System Controller Firmware Release Notes

NXP

Fri Dec 20 2019 06:56:10

Chapter 1

Introduction

This document contains release notes for the i.MX8 System Controller Firmware (SCFW). This includes:

- Highlights
- Changes from the previous release
- Known issues
- Additional info

The table below lists the release information:

Release Info	
Release name	imx_4.14.98_ga2
Previous release	imx_4.14.98_ga
Branch	imx_4.14.98_2.3.0
Build number	4112
Commit ID	5a35e406
Build date	Dec 20 2019
API Version	1.16
Supported devices	i.MX8QM (B0), i.MX8QXP (B0), i.MX8QXP (C0), and phantoms

1.1 Highlights

- Add support for i.MX8QXP C0
- Support PF8X00 C1 PMIC revision
- Support SECO FW version 2.5.4
- Support new i.MX8DXL phantom board
- Added spread spectrum support

- Implement LPDDR4 DQS2DQ training
- Support security violation and tamper detect
- Removed support for i.MX8QM A0

Many changes in this release were also delivered in patch releases to previous releases.

Note SECO firmware version 2.5.4 or higher is **REQUIRED** for correct operation!

Chapter 2

Change List

2.1 4.14.98 GA2 Change List

Below is a list of changes between the previous release (imx_4.14.98_ga) and this release (imx_4.14.98_ga2).

2.1.1 New Feature

Key	Summary	Patch	QM (B0)	QXP (B0)	QXP (C0)
SCF-186	Add support for LVDS PLL spread spectrum [detail]	p3	Y	Y	Y
SCF-227	Support i.MX8QXP C0 [detail]				Y
SCF-362	Add support for i.MX8DXL phantom board [detail]	p4		Y	Y
SCF-386	Add API call to allow starting of RNG [detail]		Y	Y	Y
SCF-392	Support PF8X00 C1 PMIC revisions	p1	Y	Y	Y
SCF-394	Add watchdog functions to the PMIC driver [detail]		Y	Y	Y
SCF-396	Enable DDR CBT by default on reference boards		Y	Y	Y
SCF-415	Add support for additional DSP memory access controls [detail]		Y	Y	Y
SCF-422	Support SECO FW v2.4.1 [detail]		Y	Y	Y
SCF-432	Add API for software wakeup request [detail]		Y	Y	Y
SCF-447	Add API functions to access SNVS security violation and tamper registers [detail]		Y	Y	Y
SCF-456	Add voltage detect process monitor driver		Y	Y	Y
SCF-474	Add header to provide GPIO driver compatibility [detail]		Y	Y	Y
SCF-492	Add mechanism for SCFW client to reset IPC/RPC protocol [detail]		Y	Y	Y
SCF-493	Add support for CAN doze mode [detail]		Y	Y	Y
SCF-494	Add API to allow reset of a peripheral [detail]		Y	Y	Y
SCF-498	Support SECO FW v2.5.3 [detail]		Y	Y	Y
SCF-505	Add support for partition watchdog window [detail]		Y	Y	Y

2.1.2 Improvement

Key	Summary	Patch	QM (B0)	QXP (B0)	QXP (C0)
SCF-268	Adjust M4 bus clock when M4 CPU clock freq changes [detail]		Y	Y	Y
SCF-271	Update STC QoS configuration for i.MX8QXP			Y	Y
SCF-342	Add support for FlexSPI controller save/restore during LPM transition		Y	Y	Y
SCF-343	Disallow SC MU resource used in SCFW API to enter STBY/OFF [detail]		Y	Y	Y
SCF-352	Test improvements		Y	Y	Y
SCF-365	Display PMIC versions with monitor info command		Y	Y	Y
SCF-385	Update M4 VLPR implementation to use PM service APIs		Y	Y	Y
SCF-395	Clean up SOC code		Y	Y	Y
SCF-397	Improve power management during early boot scenarios		Y	Y	Y
SCF-398	Clean up DSC driver code		Y	Y	Y
SCF-404	Update to latest MCUXpresso SDK 2.0 release [detail]		Y	Y	Y
SCF-406	Optimize boot time		Y	Y	Y
SCF-410	Break up the DC unit test into smaller tests		Y	Y	Y
SCF-419	Enable WDI detection in standby for PF8x00 PMIC		Y	Y	Y
SCF-429	First M4 warm reset occurrence not reported via ASMC for TCM-based image		Y	Y	Y
SCF-434	KS1 exit latency impacted by check of SNVS wake events		Y	Y	Y
SCF-441	Enable low-power retention of IRQSTEER to facilitate IPC remote wakeup		Y	Y	Y
SCF-448	Sanitize internal API names and header files		Y	Y	Y
SCF-460	Block DDR access beyond valid range [detail]		Y	Y	Y
SCF-461	Refactor scripts used to generate RPC		Y	Y	Y
SCF-462	Fix memory region creation for emulation		Y	Y	Y
SCF-466	Enhance VPU RM test		Y	Y	Y
SCF-467	Implement PF8200 secure write functionality in P↩ MIC driver		Y	Y	Y
SCF-473	Update CBT code to align DDR_PHY and DDRC MR13 register with DCU programming		Y	Y	Y
SCF-485	Utilize unallocated TCMU to increase software stack from 2K to 4K		Y	Y	Y
SCF-487	Support partition power off and reboot [detail]		Y	Y	Y
SCF-501	Add RPC servicing fairness (sequential client servicing) [detail]		Y	Y	Y
SCF-509	Add ECC granularity for early CPU boot on QXP C0				Y
SCF-510	Add more info to the debug monitor RM memrg dump		Y	Y	Y
SCF-516	Reset clock rates associated with a resource on power down [detail]		Y	Y	Y

2.1.3 Bug

Key	Summary	Patch	QM (B0)	QXP (B0)	QXP (C0)
SCF-379	Delay configuration of PCIe internal/external clock selection	p1	Y	Y	Y
SCF-381	Memory region alloc does not free region on error	p1	Y	Y	Y
SCF-384	Fix memset() function implementation	p1	Y	Y	Y
SCF-387	Only change slice parent when divider is acceptable	p1	Y	Y	Y
SCF-389	Add completion barrier to I2C writes	p1	Y	Y	Y
SCF-399	DB SSI does not idle correctly		Y	Y	Y
SCF-401	Relock boot device subsystem PLLs only once during boot sequence		Y	Y	Y
SCF-411	Reduce imaging subsystem pixel clock to 400MHz	p4	Y	Y	Y
SCF-416	Memory repair idle check has wrong polarity	p4	Y	Y	Y
SCF-420	Ensure IEEE1588 can work on fused QX (UX) parts)			Y	Y
SCF-423	sc_seco_fuse_write() and sc_seco_patch() pass incorrect address	p4	Y	Y	Y
SCF-425	Some repeated clock/power transitions can result in SCFW crash	p4	Y	Y	Y
SCF-426	Build from porting kit with D=0 leaves on some debug functionality	p4	Y	Y	Y
SCF-427	Can't control eLCDIF clock	p4		Y	Y
SCF-431	CBT does not work in 16-bit configuration for 8DX	p4	Y	Y	Y
SCF-433	Change EVSIM voltage level to 3.0V to conform to spec		Y	Y	Y
SCF-435	Resource usage issue with D=0 and U=1 build options	p10	Y		
SCF-436	Fix LPCG address access in IMG SS		Y	Y	Y
SCF-440	SECO permissions are forced to "secure RW" when monitor is enabled	p10	Y	Y	Y
SCF-443	8QXP VPU power domain is incorrect	p4		Y	Y
SCF-451	Temp sensor alarm interrupts do not clear		Y	Y	Y
SCF-453	Update resource/pad mapping between AP/M4 for DXL phantom board	p6		Y	Y
SCF-457	Ensure the correct parent returned for DC pixel clocks		Y	Y	Y
SCF-459	sc_pm_set_sys_power_mode() allows the system to enter an unsupported state [detail]		Y		
SCF-463	Modify the timing parameters to RAM (A53/A72) based on frequency and voltage	p7	Y		
SCF-471	SCFW fails when GPU power cycled on i.MX8QP	p8	Y		
SCF-489	RTC alarm can't wake up the system from KS0		Y	Y	Y
SCF-496	Critical section violation during boot (D=1 only) [detail]	p10	Y	Y	Y
SCF-502	Latch early wake events for SC_PM_WAKE_SR↩ C_SCU configurations [detail]		Y	Y	Y
SCF-503	Resources disabled in board_rsrc_avail() remain accessible	p10	Y	Y	Y

Key	Summary	Patch	QM (B0)	QXP (B0)	QXP (C0)
SCF-507	Issue with FlexSPI1 access	p10	Y	Y	Y
SCF-512	Disable PI PLL0 by default to allow the rate to be configured [detail]			Y	Y
SCF-514	DRC1 clock rate not retained across enter/exit DDR retention mode	p10	Y		

2.1.4 Silicon Workaround

These are a mix of silicon errata workarounds and recommended usage changes.

Key	Summary	Patch	QM (B0)	QXP (B0)	QXP (C0)
SCF-57	Implement LPDDR4 DQS2DQ training [detail]		Y	Y	
SCF-364	Add support for new reference voltage calibration	p1	Y		
SCF-390	DDR derating logic does not consider the temperature	p1	Y	Y	Y
SCF-393	Add matching function for PMIC OTP	p1	Y		
SCF-403	Support SECO FW v2.3.1	p2	Y	Y	Y
SCF-408	Disable MSI posted writes for all SCU MSI rings	p4	Y	Y	Y
SCF-417	Synchronize DQS2DQ with ISI data frames to avoid bus contention		Y	Y	
SCF-442	Keep REFGEN on during power transitions	p4	Y		
SCF-450	Fix LDO for connectivity and HDMI RX/TX if TRIM_LDO fuse set	p5	Y		
SCF-452	Fix the LVDS PLL spread spectrum saturation issue	p5	Y	Y	Y
SCF-464	Limit PLL div-1 freq to the max supported by vd-detect circuit in SS with HP-PLL		Y	Y	Y
SCF-468	Add board parameter to override ISI pixel clock frequency [detail]	p8	Y	Y	Y
SCF-477	Implement read DBI deskew training work around	p8	Y	Y	Y
SCF-479	DDRIO power reduction in KS1 suspend mode	p8	Y	Y	Y
SCF-480	Update 8QM DDR config to RPA v18 with ODT update	p8	Y	Y	Y
SCF-490	Add support for i.MX8DX KS1 voltage as 0.7v	p8		Y	Y
SCF-504	Drop the DC AXI internal clock to 375MHz for QXP B0 and C0 [detail]			Y	Y
SCF-506	Audio PLL instability	p9	Y	Y	Y
SCF-508	Add option for selecting LVDS PLL SSC frequency spread [detail]	p10	Y	Y	Y
SCF-517	Support SECO FW v2.5.4		Y	Y	Y

2.1.5 Task

Key	Summary	Patch	QM (B0)	QXP (B0)	QXP (C0)
SCF-224	Remove i.MX8QM A0 build option				
SCF-247	Remove i.MX8QM A0 source code				

Key	Summary	Patch	QM (B0)	QXP (B0)	QXP (C0)
SCF-312	Remove dynamic delay required to support SECO FW v1.1.0		Y	Y	Y
SCF-413	Remove SCFW DDR stress unit test		Y	Y	Y
SCF-421	Remove deprecated security functions from the MISC API		Y	Y	Y

2.1.6 Documentation

Key	Summary	Patch	QM (B0)	QXP (B0)	QXP (C0)
SCF-303	Update boot times referenced in SCFW Porting Guide		Y	Y	Y
SCF-391	Improve documentation on reference board ports		Y	Y	Y
SCF-405	Document that sc_pm_get_clock_rate() may return wrong rate in low power states [detail]		Y	Y	Y
SCF-454	Document methods to configure WDOG reset events using board code		Y	Y	Y
SCF-472	Add return possibilities to SECO service API documentation		Y	Y	Y

2.2 Details

This section provides details for select changes.

2.2.1 SCF-57: Implement LPDDR4 DQS2DQ training

Requires customers update board.c to:

- Define board_ddr_period_ms
- Call soc_ddr_dqs2dq_init() from board_init_ddr()
- Call soc_ddr_dqs2dq_periodic() from board_ddr_config()

See reference board.c implementations for NXP boards.

2.2.2 SCF-186: Add support for LVDS PLL spread spectrum

Added new board parameters to enable/disable spread spectrum on a per DC PLL basis:

- BOARD_PARM_DC0_PLL0_SSC - DC0 PLL0 spread spectrum enable
- BOARD_PARM_DC0_PLL1_SSC - DC0 PLL1 spread spectrum enable
- BOARD_PARM_DC1_PLL0_SSC - DC1 PLL0 spread spectrum enable
- BOARD_PARM_DC1_PLL1_SSC - DC1 PLL1 spread spectrum enable

Parameter value returned by board_parameter() in board.c.

2.2.3 SCF-227: Support i.MX8QXP C0

Added new function:

- sc_misc_get_boot_container()

API version updated to 1.6.

2.2.4 SCF-268: Adjust M4 bus clock when M4 CPU clock freq changes

Prior to this change, sc_pm_get_clock_rate() would incorrectly returned the previously set rate of the clock independent of the actual hardware rate of the clock. After this change, it will always return the actual rate of the clock set in hardware.

SCFW does not cache clock rates of resources. The clock rate is calculated each time a call to sc_pm_get_clock_rate() is made. When the resource power state is moved to SC_PM_PW_MODE_LP, its clock source is switched from PLL to XTAL (if its not already sourced from XTAL). Calling sc_pm_get_clock_rate() when the resource is in LP mode, will result in returning the clock rate with XTAL (24MHz) as the clock sourcing the resource. The user should be aware of the power state of the resource when making sc_pm_get_clock_rate() call and make appropriate accommodations as needed.

2.2.5 SCF-343: Disallow SC MU resource used in SCFW API to enter STBY/OFF

A call to sc_pm_set_resource_power_mode() on an MU resource will return SC_ERR_PARM if the MU is the one used to communicate the request. This is because if powered off then the SCFW cannot send back a response.

2.2.6 SCF-362: Add support for i.MX8DXL phantom board

i.MX8DXL Phantom MEK board added to accelerate DXL development with a phantom i.MX8QXP part.

To build use "make qx R=B0 B=dxl_phantom". Use U=2 if you would like to use the UART output.

2.2.7 SCF-386: Add API call to allow starting of RNG

Added new API call:

- sc_seco_start_rng()

2.2.8 SCF-394: Add watchdog functions to the PMIC driver

Added PF8100 driver functions:

- pf8100_pmic_wdog_enable()
- pf8100_pmic_wdog_disable()
- pf8100_pmic_wdog_set_timeout()
- pf8100_pmic_wdog_service()

2.2.9 SCF-404: Update to latest MCUXpresso SDK 2.0 release

The SDK GPIO driver has been replaced with the RGPIO driver. This might require some board.c implementations to change the include, config declaration, and configuration constants. See NXP board.c implementations for an example.

2.2.10 SCF-405: Document that sc_pm_get_clock_rate() may return wrong rate in low power states

This issue adds documentation associated with SCF-268.

Prior to this change, sc_pm_get_clock_rate() would incorrectly returned the previously set rate of the clock independent of the actual hardware rate of the clock. After this change, it will always return the actual rate of the clock set in hardware.

SCFW does not cache clock rates of resources. The clock rate is calculated each time a call to sc_pm_get_clock_rate() is made. When the resource power state is moved to SC_PM_PW_MODE_LP, its clock source is switched from PLL to XTAL (if its not already sourced from XTAL). Calling sc_pm_get_clock_rate() when the resource is in LP mode, will result in returning the clock rate with XTAL (24MHz) as the clock sourcing the resource. The user should be aware of the power state of the resource when making sc_pm_get_clock_rate() call and make appropriate accommodations as needed.

2.2.11 SCF-415: Add support for additional DSP memory access controls

Added two new controls for the DSP resource:

- SC_C_OFS_AUDIO_ALT - System alt address offset of AUDIO
- SC_C_DSP_BYP - DSP to DMA bypass

Updated the API version to 1.8.

2.2.12 SCF-422: Support SECO FW v2.4.1

Added new function to the SECO service:

- sc_seco_sab_msg()

This function sends a generic signed message to the SECO SHE/HSM components.

2.2.13 SCF-432: Add API for software wakeup request

New IRQ type added:

- SC_IRQ_SW_WAKE

New PM function added:

- sc_pm_partition_wake()

This function will send an SC_IRQ_SW_WAKE interrupt to all MUs owned by the partition that have this interrupt enabled. The CPU using an MU will exit a low-power state to service the MU interrupt

2.2.14 SCF-447: Add API functions to access SNVS security violation and tamper registers

Added two new functions to the SECO service:

- sc_seco_secvio_enable()
- sc_seco_secvio_config()

Added a new resource:

- SC_R_SECVIO

Added a new interrupt:

- SC_IRQ_SECVIO

These functions are used to enable the security violation interrupt and to read/write SNVS security violation and tamper registers via the SECO Manage SNVS message. All register IDs are allowed except 0x68-0x9C. Access requires the caller own SC_R_SECVIO. See the Porting Guide for more info.

SCFW API updated to version 1.13.

2.2.15 SCF-459: sc_pm_set_sys_power_mode() allows the system to enter an unsupported state

In board.c, board_power() was being used for two functions. To allow the user to power off the board and to allow the SCFW to transition the board into various low-power modes. board_power() is called directly by sc_pm_set_sys↩ _power_mode() so users could force an LPM transition directly rather than as part of a transition sequence and this results in a system failure. The solution is to separate these functions. board_power() is only for powering off the board. A new function is added to do LPM transitions.

- board_lpm()

For some board.c implementations on i.MX8QM (and phantoms), this means some functionality must be moved from board_power() to board_lpm(). See the NXP implementations for an example.

2.2.16 SCF-460: Block DDR access beyond valid range

The default memory regions created by the SCFW cover the entire potential DDR address space. There are two DDR address ranges: a lower 2GB DDR space and an upper 2-6GB space, depending on the SoC. Some DRC configurations result in mirroring of DDR at higher addresses. Access needs to be blocked to these out-of-rage addresses otherwise a security hole may exist.

This change is to all the board.c implementations for NXP boards. Customers should make similar changes. .

If the board has exactly 2GB DDR, just delete the upper region:

- rm_find_memreg(pt_boot, &mr_temp, 0x880000000ULL, 0x880000000ULL);
- rm_memreg_free(pt_boot, mr_temp);

If the board has less than 2GB DDR, delete the upper region as above and fragment and delete the top part of the lower region:

- rm_memreg_frag(pt_boot, &mr_temp, last address + 1, 0xFFFFFFFFULL);
- rm_memreg_free(pt_boot, mr_temp);

If the board has more than 2GB DDR, fragment and delete the top part of the upper region:

- rm_memreg_frag(pt_boot, &mr_temp, last address + 1, 0xFFFFFFFFFULL);
- rm_memreg_free(pt_boot, mr_temp);

See the NXP board.c implementations as examples.

2.2.17 SCF-468: Add board parameter to override ISI pixel clock frequency

This change is to support ISI pixel clock tuning required to implement a workaround for errata e050066. It adds a new board parameter:

- BOARD_PARM_ISI_PIX_FREQ

This parameter can be used in the implementation of board_parameter() in board.c to return a frequency (in Hz) for the imaging subsystem ISI pixel clock. If not specified the default is 400MHz (the max).

For example, adding the following to board_parameter() will reduce the frequency to 200MHz:

```
case BOARD_PARM_ISI_PIX_FREQ :

    rtn = SC_200MHZ;

    break;
```

2.2.18 SCF-474: Add header to provide GPIO driver compatibility

Added header file: drivers/gpio/fsl_gpio.h. This has defines to map legacy GPIO functions to the new RGPIO driver (drivers/rgpio/fsl_rgpio.h). This provides backwards compatibility, but best if code calls RGPIO functions rather than GPIO functions as this header cannot be used at the same time as the IGPIO driver (drivers/igpio/fsl_gpio.h). IGPIO has conflicting functions that start with GPIO. Note both of these drivers come from the MCUXpresso SDK.

2.2.19 SCF-487: Support partition power off and reboot

The sc_pm_set_partition_power_mode() function would hang or reboot the system when used. That's because most power transitions should be done using the HPM functions and WFI.

This function has been modified:

- Cannot be called to reset the partition making the call. This is to avoid issues with attempting to send a response to an MU that is powered off.
- Can only be called to power off a partition (mode = SC_PM_PW_MODE_OFF).

To restart the partition, the caller should call sc_pm_reboot_partition() or sc_pm_boot().

2.2.20 SCF-492: Add mechanism for SCFW client to reset IPC/RPC protocol

This change allows the MU and IPC protocol to be reset by the client. Setting the MU GIR0 will generate an interrupt to the SCFW to ask for reset. The client should then poll for the MU.CR GIR0 bit to clear. Then the MU needs to be reconfigured. See the RPC protocol section of the SCFW API Reference Guide for more information.

2.2.21 SCF-493: Add support for CAN doze mode

The SC_C_IPG_DOZE control is now available for the SC_R_CANx resources. This control can be used to set/clear the IPG_DOZE signal to the CAN controllers.

2.2.22 SCF-494: Add API to allow reset of a peripheral

Added a new PM API:

- sc_pm_resource_reset()

This function will reset a resource. Most resources cannot be reset unless the SoC design specifically allows it. In the case on MUs, the IPC/RPC protocol is also reset. Note a caller cannot reset an MU that this API call is sent on.

API version incremented to 1.14.

2.2.23 SCF-496: Critical section violation during boot (D=1 only)

SECO related API requests during early boot can conflict with SCFW main attempting to read SECO FW version for display. This is only possible in non-production builds (D=1). In production builds (D=0), main does not call to check the SECO FW version.

This fix adds a critical section around the SECO FW version call for protection.

2.2.24 SCF-498: Support SECO FW v2.5.3

Added function to manage the SNVS DGO:

- sc_seco_secvio_dgo_config()

SCFW API version updated to v1.15.

2.2.25 SCF-501: Add RPC servicing fairness (sequential client servicing)

SCFW RPC servicing is subject to prioritization among non-preemptive MU interrupts. By default these interrupts are set to the same priority level. For scenarios where multiple non-preemptive interrupts are pending, the SCU will execute interrupts based only on the interrupt number. This implies some clients can deny service to other clients depending on the MU interrupt connection. This change adds dynamic prioritization which adjusts interrupt priorities to ensure fairness.

2.2.26 SCF-502: Latch early wake events for SC_PM_WAKE_SRC_SCU configurations

SC_PM_WAKE_SRC_SCU wake events that occur between the time a CPU requests low-power mode (sc_pm_req_↩ cpu_low_power_mode) and the respective WFI signal is recognized by the SCFW are not latched. Such events will not result in the CPU being awakened. Additional wake events that arrive after the WFI is recognized will result in a CPU wakeup.

Typically the delay between the execution of sc_pm_req_cpu_low_power_mode and execution of WFI will be minimal. The recognition of the respective WFI signal may be delayed by outstanding SCFW requests from other clients which are typically handled within a few msec. Therefore in practice it is difficult to hit a case where the wake event is missed.

2.2.27 SCF-504: Drop the DC AXI internal clock to 375MHz for QXP B0 and C0

ECO fix done in QXP C0 for errata 'ERR050060: DC: PRG on the fly bypass switch issue' mandates that the axi_internal_clk frequency be limited to 375MHz. Per designer and verification team there is no performance drop in B0 due to this lower frequency.

2.2.28 SCF-505: Add support for partition watchdog window

New timer service function added:

- sc_timer_set_wdog_window()

This function can be used to define a service window. Any attempt to ping (kick) the watchdog using sc_timer_ping_↩ wdog() before the window time will result in the same watchdog action that occurs if not serviced before expiration.

SCFW API version updated to 1.16.

2.2.29 SCF-508: Add option for selecting LVDS PLL SSC frequency spread

Added an option to select the percentage of frequency spread between 0 and 2% for LVDS PLL spread spectrum. Support values are:

- BOARD_PARM_RTN_NOT_USED
- BOARD_PARM_SSC_N_0P4
- BOARD_PARM_SSC_N_1P0
- BOARD_PARM_SSC_N_1P4
- BOARD_PARM_SSC_N_2P0

Returned from board_parameter() function for BOARD_PARM_DCx_PLLx_SSC param.

2.2.30 SCF-512: Disable PI PLL0 by default to allow the rate to be configured

Prior to this fix, the rate of PI_PLL_0 can only be configured if the entire PI_0 SS was powered down.

This patch allows for the PI_PLL_0 to be configured when ever the PLL and all the clocks sourced from it are disabled.

2.2.31 SCF-516: Reset clock rates associated with a resource on power down

Prior to this patch the rates and parents of clocks associated with any resource were only reset when the entire subsystem underwent a power transition to SC_PM_PW_MODE_OFF state.

This led to an incorrect assumption that SCFW was storing the parent/rates of clocks even after the resource was powered off.
This patch fixes this issue and all rates/parents of clocks associated with a resource are reset when the resource is powered off immaterial of the state of other resources in the same SS.

This patch will require users to set the rate and parent of clocks (clocks not auto-enabled by SCFW) anytime the resource has gone through an OFF power transition before the clock is enabled.

Chapter 3

Known Issues

Below is a list of known outstanding issues in this release (imx_4.14.98_ga2).

3.1 New Feature

Key	Summary	QM (B0)	QXP (B0)	QXP (C0)
SCF-240	Create new PERF resource to control access to performance counters			
SCF-333	Create CAAM unit test	Y	Y	Y
SCF-515	Add export for SCFW API header files only	Y	Y	Y

3.2 Silicon Workaround

These are a mix of silicon errata workarounds and recommended usage changes.

Key	Summary	QM (B0)	QXP (B0)	QXP (C0)
SCF-500	Disable LPDDR4 derating errata workaround for i.MX8QXP C0			Y

Chapter 4

Additional Notes

This section details any additional notes about the original release. These do not cover changes in patch releases.

4.1 General

When the SCFW is compiled for release into production devices, it is critical that this is done without debug (default is debug enabled, D=1) and without the debug monitor (default is no monitor, M=0). For example:

```
make qm R=B0 D=0 M=0
```

Turning off debug will eliminate the linking of the standard C library. See the porting guide for more information.

The porting kit contains separate tar files for each SoC/version combination. These can be combined using the following command:

```
find scfw_export_mx8*.gz -exec tar --strip-components 1 --one-top-level=scfw_export_mx8 -xzvf {} \;
```

Note i.MX8QXP B0 and C0 use the same code so the only source tarball is for B0. Also, binaries created for i.MX8QXP are run-time compatible to both silicon versions so always build with R=B0.

4.2 SCFW API Changes

The client API is backwards compatible at the compile-time API level and at the RPC/IPC level. The changes below are all additions. The only exception is that the sc_misc_seco_* APIs deprecated in the last release have been removed (exist now only in the SECO service).

4.2.1 Interrupt (IRQ) Service

- Added SC_IRQ_SW_WAKE interrupt type
- Added SC_IRQ_SECVIO interrupt type

4.2.2 Miscellaneous (MISC) Service Changes

- Added sc_misc_seco_auth_cmd_t type and defines
- Removed deprecated SECO functions. These are now only in the SECO service.
- Added sc_misc_get_boot_container()

4.2.3 Pad Service Changes

None

4.2.4 Power Management (PM) Service

- Behavior of sc_pm_set_partition_power_mode() changed
- Added sc_pm_partition_wake()
- Added sc_pm_resource_reset()

4.2.5 Resource Management (RM) Service

None

4.2.6 SECO Service Changes

- Added SC_SECO_EVERIFY_IMAGE authentication define
- Added sc_seco_rng_stat_t type
- Added sc_seco_enh_authenticate()
- Added sc_seco_start_rng()
- Added sc_seco_sab_msg()
- Added sc_seco_secvio_enable()
- Added sc_seco_secvio_config()
- Added sc_seco_secvio_dgo_config()

4.2.7 Timer Service Changes

- Added sc_timer_set_wdog_window()

4.3 Resource Changes

- Added additional frequency defines (e.g. SC_114MHZ)
- Some unused resources (SC_R_UNUSEDx) repurposed for future devices
- Some old VPU resources (e.g. SC_R_VPUCORE) repurposed for future devices
- Added SC_R_SECVIO

4.4 Clock Changes

- i.MX8QXP DC AXI clock changed from 400MHz to 375MHz (B0 and C0)

4.5 Control Changes

- Added SC_C_OFS_AUDIO_ALT and SC_C_DSP_BYP required for DSP optimization

4.6 Board Interface Changes

- New BOARD_PARM_RTN_VDD_MEMC_NOM and BOARD_PARM_RTN_VDD_MEMC_OD board parm returns added
- New BOARD_PARM_DCx_PLLx_SSC PLL spread spectrum board parameters added
- New BOARD_PARM_ISI_PIX_FREQ parameter added
- New BOARD_PARM_VDD_MEMC parameter added
- New BOARD_BFAULT_BRD_FAIL, BOARD_BFAULT_TEST_FAIL, and BOARD_BFAULT_DDR_INIT_FAIL board fault types added
- New DDR BOARD_DDR_DERATE_PERIODIC action added
- New BOOT_PHASE_SYS_INIT boot phases added for board_init()
- board_system_config() returns void instead of sc_err_t
- Added board_lpm()
- board_trans_resource_power() returns void instead of sc_err_t
- Not an interface change but the GPIO driver has been replaced with the RGPIO driver

Chapter 5

Disclaimer

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. While NXP has implemented advanced security features, all products may be subject to unidentified vulnerabilities. Customers are responsible for the design and operation of their applications and products to reduce the effect of these vulnerabilities on customer's applications and products, and NXP accepts no liability for any vulnerability that is discovered. Customers should implement appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, COOLFLUX, EMBRACE, GREE↩ NCHIP, HITAG, I2C BUS, ICODE, JCOP, LIFE VIBES, MIFARE, MIFARE CLASSIC, MIFARE DESFire, MIFARE P↩ LUS, MIFARE FLEX, MANTIS, MIFARE ULTRALIGHT, MIFARE4MOBILE, MIGLO, NTAG, ROADLINK, SMARTLX, SMARTMX, STARPLUG, TOPFET, TRENCHMOS, UCODE, Freescale, the Freescale logo, AltiVec, C 5, CodeTEST, CodeWarrior, ColdFire, ColdFire+, C Ware, the Energy Efficient Solutions logo, Kinetis, Layerscape, MagniV, mobile↩ GT, PEG, PowerQUICC, Processor Expert, QorIQ, QorIQ Qonverge, Ready Play, SafeAssure, the SafeAssure logo, StarCore, Symphony, VortiQa, Vybrid, Airfast, BeeKit, BeeStack, CoreNet, Flexis, MXC, Platform in a Package, QUICC Engine, SMARTMOS, Tower, TurboLink, and UMEMS are trademarks of NXP B.V. All other product or service names are the property of their respective owners. Arm, AMBA, Arm Powered, Artisan, Cortex, Jazelle, Keil, SecurCore, Thumb, TrustZone, and ?Vision are registered trademarks of Arm Limited (or its subsidiaries) in the EU and/or elsewhere. Arm7, Arm9, Arm11, big.LITTLE, CoreLink, CoreSight, DesignStart, Mali, Mbed, NEON, POP, Sensinode, Socrates, ULINK and Versatile are trademarks of Arm Limited (or its subsidiaries) in the EU and/or elsewhere. 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.