48000000 ${m7_image};cp.b 0x48000000 0x7e0000
20000;bootaux 0x7e0000;'
setenv m7_image coremark.bin
saveenv
```

- Start coremark in uboot:

```
run m7_boot
```

6.13.8 Display stress test

- Move cursor to HDMI, run 720p60 HEVC video, decode and display:

```
gst-launch-1.0 filesrc location=<stream_path> ! qtdemux ! vpudec ! waylandsink
```

- Move cursor to MIPI-DSI, run 1080p60 ISP camera capture and display:

```
gst-launch-1.0 v4l2src device=<device0> ! waylandsink
```

- Move cursor to LVDS, run GPU tutorial examples:

```
/usr/bin/weston-simple-egl
```

Example for copying caps filter:

Here is the example for stream sender command:

```
gst-launch-1.0 -v filesrc location=media/bbb_sunflower_1080p_30fps_normal.mp4 ! qtdemux ! video/x-
h264 ! rtph264pay pt=96 ! udpsink host=192.168.1.72 port=8654
Setting pipeline to PAUSED ...
Pipeline is PREROLLING ...
/GstPipeline:pipeline0/GstCapsFilter:capsfilter1: caps = video/x-h264
/GstPipeline:pipeline0/GstCapsFilter:capsfilter1.GstPad:src: caps = video/x-h264, stream-
format=(string)avc, alignment=(string)au, level=(string)4.1, profile=(string)high,
codec_data=(buffer)01640029ffe1001b67640029acca501e0089f97011000003000100000300 3c8f18319601000568e93b2
c8b, width=(int)1920, height=(int)1080, framerate=(fraction)30/1, pixel-aspect-ratio=(fraction)1/1
/GstPipeline:pipeline0/GstRtpH264Pay:rtph264pay0.GstPad:src: caps = application/x-rtp,
media=(string)video, clock-rate=(int)90000, encoding-name=(string)H264, packetization-mode=(string)1,
```

```
profile-level-id=(string)640029, sprop-parameter-
sets=(string)"Z2QAKazKUB4AiflwEQAAAwABAAADADyPGDGW\,aOk7LIs\=", payload=(int)96, ssrc=(uint)39081554,
timestamp-offset=(uint)2447567668, seqnum-offset=(uint)9569, a-framerate=(string)30
/GstPipeline:pipeline0/GstUDPSink:udpsink0.GstPad:sink: caps = application/x-rtp, media=(string)video,
clock-rate=(int)90000, encoding-name=(string)H264, packetization-mode=(string)1, profile-level-
id=(string)640029, sprop-parameter-sets=(string)"Z2QAKazKUB4AiflwEQAAAwABAAADADyPGDGW\,aOk7LIs\=",
payload=(int)96, ssrc=(uint)39081554, timestamp-offset=(uint)2447567668, seqnum-offset=(uint)9569, a-
framerate=(string)30
```

Here the example for receiver command after copying caps filter:

```
gst-launch-1.0 udpsrc port=8654 caps="application/x-rtp, media=(string)video, clock-
rate=(int)90000, encoding-name=(string)H264, packetization-mode=(string)1, profile-level-
id=(string)640029, sprop-parameter-sets=(string)\"Z2QAKazKUB4AiflwEQAAAwABAAADADyPGDGW\,aOk7LIs\=\",
payload=(int)96, ssrc=(uint)39081554, timestamp-offset=(uint)2447567668, seqnum-offset=(uint)9569, a-
framerate=(string)30"  ! rtpjitterbuffer latency=1000 ! rtph264depay ! vpudec ! waylandsink
```

6.13.9 Display stress test

Note: Change the media file name/location into the scripts before running the script.

1. Run `Displaytest_HDMI.sh` over HDMI.
2. Run `Displaytest_MIPIDSI.sh` over MIPI DSI.
3. Run `/usr/bin/Weston-simple-egl` over LVDS display.

6.14 Important commands

1. Before running a use case, `<configuration_script>.sh` script must be run to configure the environment. These scripts are; *setup.sh, setup_default.sh, setup_video.sh, setup_video_stream.sh, DRC_266MHz_setup.sh, DRC_25MHz_setup.sh*. See the details below.

   - `setup.sh`: The CPU frequency is set to the maximum value 1800 MHz to achieve the best performance. Disable the Ethernet, stop the Weston service, and blank the display. Set the maximum amount of data that the kernel reads ahead for a single file to 512 kB.

     ```
     #!/bin/bash
     systemctl stop weston.service
     cpufreq-set -g performance
     echo 1 > /sys/class/graphics/fb0/blank
     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'|awk {'print $1'}`
     for eth in $eth_int;do
     ifconfig $eth down
     done
     ```

   - `setup_default.sh`: CPU frequency governor is in ondemand mode (default mode). CPU frequency will change between Cortex®-A53's supported operating points and will be set based on current system load. Disable the Ethernet, stop the Weston service, and blank the display. Set the maximum amount of data that the kernel reads ahead for a single file to 512 kB.

     ```
     #!/bin/bash
     systemctl stop weston.service
     echo 1 > /sys/class/graphics/fb0/blank
     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'|awk {'print $1'}`
for eth in $eth_int;do
ifconfig $eth down
done
```

- `setup_video.sh`: The CPU frequency is set to the maximum value 1800 MHz to achieve the best performance. Disable the Ethernet, stop the Weston service, and awake the display. Set the maximum amount of data that the kernel reads ahead for a single file to 512 kB.

```
#!/bin/bash
export WL_EGL_SWAP_INTERVAL=0
cpufreq-set -g performance
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'|awk {'print $1'}`
for eth in $eth_int;do
ifconfig $eth down
done
echo 1 > /sys/class/graphics/fb0/blank
echo 0 > /sys/class/graphics/fb0/blank
```

- `setup_video_stream.sh`: The CPU frequency is set to the maximum value 1800 MHz to achieve the best performance. Open the Ethernet to play the video online. Stop the Weston service and awake the display. Set the maximum amount of data that the kernel reads ahead for a single file to 512 kB.

```
#!/bin/bash
export WL_EGL_SWAP_INTERVAL=0
cpufreq-set -g performance
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'|awk {'print $1'}`
for eth in $eth_int;do
ifconfig $eth up
done
echo 1 > /sys/class/graphics/fb0/blank
echo 0 > /sys/class/graphics/fb0/blank
```

- `DDRC_25MHz_setup.sh`: After running below shell scripts, you will see logs that the DDR frequency switches between high bus mode 2000 MHz and low bus mode 50 MHz, due to DDR DVFS. CPU frequency is set to the minimum value 1200 MHz. Both of DDR DVFS and CPU powersave are aimed at saving power. Disable the Ethernet, stop the Weston service, and blank the display.)

```
#!/bin/bash
systemctl stop weston@root.service
echo 1 > /sys/class/graphics/fb0/blank
cpufreq-set -g powersave
rmmod qca9377
#echo 8 > /proc/sys/kernel/printk;
eth_int=`ifconfig -a|grep 'eth\|can\|sit'|awk {'print $1'}`
for eth in $eth_int;do
            ifconfig $eth down
done
echo 1 > /sys/devices/platform/busfreq/enable
```

```
echo auto > /sys/bus/platform/devices/32f10100.usb/38100000.dwc3/power/control
echo auto > /sys/bus/platform/devices/32f10108.usb/38200000.dwc3/power/control
echo auto > /sys/bus/platform/devices/32f10108.usb/38200000.dwc3/xhci-hcd.1.auto/power/
control
sleep 5
```

- `coremark_cpu_limit.sh`: This script is used for running product use cases in Product use cases. The script starts Cormark application to stress A53:

```
#!/bin/bash
while true; do
./coremark.exe &
pid=$!
taskset -cp 3 $pid
./cpulimit -l $1 -p $pid
done
```

- `ML.sh`: This script is used for running product use cases in Product use cases. The script starts machine learning example:

```
#!/bin/bash
rep=$1
cd /usr/bin/tensorflow-lite-2.3.1/examples
for i in $( eval echo {1..$rep} )
do
        /usr/bin/tensorflow-lite-2.3.1/examples/benchmark_model --
graph=mobilenet_v1_1.0_224_quant.tflite --num_runs=22000 --use_nnapi=true &
done
```

2. In the U-Boot console:
   - `printenv`: Displays the environment variables.
   - `setenv`: Updates the environment variables.
     - `setenv <name> <value> ...`
     - Sets the environment variable “name” to “value ...”.
     - `setenv <name>`
     - Deletes the environment variable “name”.
   - `saveenv`: Saves the updates to the environment variables.
   - `bootargs`: Passes to the kernel, which are called kernel command lines.
3. In the Linux OS console:
   - `cat /proc/cmdline`: Displays the command line.
   - `cat /sys/devices/virtual/thermal/thermal_zone0/temp`: Prints the temperature to the screen (the chip should be calibrated).

     NOTE

     The die temperature value was logged (written) externally, not on the SD card, for not impacting power consumption.

   - `cat /sys/kernel/debug/clk/clk_summary`: Prints all clocks to the screen.
4. Miscellaneous:

Script to get the current CPU usage:

```
#!/bin/bash

by Paul Colby (http://colby.id.au), no rights reserved
PREV_TOTAL=0
PREV_IDLE=0
```

While true; do:

a. Get the total CPU statistics, discarding the 'cpu ' prefix:

```
CPU=(`sed -n 's/^cpu\s//p' /proc/stat`)
IDLE=${CPU[3]} # Just the idle CPU time.
```

b. Calculate the total CPU time:

```
TOTAL=0
for VALUE in "${CPU[@]}"; do
let "TOTAL=$TOTAL+$VALUE"
done
```

c. 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"
```

d. Remember the total and idle CPU times for the next check:

```
PREV_TOTAL="$TOTAL"
PREV_IDLE="$IDLE"
```

e. Wait before checking again:

```
sleep 1
done
```

Below steps to determine DDR usage:

```
perf stat -I 1000 -a -M imx8mp-ddr0-all-r,imx8mp-ddr0-all-w &> ddr_utilization_$1_raw.txt &
pid=$!
sleep $2
kill 9 $pid
python /run/media/mmcblk1p3/8MP-complex-use-cases/8MP-complex-use-cases/ddr_8mp.py
ddr_utilization_$1_raw.txt >> stats_$1.txt

Script for ddr_8mp.py

#!/usr/bin/python

import os
import sys

def nonblank_lines(f):
for l in f:
line = l.rstrip()
if line:
```

```
yield line

temp= 0.0

i2 = 0
rail2 = 0.0
i3 = 0
rail3 = 0.0

rlist2 = list()
rlist3 = list()

with open(sys.argv[1], 'r') as f:
for line in nonblank_lines(f):
linedata = line.split()
print linedata

if len(linedata) >= 3:
if linedata[2] == "imx8_ddr0/axid-read,axi_mask=0xffff,axi_id=0x0000/" :
temp = int(linedata[1])
rlist2.append(temp);

if linedata[2] == "imx8_ddr0/axid-write,axi_mask=0xffff,axi_id=0x0000/" :
temp = int(linedata[1])
rlist3.append(temp);

print "\n"

print "\n"
print "ddr0/read-cycles/ = "
print "Max =", max(rlist2), " Min =", min(rlist2), " Avg =", sum(rlist2)/len(rlist2),"Count = ",
len(rlist2)
print "\n"

print "ddr0/write-cycles/ = "
print "Max =", max(rlist3), " Min =", min(rlist3), " Avg =", sum(rlist3)/len(rlist3),"Count = ",
len(rlist3)
print "\n"

print "% DDR use =", (float)((sum(rlist2)/len(rlist2) + sum(rlist3)/len(rlist3))*100)/
(16000000000)
print "\n"
```

7 Revision history

Table 64 summarizes the changes done to this document since the initial release.

Table 64. Revision history

Revision number	Date	Substantive changes
0	04/2021	Initial release

How To Reach Us

Home Page:

nxp.com

Web Support:

nxp.com/support

Limited warranty and liability — Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including “typicals,” must be validated for each customer application by customer's technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.

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.

Security — Customer understands that all NXP products may be subject to unidentified or documented vulnerabilities. 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, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, COOLFLUX,EMBRACE, GREENCHIP, HITAG, ICODE, JCOP, LIFE, VIBES, MIFARE, MIFARE CLASSIC, MIFARE DESFire, MIFARE PLUS, MIFARE FLEX, MANTIS, MIFARE ULTRALIGHT, MIFARE4MOBILE, MIGLO, NTAG, ROADLINK, SMARTLX, SMARTMX, STARPLUG, TOPFET, TRENCHMOS, UCODE, Freescale, the Freescale logo, AltiVec, CodeWarrior, ColdFire, ColdFire+, the Energy Efficient Solutions logo, Kinetis, Layerscape, MagniV, mobileGT, PEG, PowerQUICC, Processor Expert, QorIQ, QorIQ Qonverge, SafeAssure, the SafeAssure logo, StarCore, Symphony, VortiQa, Vybrid, Airfast, BeeKit, BeeStack, CoreNet, Flexis, MXC, Platform in a Package, QUICC Engine, Tower, TurboLink, EdgeScale, EdgeLock, eIQ, and Immersive3D are trademarks of NXP B.V. All other product or service names are the property of their respective owners. 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 or registered trademarks of Arm Limited (or its subsidiaries) 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. Oracle and Java are registered trademarks of Oracle and/or its affiliates. The Power Architecture and Power.org word marks and the Power and Power.org logos and related marks are trademarks and service marks licensed by Power.org.

© NXP B.V. 2021. **All rights reserved.**

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 04/2021
Document identifier: AN13054

arm