Technical Reference Manual

TPS6594-Q1 Technical Reference Manual

Table of Contents

Read This First

This Technical Reference Manual (TRM) details the integration, the environment, the functinoal description, and the programming models for each peripheral and subsytem in the device.

About This Manual

For a complete listing of releated documentation and development-support tools for the device, visit the Texas Instruments website at www.ti.com.

Notational Conventions

This document uses the following conventions.

- Hexadecimal numbers may be shown with the suffix h or the prefix 0x. For example, the following number is 40 hexadecimal (decimal 64): 40h or 0x40.
- Registers in this document are shown in figures and described in tables.
  - Each register figure shows a rectangle divided into fields that represent the fields of the register. Each field is labeled with its bit name, its beginning and ending bit numbers above, and its read/write properties with default reset value below. A legend explains the notation used for the properties.
  - Reserved bits in a register figure can have one of multiple meanings:
    - Not implemented on the device
    - Reserved for future device expansion
    - Reserved for TI testing
    - Reserved configurations of the device that are not supported
  - Writing nondefault values to the Reserved bits could cause unexpected behavior and should be avoided.

Glossary

TI Glossary This glossary lists and explains terms, acronyms, and definitions.

Related Documentation

For product information, visit the Texas Instruments website at http://www.ti.com.

STDZ039— Peripheral Reference Guide Template for SOC DSPs. Describes the peripheral reference guide template for the SOC DSPs.

Support Resources

TI E2E™ support forums are an engineer's go-to source for fast, verified answers and design help — straight from the experts. Search existing answers or ask your own question to get the quick design help you need.

Linked content is provided "AS IS" by the respective contributors. They do not constitute TI specifications and do not necessarily reflect TI's views; see TI's Terms of Use.

Trademarks

TI E2E™ is a trademark of Texas Instruments.
All trademarks are the property of their respective owners.

1 Introduction

(1)

1.1 Overview

The TPS65941-Q1 is an integrated power-management device for automotive and industrial applications. The device provides four flexible multi-phase configurable step-down converters with 3.5 A per phase, andone additional step-down converted with 2 A capability. All of the bucks can be synchronized to an internal 2.2 MHz or 4.4 MHz or an external 1 MHz, 2 MHz, or 4 MHz clock signal. To improve EMC performanceof the device, an integrated spread-spectrum modulation can be added to the synchronized buck switchingclock signal, which can also be made available to external devices through a GPIO output pin. The device 2 provides four LDOs; three with 500 mA capability which can be configured as load switches, one with 300 mA capability and low-noise performance.

Non-volatile memory (NVM) is used to control the default power sequences, as well as defaultconfigurations such as output voltage and GPIO configurations. The NVM is factory-programmed to allowstart-up without external programming. Most static settings can be changed from the default through SPI or I2C registers to configure the device to meet many different system needs. As a safety feature, the NVM contains a bit-integrity-error detection feature to stop the power-up sequence if an error is detected, preventing the system from starting in an unknown state.

The TPS65941-Q1 includes a 32 kHz crystal oscillator, which generates the accurate 32 kHz clock for the integrated RTC module. A backup-battery management provides power to the crystal oscillator and the RTC module from a coin cell battery or a super-cap in the event of power loss from the main supply.

The TPS65941-Q1 device includes a Q&A watchdog to monitor for software lockup, and 2 system error monitoring inputs with fault injection options to monitor the error signals from the attached SoC or MCU. The device also includes protection and diagnostic mechanisms such as short-circuit protection, thermal monitoring and shutdown. The PMIC can notify the processor of these events through the interrupt handler, allowing the processor to take action in response.

1.2 Functional Diagram

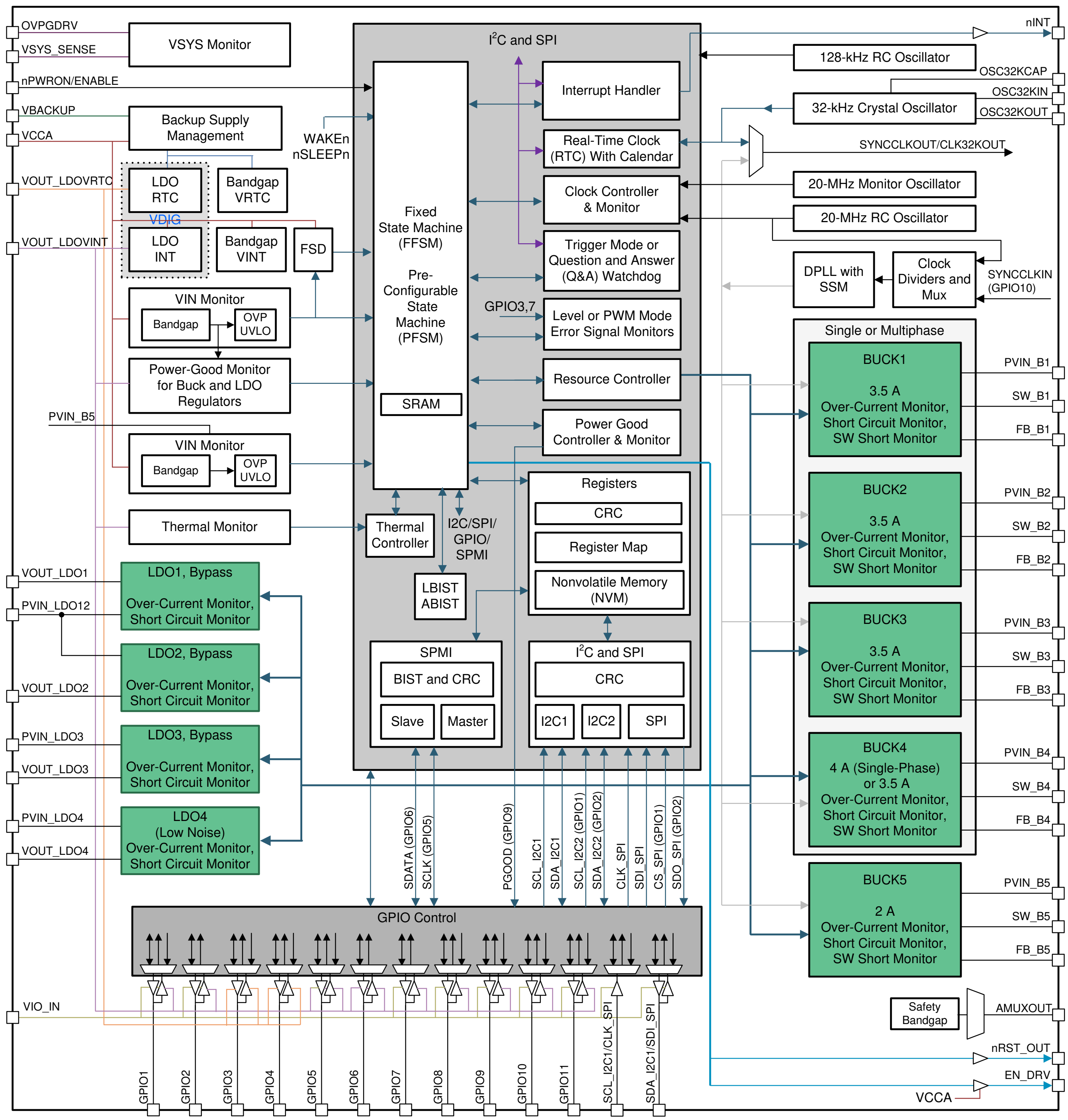


Figure 1-1. Functional Diagram

1.3 Digital Description Signals

Table 1-1. Signal Descriptions

SIGNAL NAME	I/O	Threshold Level	INPUT TYPE SELECTION		OUTPUT TYPE SELECTION		Internal PU/ PD[2]	RECOMMENDED EXTERNAL PU/PD[2]	Control Register Bits
			Power Domain	DEGLITCH TIME[5]	Power Domain	Push-pull/ Open-drain[4]			
nPWRON (Selectable function of nPWRON/ ENABLE pin)[1]	Input	$V_{IL(VCCA)}$, $V_{IH(VCCA)}$	VRTC	50 ms			400 kΩ PU to VCCA	None	NPWRON_SEL
ENABLE (Selectable function of nPWRON/ ENABLE pin)[1]	Input	$V_{IL(VCCA)}$, $V_{IH(VCCA)}$	VRTC	8 µs			400 kΩ SPU to VCCA, or 400 kΩ SPD to GND	None	NPWRON_SEL ENABLE_POL ENABLE_DEGLITCH_EN ENABLE_PU_PD_EN ENABLE_PU_SEL
EN_DRV	Output	$V_{OL(EN\_DRV)}$			VCCA/ PVIN_B1	PP	10 kΩ High-side to VCCA	None	ENABLE_DRV
SCL_I2C1 (Selectable function of SCL_I2C1/ SCK_SPI pin)[1]	Input	$V_{IL(DIG)}$, $V_{IH(DIG)}$	VINT	High-speed mode: 10 ns All other modes: 50 ns			None	PU to VIO	I2C_SPI_SEL[6] I2C1_HS
SDA_I2C1 (Selectable function of SDA_I2C1/ SDI_SPI pin)[1]	Input/output	$V_{IL(DIG)}$, $V_{IH(DIG)}$, $V_{OL(VIO)\_20mA}$	VINT	High-speed mode: 10 ns All other modes: 50 ns	VIO	OD	None	PU to VIO	I2C_SPI_SEL[6] I2C1_HS
SCL_I2C2 (Selectable function of GPIO1)[1]	Input	$V_{IL(DIG)}$, $V_{IH(DIG)}$	VINT	High-speed mode: 10 ns All other modes: 50 ns			None	PU to VIO	I2C_SPI_SEL[6] I2C2_HS GPIO1_SEL
SDA_I2C2 (Selectable function of GPIO2)[1]	Input/output	$V_{IL(DIG)}$, $V_{IH(DIG)}$, $V_{OL(VIO)\_20mA}$	VINT	High-speed mode: 10 ns All other modes: 50 ns	VIO	OD	None	PU to VIO	I2C_SPI_SEL[6] I2C2_HS GPIO2_SEL
SCK_SPI (Selectable function of SCL_I2C1/ SCK_SPI pin)[1]	Input	$V_{IL(DIG)}$, $V_{IH(DIG)}$	VINT	None			None	None	I2C_SPI_SEL[6]

Table 1-1. Signal Descriptions (continued)

SIGNAL NAME	I/O	Threshold Level	INPUT TYPE SELECTION		OUTPUT TYPE SELECTION		Internal PU/ PD(2)	RECOMMENDED EXTERNAL PU/PD(2)	Control Register Bits
			Power Domain	DEGLITCH TIME(5)	Power Domain	Push-pull/ Open-drain(4)			
SDI_SPI (Selectable function of SDA_I2C1/ SDI_SPI pin)(1)	Input	$V_{IL(DIG)}$, $V_{IH(DIG)}$	VINT	None			None	None	I2C_SPI_SEL(6)
CS_SPI (Selectable function of GPIO1)(1)	Input	$V_{IL(DIG)}$, $V_{IH(DIG)}$	VINT	None			None	None	I2C_SPI_SEL(6) GPIO1_SEL
SDO_SPI (Selectable function of GPIO2)(1)	Output	$V_{OL(VIO)\_20mA}$, $V_{OH(VIO)}$			VIO	PP(3) / HiZ	None	None	I2C_SPI_SEL(6) GPIO2_SEL
SCLK_SPMI (Configurable function of GPIO5)(1)	Input in Slave Mode Output in Master Mode	$V_{IL(DIG)}$, $V_{IH(DIG)}$, $V_{OL(DIG)\_20mA}$, $V_{OH(DIG)}$	VINT	None	VINT	PP	400 kΩ PD to GND	None	GPIO5_SEL GPIO5_PU_PD_EN
SDATA_SPMI (Configurable function of GPIO6)(1)	Input/output	$V_{IL(DIG)}$, $V_{IH(DIG)}$, $V_{OL(DIG)\_20mA}$, $V_{OH(DIG)}$	VINT	None	VINT	PP / HiZ	400 kΩ PD to GND	None	GPIO6_SEL GPIO6_PU_PD_EN
nINT	Output	$V_{OL(nINT)}$			VCCA	OD	None	PU to VCCA	
nRSTOUT	Output	$V_{OL(nRSTOUT)}$			VCCA/ VIO	PP(3) or OD	10 kΩ Pull-Up to VIO if configured as Push-Pull	PU to VIO if Open-drain (driven low if no VINT)	NRSTOUT_OD
nRSTOUT_SoC (Configurable function of GPIO1 and GPIO11)(1)	Output	$V_{OL(nRSTOUT)}$			VCCA/ VIO	PP(3) or OD	10 kΩ Pull-Up to VIO if configured as Push-Pull	PU to VIO if Open-drain (driven low if no VINT)	GPIO1_SEL GPIO1_OD GPIO11_SEL GPIO11_OD
PGOOD (Configurable function of GPIO9)(1)	Output	$V_{OL(VIO)}$, $V_{OH(VIO)}$			VIO	PP(3) or OD	None	PU to VIO if Open-drain	GPIO9_SEL GPIO9_OD PGOOD_POL PGOOD_WINDOW PGOOD_SEL_x
nERR_MCU (Configurable function of GPIO7)(1)	Input	$V_{IL(DIG)}$, $V_{IH(DIG)}$	VINT	8 µs			400 kΩ PD to GND	None	GPIO7_SEL

Table 1-1. Signal Descriptions (continued)

SIGNAL NAME	I/O	Threshold Level	INPUT TYPE SELECTION		OUTPUT TYPE SELECTION		Internal PU/PD[2]	RECOMMENDED EXTERNAL PU/PD[2]	Control Register Bits
			Power Domain	DEGLITCH TIME[5]	Power Domain	Push-pull/ Open-drain[4]			
nERR_SoC (Configurable function of GPIO3)[1]	Input	$V_{IL(DIG)}$, $V_{IH(DIG)}$	VRTC	15 µs			400 kΩ PD to GND	None	GPIO3_SEL
DISABLE_WDOG (Configurable function of GPIO8 and GPIO9)[1]	Input	$V_{IL(DIG)}$, $V_{IH(DIG)}$	VINT	30 µs			400 kΩ PD to GND	PU to VIO	GPIO8_SEL GPIO9_SEL
TRIG_WDOG (Configurable function of GPIO2 and GPIO11)[1]	Input	$V_{IL(DIG)}$, $V_{IH(DIG)}$	VINT	30 µs			400 kΩ SPD to GND	None	GPIO2_SEL GPIO2_PU_PD_EN GPIO11_SEL GPIO11_PU_PD_EN
nSLEEP1 (Configurable function of all GPIO pins)[1]	Input	$V_{IL(DIG)}$, $V_{IH(DIG)}$	GPIO3 or 4: VRTC other GPIOs: VINT	8 µs			GPIO3 or 4: 400 kΩ SPU to VRTC GPIO5 or 6: 400 kΩ SPU to VINT all other GPIOs: 400 kΩ SPU to VIO	None	GPIOn_SEL GPIOn_PU_PD_EN NSLEEP1B
nSLEEP2 (Configurable function of all GPIO pins)[1]	Input	$V_{IL(DIG)}$, $V_{IH(DIG)}$	GPIO3 or 4: VRTC other GPIOs: VINT	8 µs			GPIO3 or 4: 400 kΩ SPU to VRTC GPIO5 or 6: 400 kΩ SPU to VINT all other GPIOs: 400 kΩ SPU to VIO	None	GPIOn_SEL GPIOn_PU_PD_EN NSLEEP2B
WKUP1 (Configurable function of all GPIO pins except GPIO3 and GPIO4)[1]	Input	$V_{IL(DIG)}$, $V_{IH(DIG)}$	VINT	8 µs			GPIO5 or 6: 400 kΩ SPU to VINT or 400 kΩ SPD to GND all other GPIOs: 400 kΩ SPU to VIO or 400 kΩ SPD to GND	None	GPIOn_SEL GPIOn_DEGLITCH_EN GPIOn_PU_PD_EN GPIOn_PU_SEL

Table 1-1. Signal Descriptions (continued)

SIGNAL NAME	I/O	Threshold Level	INPUT TYPE SELECTION		OUTPUT TYPE SELECTION		Internal PU/PD(2)	RECOMMENDED EXTERNAL PU/PD(2)	Control Register Bits
			Power Domain	DEGLITCH TIME(5)	Power Domain	Push-pull/ Open-drain(4)			
WKUP2 (Configurable function of all GPIO pins except GPIO3 and GPIO4)(1)	Input	$V_{IL(DIG)}$, $V_{IH(DIG)}$	VINT	8 µs			GPIO5 or 6: 400 kΩ SPU to VINT or 400 kΩ SPD to GND all other GPIOs: 400 kΩ SPU to VIO or 400 kΩ SPD to GND	None	GPIOn_SEL GPIOn_DEGLITCH_EN GPIOn_PU_PD_EN GPIOn_PU_SEL
LP_WKUP1 (Configurable function of GPIO3 and GPIO4)(1)	Input	$V_{IL(DIG)}$, $V_{IH(DIG)}$	VRTC	8 µs, no degltch in LP_STANDBY state			400 kΩ SPU to VRTC, or 400 kΩ SPD to GND	None	GPIO3,4_SEL GPIO3,4_DEGLITCH_EN GPIO3,4_PU_PD_EN GPIO3,4_PU_SEL
LP_WKUP2 (Configurable function of GPIO3 and GPIO4)(1)	Input	$V_{IL(DIG)}$, $V_{IH(DIG)}$	VRTC	8 µs, no degltch in LP_STANDBY state			400 kΩ SPU to VRTC, or 400 kΩ SPD to GND	None	GPIO3,4_SEL GPIO3,4_DEGLITCH_EN GPIO3,4_PU_PD_EN GPIO3,4_PU_SEL
GPIO1	Input/output	$V_{IL(DIG)}$, $V_{IH(DIG)}$, $V_{OL(VIO)\_20mA}$, $V_{OH(VIO)}$	VINT	8 µs	VIO	PP(3) or OD	400 kΩ SPU to VIO, or 400 kΩ SPD to GND	PU to VIO if Open-drain	GPIO1_DIR Input: GPIO1_DEGLITCH_EN GPIO1_PU_PD_EN GPIO1_PU_SEL Output: GPIO1_OD
GPIO2	Input/output	$V_{IL(DIG)}$, $V_{IH(DIG)}$, $V_{OL(VIO)\_20mA}$, $V_{OH(VIO)}$	VINT	8 µs	VIO	PP(3) or OD	400 kΩ SPU to VIO, or 400 kΩ SPD to GND	PU to VIO if Open-drain	GPIO2_DIR Input: GPIO2_DEGLITCH_EN GPIO2_PU_PD_EN GPIO2_PU_SEL Output: GPIO2_OD

Table 1-1. Signal Descriptions (continued)

SIGNAL NAME	I/O	Threshold Level	INPUT TYPE SELECTION		OUTPUT TYPE SELECTION		Internal PU/PD[2]	RECOMMENDED EXTERNAL PU/PD[2]	Control Register Bits
			Power Domain	DEGLITCH TIME[5]	Power Domain	Push-pull/ Open-drain[4]			
GPIO3	Input/output	$V_{IL(DIG)}$, $V_{IH(DIG)}$, $V_{OL(DIG)}$, $V_{OH(DIG)}$	VRTC	8 µs	VINT	PP or OD	400 kΩ SPU to VINT, or 400 kΩ SPD to GND	PU to VIO if Open-drain	GPIO3_DIR Input: GPIO3_DEGLITCH_EN GPIO3_PU_PD_EN GPIO3_PU_SEL Output: GPIO3_OD
GPIO4	Input/output	$V_{IL(DIG)}$, $V_{IH(DIG)}$, $V_{OL(DIG)}$, $V_{OH(DIG)}$	VRTC	8 µs	VINT	PP or OD	400 kΩ SPU to VINT, or 400 kΩ SPD to GND	PU to VIO if Open-drain	GPIO4_DIR Input: GPIO4_DEGLITCH_EN GPIO4_PU_PD_EN GPIO4_PU_SEL Output: GPIO4_OD
GPIO5	Input/output	$V_{IL(DIG)}$, $V_{IH(DIG)}$, $V_{OL(VIO)\_20mA}$, $V_{OH(VIO)}$	VINT	8 µs	VINT	PP or OD	400 kΩ SPU to VINT, or 400 kΩ SPD to GND	PU to VIO if Open-drain	GPIO5_DIR Input: GPIO5_DEGLITCH_EN GPIO5_PU_PD_EN GPIO5_PU_SEL Output: GPIO5_OD
GPIO6	Input/output	$V_{IL(DIG)}$, $V_{IH(DIG)}$, $V_{OL(VIO)\_20mA}$, $V_{OH(VIO)}$	VINT	8 µs	VINT	PP or OD	400 kΩ SPU to VINT, or 400 kΩ SPD to GND	PU to VIO if Open-drain	GPIO6_DIR Input: GPIO6_DEGLITCH_EN GPIO6_PU_PD_EN GPIO6_PU_SEL Output: GPIO6_OD
GPIO7	Input/output	$V_{IL(DIG)}$, $V_{IH(DIG)}$, $V_{OL(DIG)}$, $V_{OH(DIG)}$	VINT	8 µs	VIO	PP[3] or OD	400 kΩ SPU to VIO, or 400 kΩ SPD to GND	PU to VIO if Open-drain	GPIO7_DIR Input: GPIO7_DEGLITCH_EN GPIO7_PU_PD_EN GPIO7_PU_SEL Output: GPIO7_OD

Table 1-1. Signal Descriptions (continued)

SIGNAL NAME	I/O	Threshold Level	INPUT TYPE SELECTION		OUTPUT TYPE SELECTION		Internal PU/ PD[(2)]	RECOMMENDED EXTERNAL PU/PD[(2)]	Control Register Bits
			Power Domain	DEGLITCH TIME[(5)]	Power Domain	Push-pull/ Open-drain[(4)]			
GPIO8	Input/output	$V_{IL(DIG)}$, $V_{IH(DIG)}$, $V_{OL(DIG)}$, $V_{OH(DIG)}$	VINT	8 µs	VIO	PP[(3)] or OD	400 kΩ SPU to VIO, or 400 kΩ SPD to GND	PU to VIO if Open-drain	GPIO8_DIR Input: GPIO8_DEGLITCH_EN GPIO8_PU_PD_EN GPIO8_PU_SEL Output: GPIO8_OD
GPIO9	Input/output	$V_{IL(DIG)}$, $V_{IH(DIG)}$, $V_{OL(DIG)}$, $V_{OH(DIG)}$	VINT	8 µs	VIO	P[(3)]P or OD	400 kΩ SPU to VIO, or 400 kΩ SPD to GND	PU to VIO if Open-drain	GPIO9_DIR Input: GPIO9_DEGLITCH_EN GPIO9_PU_PD_EN GPIO9_PU_SEL Output: GPIO9_OD
GPIO10	Input/output	$V_{IL(DIG)}$, $V_{IH(DIG)}$, $V_{OL(DIG)}$, $V_{OH(DIG)}$	VINT	8 µs	VIO	PP[(3)] or OD	400 kΩ SPU to VIO, or 400 kΩ SPD to GND	PU to VIO if Open-drain	GPIO10_DIR Input: GPIO10_DEGLITCH_EN GPIO10_PU_PD_EN GPIO10_PU_SEL Output: GPIO10_OD
GPIO11	Input/output	$V_{IL(DIG)}$, $V_{IH(DIG)}$, $V_{OL(DIG)}$, $V_{OH(DIG)}$	VINT	8 µs	VIO	PP[(3)] or OD	400 kΩ SPU to VIO, or 400 kΩ SPD to GND	PU to VIO if Open-drain	GPIO11_DIR Input: GPIO11_DEGLITCH_EN GPIO11_PU_PD_EN GPIO11_PU_SEL Output: GPIO11_OD
SYNCCLKIN (Configurable function of GPIO10)[(1)]	Input	$V_{IL(DIG)}$, $V_{IH(DIG)}$	VINT	None			400 kΩ SPD to GND	None	GPIO10_SEL GPIO10_PU_PD_EN
SYNCCLKOUT (Configurable function of GPIO8, GPIO9, and GPIO10)[(1)]	Output	$V_{OL(VIO)}$, $V_{OH(VIO)}$			VIO	PP[(3)]	None	None	GPIO8_SEL GPIO9_SEL GPIO10_SEL

Table 1-1. Signal Descriptions (continued)

SIGNAL NAME	I/O	Threshold Level	INPUT TYPE SELECTION		OUTPUT TYPE SELECTION		Internal PU/ PD[2]	RECOMMENDED EXTERNAL PU/PD[2]	Control Register Bits
			Power Domain	DEGLITCH TIME[5]	Power Domain	Push-pull/ Open-drain[4]			
CLK32KOUT (Configurable function of GPIO3, GPIO4, GPIO8, and GPIO10)[1]	Output	GPIO3 or 4: $V_{OL(DIG)}$, $V_{OH(DIG)}$ GPIO8 or 10: $V_{OL(VIO)}$, $V_{OH(VIO)}$			GPIO3 or 4: VRTC GPIO8 or 10: VIO	PP[3]	None	None	GPIO3_SEL GPIO4_SEL GPIO8_SEL GPIO10_SEL

(1) Configurable function through NVM register setting.
(2) PU = Pullup, PD = Pulldown, SPU = Software-configurable pullup, SPD = Software-configurable pulldown.
(3) When VIO is not available, the push-pull pin should be configured as low output to minimize current leakage from the IO cell.
(4) PP = Push-pull, OD = Open-drain.
(5) Deglitch time is only applicable when option is enabled.
(6) I2C_SPI_SEL refers to NVM setting and cannot be override during operation.

2 Detailed Description

2.1 Overview

The TPS65941-Q1 device is a power-management integrated circuit (PMIC), available in a 56-pin, 0.5-mm pitch, 8-mm × 8-mm QFN package. It is designed for powering embedded systems or system on chip (SoC) in Automotive or Industrial applications. It provides five configurable buck converter rails, with four of the rails having the ability to combine outputs in multi-phase mode. BUCK4 has the ability to supply upto 4 A in single-phase mode, while BUCK1, BUCK2, and BUCK3 have the ability to supply up to 3.5 A insingle-phase mode. When working in multi-phase mode, each BUCK1, BUCK2, BUCK3, and BUCK4 can supply up to 3.5 A per phase, adding up to 14 A in four-phase configuration. BUCK5 is a single-phaseonly buck converter which supports up to 2 A current load. All five of the BUCK converters has thecapability to sink up to 1 A, and support dynamic voltage scaling. Double buffered voltage scaling registers enable each BUCK to transition to a different voltages during operation by SPI or I2C. A DPLL enables the BUCK converters to synchronizing to an external clock input, with phase delays between the outputs rails.

The TPS65941-Q1 device also provides three LDO rails which can supply up to 500 mA per rail and canbe configured in bypass mode to be used as a load switch. One additional low-noise LDO rail can supplyup to 300 mA. The 500 mA LDOs support 0.6 V to 3.3 V output with 50 mV step. The 300 mA low-noise LDO supports 1.2 V to 3.3 V output with 25 mV step. The output voltages of the LDOs can beprogrammable through the SPI or I2C interfaces.

Two I2C interface channels or one SPI channel can be used to program the power rails and configure thepower state of the TPS65941-Q1 device. I2C channel 1 (I2C1) is the main channel with access to theregisters which control the programmable power sequencer, the states and the outputs of power rails (including DVFS), the device operating states, and the RTC registers. I2C channel 2 (I2C2), which isavailable through GPIO1 and GPIO2 pins, is dedicated for accessing the Q&A Watchdog communication registers. When the SPI is configured instead of the two I2C interfaces, the SPI can access all of the registers, including the Q&A Watchdog registers. An NVM option is available to enable I2C1 to access all of the registers as well, including the Q&A Watchdog registers.

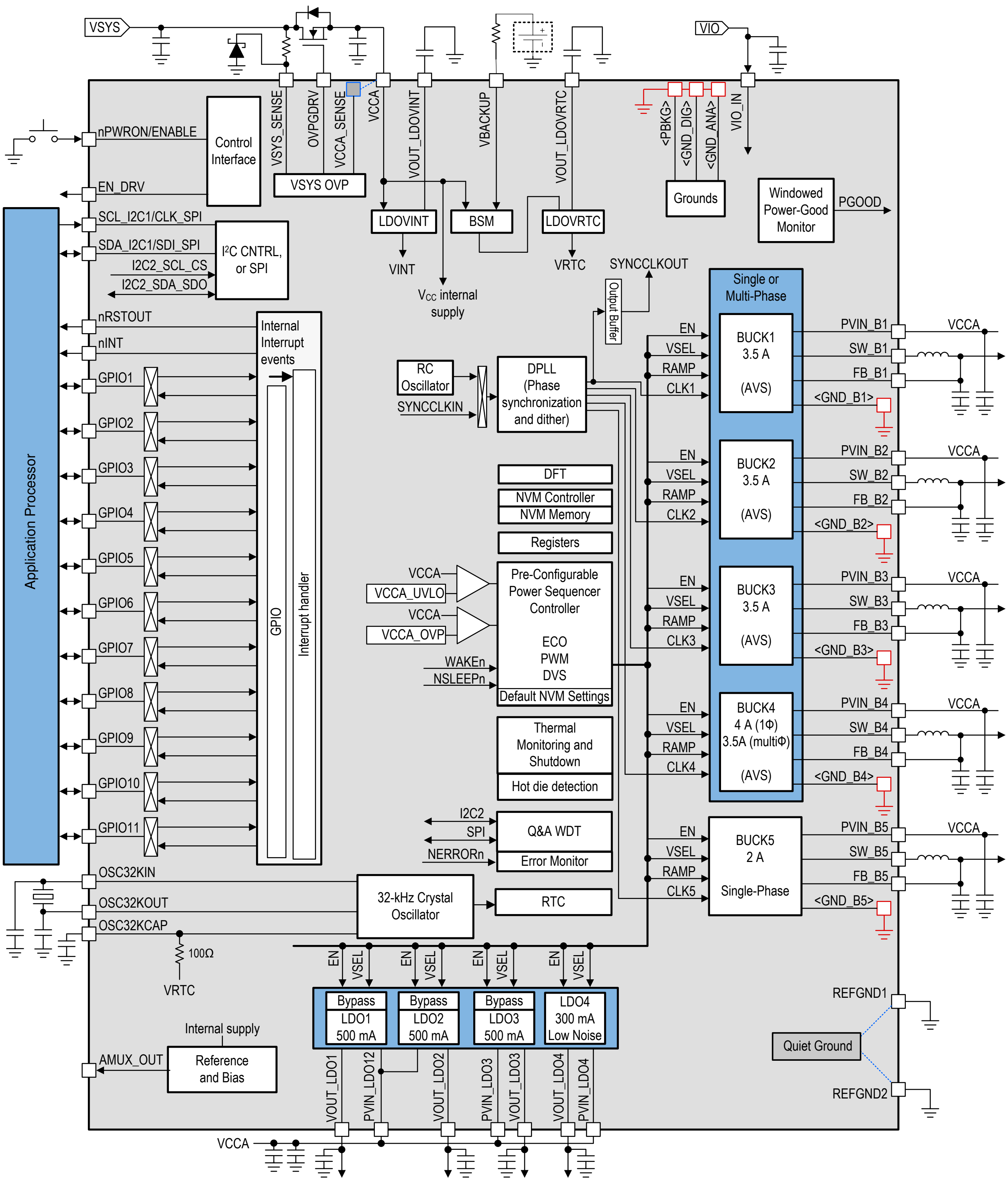
The TPS65941-Q1 device includes an internal RC oscillator to sequence all resources during power upand power down. Two internal LDOs (LDOVINT and LDOVRTC) generate the supply for the entire digitalcircuitry of the device as soon as the external input supply is available through the VCCA input. A backupbattery supply input can also be used to power the RTC block and a 32 kHz Crystal Oscillator clockgenerator in the event of a power loss from the main supply.

TPS65941-Q1 device has eleven GPIOs each with multiple functions and configurable features. All of theGPIOs, when configured as a general purpose output pin, can be included in the power-up and powerdownsequence and used as enable signals for external resources. In addition, each GPIO can beconfigured as a wake-up input or a sleep mode trigger. The default configuration of the GPIO port comesfrom the NVM memory, and can be re-programmed by software if the external connection permits.

The TPS65941-Q1 device includes a Q&A watchdog to monitor software lockup, and two system errormonitoring inputs with fault injection options to monitor the lock-step signal of the attached SoC or MCU. The device includes protection and diagnostic mechanisms such as short-circuit protection, thermalmonitoring and shutdown. The PMIC can notify the processor of these events through the interrupt signal open-drain output, allowing the processor to take action in response.

An SPMI interface is included in the TPS65941-Q1 device to distribute power state information to satellitePMICs, thus enabling synchronous power state transition across multiple PMICs in the application system.This feature allows the consolidation of IO control signals required between the application processor or MCU and any number of PMICs in the system into TPS65941-Q1 only.

2.2 Functional Diagram



* These red squares are internal pads for down-bonds to the package thermal/ground pad.

Figure 2-1. Functional Diagram

2.3 System Voltage Monitor and Over-Voltage Protection

The TPS65941-Q1 device includes an over-voltage protection mechanism through a 12 V compliant input monitor at the VSYS_SENSE pin. When an over-voltage is detected at the VSYS_SENSE pin, OVP GDRVpin is pulled low to disable the external high voltage load switch which connects the VSYS supply to theVCCA pin. To protect VSYS_SENSE pin from over-voltage condition due to possible short at the preregulatoroutput, we recommend connecting a 10 V zener diode to ground at the VSYS_SENSE pin, as well as one or more series resistors between the VSYS_SENSE pin and the output of the pre-regulator to limit thecurrent surge. The voltage slew rate at the VSYS_SENSE pin must be limited to ≤100 mV/µs to prevent possible damage to the device

In case the TPS65941-Q1 device detects a VCCA over voltage condition, the VCCA domain will be unpowered and no longer able to signal the over voltage condition to the VSYS over-voltage protection module. Therefore, a dead-lock mechanism is implemented in the VSYS domain by setting a latch to keep the external high voltage load switch (between VSYS and VCCA) open once the Leo device has detected a VCCA over voltage condition.

The comparator module which monitors the voltage on the VCCA pins controls the power state machine of the TPS65941-Q1 device. VCCA voltage detection outputs determine the power states of the device as following:

VCCA_UVLO When the voltage on the VCCA pin rises above VCCA_UVLO during initial power up, the device transitions from the NO SUPPLY state to the INIT state. When the supply at the VCCA pin falls below the VCCA_UVLO threshold, the device returns to the BACKUP state. During BACKUP state LDOVRTC is powered by the output of the Backup Supply Management (BSM) module. When the input supply of the LDOVRTC falls below the operating range, the device returns to the NO SUPPLY state and is completely shutdown. The device will notreturn to the BACKUP state from the NO SUPPLY state.

VCCA_OVP While the device is in operation, if the voltage on VCCA pin rises above the VCCA_OVP threshold despite the OVPGDRV mechanism, the device will clear the ENABLE_DRV bit and start the immediate shutdown sequence to protect itself from over-voltage input condition.

When VCCA is expected to be 5 V or 3.3 V, a separate voltage comparator can be enabled to monitor whether or not the VCCA voltage is within the expected PGOOD range. Please refer to Section 2.8 for additional detail on the operation of the PGOOD monitor function.

The Figure 2-2 shows a block diagram of the system input monitoring and over-voltage protection mechanism, and the generation of the VCCA UVLO and VCCA_OVP power state control signals.

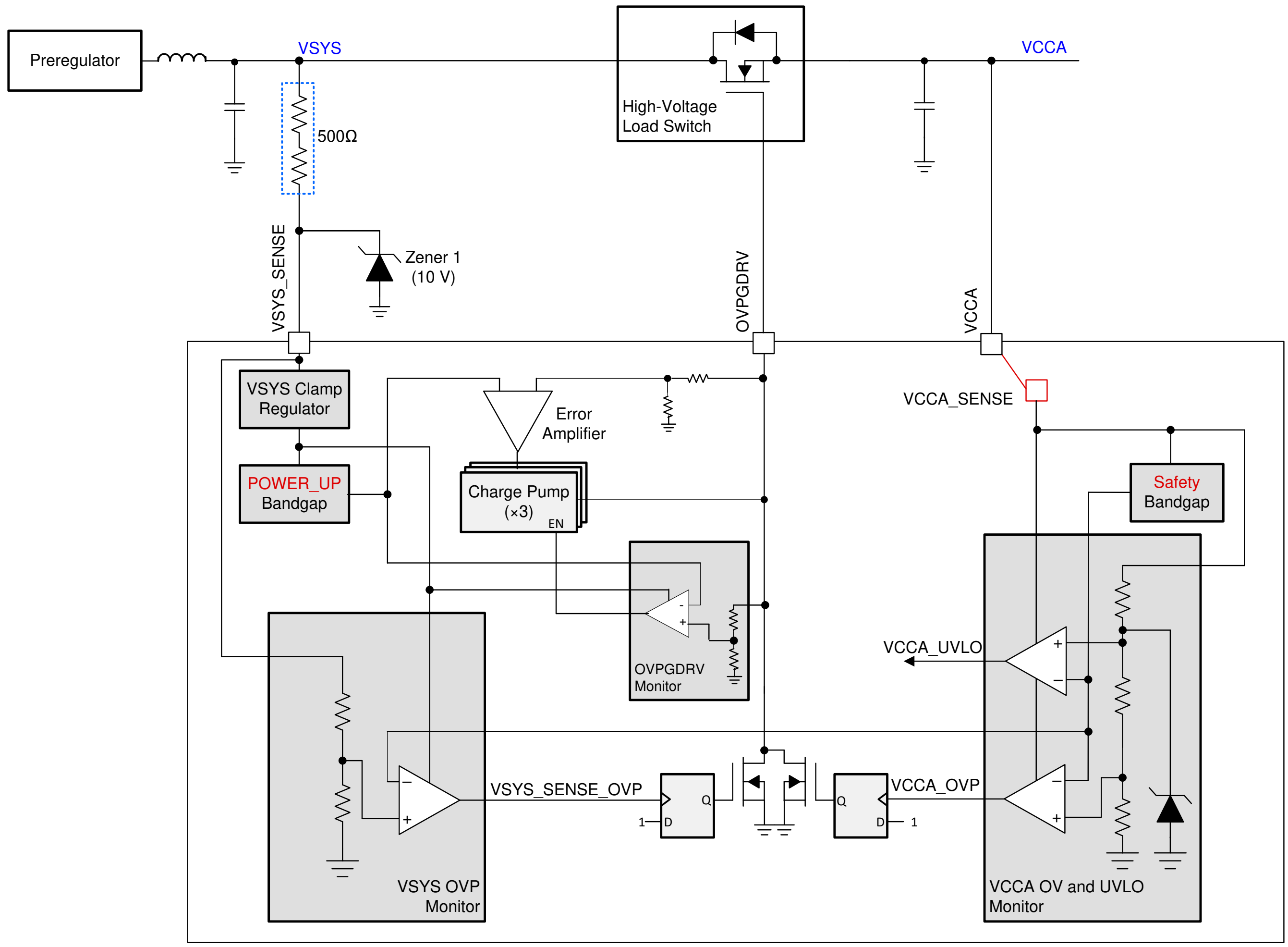


Figure 2-2. VSYS Monitor and OVPGDRV Output Generation

2.4 Device State Machine

The TPS6594-Q1 device integrates a finite state machine (FSM) engine, which manages the state of the device during operating state transitions. It supports EEPROM configurable mission states with configurable input triggers for transitions between states. Any resources, including the 5 bucks, 4 LDOs, and all of the digital IO pins including the 11 GPIO pins on the device, can be controlled during power sequencing. When a resource is not controlled or configured through a power sequence, the resource is left in the default state as pre-configured by the NVM.

Each resource can be pre-configured through the NVM configuration, or re-configured through register bits. Therefore, the user can statically control the resource through the control interfaces (I^2C or SPI), or the FSM can automatically control the resource during state sequences.

The FSM is powered by an internal LDO which is automatically enabled when VCCA supply is available to the device. Ensuring that the VCCA supply is the first supply available to the device is important to ensure proper operation of all the power resources as well as the control interface and device IOs.

There are 3 parts of the FSM which control the operational modes of the TPS6594-Q1 device:

- Fixed Device Power Finite State Machine (FFSM)
- Pre-Configurable Mission Finite State Machine (PFSM)
- Error Handling Operations

The PFSM works in complimentry to the traditional FFSM in order to draw from the strengths of both designs and to reduce the high cost of a completely configurable FSM. The PFSM provides configurable rail sequencing utilizing instructions in configuration memory. This flexibility enables customers to alter power-up sequences on a platform basis. The FFSM handles the majority of fixed functionality that is internally mandated and common to all platforms.

2.4.1 Fixed Device Power States

The Fixed Device Power States portion of the FSM engine (FFSM) manages to power up the device before the power rails are fully enabled and ready to power external loadings, and to power down the device when in the events of insufficient power supply or device/system error conditions. While the device is in one of the Hardware Device Powers states, the ENABLE_DRV bit remains low.

The definitions and transition triggers of the Device Power States are fixed and cannot be reconfigured.

Following are the definitions of the Device Power states:

No Supply The device is not powered by a valid energy source on the system power rail. The device is completely powered off.

BACKUP (RTC backup battery) The device is not powered by a valid supply on the system power rail (VCCA < VCCA_UVLO). However a backup power source is present and is within the operating range of the LDO VRTC. The RTC clock counter remains active in this state if it has been previously activated by appropriate register enable bit. The calendar function of the RTC block is activated, butnot accessible in this state. Customer has the option to enable the shelf mode by setting the LDORTC_DIS bit to 1 while the I2C is in operation. This bit will force the device to skip the BACKUP state and enters the NO SUPPLY state under VCCA_UVLO condition.

LP_STANDBY The device can enter this state from a mission state after receiving a valid OFF requestor an I2C trigger, and the LP_STANDBY_SEL= 1. When the device is in this state, the RTC clockcounter and the RTC Alarm or Timer Wake up functions are active if they have been previously activated by appropriate register enable bit. Low Power Wake-up input monitor in the LDOVRTC domain (LP_WKUP secondary function through GPIO3 or GPIO4) and the on request monitors are also enabled in this state. When a logic level transition from high-to-low or low-to-high with aminimum pulse length of tLP_WKUP is detection on the assigned LP_WKUP pin, or if the devicedetects an valid on-request or a wake up signal from the RTC block, the device will proceed toexecute the power up sequence and reach the default mission state. More details regarding the LP_WAKE function can be found in Section 2.4.2.3.2.3.

INIT The device is powered by a valid supply on the system power rail (VCCA ≥ VCCA_UV) and have received an external wake-up signal such as CAN WAKE-UP, the RTC alarm or timer wake-up signal, or an On Request from the nPWRON/ENABLE pin. Device digital and monitor circuits are powered up. The PMIC reads its internal NVM memory in this state and configures default values to registers, IO configuration and FSM

BOOST BIST The device is running the built-in self test routine which includes both the LBIST (around 3 ms run time) and the ABIST/CRC (around 100 µs run time). An option is available to shorten the device power up time from the NO_SUPPLY state by setting the NVM bit FAST_BOOT_BIST = '1''to skip the LBIST. Software can also set the FAST_BIST = '1' to skip LBIST after the device wakesup from the LP STANDBY state. When the device arrives this state from the SAFE_RECOVERY state, LBIST is automatically skipped if it has not previously failed. If LBIST failed, but passed aftermultiple re-tries before exceeding the recovery counter limit, the device will be powered upnormally. The following NVM bits are additional options which can be set to disable parts of the ABIST/CRC tests if further sequence time reduction is required. Note: the BIST tests are executed as parallel processes, and the longest process determines the total BIST duration.Therefore the following options does not guarantee a certain amount of time saving from the total BIST duration:

- REG_CRC_EN = '0': disables the register map and SRAM CRC check
- VMON_ABIST_EN = '0': disables the ABIST for the VMON OV/UV function
- SPMI_WD_EN = '0': disables the SPMI Watchdog operation for the SPMI WD function

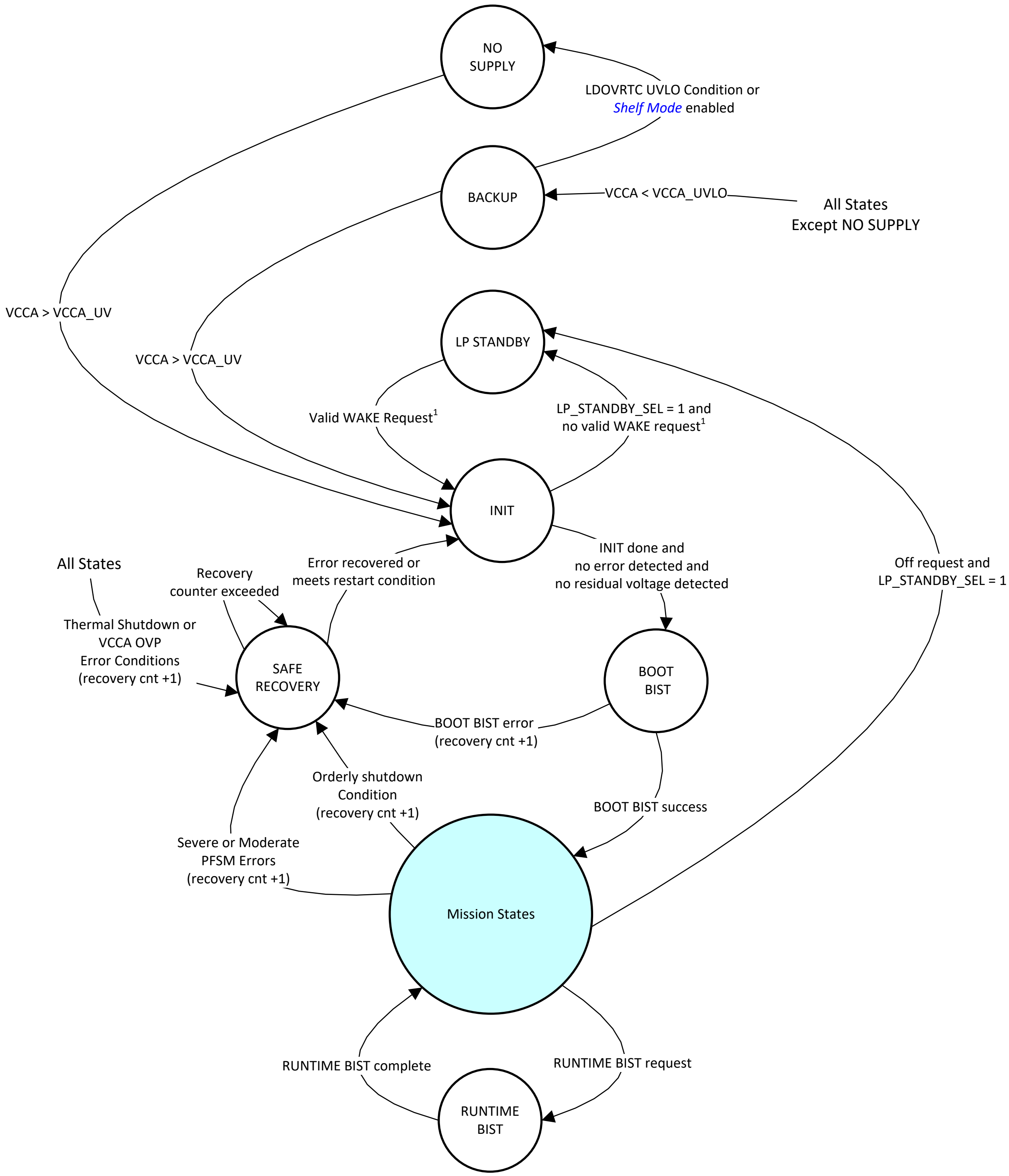
RUNTIME BIST MCU writes to the TRIGGER_I2C_1 register to exercise runtime BIST on the device. No rails are modified and all external signals, include all I2C or SPI interface communications, areignored during BIST. If the device passed BIST, it will resume the previous operation. if the device failed BIST, it will shut down all of the regulator outputs and proceed to the SAFE RECOVERY state. In order to avoid a register CRC error, all register write must be avoid after the request for the BIST operation until an interrupt is asserted to indicate the completion of BIST. The results of the BIST are indicated by the BIST_PASS_INT or the BIST_FAIL_INT bits.

SAFE RECOVERY The device meets the qualified error condition for immediate or ordered shutdown request. If the error is recovered within the recovery time interval, the device will increment therecovery count, and return to INIT state if the recovery count does not exceeded the threshold of the counter. If the recovery count exceeded the threshold or if the error cannot be recovered, suchas the die temperature cannot be reduced to < TWARN, or if VCCA stays above OVP threshold, the device will stay in SAFE RECOVERY state until supply power cycle occurs.

When multiple system conditions occur simultaneously which demands power state arbitration, the device will go to the higher priority state according to the following priority order:

1. NO SUPPLY
2. BACKUP
3. SAFE_RECOVERY
4. LP_STANDBY
5. MISSION STATES

Figure 2-3 shows the power transition states of the FSM engine.



[1] A valid WAKE request consist of:

- nPWRON/ENABLE on request detection if the device arrived the LP_STANDBY state through the long key-press of the nPWRON pin or by disabling the ENABLE pin, or
- RTC Alarm, RTC Timer, LP_WKUP1 or LP_WKUP2 detection if the device arrived the LP_STANDBY state through writing to a TRIGGER_I2C_0 bit.

Figure 2-3. State Diagram for Device Power States

2.4.1.1 Register Resets and EEPROM read at INIT State

When the device transitions from the LP_STANDBY or the SAFE_RECOVERY to the INIT state, the registers are reset and EEPROM is read based on FIRST_STARTUP_DONE bit. When the FIRST_STARTUP_DONE is '0', all of the registers are reset, and all of the EEPROM registers, includingthe ones in the RTC domains, will be loaded from the EEPROM. Once the FIRST_STARTUP_DONE bit isto '1', typically after the initial power up from a supply power cycle, the registers in the RTC domain will notbe reset, and the EEPROM registers in the RTC comain will no longer be loaded from the EEPROM. This prevents the control and status bits stored in the RTC domain registers from being over written.

Table 2-1. Register Resets and EEPROM Read at INIT State

FIRST_STARTUP_DONE	EEPROM Registers in RTC Domain	Non-EEPROM Registers in RTC Domain	Other EEPROM Registers	Other Non-EEPROM Registers
0	Defaults read from EEPROM	No changes	Reset and defaults read from EEPROM	Reset
1	No changes	No changes	Reset and defaults read from EEPROM	Reset

The bits in the RTC domain include the following:

- GPIO3_CONF and GPIO4_CONF registers, except the GPIOn_DEGLITCH_EN bits
- GPIO3_RISE_MASK, GPIO3_FALL_MASK, GPIO4_RISE_MASK, and GPIO4_FALL_MASK bits
- NPWRON_CONF register except ENALBE_DEGLITCH_EN and NRSTOUT_OD bits
- FSD_MASK, ENABLE_MASK, NPWRON, START_MASK, and NPWRON_LONG_MASK bits
- FIRST_STARTUP_DONE, STARTUP_DEST, FAST_BIST, LP_STANDBY_SEL, XTAL_SEL, and XTAL_EN bits
- SCRATCH_PAD_n, PFSM_DELAYn, and RTC_SPARE_n bits
- All of RTC control and configuration registers

2.4.2 Pre-Configurable Mission States

When the device arrives a mission state, all rail sequencing is controlled by the pre-configurable FSM engine (PFSM) through the configuration memory. The configuration memory allows configurations of the triggers and the operation states which together form the configurable sub state machine within the scope of mission states. This sub state machine could be used to control and sequence the different voltage outputs as well as any GPIO outputs that can be used as enable for external rails. Various forms of register writes provide the means of controlling the external environment. When the device is in a mission state, it has the capacity to supply the processor and other platform modules depending on the power rail configuration. The definitions and transition triggers of the mission states are configurable through the NVM configuration.

When the device is in any one of the Mission States, the state with the higher power level has the higher priority. For instance, if the Mission States consist of ACTIVE; MCU ONLY; DEEP SLEEP; S2R; and STANDBY states, then the priority order of these states should be in the following order:

1. ACTIVE
2. MCU ONLY
3. DEEP SLEEP / S2R
4. STANDBY

2.4.2.1 Mission State Configuration

The Mission States portion of the FSM engine manages the sequencing of power rails and external outputs in the user defined states. The rest of Section 2.4 will use Figure 2-4 as an example state machine which is defined through the configuration memory using the configuration FSM instructions.

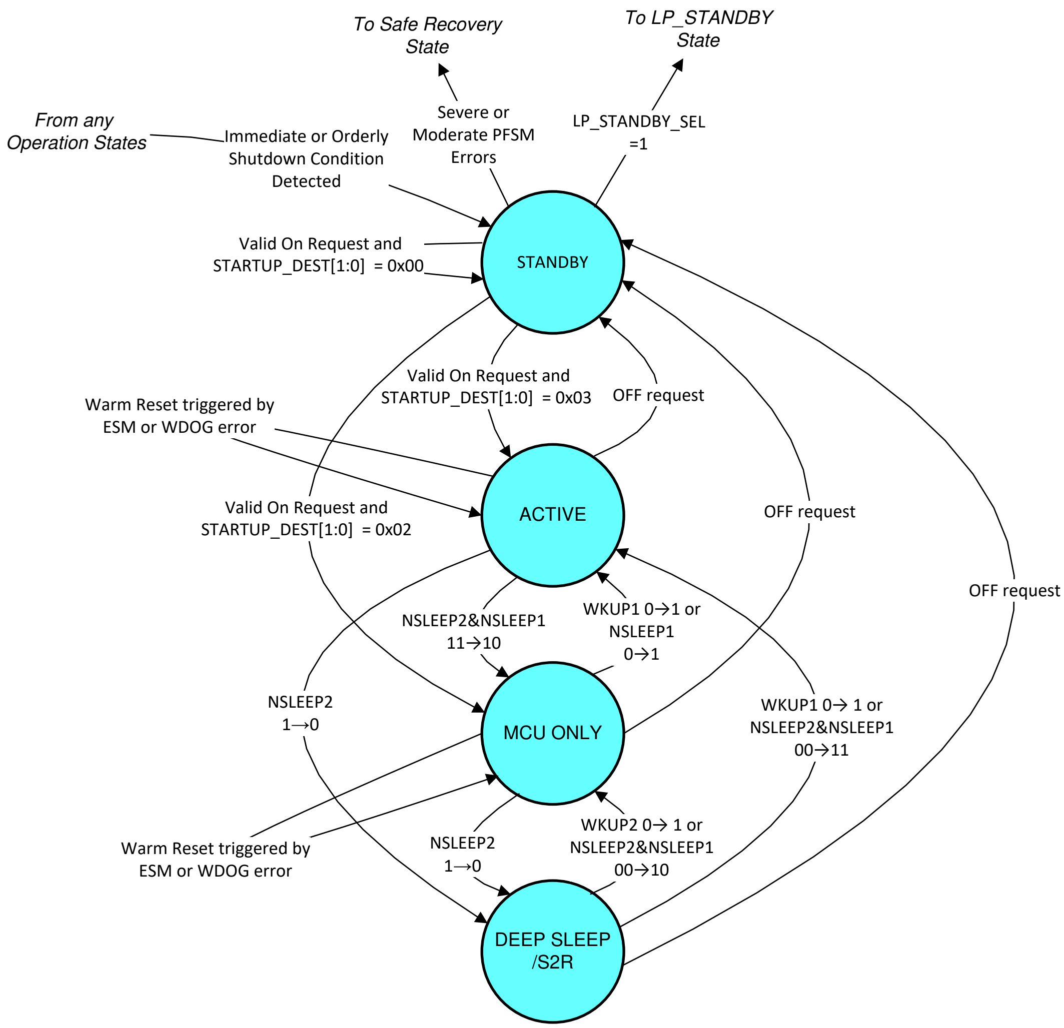


Figure 2-4. Example of a Mission State-Machine

Each power state (light blue bubbles in Figure 2-4) defines the ON or OFF state and the sequencing timing of the external regulators and GPIO outputs. This example defines 4 power states: STANDBY, ACTIVE, MCU ONLY, and DEEP_SLEEP/S2R states. The priority order of these states will be the following:

1. ACTIVE
2. MCU ONLY
3. DEEP SLEEP/S2R
4. STANDBY

The transitions between each power state is determined by the trigger signals source pre-selected from Section 2.4.2.2. These triggers are then placed in the order of priority through the trigger ID assignment of each trigger source. The critical error triggers are placed first, some specified as immediate triggers that can interrupt an on-going sequence. The non-error triggers which are used to enable state transtions during normal device operation are then placed according to the priority order of the state the device is transitioning to. Table 2-2 lists the trigger signal sources, in the order of priority, used to define the power states and transitions of the example mission state machine shown in Figure 2-4. This table also helps to determine which triggers should be masked by the TRIG_MASK command upon arriving a pre-defined power state to produce the desired PFSM behavior.

Table 2-2. List of Trigger Used in Example Mission State Machine

Trigger ID	Trigger Signal	State Transitions	Trigger Masked In Each User Defined Power State			
			STANDBY	ACTIVE	MCU ONLY	DEEP SLEEP / S2R
0	IMMEDIATE_SHUTDOWN (1)	From any state to SAFE RECOVERY				

Table 2-2. List of Trigger Used in Example Mission State Machine (continued)

Trigger ID	Trigger Signal	State Transitions	Trigger Masked In Each User Defined Power State			
			STANDBY	ACTIVE	MCU ONLY	DEEP SLEEP / S2R
1	MCU_POWER_ERROR (1)	From any state to SAFE RECOVERY				
2	ORDERLY_SHUTDOWN (1)	From any state to SAFE RECOVERY				
3	TRIGGER_FORCE_STANDBY	From any state to STANDBY or LP_STANDBY	Masked			
4	WD_ERROR	Perform warm reset of all power rails and return to ACTIVE	Masked		Masked	Masked
5	ESM_MCU_ERROR	Perform warm reset of all power rails and return to ACTIVE	Masked		Masked	Masked
6	ESM_SOC_ERROR	Perform warm reset of power rails in SOC domain and return to ACTIVE	Masked		Masked	Masked
7	WD_ERROR	Perform warm reset of all power rails and return to MCU ONLY	Masked	Masked		Masked
8	ESM_MCU_ERROR	Perform warm reset of all power rails and return to MCU ONLY	Masked	Masked		Masked
9	SOC_POWER_ERROR	ACTIVE to MCU ONLY	Masked		Masked	Masked
10	TRIGGER _I2C_1 (self-cleared)	Start RUNTIME_BIST	Masked			Masked
11	TRIGGER_I2C_2 (self-cleared)	Enable I2C CRC Function	Masked			Masked
12	TRIGGER_SU_ACTIVE	STANDBY to ACTIVE			Masked	Masked
13	TRIGGER_WKUP1	Any State to ACTIVE				
14	TRIGGER_A (NSLEEP2&NSLEEP1 = '11')	MCU ONLY or DEEP SLEEP/S2R to ACTIVE	Masked			
15	TRIGGER_SU_MCU_ONLY	STANDBY to MCU ONLY		Masked		Masked
16	TRIGGER_WKUP2	STANDBY or DEEP SLEEP/S2R to MCU ONLY		Masked		
17	TRIGGER_B (NSLEEP2&NSLEEP1 = '10')	ACTIVE or DEEP SLEEP/S2R to MCU ONLY	Masked			
18	TRIGGER_D or TRIGGER_C (NSLEEP2 = '0')	ACTIVE or MCU ONLY to DEEP SLEEP/S2R	Masked			Masked
19	TRIGGER_I2C_0 (self-cleared)	Any state to STANDBY	Masked			Masked
20	Always '1' (2)	STANDBY to SAFE RECOVERY	Mask	Masked	Masked	Masked
21	Not Used		Mask	Masked	Masked	Masked
22	Not Used		Mask	Masked	Masked	Masked
23	Not Used		Mask	Masked	Masked	Masked
24	Not Used		Mask	Masked	Masked	Masked
25	Not Used		Mask	Masked	Masked	Masked
26	Not Used		Mask	Masked	Masked	Masked
27	Not Used		Mask	Masked	Masked	Masked
28-bit TRIG_MASK Value in Hex format:			0xFFE4FF8	0xFF18180	0xFF01270	0xFFC9FF0

(1) This is an immediate trigger.

(2) When an error occurs, which requires the device to enter directly to the SAFE RECOVERY state, the mask for this trigger must be removed while all other non-immediate triggers are masked. The device will leave mission states and the FFSM state machine will take over control of the device power states once this trigger is executed.

2.4.2.2 Configuration Memory Organization and Sequence Execution

The configuration memory is loaded from EEPROM into an SRAM. Figure 2-5 shows an example configuration memory with only two configured sequences.

```
pfsm_start:
TRIG_SET DEST=sequence_name1 ID=0 SEL=trigger_name TYPE=high/low/rise/fall IMM=0/1 EXT=0/1
TRIG_SET DEST=sequence_name2 ID=1 SEL=trigger_name TYPE=high/low/rise/fall IMM=0/1 EXT=0/1
TRIG_SET DEST=sequence_name3 ID=2 SEL=trigger_name TYPE=high/low/rise/fall IMM=0/1 EXT=0/1
TRIG_SET DEST=sequence_name4 ID=4 SEL=trigger_name TYPE=high/low/rise/fall IMM=0/1 EXT=0/1
TRIG_SET DEST=sequence_name5 ID=4 SEL=trigger_name TYPE=high/low/rise/fall IMM=0/1 EXT=0/1
……
TRIG_MASK 0xFFFFFF0
END
sequence_name1
REG_WRITE_MASK_IMM ADDR=register_addr DATA=data MASK=mask_setting
REG_WRITE_MASK_IMM ADDR=register_addr DATA=data MASK=mask_setting
REG_WRITE_MASK_IMM ADDR=register_addr DATA=data MASK=mask_setting
REG_WRITE_VCTRL_IMM REGULATOR=regulator VCTRL=ctrl_setting MASK=mask_setting DELAY=delay_time
REG_WRITE_VCTRL_IMM REGULATOR=regulator VCTRL=ctrl_setting MASK=mask_setting DELAY=delay_time
DELAY_IMM delay_time
REG_WRITE_MASK_IMM ADDR=register_addr DATA=data MASK=mask_setting
REG_WRITE_VCTRL_IMM REGULATOR=regulator VCTRL=ctrl_setting MASK=mask_setting DELAY=delay_time
REG_WRITE_MASK_IMM ADDR=register_addr DATA=data MASK=mask_setting
TRIG_MASK 0xFC00EDF
END
……
sequence_name4
REG_WRITE_MASK_IMM ADDR=register_addr DATA=data MASK=mask_setting
DELAY_IMM delay_time
REG_WRITE_MASK_IMM ADDR=register_addr DATA=data MASK=mask_setting
REG_WRITE_VCTRL_IMM REGULATOR=regulator VCTRL=ctrl_setting MASK=mask_setting DELAY=delay_time
REG_WRITE_VCTRL_IMM REGULATOR=regulator VCTRL=ctrl_setting MASK=mask_setting DELAY=delay_time
DELAY_IMM delay_time
REG_WRITE_VCTRL_IMM REGULATOR=regulator VCTRL=ctrl_setting MASK=mask_setting DELAY=delay_time
REG_WRITE_MASK_IMM ADDR=register_addr DATA=data MASK=mask_setting
REG_WRITE_VCTRL_IMM REGULATOR=regulator VCTRL=ctrl_setting MASK=mask_setting DELAY=delay_time
REG_WRITE_MASK_IMM ADDR=register_addr DATA=data MASK=mask_setting
TRIG_MASK 0xFEF6EDC
END
```

These TRIG_SET instructions are used to define the trigger types which initiates each power state sequence. There are a total of 28 TRIG_SET available for each PMIC. TYPE parameter defines the type of trigger as:

High: active high (level sensitive)
Low: active low (level sensitive)
Rise: active high (edge sensitive)
Fall: active low (edge sensitive)

Figure 2-5. Configuration Memory Script Example

As soon as the PMIC state reaches the mission states, it will start reading from the configuration memory until it hits the first END command. Setting up the triggers (1-28) should be the first section of the configuration memory, as well as the first set of trigger configurations. The trigger configurations are read and mapped to an internal lookup table, which contains the starting address associated with each trigger in the configuration memory. If the trigger destination is an FFSM state then the address contains the fixed state value. After the trigger configurations are read and mapped into the SRAM, these triggers control the execution flow of the state transitions. The signal source of each trigger is listed under Table 2-3.

When a trigger or multiple triggers are activated, the PFSM execution engine looks up the starting address associated with the highest priority trigger which is unmasked, and starts executing commands until it hits an END command. The last commands before END statement will generally be the TRIG_MASK command, which direct the PFSM to a new set of unmasked trigger configurations, and the trigger with the highest priority in the new set will be serviced next. Trigger priority is determined by the Trigger ID associated with each trigger. The priority of the trigger decreases as the associated trigger ID increases. As a result the critical error triggers are usually located at the lowest trigger IDs.

The TRIG_SET commands specify if a trigger is immediate or non-immediate. Immediate triggers are serviced immediately, which involves branching from the current sequence of commands to reach a new target destination. The non-immediate triggers are accumulated and serviced in the order of priority through the execution of each given sequence until the END command in reached. Therefore the trigger ID assignment for each trigger can be arranged to produce the desired PFSM behavior.

The TRIG_MASK command determines which triggers are active at the end of each sequence, and is usually placed just before the END instruction. The TRIG_MASK takes a 28 bit input to allow any combination of triggers to be enabled with a single command. Through the definition of the active triggers after each seqeucne execution the TRIG_MASK command can be conceptualized as establishing a power state.

The above sequence of waiting for triggers and executing the sequence associated with an activated trigger is the normal operating condition of the PFSM execution engine when the PMIC is in the MISSION state. The FFSM state machine will take over control from the execution engine any time an event occurs that requires a transition from the MISSION state of the PMIC to a fixed device state.

Table 2-3. PFSM Trigger Selections

Trigger Name	Trigger Source
IMMEDIATE_SHUTDOWN	An error event causes one of the triggers defined in the FSM_TRIG_SEL_1/2 register to activate, and the inteded action for the activated trigger is to *immediate shutdown* the device
MCU_POWER_ERROR	Output failure detection from a regulator which is assigned to the MCU rail group (x_GRP_SEL = '01')
ORDERLY_SHUTDOWN	An event which causes MODERATE_ERR_INT = '1'
FORCE_STANDBY	nPWRON long-press event when NPOWRON_SEL = '01', or ENABLE = '0' when NPOWERON_SEL = '00'
SPMI_WD_BIST_DONE	Completion of SPMI WatchDog BIST
ESM_MCU_ERROR	An event which causes ESM_MCU_FAIL_INT
WD_ERROR	An event which causes WD_INT
SOC_POWER_ERROR	Output failure detection from a regulator which is assigned to the SOC rail group (x_GRP_SEL = '10')
ESM_SOC_ERROR	An event which causes ESM_SOC_FAIL_INT
A	NSLEEP2 and NSLEEP1 = '11'. More information regarding the NSLEEP1 and NSLEEP2 functions can be found under Section 2.4.2.3.2.1
WKUP1	A rising or falling edge detection on a GPIO pin which is configured as WKUP1 or LP_WKUP1
SU_ACTIVE	A valid On-Request detection when STARTUP_DEST = '11'
B	NSLEEP2 and NSLEEP1 = '10'. More information regarding the NSLEEP1 and NSLEEP2 functions can be found under Section 2.4.2.3.2.1
WKUP2	A rising or falling edge detection on a GPIO pin which is configured as WKUP2 or LP_WKUP2
SU_MCU_ONLY	A valid On-Request detection when STARTUP_DEST = '10'
C	NSLEEP2 and NSLEEP1 = '01', More information regarding the NSLEEP1 and NSLEEP2 functions can be found under Section 2.4.2.3.2.1
D	NSLEEP2 and NSLEEP1 = '00'. More information regarding the NSLEEP1 and NSLEEP2 functions can be found under Section 2.4.2.3.2.1
SU_STANDBY	A valid On-Request detection when STARTUP_DEST = '00'
SU_X	A valid On-Request detection when STARTUP_DEST = '01'
WAIT_TIMEOUT	PFSM WAIT command condition timed out.
GPIO1	Input detection at GPIO1 pin
GPIO2	Input detection at GPIO2 pin
GPIO3	Input detection at GPIO3 pin
GPIO4	Input detection at GPIO4 pin
GPIO5	Input detection at GPIO5 pin
GPIO6	Input detection at GPIO6 pin
GPIO7	Input detection at GPIO7 pin
GPIO8	Input detection at GPIO8 pin
GPIO9	Input detection at GPIO9 pin
GPIO10	Input detection at GPIO10 pin
GPIO11	Input detection at GPIO11 pin
I2C_0	Input detection of TRIGGER_I2C_0 bit
I2C_1	Input detection of TRIGGER_I2C_1 bit
I2C_2	Input detection of TRIGGER_I2C_2 bit
I2C_3	Input detection of TRIGGER_I2C_3 bit
I2C_4	Input detection of TRIGGER_I2C_4 bit
I2C_5	Input detection of TRIGGER_I2C_5 bit

Table 2-3. PFSM Trigger Selections (continued)

Trigger Name	Trigger Source
I2C_6	Input detection of TRIGGER_I2C_6 bit
I2C_7	Input detection of TRIGGER_I2C_7 bit
SREG0_0	Input detection of SCRATCH_PAD_REG_0 bit 0
SREG0_1	Input detection of SCRATCH_PAD_REG_0 bit 1
SREG0_2	Input detection of SCRATCH_PAD_REG_0 bit 2
SREG0_3	Input detection of SCRATCH_PAD_REG_0 bit 3
SREG0_4	Input detection of SCRATCH_PAD_REG_0 bit 4
SREG0_5	Input detection of SCRATCH_PAD_REG_0 bit 5
SREG0_6	Input detection of SCRATCH_PAD_REG_0 bit 6
SREG0_7	Input detection of SCRATCH_PAD_REG_0 bit 7
0	Always '0'
1	Always '1'

2.4.2.3 PMSM States and Transitions

The PFSM operation has a high degree of flexibility with the use of the PFSM instructions set. The operation described below pertains to the default configuration of PFSM. This PFSM configuration can be used directly or may serve as the a template for a custom configuration.

2.4.2.3.1 On Requests

ON requests are used to switch on the device, which transitions the device from the STANDBY or the LP_STANDBY to the state specified by STARTUP_DEST[1:0].

After the device arrives at the corresponding STARTUP_DEST[1:0] operation state, the MCU must setup the NSLEEP1 and NSLEEP2 signals accordingly before clearing the STARTUP_INT interrupt. Once the interrupt is cleared, the device will stay or move to the next state corresponding to the NSLEEP signals state assignment as specified in Table 2-7.

Table 2-4 lists the available ON requests.

Table 2-4. ON Requests

EVENT	MASKABLE	COMMENT	DEBOUNCE
nPWRON (pin)	Yes	Edge sensitive	50 ms
ENABLE (pin)	Yes	Level sensitive	8 µs
First Supply Detection (FSD)	Yes	VCCA > VCCA_UV and FSD unmasked	N/A
RTC ALARM Interrupt	Yes		N/A
RTC TIMER Interrupt	Yes		N/A
WKUP1 or WKUP2 Detection	Yes	Edge sensitive	8 µs
LP_WKUP1 or LP_WKUP2 Detection	Yes	Edge sensitive	N/A
Recovery from Immediate and Orderly Shutdown	Yes	Recover from system errors which caused immediate or orderly shut down of the device	N/A

If one of the events listed in Table 2-4 occurs, then the event powers on the device unless one of the gating conditions listed in Table 2-5 is present.

Table 2-5. ON Requests Gating Conditions

EVENT	MASKABLE	COMMENT
VCCA_OVP (event)	No	VCCA_SENSE > VCCA_OVP, VSYS_DEAD_LOCK_EN = 1
VCCA_UVLO (event)	No	VCCA < VCCA_UVLO
VINT_OVP (event)	No	LDOVINT > 1.98 V

Table 2-5. ON Requests Gating Conditions (continued)

EVENT	MASKABLE	COMMENT
VINT_UVLO (event)	No	LDOVINT < 1.62 V
TSD (event)	No	Device stays in SAFE RECOVERY until temperature decreases below TWARN level

The NPWRON_SEL NVM register bit determines whether the nPWRON/ENABLE pin should be treated as a power on press button or a level sensitive enable switch. When this pin is configured as the nPWRON button, a short button press detection will be latched internally as a device enable signal until the NPWRON_START_INT is cleared, or a long press key event is detected. The short button press detection occurs when an falling edge is detected at the nPWRON pin.

The pin is a level sensitive pin when it is configured as an ENABLE pin, and an assertion will enable the device until the pin is released.

2.4.2.3.2 Off Requests

An OFF request is used to orderly switch off the device. OFF requests initiate transition from any other mission state to the STANDBY state or the LP_STANDBY state depending on the setting of the LP_STANDBY_SEL bit. Table 2-6 lists the conditions to generate the OFF requests and the corresponding destination state.

Table 2-6. OFF Requests

EVENT	DEBOUNCE	LP_STANDBY_SEL BIT SETTING	DESTINATION STATE
nPWRON (pin) (long press key event)	8 s	LP_STANDBY_SEL = 0	STANDBY
		LP_STANDBY_SEL = 1	LP_STANDBY
ENABLE (pin)	8 µs	LP_STANDBY_SEL = 0	STANDBY
		LP_STANDBY_SEL = 1	LP_STANDBY
I2C_TRIGGER_0	NA	LP_STANDBY_SEL = 0	STANDBY
		LP_STANDBY_SEL = 1	LP_STANDBY

The long press key event occurs when the nPWRON pin stays low for longer than t_{LPK_TIME} while the device is in a mission state.

Using the I2C_TRIGGER_0 bit as the OFF request will enable the device to wake up from the STANDBY or the LP_STANDBY states through the detection of LP_WKUPn/WKUPn pins, as well as RTC alarm or timer interrupts. To enable this feature, the device must set the I2C_TRIGGER_0 bit to '1' while the NSLEEPn signals are masked, and the ON request (initialized by the nPWRON or ENABLE pins) remains active.

2.4.2.3.2.1 NSLEEP1 and NSLEEP2 Functions

The SLEEP requests are activated through the assertion of nSLEEP1 or nSLEEP2 pins, which are the secondary functions of the 11 GPIO pins and can be selected through GPIO configuration using the GPIOx_SEL register bits. If the nSLEEP1 or nSLEEP2 pins are not available, the NSLEEP1B and NSLEEP2B register bits can be configured in place for their functions. The input of nSLEEP1 pin and the state of the NSLEEP1B register bit are combined to create the NSLEEP1 signal through an *OR* function. Similarly for the input of the nSLEEP2 pin and the NSLEEP2B register bit as they are combined to create the NSLEEP2 signal.

A 1 → 0 logic level transition of the NSLEEP signal generates a sleep request, while a 0 → 1 logic level transition reverses the sleep request in the example PFSM from Figure 2-4. When a NSLEEPn signal transitions from 1 → 0, it generates a sleep request to go from a higher power state to a lower power state. When the signal transitions from 0 → 1, it reverses the sleep request and returns the device to the higher power state.

The NSLEEPn_MASK bit can be used to mask the sleep request associated with the corresponding NSLEEPn signal. When the NSLEEPn_MASK = 1, the corresponding NSLEEPn signal will be ignored. The combination of the NSLEEPn signals and NSLEEPn_MASK bits creates triggers A/B/C/D to the FSM to control the power state of the device as shown in Table 2-7.

The states of the resources during ACTIVE, SLEEP, and DEEP SLEEP/S2R states are defined in the LDOn_CTRL and BUCKn_CTRL registers. For each resource, a transition to the MCU ONLY or the DEEP

SLEEP/S2R states is controlled by the FSM when the resource is associated to the SLEEP or DEEP SLEEP/S2R states.

Table 2-7 shows the corresponding state assignment based on the state of the NSLEEPn and their corresponding mask signals using the example PFSM from Figure 2-4.

Table 2-7. NSLEEPn Transitions and Mission State Assignments

Current State	NSLEEP1	NSLEEP2	NSLEEP1 MASK	NSLEEP2 MASK	Trigger to FSM	Next State
DEEP SLEEP/S2R	0	0 → 1	0	0	TRIGGER B	MCU ONLY
DEEP SLEEP/S2R	0 → 1	0 → 1	0	0	TRIGGER A	ACTIVE
DEEP SLEEP/S2R	Don't care	0 → 1	1	0	TRIGGER A	ACTIVE
DEEP SLEEP/S2R or MCU ONLY	0 → 1	Don't care	0	1	TRIGGER A	ACTIVE
MCU ONLY	0 → 1	1	0	0	TRIGGER A	ACTIVE
MCU ONLY	0	1 → 0	0	0	TRIGGER D	DEEP SLEEP or S2R
MCU ONLY	Don't care	1 → 0	1	0	TRIGGER D	DEEP SLEEP or S2R
ACTIVE	1 → 0	1	0	0	TRIGGER B	MCU ONLY
ACTIVE	1 → 0	1 → 0	0	0	TRIGGER D	DEEP SLEEP or S2R
ACTIVE	Don't care	1 → 0	1	0	TRIGGER D	DEEP SLEEP or S2R
ACTIVE	1 → 0	Don't care	0	1	TRIGGER B	MCU ONLY

2.4.2.3.2.2 WKUP1 and WKUP2 Functions

The WKUP1 and WKUP2 functions are activated through the edge detection on all GPIO pins. Any one of these GPIO pins when configured as an input pin can be configured to wake up the device by setting GPIOn_SEL bit to select the WKUP1 or WKUP2 functions. In the example PFSM depicted in Figure 2-4, when a GPIO pin is configured as a WKUP1 pin, a rising or falling edge detected at the input of this pin (configurable by the GPIOn_FALL_MASK and the GPIOn_RISE_MASK bits) will wake up the device to the ACTIVE state. Likewise if a GPIO pin is configured as a WKUP2 pin, a detected edge will wake up the device to the MCU ONLY state. If multiple edge detections of WKUP signals occur simultaneous, the device will go to the state in the following priority order:

1. ACTIVE
2. MCU ONLY

When a valid edge is detected at a WKUP pin, an interrupt will be generated by the nINT pin to signal the MCU of the wake-up event, and the GPIOx_INT interrupt bit will be set. The wake request will remain active until the GPIOx_INT bit is cleared by the MCU. While the wake request is executing, the device will ignore any sleep request to go to a lower power state until the corresponding GPIOx_INT interrupt bit is cleared to deactivate the wake request. After the wake request is deactivated, the device will return to the state indicated by the NSLEEP1 and NSLEEP2 signals as shown in Table 2-7.

2.4.2.3.2.3 LP_WKUP Pins for Waking Up from LP STANDBY

The LP_WKUP functions are activated through the edge detection of LP_WKUP pins, configurable as secondary functions of GPIO3 and GPIO4. They are specially designed to wake the device up from the LP STANDBY state when a high speed wake up signal is detected. Similar to the WKUP1 and WKUP2 pins, when GPIO3 or GPIO4 pin is configured as a LP_WKUP1 pin, a rising or falling edge detected at the input of this pin (configurable by the GPIOn_FALL_MASK and the GPIOn_RISE_MASK bits) will wake up the device to the ACTIVE state. Likewise if the pin is configured as a LP_WKUP2 pin, a detected edge will wake up the device to the MCU ONLY state. If multiple edge detections of LP_WKUP signals occur simultaneous, the device will go to the state in the following priority order:

1. ACTIVE
2. MCU ONLY

The TPS6594-Q1 device supports limited CAN Wake-up capability through the LP_WKUP1/2 pins. When an input signal (without deglitch) with logic level transition from high-to-low or low-to-high with a minimum pulse width of $t_{WK\_PW\_MIN}$ is detection on the assigned LP_WKUP1/2 pins, the device will wake up asynchronously and execute the power up sequence. CAN-transceiver RXD- or INH-outputs can be connected to the LP_WKUP pin. If RXD-output is used it is assumed the transceiver RXD-pin IO is powered by the transceiver itself from an external supply when TPS6594-Q1 is in LP_STANDBY state. If INH-signal is used it has to be scaled down to the recommenced GPIO input voltage level specified in the electrical characteristics table.

In this PFSM example, the device can wake up from the LP_STANDBY state through the detection of LP_WKUP pins only if it enters the LP_STANDBY state through the TRIGGER_I2C_0 OFF request while the NSLEEPn signals are masked, and the On request initialized by the nPWRON/ENABLE pin remains active. Once a valid wake-up signal is detected at the LP_WKUP pin, it is handled as a WAKE request. An interrupt will be generated by the nINT pin to signal the MCU of the wake-up event, and the corresponding GPIOx_INT interrupt bit will be set. The wake request will remain active until the interrupt bit is cleared by the MCU. After the wake request is deactivated, the device will return to the state indicated by the NSLEEP1 and NSLEEP2 signals as shown in Table 2-7.

Figure 2-6 illustrates the valid wake-up signal at the LP_WKUP1/2 pins, and the generation and clearing of the internal wake-up signal.

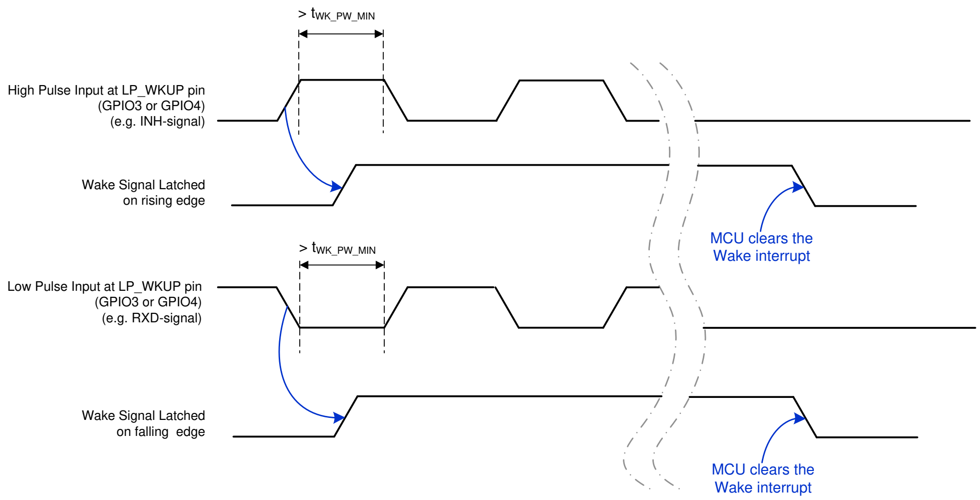


Figure 2-6. CAN Wake-up Timing Diagram

2.4.3 Error Handling Operations

The FSM engine of the TPS6594-Q1 device is designed to handle the following types of errors throughout the operation:

- Power Rail Output Error
- Boot BIST Error
- Runtime BIST Error
- Catastrophic Error
- Watchdog Error
- Error Signal Monitor (ESM) Error
- Warnings

2.4.3.1 Power Rail Output Error

A power rail output error occurs when an error condition is detected from the output rails of the device, which are used to power the attached MCU or SoC. These errors include the following:

- Rails not reaching or maintaining within the power good voltage level threshold.
- A short condition that is detected at a regulator output.
- The load current that exceedes the forward current limit.

The BUCKn_GRP_SEL, LDOn_GRP_SEL, and VCCA_GRP_SEL registers are used to configure the rail group for all of the Bucks, LDOs, and the voltage monitors which are available for external rails. The selectable rail groups are MCU rail group, SoC rail group, or other rail group. The TPS6594-Q1 device is designed to react differently when an error is detected from a power resource assigned to the different rail groups.

The SOC_RAIL_TRIG[1:0], MCU_RAIL_TRIG[1:0], and OTHER_RAIL_TRIG[1:0] registers are used as the *Immediate Shutdown Trigger Mask*, *Orderly Shutdown Trigger Mask*, *MCU Power Error Trigger Mask*, or the *SoC Power Error Trigger Mask*. The settings of these register bits determine the error handling sequence which the assigned groups of rails should take in an event of output error. The PFSM engine can be configured to execute the appropriate error handling sequence for the following error handling sequence options: *immediate shutdown*, *orderly shutdown*, *MCU power error*, or *SOC power error*. For example, if an *immediate shutdown* sequence is assigned to the MCU rail group through the MCU_RAIL_TRIG[1:0], any failure detected in this group of rails will cause the IMMEDIATE_SHUTDOWN trigger to be executed. This trigger is expected to start the immediate shutdown sequence and cause the device to enter the SAFE RECOVERY state. The device will immediately reset both the attached MCU and SoC by de-asserting the nRSTOUT and nRSTOUT_SoC (GPO1) pins. All of the power resources assigned to both the MCU and the SOC will be shutdown immediately without sequencing order. The nINT pin will signal a MCU_PWR_ERR_INT interrupt event has occurred and the EN_DRV pin will be forced low. If the error is recoverable within the recovery time interval, the device will increment the recovery count, return to INIT state and re-attempt the power up sequence if the recovery count has not exceeded the counter threshold. If the recovery count has already exceeded the threshold, the device will stay in the SAFE RECOVERY state until VCCA voltage is below the VCCA_UVLO threshold and the device is power cycled.

The power resources assigned to the SoC rail group are typically assigned to the SOC power error handling sequence. When a power resource in this group is detected, the PFSM will typically cause the device to execute the shutdown of all the resources assigned to the SoC rail group, and the device will enter the MCU ONLY state. The device will immediately reset the attached SoC by toggling the nRSTOUT_SoC (GPO1) pin. The reset output to the MCU and the resources assigned to the MCU rail group will remain unchanged. The EN_DRV pin will also remain unchanged, and the nINT pin will signal a SOC_PWR_ERR_INT interrupt event has occurred. To recover from the MCU_ONLY state after a SOC power error, the MCU software must set NSLEEP1 signal to '0' while NSLEEP2 signal remains '1'. This action signals TPS6594-Q1 that MCU has acknowleged the SOC power error, and is ready to return to normal operation. MCU can then set the NSLEEP1 signal back to '1' for the device to return to ACTIVE state and reattemp the SoC power up.

2.4.3.2 Boost BIST Error

Boot BIST error occurs when the device is not able to pass the BOOT BIST during device power up. Every failure of the BOOT BIST attempt will cause the recovery count to increment as the device enters the SAFE RECOVERY state. If the count value is smaller than the counter threshold, the device will attempt to enter the INIT state again and re-attempt the BOOT BIST until the recovery count reaches the maximum threshold. When this occurs the device will stay in SAFE RECOVERY state until VCCA voltage is below the VCCA_UVLO threshold and the device is power cycled.

2.4.3.3 Runtime BIST Error

Runtime BIST error occurs when the device is not able to pass the Runtime BIST while the device is in an operation state. This error creates an immediate shutdown condition, which will cause the device to execute the immediate shutdown sequence and enter the SAFE RECOVERY state. The device will immediately reset both the attached MCU and SoC by de-asserting the nRSTOUT and nRSTOUT_SoC (GPO1 or GPIO11)) pins. All of the power resources assigned to both the MCU and the SOC will be immediately shutdown. The EN_DRV pin will be forced low, and the nINT pin will be de-asserted to signal an interrupt event has occurred.

2.4.3.4 Catastrophic Error

Catastrophic errors are errors that affect multiple power resources such as errors detected in supply voltage, LDOVINT supply for control logic, clocks monitors, as well as device temperature passing the thermal shutdown threshold, or error detected in the SPMI communication network. These error are grouped as the severe errors. By setting the SEVERE_ERR_TRIG[1:0] to creates an immediate or orderly shutdown condition, the PFSM will execute the corresponding sequence for the IMMEDIATE_SHUTDOWN trigger or the ORDERLY_SHUTDOWN trigger and enter the SAFE RECOVERY state. The device will reset both the attached MCU and SoC by de-asserting the nRSTOUT and nRSTOUT_SoC (GPO1 or GPIO11) pins. All of the power resources assigned to both the MCU and the SOC will be shutdown. The nINT pin will be de-asserting to signal an interrupt event has occurred , and the EN_DRV pin will be forced low.

2.4.3.5 Watchdog (WDOG) Error

Watch (WDOG) errors detection mechanisms are described in detail under Section 2.4.3.5.

2.4.3.6 Error Signal (ESM) Error

There are two Error Signal Monitors (ESM) available for the TPS6594-Q1 device, one designed to detect and handle the error signals received from the attached SoC, while the other one for the attached MCU. The error detection mechanisms for both monitors are described in detail under Section 2.4.3.6.

2.4.3.7 Warnings

Warning are non-catastrophic errors. When such an error occurs while the device is in the operating states, the device detects the error and handles the error through the interrupt handler. These are errors such as thermal warnings, I2C, or SPI communication errors, or power resource over current limit detection while the output voltage still maintains within the power good threshold. When these errors occur, the PFSM will pull the nINT pin low to signal an interrupt event has occurred. The device will remain in the operation state and no changes will be applied to the state of the EN_DRV pin, the power resources, nor the reset outputs.

2.4.4 Device Startup Timing

Figure 2-7 shows the timing diagram of the TPS6594-Q1 after the first supply detection.

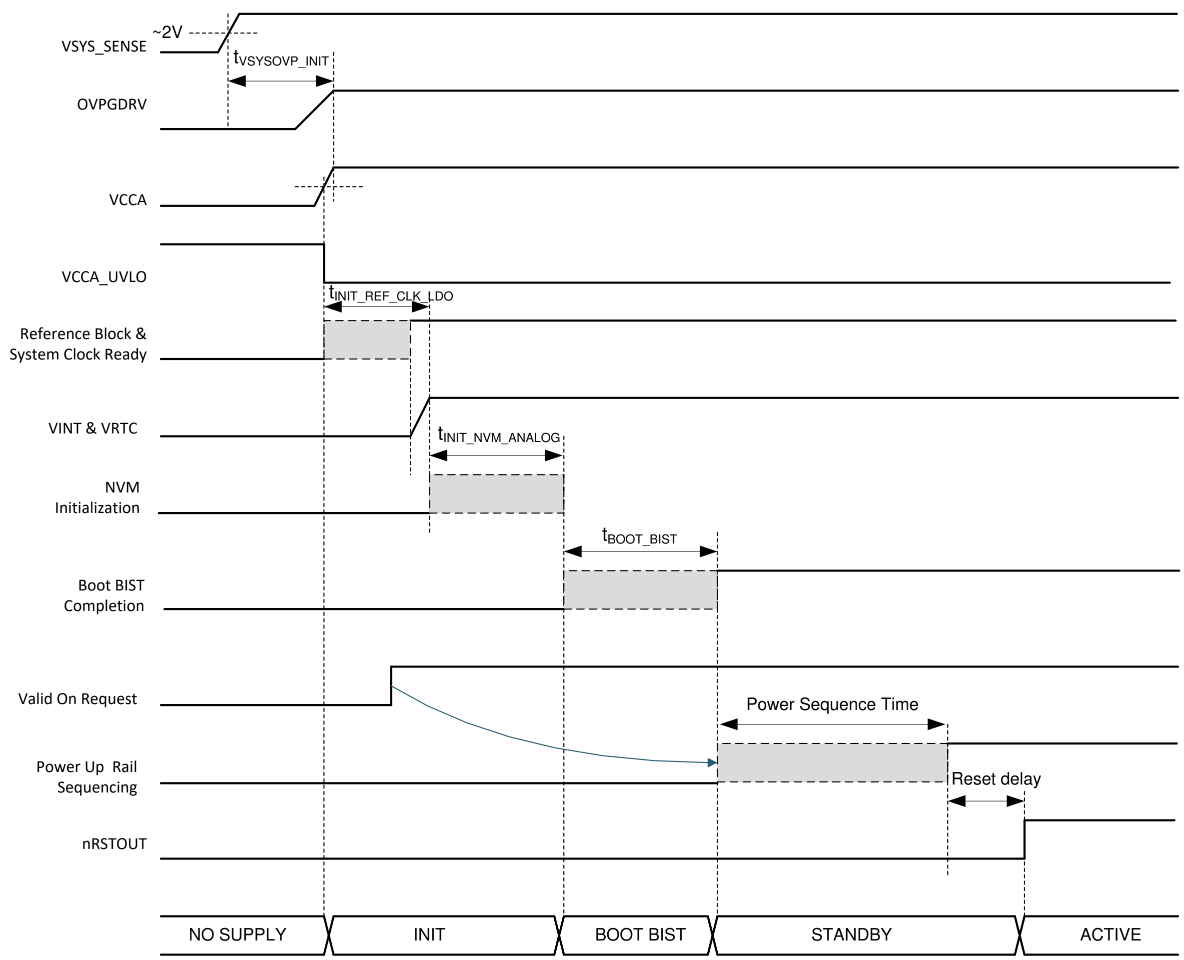


Figure 2-7. Device Startup Timing Diagram

$t_{VSYSOVP\_INIT}$ is the time between VSYS detection and when the VSYS Over Voltage Protection Module is in operation and the external protection FET connects the VSYS_SENSE to VCCA and the PVINx pins.

$t_{INIT\_REFCLK\_LDO}$ is the start up time for the reference block. $t_{INIT\_NVM\_ANALOG}$ is the time for the device to load the default values of the NVM configurable registers from the NVM memory, and the start up time for the analog circuits in the device. Both $t_{INIT\_REFCLK\_LDO}$ and $t_{INIT\_NVM\_ANALOG}$ are defined in the electrical characterization table.

t_{BOOT_BIST} is the sum of $t_{ABISTrun}$ and $t_{LBISTrun}$, which are defined in the electrical characterization tables.

The Power Sequence time is the total time for the device to complete the power up sequence. Please refer to Section 2.4.5 for more detail.

The Reset delay time is a configurable wait time for the nRSTOUT and the nRSTOUT_SoC release after the power up sequence is completed.

2.4.5 Power Sequences

A power sequence is an automatic preconfigured sequence the TPS6594-Q1 device applies to its resources, which include the states of the BUCKs, LDOs, 32-kHz clock, and the GPIO output signals. For a detailed description of the GPIOs signals, please refer to Section 2.9.

Figure 2-8 shows an example of a power up transition followed by a power down transition. The power up sequence is triggered through a valid on request, and the power down sequence is trigger by a valid off request. The resources controlled (for this example) are: BUCK3, LDO1, BUCK2, LDO2, GPIO1, LDO4, and LDO3. The time between each resource enable and disable (TinstX) is also part of the preconfigured sequence definition

When a resource is not assigned to any power sequence, it remains in off mode. The MCU can enable and configure this resource independently when the power sequence completes.

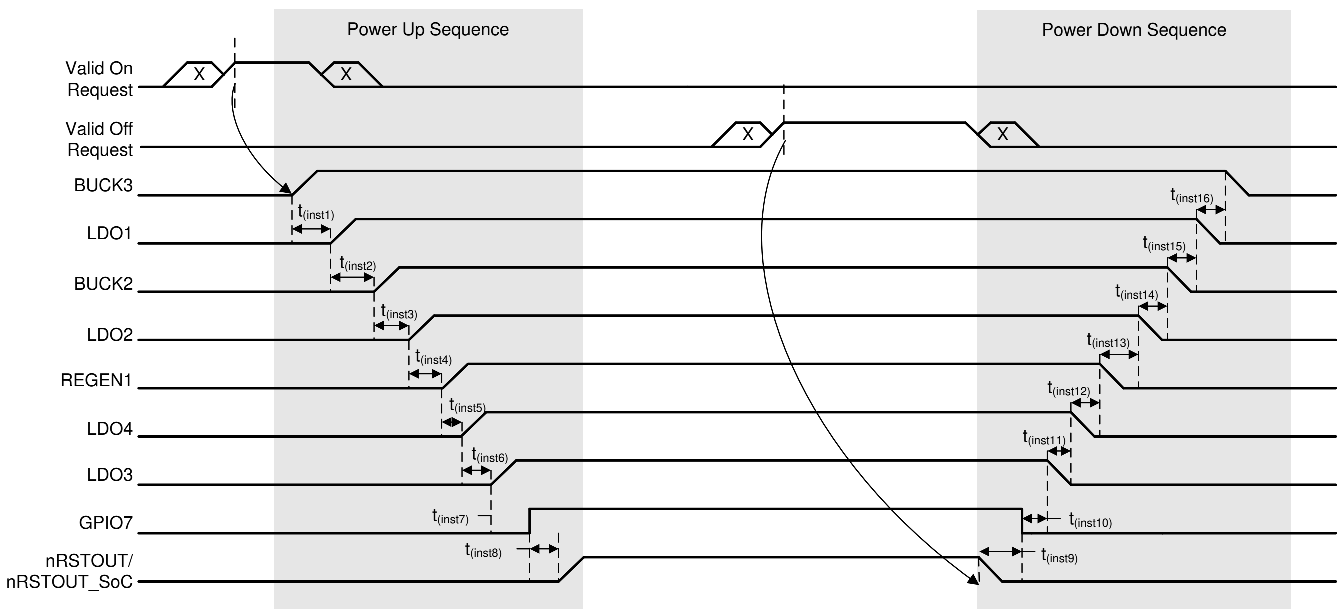


Figure 2-8. Power Sequence Example

As the power sequences of the TPS6594-Q1 device are defined according to the processor requirements, the total time for the completion of the power sequence will vary across various system definitions.

2.4.6 First Supply Detection

The TPS6594-Q1 device can be configured to automatic start up from a first supply-detection (FSD) event detection. This feature is enabled by setting the FSD_MASK register bit to '0', and setting the NPWRON_SEL[1:0] registers bits to '10' or '11' to mask the functionality of nPWRON/ENABLE pin. When the device is powered up from the NO SUPPLY state, the FSD detection is validated after the NVM default for this feature is loaded into the device memory.

When the FSD feature is enabled, the PMIC powers up from the NO SUPPLY state immediately to an operation state configured by STARTUP_DEST[1:0] bits when VCCA > VCCA_UV, while VCCA_UV gating is performed only when VCCA voltage monitoring is enabled (VCCA_VMON_EN = 1). After the device arrives the corresponding STARTUP_DEST[1:0] operation state, the MCU must setup the NSLEEP1 and NSLEEP2 signals accordingly before clearing the FSD_INT interrupt. Once the interrupt is cleared, the device will stay or move to the destination state according to the state of the NSLEEP1/2 signals as specified in Table 2-7.

2.4.7 Register Power Domains and Reset Levels

The TPS6594-Q1 registers are defined by the following categories:

- LDOVINT registers
- LDOVRTC registers

LDOVINT registers The LDOVINT registers are powered by the internal LDOVINT, and retain their values until the device enters LP_STANDBY state or the BACKUP state after the device was fully powered up and in operation. When this occurs LDOVINT is powered off, and the content of all LDOVINT registers will be lost, including the VSET registers which stores the default output voltage levels for all of the external power rails. As the device re-enters the INIT state from a wake up signal or an On-request, the registers powered by the LDOVINT will be re-written with the default values. All registers in the device except the LDOVRTC registers are powered by LDOVINT.

LDOVRTC registers The LDOVRTC registers retains their values until a Power-On-Reset (POR) occurs. POR occurs when the device lost supply power and enters the NO SUPPLY state. When this occurs LDOVRTC is powered off, and the content of all LDOVRTC registers will be lost.

Following are the LDOVRTC registers:

- All RTC registers
- RTC and Crystal Oscillator bits
- Status registers for the following events: TSD and RTC reset
- Control registers for PWRON/ENABLE, GPIO3, and GPIO4 pins (for wake signal monitor during LP_STANDBY state)
- Following interrupt registers:
  - FSD_INT
  - RECOV_CNT_INT
  - TSD_ORD_INT
  - TSD_IMM_INT
  - PFSM_ERR_INT
  - VCCA_OVP_INT
  - ESM_MCU_RST_INT
  - ESM_SOC_RST_INT
  - WD_RST_INT
  - WD_LONGWIN_TIMEOUT_INT
  - NPWRON_LONG_INT

2.5 Power Resources (Bucks and LDOs)

The power resources provided by the TPS65941-Q1 device include inductor-based bucks and linearLDOs. These supply resources provide the required power to the external processor cores, external components, and to modules embedded in the device. The supply of the bucks, the PVIN_Bx pins, must connect to the VCCA pin externally. The supply of the LDOs, the PVIN_LDOx pins, may connect to theVCCA pin or a buck output which is at a lower voltage level than the VCCA.

The voltage output of each power resources are continuously monitored by a dedicated analog monitor on an independent reference voltage domain. An un-used regulator can also be used as a voltage monitor foran external rail by connected the external rail to the VOUT_Bn or the VOUT_LDOn pin. A residual voltage checking option is also available for each power resource to ensure the output voltage has dropped below150 mV before it can be powered up again.

Table 2-8 lists the power resources provided by the TPS65941-Q1 device.

Table 2-8. Power Resources

RESOURCE	TYPE	VOLTAGE	CURRENT CAPABILITY	COMMENTS
BUCK1, BUCK2,BUCK3	BUCK	0.3 to 0.6 V, 20-mV steps 0.6 to 1.1 V, 5-mV steps 1.1 to 1.66 V, 10-mV steps 1.66 to 3.34 V, 20-mV steps (ginle-phase mode only)	3.5 A	Can be configured in multi-phase mode or stand-alone in single-phase mode.
BUCK4	BUCK	0.3 to 0.6 V, 20-mV steps 0.6 to 1.1 V, 5-mV steps 1.1 to 1.66 V, 10-mV steps 1.66 to 3.34 V, 20-mV steps (ginle-phase mode only)	4 A in single-phase mode 3.5 A in multi-phase mode	Can be configured in multi-phase mode or stand-alone in single-phase mode.
BUCK5	BUCK	0.3 to 0.6 V, 20-mV steps 0.6 to 1.1 V, 5-mV steps 1.1 to 1.66 V, 10-mV steps 1.66 to 3.34 V, 20-mV steps (ginle-phase mode only)	2 A	Only in single-phase mode.
LDO1, LDO2, LDO3	LDO	0.6 V to 3.3 V, 50-mV steps	500 mA	Bypass mode configurable
LDO4	LDO	1.2 V to 3.3 V, 25-mV steps	300 mA	Low-noise performance

2.5.1 Buck Regulators

2.5.1.1 Overview

The TPS6594-Q1 includes five synchronous buck converters, four of which can be combined in multi-phase configuration. All of the buck converters support the following features:

- Automatic mode control based on the loading (PFM or PWM mode) or Forced-PWM mode operation
- External clock synchronization option to minimize crosstalk
- Optional spread spectrum technique to reduce EMI
- Soft start
- AVS support with configurable slew-rate
- Windowed undervoltage and overvoltage monitora with configurable threshold
- Windowed voltage monitor for external supply when the buck converter is disabled

When the outputs of converters are combined in multi-phase configuration, it also supports the following features:

- Current balancing between the phases of the converter
- Differential voltage sensing from point of the load
- Phase shifted outputs for EMI reduction
- Optional dynamic phase shedding or adding

There are two modes of operation for the converter, depending on the output current required: pulse-width modulation (PWM) and pulse-frequency modulation (PFM). The converter operates in PWM mode at high load currents of approximately 600 mA or higher. Lighter output current loads cause the converter to automatically switch into PFM mode for reduced current consumption. When forced-PWM mode is selected (BUCKn_FPWM = 1), the device avoids pulse skipping and allows easy filtering of the switch noise by external filter components. The drawback of this mode is the higher quiescent current at low output current levels.

When operating in PWM mode the phases of a multi-phase regulator are automatically added or shed based on the load current level. The forced multi-phase mode can be enabled for lower ripple at the output.

A multi-phase synchronous BUCK converter offers several advantages over a single power stage converter. For application processor power delivery, lower ripple on the input and output currents and faster transient response to load steps are the most significant advantages. With the even distribution of the load current in a multi-phase output configuration, the heat generated is greatly reduced for each channel due to the fact that power loss is proportional to the square of current. The physical size of the output inductor shrinks significantly due to this heat reduction.

A block diagram of a single core is shown in Figure 2-9.

Interleaving switching action of the multi-phase converters is shown in Figure 2-10.

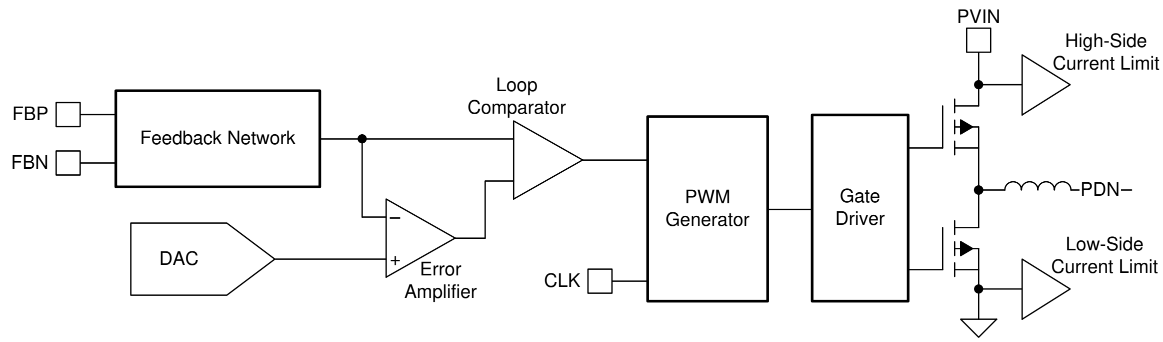


Figure 2-9. Buck Core Block Diagram

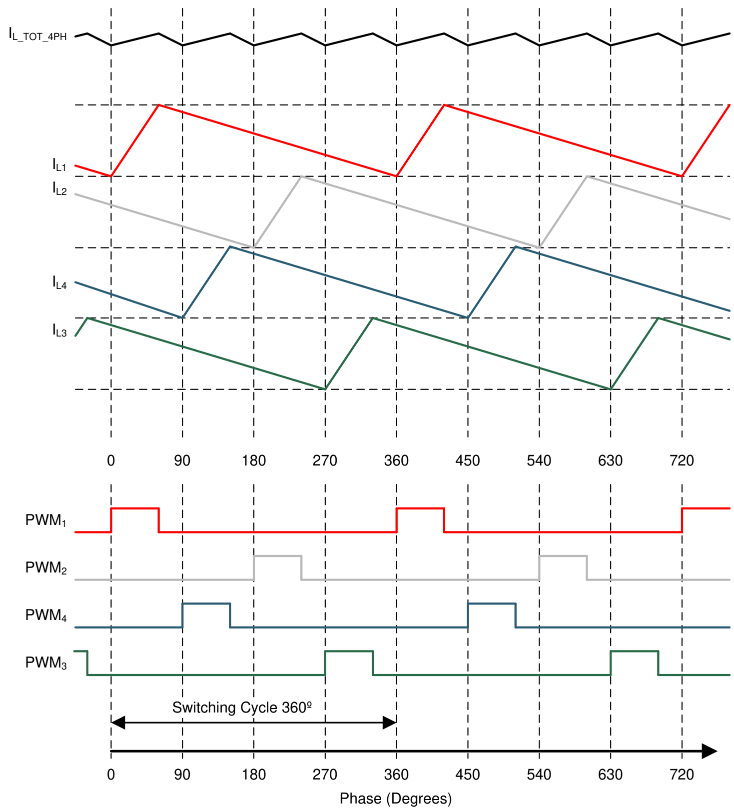


Figure 2-10. Example of PWM Timings, Inductor Current Waveforms, and Total Output Current in 4-Phase Configuration. [1]

2.5.1.2 Multi-Phase Operation and Phase-Adding/Shedding

The 4-phase converters (Buck1, Buck2, Buck3, and Buck4) switches each channel 90° apart under heavy load conditions. As a result, the 4-phase converter has an effective ripple frequency four times greater than the switching frequency of any one phase. In the same way 3-phase converter has an effective ripple frequency three times greater and 2-phase converter has an effective ripple frequency two times greater than the switching frequency of any one phase. However, the parallel operation decreases the efficiency at light load conditions. In order to overcome this operational inefficiency, the TPS6594-Q1 can change the number of active phases to

[1] Graph is not in scale and is for illustrative purposes only.

optimize efficiency for the variations of the load. This is called phase adding or shedding. The concept is shown in Figure 2-11.

The converter can be forced to multi-phase operation by the BUCKn_FPWM_MP bit in BUCKn_CTRL1 register. If the regulator operates in forced multi-phase mode (two phases in the dual-phase configuration, three phases in three-phase configuration and four phases in a four-phase configuration) the forced-PWM operation is automatically used. If the multi-phase operation is not forced, the number of phases are added and shedded automatically to follow the required output current.

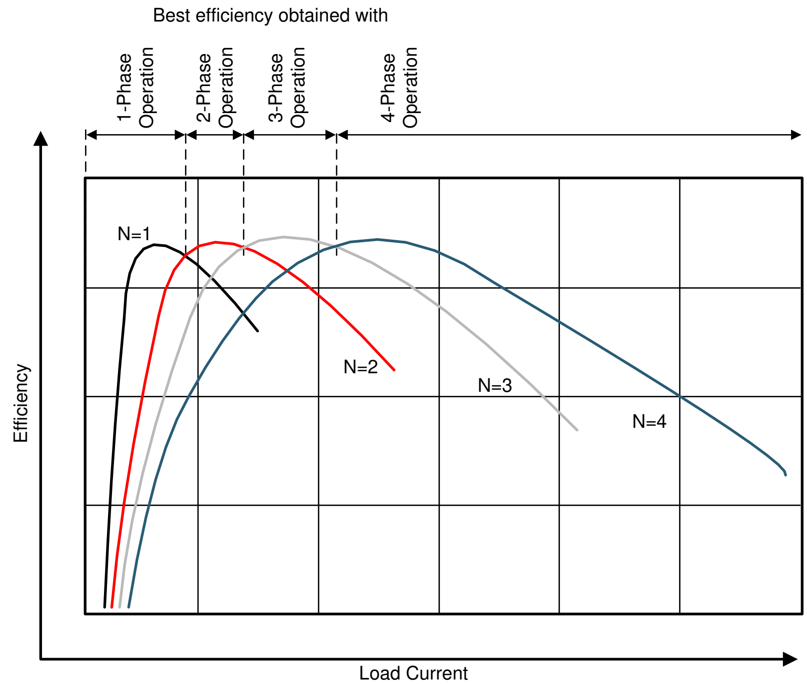


Figure 2-11. Multiphase Buck Converter Efficiency vs Number of Phases (Converters in PWM Mode) [2]

2.5.1.3 Transition Between PWM and PFM Modes

Force PWM mode operation with phase-adding or shedding optimizes efficiency at mid-to-full load. The TPS6594-Q1 converter operates in PWM mode at load current of about 600 mA or higher. At lighter load-current levels the device automatically switches into PFM mode for reduced current consumption when forced-PWM mode is disabled (BUCKn_FPWM = 0). By combining the PFM and the PWM modes a high efficiency is achieved over a wide output-load-current range.

2.5.1.4 Multi-Phase Buck Regulator Configurations

In the multi-phase configuration the control of the multi-phase regulator settings is done using the control registers of the master buck. The following slave registers are ignored:

- BUCKn_CTRL register, except BUCKn_VMON_EN and BUCKn_RV_SEL
- BUCKn_CONF register
- BUCKn_VOUT_1 and BUCKn_VOUT_2 registers
- BUCKn_PG_WINDOW register
- Interrupt bits related to the slave buck, except BUCKn_ILIM_INT, BUCKn_ILIM_MASK and BUCKn_ILIM_STAT

Table 2-9 shows the supported Multi-Phase buck regulator configurations and the assigned master buck in each configuration.

[2] Graph is not in scale and is for illustrative purposes only.

Table 2-9. Master Buck Assignment for Supported Multi-phase Configuration

Supported Multi-Phase Buck Regulator Configuration	Master Buck Assignment
4-Phase: BUCK1 + BUCK2 + BUCK3 + BUCK4	BUCK1
3-Phase: BUCK1 + BUCK2 + BUCK3	BUCK1
2-Phase: BUCK1 + BUCK2	BUCK1
2-Phase: BUCK3 + BUCK4	BUCK3

When the bucks are configured in 3-phase or 4-phase configurations, there are exceptions to the above list of slave registers which are ignored. The configuration registers for the voltage monitor function on Buck3 and Buck4 in a 4-phase configuration, and Buck3 in a 3-phase configuration, are user configurable. This is because the FB_Bn pins of these bucks can be used as voltage monitor pins for external supplies. The following list of registers and register bits for Buck3 and Buck4 can be used to enable and set the target voltage for the external voltage monitoring function under such configuration:

- BUCKn_VMON_EN bit
- BUCKn_RV_SEL bit
- BUCKn_VSEL bit
- BUCKn_SLEW_RATE
- BUCKn_VOUT_1 and BUCKn_VOUT_2 registers
- BUCKn_PG_WINDOW register

Customer is responsible for the values set in these registers when using Buck3 or Buck4 to monitor an external supply under the 3-phase or 4-phase configuration. If the voltage monitor function is not used under such scenario, the FB_Bn pins must be connected to the reference ground, and the BUCKn_VMON_EN and BUCKn_RV_SEL bits must be set to '0'.

2.5.1.5 Spread-Spectrum Mode

The TPS6594-Q1 device supports spread-spectrum modulation of the switching clocks of the buck regulators. Three factory-selectable modulation modes are available. The first mode is modulation from external input clock at the SYNCCLKIN pin. The second mode is modulating the input clock at the SYNCCLKIN pin using the DPLL. The third mode is modulating the internal 20 MHz RC Oscillator clock using the DPLL.

This is a fixed NVM option and changing modulation setting during operation is not supported.

The modulation frequency range is limited by the DPLL bandwidth. The max frequency spread for the input clock to the DPLL is ±18% to secure parametric compliance of the buck output performance.

The internal modulation is disabled by default and can be enabled and configured after power up. Internal modulation is activated by setting the SS_EN control bit. The internal modulation must be disabled (SS_EN = 0) when changing the following parameter:

- SS_DEPTH[1:0] - Spread Spectrum modulation depth

When internal modulation is enabled and configured, it can be disabled by the system MCU during operation. The device transition to different mission states does not impact internal modulation when it is enabled and configured.

2.5.1.6 Adaptive Voltage Scaling (AVS) and Dynamic Voltage Scaling (DVS) Support

An AVS or a DVS voltage value can be configured by the attached MCU after the buck regulator is powered up to the default output voltage selected in register BUCKn_VSET1, which loads its default value from NVM. The purpose of the AVS/DVS voltage is to set the buck output voltage to enable optimal efficiency and performance of the attached SoC.

All of bucks on the TPS6594-Q1 device support AVS and DVS voltage scaling changes. Once the AVS/DVS voltage value is written into the BUCKn_VSET1 or BUCKn_VSET2 register, and the MCU sets the BUCKn_VSEL register to select the AVS/DVS voltage, the output of the buck will maintain at the AVS/DVS voltage level instead of the default voltage from NVM until any one of the following event occurs:

- Error that causes the device to re-initialize itself through a power cycle after reaching the SAFE RECOVERY state

- Error that causes the device to execute warm reset
- MCU configures the device to enter the LP STANDBY state

Figure 2-12 shows the arbitration scheme for loading the buck output level from the AVS register using the BUCKn_VSET control registers.

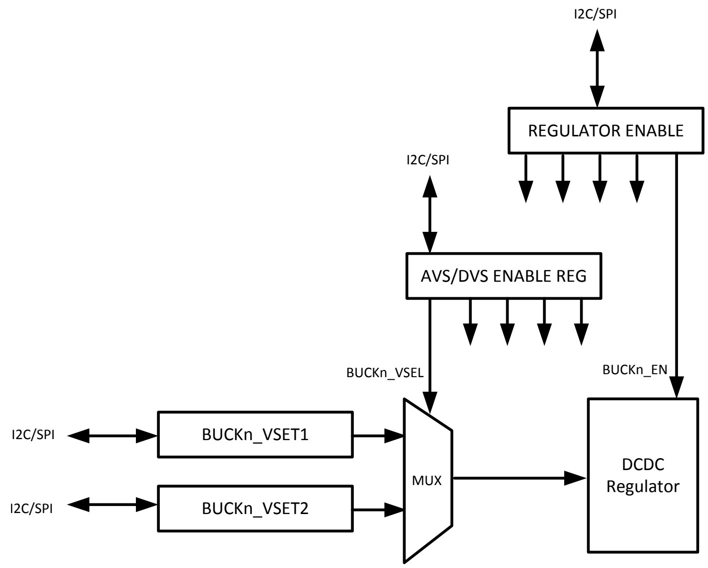


Figure 2-12. AVS/DVS Configuration Register Arbitration Diagram

The OV and UV threshold of the buck output voltage monitor will be updated automatically by the digital control block during the AVS or DVS voltage change. When the output voltage is increased, the OV threshold is updated at the same time the BUCKn_VSETx is updated to the AVS voltage level, while the UV threshold is updated after a delay calculated by Equation 2.

When the output voltage is decreased, the UV threshold is updated at the same time the BUCKn_VSETx is updated to the AVS voltage level, while the OV threshold is updated after a delay calculated by Equation 2.

$$t_{PG\_OV\_UV\_DELAY} = (dV / BUCKn\_SLEW\_RATE) + t_{settle\_Bx} \quad (2)$$

In order to prevent erroneous voltage monitoring, the digital block also temporarily masks the results of the OV and UV monitor from the regulator output when the buck is enabled and the voltage is rising to the BUCKn_VSETx level. The duration of the mask starts from the time the buck is enabled. The buck OV monitor output is masked for a fixed delay time of $t_{PG\_OV\_GATE}$, which is approximately 115 µs – 128 µs. The UV monitor output is masked for the time duration calculated by Equation 3. The 370 µs additional delay time in the formula includes the start-up delay of the buck, the fixed delay after the ramp, and the time for the BIST operation of the OV and UV monitors.

$$t_{PG\_UV\_GATE} = (BUCKn\_VSEL / BUCKn\_SLEW\_RATE) + 370\ \mu s \quad (3)$$

Figure 2-13 and Figure 2-14 are timing diagrams illustrating the voltage change for AVS and DVS enabled bucks and the corresponding OV and UV monitor threshold changes.

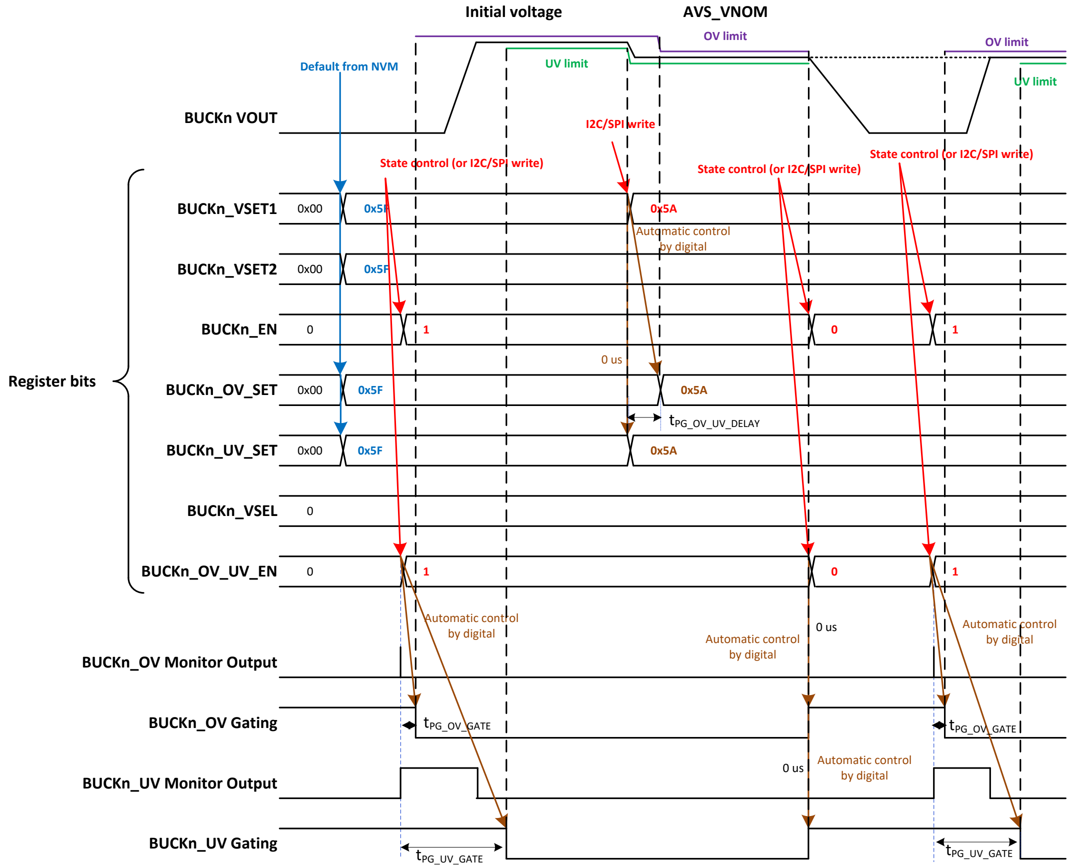


Figure 2-13. AVS Voltage and OV UV Threshold Level Change Timing Diagram

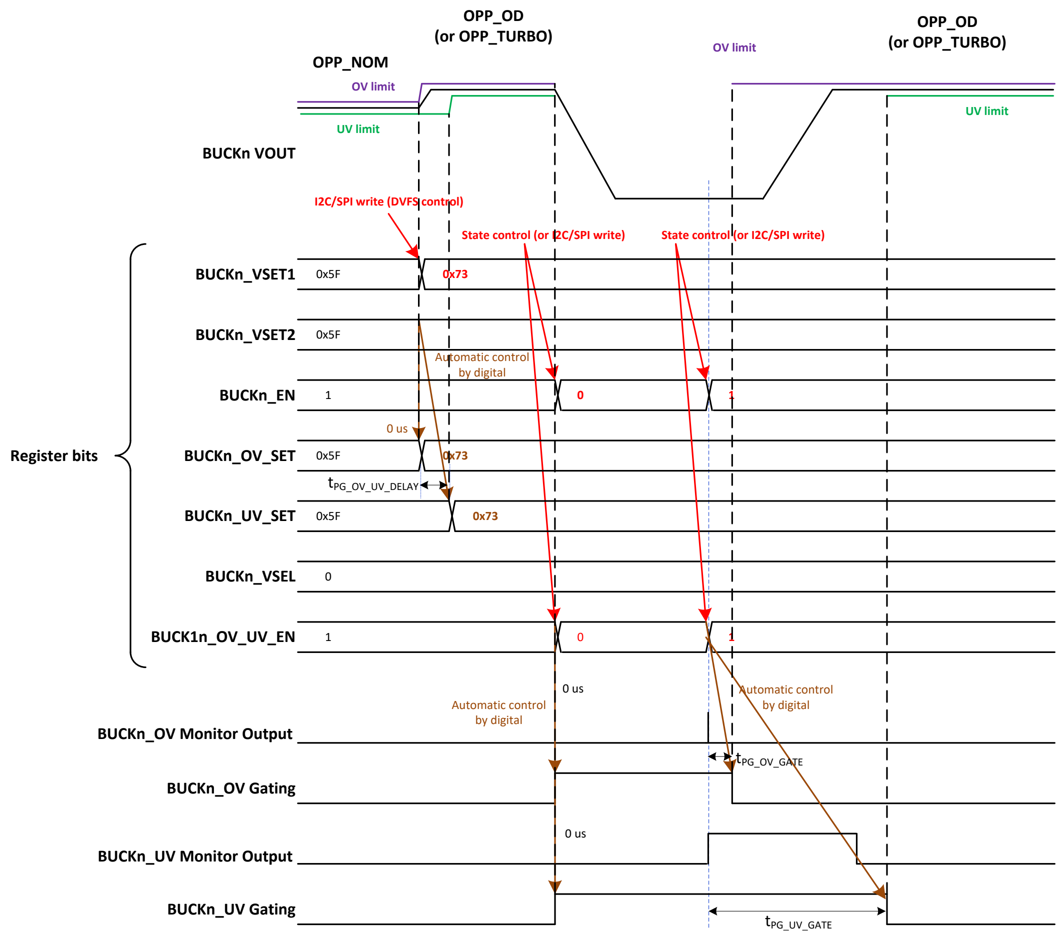


Figure 2-14. DVS Voltage and OV UV Threshold Level Change Timing Diagram

2.5.1.7 Buck Output Voltage Setting

The buck output voltage is selected using the coding shown in Table 2-10.

Table 2-10. Output Voltage Selection for Buck Regulators

BUCKn_ VSETn	Output Voltage [V] 20 mV steps	BUCKn_ VSETn	Output Voltage [V] 5 mV steps	BUCKn_ VSETn	Output Voltage [V] 5 mV steps	BUCKn_ VSETn	Output Voltage [V] 10 mV steps	BUCKn_ VSETn	Output Voltage [V] 20 mV steps	BUCKn_ VSETn	Output Voltage [V] 20 mV steps
0x00	0.3	0x0F	0.6	0x41	0.85	0x73	1.1	0xAB	1.66	0xD6	2.52
0x01	0.32	0x10	0.605	0x42	0.855	0x74	1.11	0xAC	1.68	0xD7	2.54
0x02	0.34	0x11	0.61	0x43	0.86	0x75	1.12	0xAD	1.7	0xD8	2.56
0x03	0.36	0x12	0.615	0x44	0.865	0x76	1.13	0xAE	1.72	0xD9	2.58
0x04	0.38	0x13	0.62	0x45	0.87	0x77	1.14	0xAF	1.74	0xDA	2.6
0x05	0.4	0x14	0.625	0x46	0.875	0x78	1.15	0xB0	1.76	0xDB	2.62
0x06	0.42	0x15	0.63	0x47	0.88	0x79	1.16	0xB1	1.78	0xDC	2.64
0x07	0.44	0x16	0.635	0x48	0.885	0x7A	1.17	0xB2	1.8	0xDD	2.66
0x08	0.46	0x17	0.64	0x49	0.89	0x7B	1.18	0xB3	1.82	0xDE	2.68
0x09	0.48	0x18	0.645	0x4A	0.895	0x7C	1.19	0xB4	1.84	0xDF	2.7
0x0A	0.5	0x19	0.65	0x4B	0.9	0x7D	1.2	0xB5	1.86	0xE0	2.72
0x0B	0.52	0x1A	0.655	0x4C	0.905	0x7E	1.21	0xB6	1.88	0xE1	2.74
0x0C	0.54	0x1B	0.66	0x4D	0.91	0x7F	1.22	0xB7	1.9	0xE2	2.76
0x0D	0.56	0x1C	0.665	0x4E	0.915	0x80	1.23	0xB8	1.92	0xE3	2.78

Table 2-10. Output Voltage Selection for Buck Regulators (continued)

BUCKn_VSETn	Output Voltage [V] 20 mV steps	BUCKn_VSETn	Output Voltage [V] 5 mV steps	BUCKn_VSETn	Output Voltage [V] 5 mV steps	BUCKn_VSETn	Output Voltage [V] 10 mV steps	BUCKn_VSETn	Output Voltage [V] 20 mV steps	BUCKn_VSETn	Output Voltage [V] 20 mV steps
0x0E	0.58	0x1D	0.67	0x4F	0.92	0x81	1.24	0xB9	1.94	0xE4	2.8
		0x1E	0.675	0x50	0.925	0x82	1.25	0xBA	1.96	0xE5	2.82
		0x1F	0.68	0x51	0.93	0x83	1.26	0xBB	1.98	0xE6	2.84
		0x20	0.685	0x52	0.935	0x84	1.27	0xBC	2	0xE7	2.86
		0x21	0.69	0x53	0.94	0x85	1.28	0xBD	2.02	0xE8	2.88
		0x22	0.695	0x54	0.945	0x86	1.29	0xBE	2.04	0xE9	2.9
		0x23	0.7	0x55	0.95	0x87	1.3	0xBF	2.06	0xEA	2.92
		0x24	0.705	0x56	0.955	0x88	1.31	0xC0	2.08	0xEB	2.94
		0x25	0.71	0x57	0.96	0x89	1.32	0xC1	2.1	0xEC	2.96
		0x26	0.715	0x58	0.965	0x8A	1.33	0xC2	2.12	0xED	2.98
		0x27	0.72	0x59	0.97	0x8B	1.34	0xC3	2.14	0xEE	3.0
		0x28	0.725	0x5A	0.975	0x8C	1.35	0xC4	2.16	0xEF	3.02
		0x29	0.73	0x5B	0.98	0x8D	1.36	0xC5	2.18	0xF0	3.04
		0x2A	0.735	0x5C	0.985	0x8E	1.37	0xC6	2.2	0xF1	3.06
		0x2B	0.74	0x5D	0.99	0x8F	1.38	0xC7	2.22	0xF2	3.08
		0x2C	0.745	0x5E	0.995	0x90	1.39	0xC8	2.24	0xF3	3.1
		0x2D	0.75	0x5F	1.0	0x91	1.4	0xC9	2.26	0xF4	3.12
		0x2E	0.755	0x60	1.005	0x92	1.41	0xCA	2.28	0xF5	3.14
		0x2F	0.76	0x61	1.01	0x93	1.42	0xCB	2.3	0xF6	3.16
		0x30	0.765	0x62	1.015	0x94	1.43	0xCC	2.32	0xF7	3.18
		0x31	0.77	0x63	1.02	0x95	1.44	0xCD	2.34	0xF8	3.2
		0x32	0.775	0x64	1.025	0x96	1.45	0xCE	2.36	0xF9	3.22
		0x33	0.78	0x65	1.03	0x97	1.46	0xCF	2.38	0xFA	3.24
		0x34	0.785	0x66	1.035	0x98	1.47	0xD0	2.4	0xFB	3.26
		0x35	0.79	0x67	1.04	0x99	1.48	0xD1	2.42	0xFC	3.28
		0x36	0.795	0x68	1.045	0x9A	1.49	0xD2	2.44	0xFD	3.3
		0x37	0.8	0x69	1.05	0x9B	1.5	0xD3	2.46	0xFE	3.32
		0x38	0.805	0x6A	1.055	0x9C	1.51	0xD4	2.48	0xFF	3.34
		0x39	0.81	0x6B	1.06	0x9D	1.52	0xD5	2.5		
		0x3A	0.815	0x6C	1.065	0x9E	1.53				
		0x3B	0.82	0x6D	1.07	0x9F	1.54				
		0x3C	0.825	0x6E	1.075	0xA0	1.55				
		0x3D	0.83	0x6F	1.08	0xA1	1.56				
		0x3E	0.835	0x70	1.085	0xA2	1.57				
		0x3F	0.84	0x71	1.09	0xA3	1.58				
		0x40	0.845	0x72	1.095	0xA4	1.59				
						0xA5	1.6				
						0xA6	1.61				
						0xA7	1.62				
						0xA8	1.63				
						0xA9	1.64				
						0xAA	1.65				

2.5.2 Sync Clock Functionality

The TPS6594-Q1 device contains a SYNCCLKIN (GPIO10) input to synchronize switching clock of the buck regulator with the external clock. The block diagram of the clocking and PLL module is shown in Figure 2-15. The external clock is selected when the external clock is available, and SEL_EXT_CLK = '1'. The nominal frequency of the external input clock is set by EXT_CLK_FREQ[1:0] bits in the NVM and it can be 1.1 MHz, 2.2 MHz, or 4.4 MHz. The external SYNCCLKIN clock must be inside accuracy limits (–18%/+18%) of the typical input frequency for valid clock detection.

The EXT_CLK_INT interrupt is generated in cases the external clock is expected (SEL_EXT_CLK = 1), but it is not available or the clock frequency is not within the valid range.

The TPS6594-Q1 device can also generate a clock signal, SYNCCLKOUT, for external device use. The SYNCCLKOUT_FREQ_SEL[1:0] selects the frequency of the SYNCCLKOUT. Please note that SYNCCLKOUT_FREQ_SEL[1:0] must stay static while SYNCCLKOUT is used, as changing the output frequency selection may cause glitches on the clock output. The SYNCCLKOUT is available through GPIO8, GPIO9, or GPIO10.

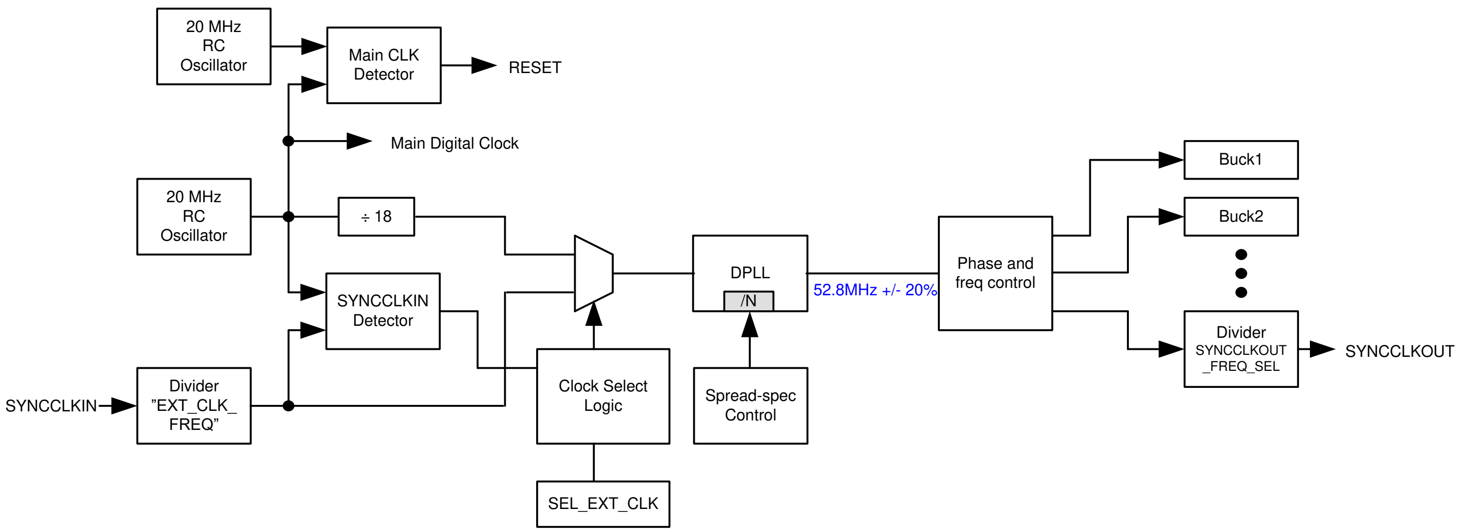


Figure 2-15. Sync Clock and DPLL Module

2.5.3 Low Dropout Regulators (LDOs)

All of the LDO regulators in the TPS6594-Q1 device can be supplied by the system supply or another pre-regulated voltage source which are within the specified VIN range. The PVIN_LDOn voltage level must be equal or less than the VCCA voltage level to ensure proper operation of the LDOs. The default output voltages of all LDOs are loaded from the NVM memory and can be configured by the LDOn_VSET[7:0]. There is no hardware protection to prevent software from selecting an improper output voltage if the minimum level of PVIN_LDOn is lower than the dropout voltage of the LDO regulator in addition to the configured LDO output voltage. In such conditions, the output voltage will droop to near the PVIN_LDOn level.

Note

Writing a *RESERVED* value to the LDOn_VSET[7:0] register bits will trigger a LDOn_OV_INT or LDOn_UV_INT interrupt.

LDO regulators do not have slew rate control for voltage ramp. However, by setting the LDOn_SLOW_RAMP bit to '1' will slow down the ramp up speed of the regulator output voltage to < 3 V/ms.

If an LDO is not needed, it can be used as a voltage monitor for an external rail by connecting the external rail to the VOUT_LDOn pin. The voltage output level to be monitored must be within the PGOOD monitor range of the LDOn_VSET[7:0] of the LDO. If external resistor divider is necessary in this case, the user must take into account of the input impedance at the VOUT_LDOn pin as shown in Figure 2-16, and adjust the resistor values to compensate for the voltage shift.

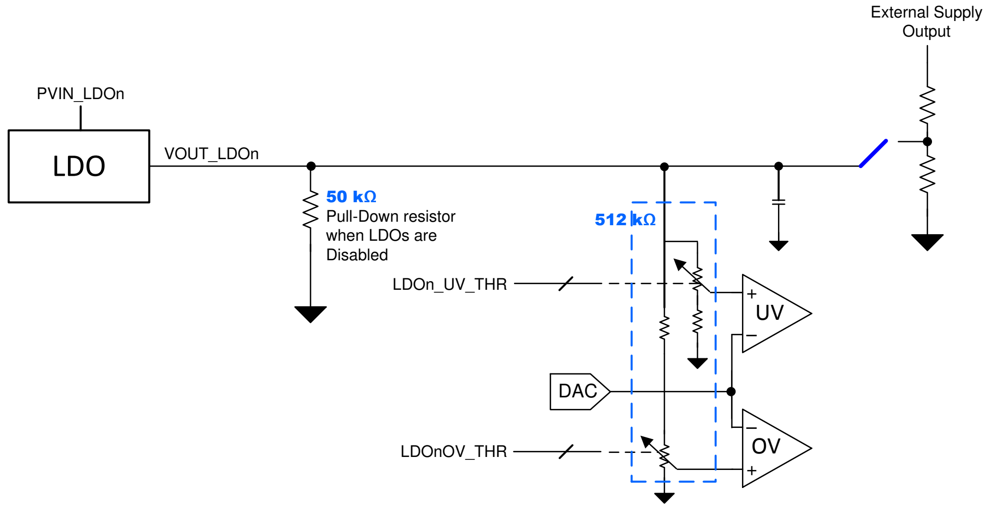


Figure 2-16. Impedance at the VOUT_LDOn pins

2.5.3.1 LDOVINT

The LDOVINT voltage regulator is dedicated to supply the digital and analog functions of the TPS6594-Q1 device which are not required to be always-on and can be turned-off when the device is in low power states. The LDOVINT regulator is automatically enabled and disabled as needed if LP_STANDBY_SEL = '1'. The automatic control optimizes the overall current consumption when the device is in low power LP_STANDBY state.

The LDOVINT is dedicated for internal use only for the TPS6594-Q1 device. It cannot be used to support external loads. The VOUT_LDOVINT pin should only be connected to the output filtering capacitor for the regulator and nothing else.

2.5.3.2 LDOVRTC

The LDOVRTC regulator supplies always-on functions, such as wake-up functions. This power resource is active as soon as a valid VCCA is present. The LDOVRTC is dedicated for internal use only for the TPS6594-Q1 device. It cannot be used to support external loads. The VOUT_LDOVRTC pin should only be connected to the output filtering capacitor for the regulator and nothing else.

This resource runs in normal mode or backup mode. The LDOVRTC regulator functions in normal mode when supplied from the main system power rail and is able to supply the input buffers of GPIO3/4 pins, the digital components, the crystal and the RTC calendar module of the TPS6594-Q1 device. The LDOVRTC regulator remains on in BACKUP state when VCCA is below the VCCA_UVLO level, and the backup power source is above the LDOVRTC_UVLO level.

In BACKUP state, only the 32 kHz crystal and the RTC counter are activated. In the LP STANDBY state, the RTC calendar function will remain active, but the interrupt functions are reduced to maintaining the wake up functions only. In the mission states, the RTC calendar and interrupt functions are fully activated.

Customer has the option to enable the *shelf mode* by setting the LDORTC_DIS bit to 1 while the device is in MISSION state and the I2C bus is in operation, and ramp down VCCA to 0V immediately after the I2C write has completed. This bit will force the device to skip the BACKUP state and enters the NO SUPPLY state under VCCA_UVLO condition. This mode is useful to prevent the continual draining of the back up power source when the 32 KHz crystal and RTC counter functions are no longer needed.

2.5.3.3 LDO1, LDO2, and LDO3

The LDO1, LDO2 and LDO3 regulators can deliver up to 500 mA of current, with a configurable output range of 0.6 V to 3.3 V in 50 mV steps. These 3 LDO regulators also support bypass mode, which allows an input voltage at the PVIN_LDOn to show up at the VOUT_LDOn pin. This feature allows the LDOs to be configured as load switches with power sequencing control. Similar to the buck regulatros mentioned in Section 2.5.1.4, an un-used regulator can also be used as a voltage monitor for an external rail by connected the external rail to the VOUT_LDOn pin.

The bypass capability to connect the input voltage to the output in bypass mode is supported when the input voltage is within the 1.7 V to 3.5 V range. For an SD card I/O supply, this bypass capability also allows the LDO to switch from 3.3 V in bypass mode to 1.8 V in LDO mode, or switch from 1.8 V in LDO mode to 3.3 V in bypass mode.

When changing the LDO output voltage setting, it is important to wait until the LDO has settled on the target voltage from the previous change. The worst case voltage scaling time for LDO1, LDO2, and LDO3 is 63 µs x (7 + the number of 50 mV steps to the new target voltage).

The output voltage for LDO1, LDO2, and LDO3 is selected using the coding shown in Table 2-11

Table 2-11. Output Voltage Selection for LDO1, LDO2, and LDO3

LDOx_VSET	Output Voltage [V]	LDOx_VSET	Output Voltage [V]	LDOx_VSET	Output Voltage [V]	LDOx_VSET	Output Voltage [V]
0x00	Reserved	0x10	1.20	0x20	2.00	0x30	2.80
0x01	Reserved	0x11	1.25	0x21	2.05	0x31	2.85
0x02	Reserved	0x12	1.30	0x22	2.10	0x32	2.90
0x03	Reserved	0x13	1.35	0x23	2.15	0x33	2.95
0x04	0.60	0x14	1.40	0x24	2.20	0x34	3.00
0x05	0.65	0x15	1.45	0x25	2.25	0x35	3.05
0x06	0.70	0x16	1.50	0x26	2.30	0x36	3.10
0x07	0.75	0x17	1.55	0x27	2.35	0x37	3.15
0x08	0.80	0x18	1.60	0x28	2.40	0x38	3.20
0x09	0.85	0x19	1.65	0x29	2.45	0x39	3.25
0x0A	0.90	0x1A	1.70	0x2A	2.50	0x3A	3.30
0x0B	0.95	0x1B	1.75	0x2B	2.55	0x3B	Reserved
0x0C	1.00	0x1C	1.80	0x2C	2.60	0x3C	Reserved
0x0D	1.05	0x1D	1.85	0x2D	2.65	0x3D	Reserved
0x0E	1.10	0x1E	1.90	0x2E	2.70	0x3E	Reserved
0x0F	1.15	0x1F	1.95	0x2F	2.75	0x3F	Reserved

2.5.3.4 Low-Noise LDO (LDO4)

The LDO4 regulator can deliver up to 300 mA of current, with a configurable output range of 1.2 V to 3.3 V in 25 mV steps. This LDO is specifically designed to supply noise sensitive circuits. This supply can be used to power circuits such as PLLs, oscillators, or other analog modules that require low noise on the supply. LDO4 does not support bypass mode. However it can also be used as a external voltage monitor if its regulator function is not needed.

The output voltage for LDO4 is elected using the coding shown in Table 2-12

Table 2-12. Output Voltage Selection for LDO4

LDO4_VSET	Output Voltage [V]	LDO4_VSET	Output Voltage [V]	LDO4_VSET	Output Voltage [V]	LDO4_VSET	Output Voltage [V]
0x00	Reserved	0x20	1.200	0x40	2.000	0x60	2.800
0x01	Reserved	0x21	1.225	0x41	2.025	0x61	2.825
0x02	Reserved	0x22	1.250	0x42	2.050	0x62	2.850
0x03	Reserved	0x23	1.275	0x43	2.075	0x63	2.875
0x04	Reserved	0x24	1.300	0x44	2.100	0x64	2.900
0x05	Reserved	0x25	1.325	0x45	2.125	0x65	2.925
0x06	Reserved	0x26	1.350	0x46	2.150	0x66	2.950
0x07	Reserved	0x27	1.375	0x47	2.175	0x67	2.975
0x08	Reserved	0x28	1.400	0x48	2.200	0x68	3.000
0x09	Reserved	0x29	1.425	0x49	2.225	0x69	3.025
0x0A	Reserved	0x2A	1.450	0x4A	2.250	0x6A	3.050

Table 2-12. Output Voltage Selection for LDO4 (continued)

LDO4_VSET	Output Voltage [V]	LDO4_VSET	Output Voltage [V]	LDO4_VSET	Output Voltage [V]	LDO4_VSET	Output Voltage [V]
0x0B	Reserved	0x2B	1.475	0x4B	2.275	0x6B	3.075
0x0C	Reserved	0x2C	1.500	0x4C	2.300	0x6C	3.100
0x0D	Reserved	0x2D	1.525	0x4D	2.325	0x6D	3.125
0x0E	Reserved	0x2E	1.550	0x4E	2.350	0x6E	3.150
0x0F	Reserved	0x2F	1.575	0x4F	2.375	0x6F	3.175
0x10	Reserved	0x30	1.600	0x50	2.400	0x70	3.200
0x11	Reserved	0x31	1.625	0x51	2.425	0x71	3.225
0x12	Reserved	0x32	1.650	0x52	2.450	0x72	3.250
0x13	Reserved	0x33	1.675	0x53	2.475	0x73	3.275
0x14	Reserved	0x34	1.700	0x54	2.500	0x74	3.300
0x15	Reserved	0x35	1.725	0x55	2.525	0x75	Reserved
0x16	Reserved	0x36	1.750	0x56	2.550	0x76	Reserved
0x17	Reserved	0x37	1.775	0x57	2.575	0x77	Reserved
0x18	Reserved	0x38	1.800	0x58	2.600	0x78	Reserved
0x19	Reserved	0x39	1.825	0x59	2.625	0x79	Reserved
0x1A	Reserved	0x3A	1.850	0x5A	2.650	0x7A	Reserved
0x1B	Reserved	0x3B	1.875	0x5B	2.675	0x7B	Reserved
0x1C	Reserved	0x3C	1.900	0x5C	2.700	0x7C	Reserved
0x1D	Reserved	0x3D	1.925	0x5D	2.725	0x7D	Reserved
0x1E	Reserved	0x3E	1.950	0x5E	2.750	0x7E	Reserved
0x1F	Reserved	0x3F	1.975	0x5F	2.775	0x7F	Reserved

2.6 Backup Supply Power-Path

LDOVRTC is supplied from either the VBACKUP (backup supply from either coin-cell or super-cap) input or VCCA. The power-path is designed to prioritize VCCA to maximize the life of the backup supply.

When VCCA drops below the VCCA_UVLO threshold, the device shuts down all rails except LDOVRTC and enters BACKUP mode. At this point the Backup Supply Power-Path switches to the VBACKUP as the input of LDOVRTC. When the voltage of VCCA returns to level above the VCCA_UVLO threshold level, the power-path switches the input of LDOVRTC back to VCCA.

When both the VCCA voltage drop below the VCCA_UVLO threshold, and the VBACKUP voltage drops below the RTC_LDO_UVLO threshold, LDOVRTC is turned OFF and the digital core is reset, forcing the device into NO SUPPLY state.

Note that backup supply is not required for the device to operate. The device will skip BACKUP state if the VBACKUP pin is grounded.

2.7 Residual Voltage Checking

The residual voltage (RV) checking feature ensures the voltage level at the buck or LDO regulators is below $V_{TH\_SC\_RV}$ before it can be ramped up the target output voltage. If BUCKn/LDOn_RV_SEL=1 by default, residual voltage is also checked before the device enters BOOT_BIST state. If the residual voltage at the output of the regulators is greater than $V_{TH\_SC\_RV}$, the device waits until voltage goes below $V_{TH\_SC\_RV}$ before starting BOOT_BIST or the voltage ramp up.

This feature is enabled by the BUCKn_VMON_EN and BUCKn_RV_SEL bits for each buck regulator, and by the LDOn_VMON_EN and LDOn_RV_SEL bits for each LDO regulator. When this feature is enabled, the VMON of the corresponding regulator will remain on after the regulator is disabled, and remain on for the RV_TIMEOUT period. After the RV Timeout period elapses the output voltage of the regulator will be compared to the short circuit (SC) threshold of $V_{TH\_SC\_RV}$, and assert the corresponding BUCKn_SC_INT or LDOn_SC_INT interrupt bits if the residual voltage is still higher than the threshold voltage. The RV timeout period for the BUCK regulators is automatically calculated by the digital controller inside the device by Equation 4. The RV timeout period of the LDO regulator is configured by the LDOn_RV_TIMEOUT[3:0].

$$t_{BUCK\_RV\_TIMEOUT} = BUCKn\_VSET / BUCKn\_SLEW\_RATE + 100\ \mu s \quad (4)$$

The residual voltage check can also be performed on external rails when they are connected to unused LDO regulator outputs.

Figure 2-17 shows the timing diagram of the residual voltage checking operation which results in pass or fail results.

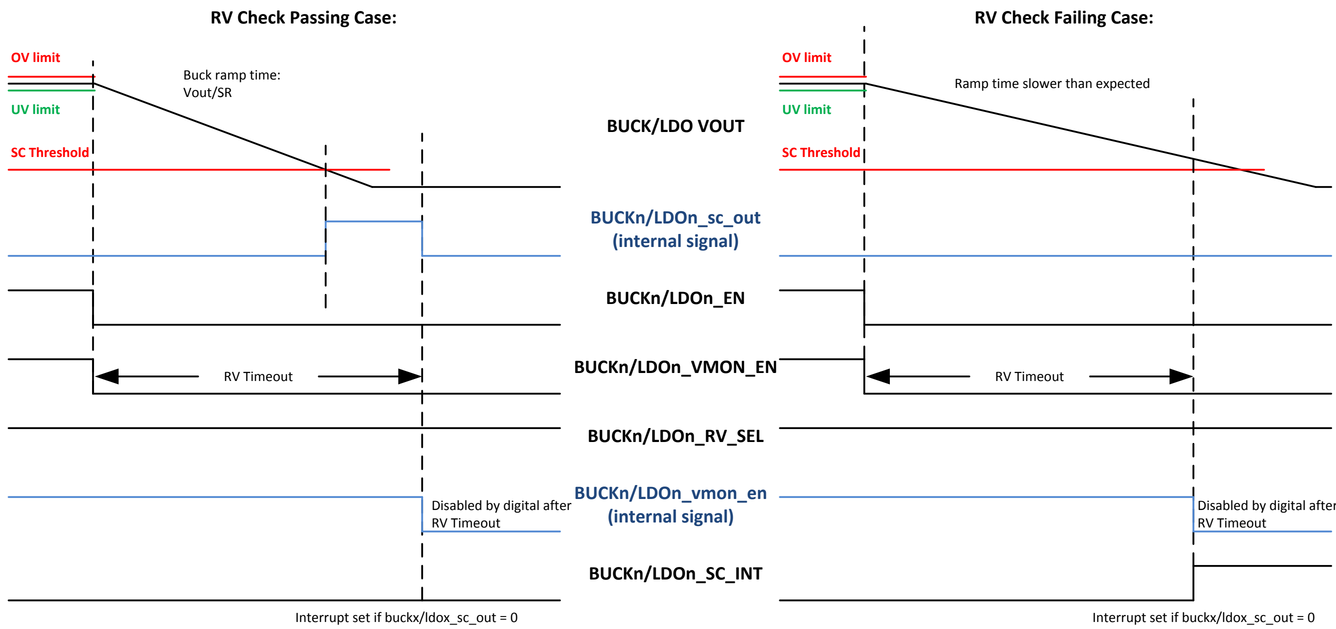


Figure 2-17. Residual Voltage Check Timing Diagram

2.8 Output Voltage Monitor and PGOOD Generation

The TPS6594-Q1 device monitors the under-voltage (UV) and over-voltage (OV) conditions of the output voltage of the bucks and LDOs, as well as VCCA when it is expected to be 5 V or 3.3 V, and has the option to indicate result with PGOOD signal. Thermal warning can also be included in the result of the PGOOD monitor if it is not masked. Either voltage and current monitoring or only voltage monitoring can be selected for PGOOD indication. This selection is set by the PGOOD_SEL_BUCKn register bits for each buck regulator (select master phase for multi-phase regulator), and is set by the PGOOD_SEL_LDOn register bits for each LDO regulator. When both voltage and current are monitored, PGOOD signal active indicates that the regulator output is inside the Power-Good voltage window and that load current is below the current limit. If only voltage is monitored, then the current monitoring is ignored for the PGOOD signal.

The BUCKn_VMON_EN bit enables the OV and UV, Short-circuit and current limit comparators. For LDO regulators, the LDOn_VMON_EN bit enables the OV and UV, Short-circuit and current limit comparators. When a buck or an LDO is not needed as a regulated output, it can be used as a voltage monitor for an external rail. For buck converters, if the BUCKn_VMON_EN bit remains '1' while the BUCKn_EN bit is '0', it can be used as a voltage monitor for an external rail which is connected to the buck converter's FB_Bn pin. For LDO regulators, if the LDOn_VMON_EN bit remains '1' while the LDOn_EN bit is '0', it can be used as a voltage monitor for an external rail which is connected to the VOUT_LDOn pin.

When the monitor for a buck or a LDO regulator is disabled, the output of the corresponding monitor is automatically masked to prevent it from forcing PGOOD inactive. This allows PGOOD to be connected to other open-drain power good signals in the system.

The VCCA_VMON_EN bit enables the monitoring of the VCCA input voltage. It can be enabled as an NVM default setting, which will start the monitoring of the VCCA voltage after the voltage monitor passes ABIST during the BOOT BIST state. The reference voltage for the VCCA monitor can be set by the VCCA_PG_SET bit to either 3.3 V or 5 V. The PGOOD_SEL_VCCA register bit selects whether or not the result of the VCCA monitor will be included in the PGOOD monitor output signal.

An NVM option is available to gate the PGOOD output with the nRSTOUT and the nRSTOUT_SoC signals, the intended reset signals for the safety MCU and the SoC respectively. When PGOOD_SEL_NRSTOUT = '1', the PGOOD pin is gated by the nRSTOUT signal. When PGOOD_SEL_NRSTOUT_SOC = '1', the PGOOD pin is gated by the nRSTOUT_SoC signal. This option allows the PGOOD output to be used as an enable signal for external peripherals.

The monitoring from all the output rails are combined, and PGOOD is active only if all the sources shows active status.

The type of output voltage monitoring for PGOOD signal is selected by PGOOD_WINDOW bit. If the bit is 0, only undervoltage is monitored; if the bit is 1, both undervoltage and over-voltage are monitored.

The polarity and the output type (push-pull or open-drain) are selected by PGOOD_POL and GPIO9_OD bits.

The Power-Good generation block diagram is shown in Figure 2-18. The Power-Good waveforms are shown in Figure 2-19.

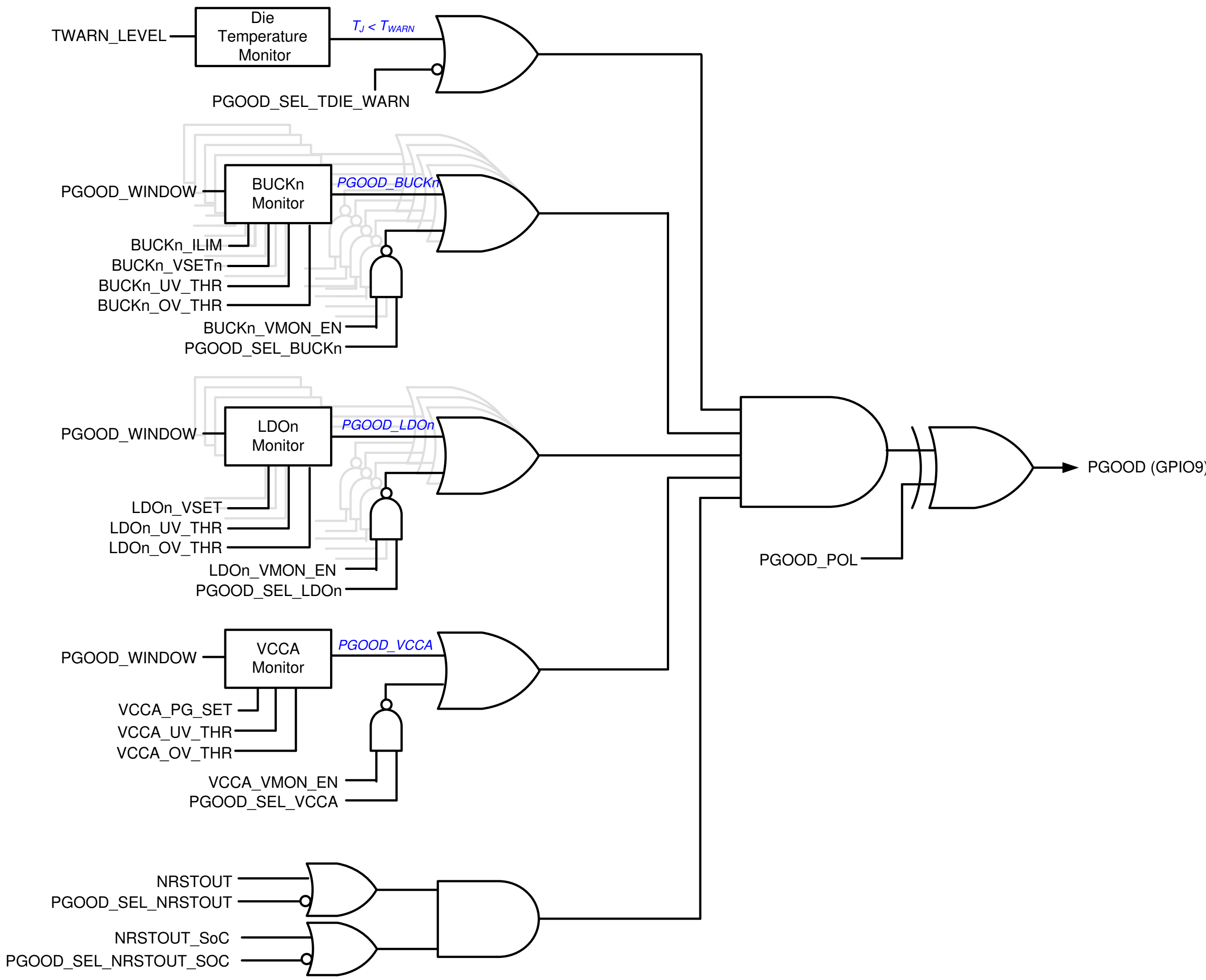


Figure 2-18. PGOOD Block Diagram

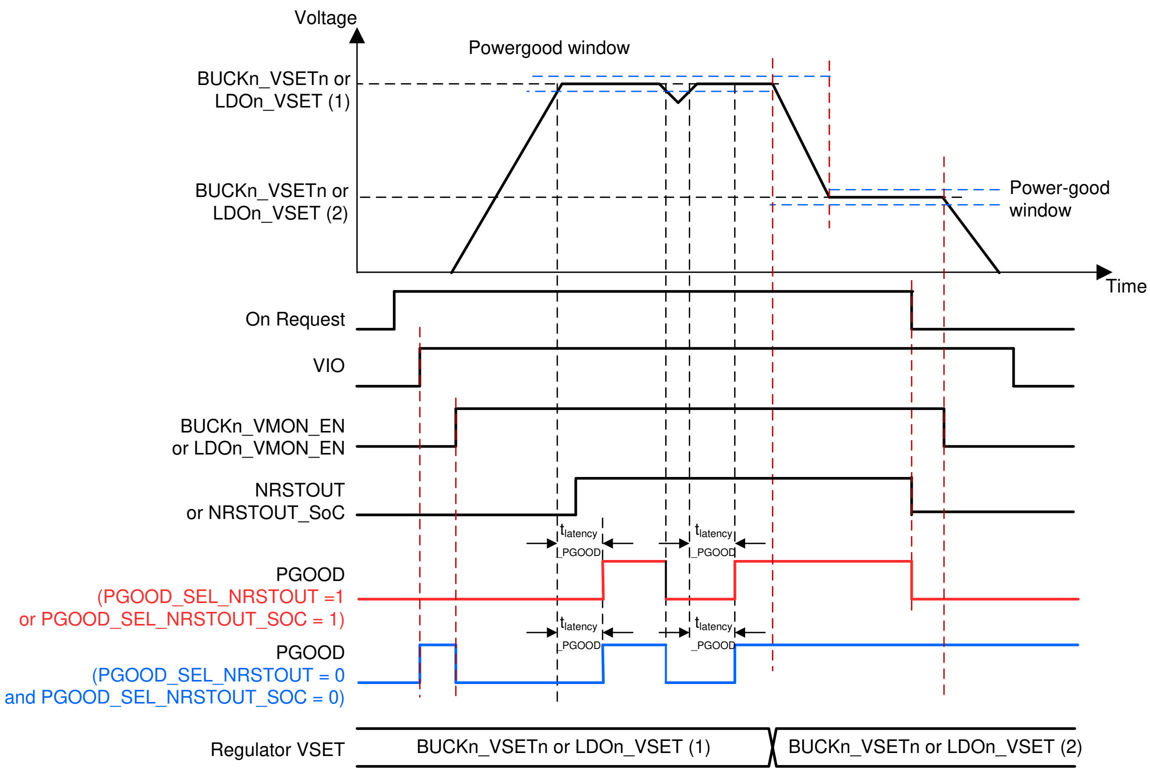


Figure 2-19. PGOOD Waveforms

The OV and UV threshold of the buck and LDO output voltage monitor are updated automatically by the digital control block when the output voltage setting changes. When the output voltage is increased, the OV threshold is updated at the same time the VSET of the regulator is changed. The UV threshold is updated after a delay calculated by the delta voltage change and the slew rate setting. When the output voltage is decreased, the UV threshold is updated at the same time the VSET of the regulator is changed. The OV threshold is updated after a delay calculated by the delta voltage change and the slew rate setting. The OV and UV threshold of the buck and LDO output voltage monitors are calculated based on the target output voltage set by the corresponding BUCKn_VSET1, BUCKn_VSET2, or LDOn_VSET registers, and the deviation from the target output voltage set by the corresponding BUCKn_UV_THR, BUCKn_OV_THR, LDOn_UV_THR, and the LDOn_OV_THR registers. For the OV and UV threshold of buck and LDO output monitors to be updated with the correct timing, following operating procedures must be followed when updating the VSET values of the regulators to avoid detection of OV/UV fault:

- Buck and LDO regulators must be enabled at the same time as or earlier than as their VMON so that the voltage will reach target value before OV/UV self test (BIST) is done
- New voltage level must not be set before the startup has finished and OV/UV self test (BIST) is completed
- New voltage level must not be set before the previous voltage change (ramp plus settling time) has completed

It is important to note that when a regulator is enabled, a voltage monitor self test is perform to ensure proper operation. The monitoring function is disabled and gated during this time. Figure 2-20 shows the timing diagram of the buck regulator UV/OV self test. Figure 2-21 shows the timing diagram of the LDO UV/OV self test.The monitoring function will become effective after the gating period.

The self test for VCCA, Buck and LDO voltage monitors is done every time when the monitoring function is enabled and VMON_ABIST_EN=1. The self test checks that OV and UV comparators are changing their output when the input thresholds are swapped. The self test assumes that the input voltage is inside OV/UV threshold limits. If the voltage is outside the limits, the self test will fail and BIST_FAIL_INT interrupt is set. In addition, a failed self test for over-voltage comparator will set the over-voltage interrupt.

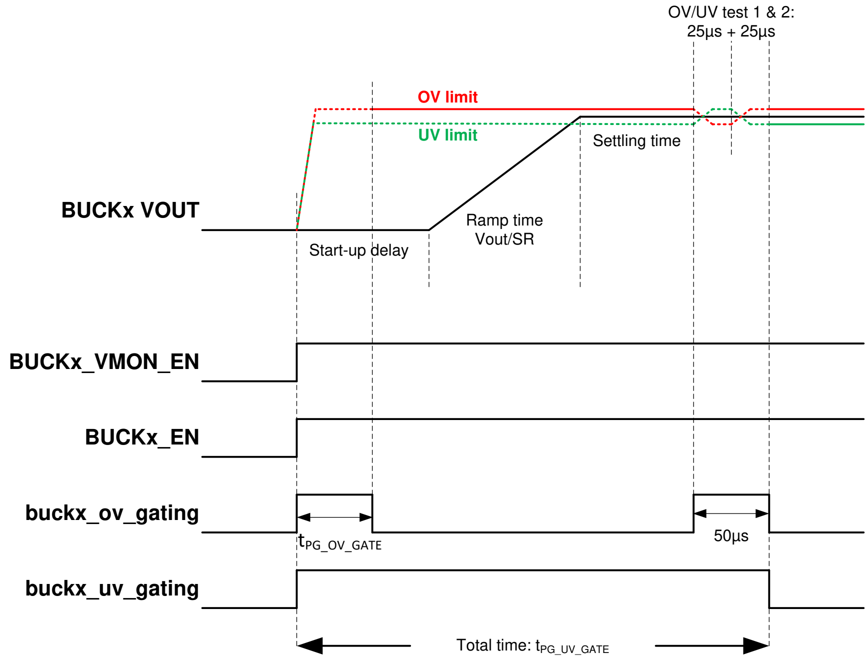


Figure 2-20. Timing of Buck Regulator UV/OV Self Test

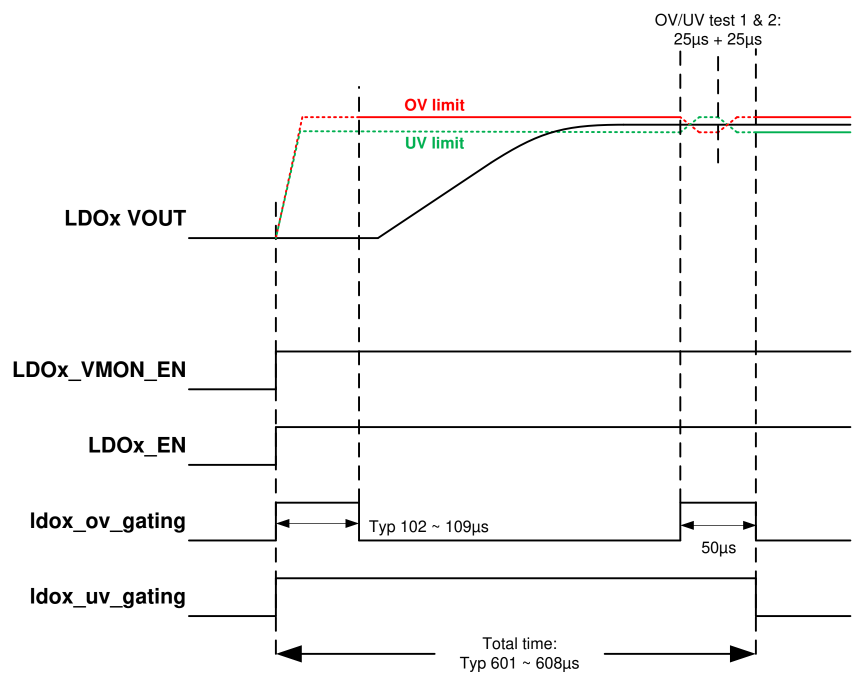


Figure 2-21. Timing of LDO Regulator UV/OV Self Test

2.9 General-Purpose I/Os (GPIO Pins)

The TPS6594-Q1 device integrates eleven configurable general-purpose I/Os that are multiplexed with alternative features as listed in the Pin Configuration and Functions section of SLVSEA7.

For GPIOs characteristics, refer to Electrical characteristics tables for Digital Input Signal Parameters and Digital Output Signal Parameters.

When configured as primary functions, all GPIOs are controlled through the following set of registers bits under the individual GPIOn_CONF register.

- GPIOn_DEGLITCH_EN: Enables the 8 µs deglitch time for each GPIO pin (input)

- GPIOn_PU_PD_EN: Enables the internal pull up or pull down resistor connected to each GPIO pin
- GPIOn_PU_SEL: Selects the pull up or the pull down resistor to be connected when GPIOn_PU_PD_EN = '1'. '1' = pull-up resistor selected, '0' = pull-down resistor selected
- GPIOn_OD: Configures the GPIO pin (output) as: '1' = open drain, '0' = push-pull
- GPIOn_DIR: Configures the input or output direction of each GPIO pin

Each GPIO event can generate an interrupt on a rising edge, falling edge, or both, configured through the GPIOn_FALL_MASK and the GPIOn_RISE_MASK register bits. A GPIO-interrupt applies when the primary function (general-purpose I/O) has been selected and also for the following alternative functions:

- nRSTOUT_SOC
- PGOOD
- nERR_MCU
- nERR_SoC
- TRIG_WDOG
- DISABLE_WDOG
- NSLEEP1, NSLEEP2
- WKUP1, WKUP2
- LP_WKUP1, LP_WKUP2

The GPIOn_SEL[2:0] register bits under the GPIOn_CONF registers control the selection between a primary and an alternative function. When a pre-defined function is selected, some predetermined IO characteristics (such as pullup, pulldown, push-pull or open drain) for the pin will be enforced regardless of the settings of the associated GPIO configuration register. Please note that if the GPIOn_SEL[2:0] is changed during device operation, a signal glitch may occur which may cause digital malfunction, especially if it involves a clock signal such as SCL_I2C2, CLK32KOUT, SCL_SPMI, SYNCCLKIN, or SYNCCLKOUT. Please refer to Section 1.3 for more detail on the predetermined IO characteristics for each pre-defined digital interface function.

All GPIOs can be configured as a wake-up input when it is configured as a WKUP1 or a WKUP2 signal. Only GPIO3 and GPIO4 can be configured as LP_WKUP1 or LP_WKUP2 signal so that they can be used to wake up the device from LP_STANDBY state. All GPIOs can also be configured as a NSLEEP1 or a NSLEEP2 input. For more information regarding the usage of the NSLEEPx pins and the WKUPx pins, please refer to Section 2.4.2.3.2.1 and Section 2.4.2.3.2.2.

2.10 Thermal Monitoring

The TPS6594-Q1 device includes several thermal monitoring functions for internal thermal protection of the PMIC.

The TPS6594-Q1 device integrates thermal detection modules to monitor the temperature of the die. These modules are placed on opposite sides of the device and close to the LDO and BUCK modules. An over-temperature condition at either module first generates a warning to the system and then, if the temperature continues to rise, a switch-off of the PMIC device can occur before damage to the die.

Three thermal protection levels are available. One of these protections is a thermal warning function described in Section 2.10.1, which sends an interrupt to software. Software is expected to close any noncritical running tasks to reduce power. The second and third protections are the thermal shutdown (TS) function described in Section 2.10.2, which begins device shutdown orderly or immediately.

Thermal monitoring is automatically enabled when any one of the buck or LDO outputs is enabled within the mission states. It is disabled in low power states, including the LP_STANDBY state, when only the internal regulators are enabled, to minimize the device power consumption. Indication of a thermal warning event is written to the TWARN_INT register.

The current consumption of the thermal monitoring can be decreased in mission states when the low power dissipation is important. If LPM_EN bit is set and the temperature is below thermal warning level in all thermal detection modules, only one thermal detection module is monitored. If the temperature rises in this module, monitoring in all thermal detection modules is started.

If the die temperature of the TPS6594-Q1 device continues to rise while the device is in mission state, an TSD_ORD_INT or TSD_IMM_INT interrupt is generated, causing a SEVERE or MODERATE error trigger (respectively) in the state machine. While the sequencing and error handling is NVM memory dependent, TI recommends a sequenced shutdown for MODERATE errors, and an immediate shutdown, using resistive discharging, for SEVERE errors to prevent damage to the device. The system cannot restart until the temperature falls below the thermal warning threshold.

2.10.1 Thermal Warning Function

The thermal monitor provides a warning to the host processor through the interrupt system when the temperature reaches within a cautionary range. The threshold value must be set to less than the thermal shutdown threshold.

The integrated thermal warning function provides the MCU an early warning of over-temperature condition. This monitoring system is connected to the interrupt controller and can send an TWARN_INT interrupt when the temperature is higher than the preset threshold. The TPS6594-Q1 device uses the TWARN_LEVEL register bit to set the thermal warning threshold temperature at 130°C or 140°C. There is no hysterisis for the thermal warning level.

When the power-management software triggers an interrupt, immediate action must be taken to reduce the amount of power drawn from the PMIC device (for example, noncritical applications must be closed).

2.10.2 Thermal Shutdown

The thermal shutdown detector monitors the temperature on the die. If the junction reaches a temperature at which damage can occur, a switch-off transition is initiated and a thermal shutdown event is written into a status register. There are two levels of thermal shutdown threshold. When the die temperature reaches the $T_{SD\_orderly}$ level, an orderly shutdown of the TPS6594-Q1 device will take place. If the die temperature raises rapidly and reaches the T_{SD_imm} level before the orderly shutdown process completes, an immediate shutdown of the device will take place to turn off all of the power resources as rapidly as possible. After the thermal shutdown takes place, the system cannot restart until the die temperature falls below the thermal warning threshold.

2.11 Interrupts

The interrupt registers in the device are organized in hierarchical fashion. The interrupts are grouped into the following categories:

BUCK ERROR	These interrupts indicate over-voltage (OV), under-voltage (UV), short-circuit (SC), residual voltage (SC) and over-current (ILIM) error conditions found on the Buck regulators.
LDO ERROR	These interrupts indicate OV, UV, and SC error conditions found on the LDO regulators, as well as OV and UV error conditions found on the VCCA supply.
SEVERE ERROR	These errors indicate severe device error conditions, such as thermal shutdown, PFSM sequencing and execution error and pre-regulator over-voltage failure, which causes the device to trigger the PFSM to execute immediate shutdown of all digital outputs, external voltage rails and monitors, and proceed to the Safe Recovery State.[3]
MODERATE ERROR	These interrupts provide warnings to the system to indicate detection of multiple WDOG Errors or ESM errors exceeding the allowed recovery count, detection of long press nPWRON button, SPMI communication error, register CRC error, BIST failure, or thermal reaching orderly shutdown level. These warning causes the device to trigger the PFSM to execute orderly shutdown of all digital outputs, external voltage rails and monitors, and proceed to the Safe Recovery State.[4]
MISCELLANEOUS WARNING	These interrupts provide information to the system to indicate detection of WDOG or ESM errors, die temperature crossing thermal warning threshold, device passing BIST test, or external sync clock availability.

[3] This error is handled in NVM memory but TI requires that the NVM pre-configurable finite state machine (PFSM) settings always follow this described error handling to meet device specifications.
[4] This error is handled in NVM memory but TI requires that the NVM pre-configurable finite state machine (PFSM) settings always follow this described error handling to meet device specifications.

STARTUP SOURCE These interrupts provides information to the system on the mechanism which caused the device to start up, which includes FSD, RTC alarm or timer interrupts, the activation of the ENABLE pin or the nPRWON pin button detection.

GPIO DETECTION These interrupts indicate the High/Rising-Edge or the Low/Falling-Edge detection at the GPIO1 through GPIO11 pins.

FSM ERROR INTERRUPT These interrupts indicate the detection of an error which causes the device mission state changes.

All interrupts are logically combined on a single output pin, nINT (active low). The host processor can read the INT_TOP register to find the interrupt registers to find out the source of the interrupt, and write '1' to the corresponding interrupt register bit to clear the interrupt. This mechanism ensures when a new interrupt occurs while the nINT pin is still active, all of the corresponding interrupt register bit will retain the interrupt source information until it is cleared by the host.

Some of the interrupts and EN_DRV status are also sent to host during SPI communication. See Section 2.17.3 for more information on SPI status signals.

Any interrupt source can be masked by setting the corresponding mask register to '1'. When an interrupt is masked, the interrupt bit is not updated when the associated event occurs, the nINT line will not be affected, and the event is not recorded. If an interrupt is masked after the event occurred, the interrupt register bit will reflect the event until the bit is cleared. While the event is masked, the interrupt register bit will not be over-written when a new event occurs.

Figure 2-22 shows the hierarchical structure of the interrupt registers according to the categories described above. The purpose of this register structure is to reduce the number of interrupt register read cycles the host has to perform in order to identify the source of the interrupt. Table 2-13 summarizes the trigger and the clearing mechanism for all of the interrupt signals. More detail descriptions of each interrupt registers can be found in Section 3.

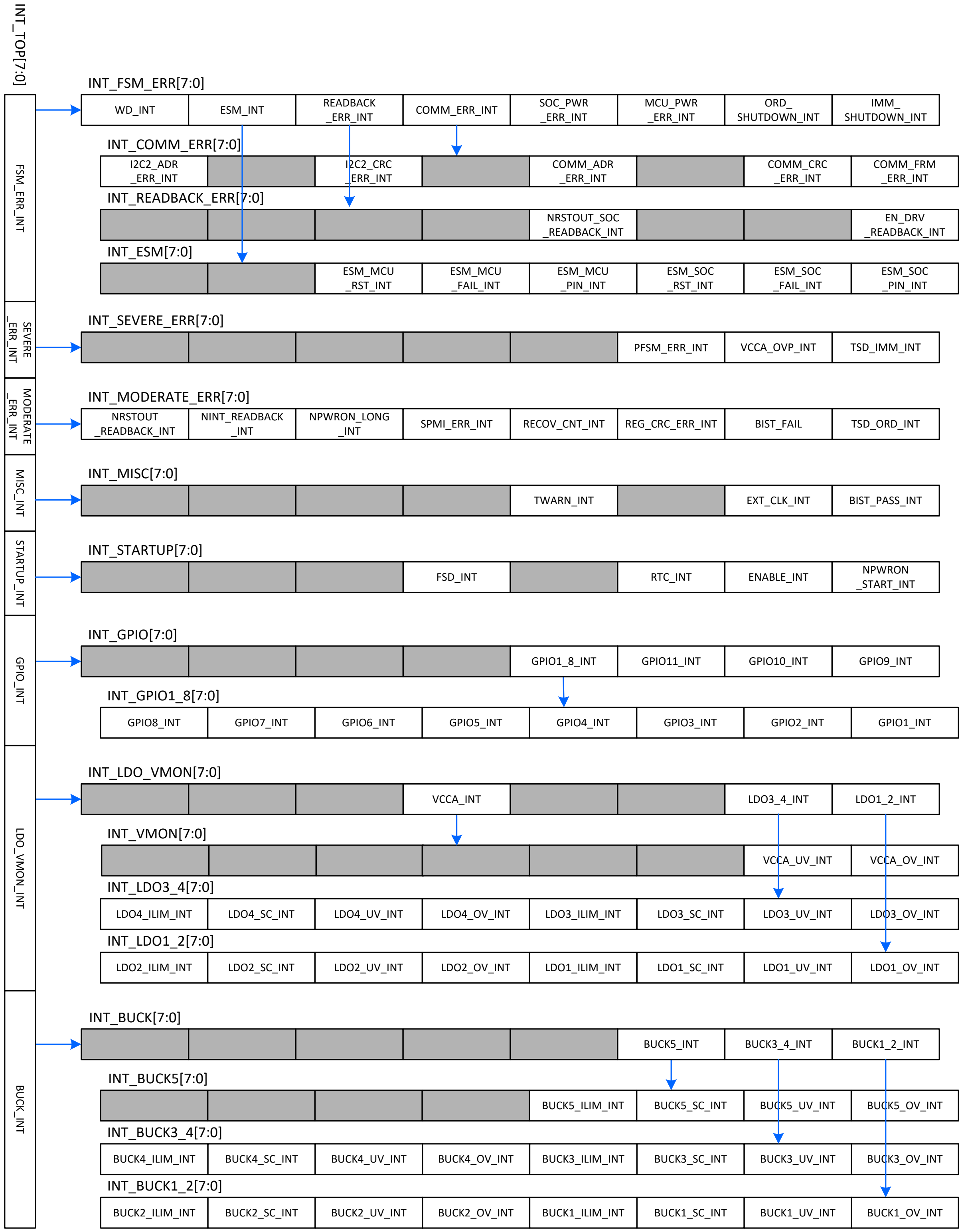


Figure 2-22. Hierarchical Structure of Interrupt Registers

Table 2-13. Summary of Interrupt Signals

EVENT	TRIGGER FOR FSM	RESULT [1]	RECOVERY	INTERRUPT BIT	MASK FOR INTERRUPT	LIVE STATUS BIT	INTERRUPT CLEAR
Buck regulator forward current limit triggered	**EN_ILIM_FSM_CTRL=1:** According to BUCKn_GRP_SEL and x_RAIL_TRIG bits **EN_ILIM_FSM_CTRL=0:** N/A	**EN_ILIM_FSM_CTRL=1:** Transition according to FSM trigger and interrupt **EN_ILIM_FSM_CTRL=0:** Interrupt only	Depends on FSM configuration, see FSM transition diagram	BUCKn_ILIM_INT = 1	BUCKn_ILIM_MASK	BUCKn_ILIM_STAT	Write 1 to BUCKn_ILIM_INT bit Interrupt is not cleared if current limit violation is active
LDO regulator current limit triggered	**EN_ILIM_FSM_CTRL=1:** According to LDOn_GRP_SEL and x_RAIL_TRIG bits **EN_ILIM_FSM_CTRL=0:** N/A	**EN_ILIM_FSM_CTRL=1:** Transition according to FSM trigger and interrupt **EN_ILIM_FSM_CTRL=0:** Interrupt only	Depends on FSM configuration, see FSM transition diagram	LDOn_ILIM_INT = 1	LDOn_ILIM_MASK	LDOn_ILIM_STAT	Write 1 to LDOn_ILIM_INT bit Interrupt is not cleared if current limit violation is active
Buck output or switch short circuit detected	According to BUCKn_GRP_SEL and x_RAIL_TRIG bits	Regulator disable and transition according to FSM trigger and interrupt	Depends on FSM configuration, see FSM transition diagram	BUCKn_SC_INT = 1	N/A	N/A	Write 1 to BUCKn_SC_INT bit
LDO output short circuit detected	According to LDOn_GRP_SEL and x_RAIL_TRIG bits	Regulator disable and transition according to FSM trigger and interrupt	Depends on FSM configuration, see FSM transition diagram	LDOn_SC_INT = 1	N/A	N/A	Write 1 to LDOn_SC_INT bit
Buck output residual voltage violation	**BUCKn_RV_SEL = 1** According to BUCKn_GRP_SEL and x_RAIL_TRIG bits **BUCKn_RV_SEL = 0** N/A	**BUCKn_RV_SEL = 1** Regulator disable and transition according to FSM trigger and interrupt **BUCKn_RV_SEL = 0** N/A	Depends on FSM configuration, see FSM transition diagram	BUCKn_SC_INT = 1	N/A	N/A	Write 1 to BUCKn_SC_INT bit
LDO output residual voltage violation	**LDOn_RV_SEL = 1** According to LDOn_GRP_SEL and x_RAIL_TRIG bits **LDOn_RV_SEL = 0** N/A	**LDOn_RV_SEL = 1** Regulator disable and transition according to FSM trigger and interrupt **LDOn_RV_SEL = 0** N/A	Depends on FSM configuration, see FSM transition diagram	LDOn_SC_INT = 1	N/A	N/A	Write 1 to LDOn_SC_INT bit
Buck regulator overvoltage	According to BUCKn_GRP_SEL and x_RAIL_TRIG bits	Transition according to FSM trigger and interrupt	Depends on FSM configuration, see FSM transition diagram	BUCKn_OV_INT = 1	BUCKn_OV_MASK	BUCKn_OV_STAT	Write 1 to BUCKn_OV_INT bit Interrupt is not cleared if it is active

Table 2-13. Summary of Interrupt Signals (continued)

EVENT	TRIGGER FOR FSM	RESULT (1)	RECOVERY	INTERRUPT BIT	MASK FOR INTERRUPT	LIVE STATUS BIT	INTERRUPT CLEAR
Buck regulator undervoltage	According to BUCKn_GRP_SEL and x_RAIL_TRIG bits	Transition according to FSM trigger and interrupt	Depends on FSM configuration, see FSM transition diagram	BUCKn_UV_INT = 1	BUCKn_UV_MASK	BUCKn_UV_STAT	Write 1 to BUCKn_UV_INT bit Interrupt is not cleared if it is active
LDO regulator overvoltage	According to LDOn_GRP_SEL and x_RAIL_TRIG bits	Transition according to FSM trigger and interrupt	Depends on FSM configuration, see FSM transition diagram	LDOn_OV_INT = 1	LDOn_OV_MASK	LDOn_OV_STAT	Write 1 to LDOn_OV_INT bit Interrupt is not cleared if it is active
LDO regulator undervoltage	According to LDOn_GRP_SEL and x_RAIL_TRIG bits	Transition according to FSM trigger and interrupt	Depends on FSM configuration, see FSM transition diagram	LDOn_UV_INT = 1	LDOn_UV_MASK	LDOn_UV_STAT	Write 1 to LDOn_UV_INT bit Interrupt is not cleared if it is active
VCCA input overvoltage monitoring	According to VCCA_GRP_SEL and x_RAIL_TRIG bits	Transition according to FSM trigger and interrupt	Depends on FSM configuration, see FSM transition diagram	VCCA_OV_INT = 1	VCCA_OV_MASK	VCCA_OV_STAT	Write 1 to VCCA_OV_INT bit Interrupt is not cleared if it is active
VCCA input undervoltage monitoring	According to VCCA_GRP_SEL and x_RAIL_TRIG bits	Transition according to FSM trigger and interrupt	Depends on FSM configuration, see FSM transition diagram	VCCA_UV_INT = 1	VCCA_UV_MASK	VCCA_UV_STAT	Write 1 to VCCA_UV_INT bit Interrupt is not cleared if it is active
Thermal warning	N/A	Interrupt only	Not valid	TWARN_INT = 1	TWARN_MASK	TWARN_STAT	Write 1 to TWARN_INT bit Interrupt is not cleared if temperature is above thermal warning level
Thermal shutdown, orderly sequenced	ORDERLY_SHUTDOWN (MODERATE_ERR_INT)	All regulators disabled and Output GPIOx set to low in a sequence and interrupt(1)	Automatic startup to STARTUP_DEST[1:0] state after temperature is below TWARN level	TSD_ORD_INT = 1	N/A	TSD_ORD_STAT	Write 1 to TSD_ORD_INT bit Interrupt is not cleared if temperature is above thermal shutdown level
Thermal shutdown, immediate	IMMEDIATE_SHUTDOWN (SEVERE_ERR_INT)	All regulators disabled with pull-down resistors and Output GPIOx set to low immediately and interrupt(1)	Automatic startup to STARTUP_DEST[1:0] state after temperature is below TWARN level	TSD_IMM_INT = 1	N/A	TSD_IMM_STAT	Write 1 to TSD_IMM_INT bit Interrupt is not cleared if temperature is above thermal shutdown level
BIST error	ORDERLY_SHUTDOWN (MODERATE_ERR_INT)	All regulators disabled and Output GPIOx set to low immediately and interrupt(1)	Automatic startup to STARTUP_DEST[1:0] state	BIST_FAIL_INT = 1	BIST_FAIL_MASK	N/A	Write 1 to BIST_FAIL_INT bit

Table 2-13. Summary of Interrupt Signals (continued)

EVENT	TRIGGER FOR FSM	RESULT [(1)]	RECOVERY	INTERRUPT BIT	MASK FOR INTERRUPT	LIVE STATUS BIT	INTERRUPT CLEAR
Register CRC error	ORDERLY_SHUTDOWN (MODERATE_ERR_INT)	All regulators disabled and Output GPIOx set to low immediately and interrupt[(1)]	Automatic startup to STARTUP_DEST[1:0] state	REG_CRC_ERR_INT = 1	REG_CRC_ERR_MASK	N/A	Write 1 to REG_CRC_ERR_INT bit
SPMI communication error	ORDERLY_SHUTDOWN (MODERATE_ERR_INT)	All regulators disabled and Output GPIOx set to low immediately and interrupt[(1)]	Automatic startup to STARTUP_DEST[1:0] state	SPMI_ERR_INT = 1	SPMI_ERR_MASK	N/A	Write 1 to SPMI_ERR_INT bit
SPI frame error	N/A	Interrupt only	Not valid	COMM_FRM_ERR_INT = 1	COMM_FRM_ERR_MASK	N/A	Write 1 to COMM_FRM_ERR_INT bit
I2C1 or SPI CRC error	N/A	Interrupt only	Not valid	COMM_CRC_ERR_INT = 1	COMM_CRC_ERR_MASK	N/A	Write 1 to COMM_CRC_ERR_INT bit
I2C1 or SPI address error	N/A	Interrupt only	Not valid	COMM_ADR_ERR_INT = 1	COMM_ADR_ERR_MASK	N/A	Write 1 to COMM_ADR_ERR_INT bit
I2C2 CRC error	N/A	Interrupt only	Not valid	I2C2_CRC_ERR_INT = 1	I2C2_CRC_ERR_MASK	N/A	Write 1 to I2C2_CRC_ERR_INT bit
I2C2 address error	N/A	Interrupt only	Not valid	I2C2_ADR_ERR_INT = 1	I2C2_ADR_ERR_MASK	N/A	Write 1 to I2C2_ADR_ERR_INT bit
PFSM error	IMMEDIATE_SHUTDOWN (SEVERE_ERR_INT)	All regulators disabled with pull-down resistors and Output GPIOx set to low immediately and interrupt[(1)]	Automatic startup to STARTUP_DEST[1:0] state. If previous PFSM_ERR_INT is pending, VCCA power cycle needed for recovery.	PFSM_ERR_INT = 1		N/A	Write 1 to PFSM_ERR_INT bit
EN_DRV pin readback error (monitoring high and low states)	N/A	Interrupt and EN_DRV = 0	Not valid	EN_DRV_READBACK_INT = 1	EN_DRV_READBACK_MASK	EN_DRV_READBACK_STAT	Write 1 to EN_DRV_READBACK_INT bit Interrupt is not cleared if it is active
NINT pin readback error (monitoring low state)	ORDERLY_SHUTDOWN (MODERATE_ERR_INT)	All regulators disabled with pull-down resistors and Output GPIOx set to low immediately and interrupt[(1)]	Automatic startup to STARTUP_DEST[1:0] state	NINT_READBACK_INT = 1	NINT_READBACK_MASK	NINT_READBACK_STAT	Write 1 to NINT_READBACK_INT bit Interrupt is not cleared if it is active
NRSTOUT pin readback error (monitoring low state)	ORDERLY_SHUTDOWN (MODERATE_ERR_INT)	All regulators disabled with pull-down resistors and Output GPIOx set to low immediately and interrupt[(1)]	Automatic startup to STARTUP_DEST[1:0] state	NRSTOUT_READBACK_INT = 1	NRSTOUT_READBACK_MASK	NRSTOUT_READBACK_STAT	Write 1 to NRSTOUT_READBACK_INT bit Interrupt is not cleared if it is active

Table 2-13. Summary of Interrupt Signals (continued)

EVENT	TRIGGER FOR FSM	RESULT [(1)]	RECOVERY	INTERRUPT BIT	MASK FOR INTERRUPT	LIVE STATUS BIT	INTERRUPT CLEAR
NRSTOUT_SOC pin readback error (monitoring low state)	N/A	Interrupt only	Not valid	NRSTOUT_SOC_READBACK_INT = 1	NRSTOUT_SOC_READBACK_MASK	NRSTOUT_SOC_READBACK_STAT	Write 1 to NRSTOUT_SOC_READBACK_INT bit Interrupt is not cleared if it is active
Fault detected by SOC ESM (level mode: low level detected, PWM mode: PWM signal timing violation)	N/A	Interrupt only	Not valid	ESM_SOC_PIN_INT = 1	ESM_SOC_PIN_MASK	N/A	Write 1 to ESM_SOC_PIN_INT bit
Fault detected by SOC ESM (level mode: low level longer than DELAY1 time, PWM mode: ESM error counter > FAIL_THR longer than DELAY1time)	N/A	Interrupt and EN_DRV = 0 (configurable)	Not valid	ESM_SOC_FAIL_INT = 1	ESM_SOC_FAIL_MASK	N/A	Write 1 to ESM_SOC_FAIL_INT bit
Fault detected by SOC ESM (level mode: low level longer than DELAY1+DELAY2 time, PWM mode: ESM error counter > FAIL_THR longer than DELAY1+DELAY2 time)	ESM_SOC_RST	Interrupt, and NRSTOUT_SOC toggle[(1)]	Automatically returns to the current operating state after the completion of SoC warm reset	ESM_SOC_RST_INT = 1	ESM_SOC_RST_MASK	N/A	Write 1 to ESM_SOC_RST_INT bit
Fault detected by MCU ESM (level mode: low level detected, PWM mode: PWM signal timing violation	N/A	Interrupt only	Not valid	ESM_MCU_PIN_INT = 1	ESM_MCU_PIN_MASK	N/A	Write 1 to ESM_MCU_PIN_INT bit
Fault detected by MCU ESM (level mode: low level longer than DELAY1 time, PWM mode: ESM error counter > FAIL_THR longer than DELAY1 time)	N/A	Interrupt and EN_DRV = 0 (configurable)	Not valid	ESM_MCU_FAIL_INT = 1	ESM_MCU_FAIL_MASK	N/A	Write 1 to ESM_MCU_FAIL_INT bit

Table 2-13. Summary of Interrupt Signals (continued)

EVENT	TRIGGER FOR FSM	RESULT [(1)]	RECOVERY	INTERRUPT BIT	MASK FOR INTERRUPT	LIVE STATUS BIT	INTERRUPT CLEAR
Fault detected by MCU ESM (level mode: low level longer than DELAY1+DELAY2 time, PWM mode: ESM error counter > FAIL_THR longer than DELAY1+DELAY2 time)	ESM_MCU_RST	Interrupt and Warm Reset (EN_DRV = 0 and NRSTOUT and NRSTOUT_SOC toggle)[(1)]	Automatically returns to the current operating state after the completion of warm reset	ESM_MCU_RST_INT = 1	ESM_MCU_RST_MASK	N/A	Write 1 to ESM_MCU_RST_INT bit
External clock is expected, but it is not available or the frequency is not in the valid range	N/A	Interrupt only	Not valid	EXT_CLK_INT = 1[(2)]	EXT_CLK_MASK	EXT_CLK_STAT	Write 1 to EXT_CLK_INT bit
BIST completed successfully	N/A	Interrupt only	Not valid	BIST_PASS_INT = 1	BIST_PASS_MASK	N/A	Write 1 to BIST_PASS_INT bit
Watchdog fail counter above fail threshold	N/A	Interrrupt and EN_DRV = 0	Clear interrupt and WD_FAIL_CNT < WD_FAIL_TH	WD_FAIL_INT = 1	N/A	N/A	Write 1 to WD_FAIL_INT bit
Watchdog fail counter above reset threshold	WD_RST (if WD_RST_EN = 1)	Interrupt and Warm Reset if WD_RST_EN = 1 (EN_DRV = 0 and NRSTOUT and NRSTOUT_SOC toggle)[(1)]	Automatically returns to the current operating state after the completion of warm reset	WD_RST_INT = 1	N/A	N/A	Write 1 to WD_RST_INT bit
Watchdog long window timeout	WD_RST	Interrupt and Warm Reset (EN_DRV = 0 and NRSTOUT and NRSTOUT_SOC toggle)[(1)]	Automatically returns to the current operating state after the completion of warm reset	WD_LONGWIN_TIMEOUT_INT = 1	N/A	N/A	Write 1 to WD_LONGWIN_TIMEOUT_INT bit
RTC alarm wake-up	TRIGGER_SU_x	Startup to STARTUP_DEST[1:0] state and interrupt[(1)]	Not valid	ALARM = 1	IT_ALARM = 0	N/A	Write 1 to ALARM bit
RTC timer wake-up	TRIGGER_SU_x	Startup to STARTUP_DEST[1:0] state and interrupt[(1)]	Not valid	TIMER = 1	IT_TIMER = 0	N/A	Write 1 to TIMER bit
Low state in NPWRON pin	TRIGGER_SU_x	Startup to STARTUP_DEST[1:0] state and interrupt[(1)]	Not valid	NPWRON_START_INT = 1	NPWRON_START_MASK	NPWRON_IN	Write 1 to NPWRON_START_INT bit
Long low state in NPWRON pin	ORDERLY_SHUTDOWN	All regulators disabled and Output GPIOx set to low in a sequence and interrupt[(1)]	Valid power-on request	NPWRON_LONG_INT = 1	NPWRON_LONG_MASK	NPWRON_IN	Write 1 to NPWRON_LONG_INT bit

Table 2-13. Summary of Interrupt Signals (continued)

EVENT	TRIGGER FOR FSM	RESULT [(1)]	RECOVERY	INTERRUPT BIT	MASK FOR INTERRUPT	LIVE STATUS BIT	INTERRUPT CLEAR
Low state in ENABLE pin	TRIGGER_FORCE_STANDBY/ TRIGGER_FORCE_LP_STANDBY	Transition to STANDBY or LP_STANDBY depending on the LP_STANDBY_SEL bit setting[(1)]	ENABLE pin rise	N/A	N/A	N/A	N/A
ENABLE pin rise	TRIGGER_SU_x	[(1)]	Not valid	ENABLE_INT = 1	ENABLE_MASK	ENABLE_STAT	Write 1 to ENABLE_INT bit
Fault causing orderly shutdown	ORDERLY_SHUTDOWN	All regulators disabled and Output GPIOx set to low in a sequence and interrupt[(1)]	Automatic startup to STARTUP_DEST[1:0] state	ORD_SHUTDOWN_INT	ORD_SHUTDOWN_MASK	N/A	Write 1 to ORD_SHUTDOWN_INT
Fault causing immediate shutdown	IMMEDIATE_SHUTDOWN	All regulators disabled with pull-down resistors and Output GPIOx set to low immediately and interrupt[(1)]	Automatic startup to STARTUP_DEST[1:0] state	IMM_SHUTDOWN_INT	IMM_SHUTDOWN_MASK	N/A	Write 1 to IMM_SHUTDOWN_INT
Power supply error for MCU	MCU_POWER_ERROR	Transition according to FSM trigger and interrupt	Depends on FSM configuration, see FSM transition diagram	MCU_PWR_ERR_INT	MCU_PWR_ERR_MASK	N/A	Write 1 to MCU_PWR_ERR_INT
Power supply error for SOC	SOC_POWER_ERROR	Transition according to FSM trigger and interrupt	Depends on FSM configuration, see FSM transition diagram	SOC_PWR_ERR_INT	SOC_PWR_ERR_MASK	N/A	Write 1 to SOC_PWR_ERR_INT
VCCA over-voltage ($VCCA_{OVP}$)	IMMEDIATE_SHUTDOWN (SEVERE_ERR_INT)	All regulators disabled with pull-down resistors and Output GPIOx set to low immediately and interrupt[(1)]	Automatic startup to STARTUP_DEST[1:0] state after VCCA voltage is below $VCCA_{OVP}$	VCCA_OVP_INT = 1	N/A	VCCA_OVP_STAT	Write 1 to INT_OVP _INT bit Interrupt is not cleared if VCCA voltage is above $VCCA_{OVP}$ level
GPIO interrupt	According to GPIOx_FSM_MASK and GPIOx_FSM_MASK_POL bits	Transition according to FSM trigger and interrupt	Not valid	GPIOx_INT = 1	GPIOx_RISE_MASK GPIOx_FALL_MASK	GPIOx_IN	Write 1 to GPIOx_INT bit
WKUP1 and LP_WKUP1 signals	WKUP1	Transition to ACTIVE state and interrupt[(1)]	Not valid	N/A	GPIOx_RISE_MASK GPIOx_FALL_MASK	GPIOx_IN	Write 1 to GPIOx_INT bit
WKUP2 and LP_WKUP2 signals	WKUP2	Transition to MCU ONLY state and interrupt[(1)]	Not valid	N/A	GPIOx_RISE_MASK GPIOx_FALL_MASK	GPIOx_IN	Write 1 to GPIOx_INT bit
NSLEEP1 signal, NSLEEP1B bit	According to NSLEEP1 and NSLEEP2	State transition based on NSLEEP1 and NSLEEP2	Not valid	N/A	NSLEEP1_MASK	GPIOx_IN	N/A

Table 2-13. Summary of Interrupt Signals (continued)

EVENT	TRIGGER FOR FSM	RESULT [(1)]	RECOVERY	INTERRUPT BIT	MASK FOR INTERRUPT	LIVE STATUS BIT	INTERRUPT CLEAR
NSLEEP2 signal, NSLEEP2B bit	According to NSLEEP1 and NSLEEP2	State transition based on NSLEEP1 and NSLEEP2	Not valid	N/A	NSLEEP2_MASK	GPIOx_IN	N/A
LDOVINT over- or undervoltage	IMMEDIATE_SHUTDOWN	All regulators disabled with pull-down resistors and Output GPIOx set to low immediately[(1)]	Valid LDOVINT voltage	N/A	N/A	N/A	N/A
Main clock outside valid frequency	IMMEDIATE_SHUTDOWN	All regulators disabled with pull-down resistors and Output GPIOx set to low immediately[(1)]	VCCA power cycle	N/A	N/A	N/A	N/A
Recovery counter limit exceeded[(3)]	ORDERLY_SHUTDOWN	All regulators disabled and Output GPIOx set to low in a sequence[(1)]	VCCA power cycle	N/A	N/A	N/A	N/A
VCCA supply falling below $VCCA_{UVLO}$	IMMEDIATE_SHUTDOWN	Immediate shutdown[(1)]	VCCA voltage rising	N/A	N/A	N/A	N/A
First supply detection, VCCA supply rising above $VCCA_{UVLO}$	TRIGGER_SU_x	Startup to STARTUP_DEST[1:0] state and interrupt[(1)]	Not valid	FSD_INT = 1	FSD_MASK	N/A	Write 1 to FSD_INT bit

(1) The results shown in this column are selected to meet functional safety assumptions and device specifications. The actual results can be configured differently in NVM memory. TI recommends reviewing of the system and device funcational safety goal and documentation before deviating from these recommendations.
(2) Interrupt is generated during clock detector operation and in case clock is not available when clock detector is enabled.
(3) This event will not occur if RECOV_CNT_THR = 0, even though RECOV_CNT will continue to accumulate and increase, and will eventually saturate when it reaches the max count of 15.

2.12 RTC

2.12.1 General Description

The RTC is driven by the 32-kHz oscillator and it provides the alarm and time-keeping functions.

The main functions of the RTC block are:

- Time information (seconds, minutes, and hours) in binary-coded decimal (BCD) code
- Calendar information (day, month, year, and day of the week) in BCD code up to year 2099
- Configurable interrupts generation; the RTC can generate two types interrupts which can be enabled and masked individually:
  – Timer interrupts periodically (1-second, 1-minute, 1-hour, or 1-day periods)
  – Alarm interrupt at a precise time of the day (alarm function)
- Oscillator frequency calibration and time correction with 1/32768 resolution

Figure 2-23 shows the RTC block diagram.

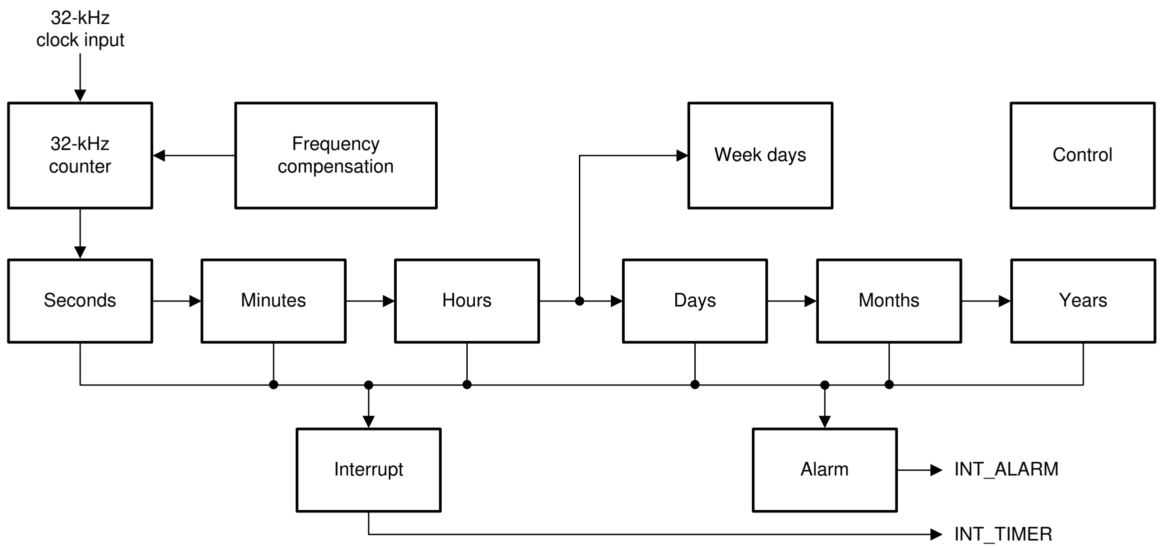


Figure 2-23. RTC Block Diagram

2.12.2 Time Calendar Registers

All the time and calendar information is available in the time calendar (TC) dedicated registers: SECONDS_REG, MINUTES_REG, HOURS_REG, DAYS_REG, WEEKS_REG, MONTHS_REG, and YEARS_REG. The TC register values are written in BCD code.

- Year data ranges from 00 to 99.
  - Leap Year = Year divisible by four (2000, 2004, 2008, 2012, and so on)
  - Common Year = Other years
- Month data ranges from 01 to 12.
- Day value ranges:
  - 1 to 31 when months are 1, 3, 5, 7, 8, 10, 12
  - 1 to 30 when months are 4, 6, 9, 11
  - 1 to 29 when month is 2 and year is a leap year
  - 1 to 28 when month is 2 and year is a common year
- Weekday value ranges from 0 to 6.
- Hour value ranges from 0 to 23 in 24-hour mode and ranges from 1 to 12 in AM or PM mode.
- Minutes value ranges from 0 to 59.
- Seconds value ranges from 0 to 59.

Example: Time is 10H54M36S PM (PM_AM mode set), 2008 September 5; previous registers values are listed in Table 2-14:

Table 2-14. RTC Time Calendar Registers Example

REGISTER	CONTENT
RTC_SECONDS	0x36
RTC_MINTURES	0x54
RTC_HOURS	0x10
RTC_DAYS	0x05
RTC_MONTHS	0x09
RTC_YEARS	0x08
RTC_WEEKS	0x06

The user can round to the closest minute, by setting the ROUND_30S register bit in the RTC_CTRL_REG register. TC values are set to the closest minute value at the next second. The ROUND_30S bit is automatically cleared when the rounding time is performed.

Example:

- If current time is 10H59M45S, round operation changes time to 11H00M00S
- If current time is 10H59M29S, round operation changes time to 10H59M00S

2.12.2.1 TC Registers Read Access

TC register read access can be done in two ways:

- A direct read to the TC registers. In this case, there can be a discrepancy between the final time read and the real time because the RTC keeps running because some of the registers can toggle in between register accesses. Software must manage the register change during the reading.
- Read access to shadowed TC registers. These registers are at the same addresses as the normal TC registers. They are selected by setting the GET_TIME bit in the RTC_CTRL_REG register. When this bit is set, the content of all TC registers is transferred into shadow registers so they represent a coherent timestamp, avoiding any possible discrepancy between them. When processing the read accesses to the TC registers, the value of the shadowed TC registers is returned so it is completely transparent in terms of register access.

2.12.2.2 TC Registers Write Access

TC registers write accesses can be done while RTC is stopped. MCU can stop the RTC by the clearing the STOP_RTC bit of the control register and checking the RUN bit of the status to be sure that RTC is frozen. MCU then updates the TC values and restarts the RTC by setting the STOP_RTC bit, which ensures that the final written values are aligned with the targeted values.

2.12.3 RTC Alarm

RTC alarm registers (ALARM_SECONDS_REG, ALARM_MINUTES_REG, ALARM_HOURS_REG, ALARM_DAYS_REG, ALARM_MONTHS_REG, and ALARM_YEARS_REG) are used to set the alarm time or date to the corresponding generated ALARM interrupts. These register values are written in BCD code, with the same data range as described for the TC registers (see Section 2.12.2).

2.12.4 RTC Interrupts

The RTC supports two types of interrupts:

- ALARM interrupt. This interrupt is generated when the configured date or time in the corresponding ALARM registers is reached. This interrupt is enabled and disabled by setting the IT_ALARM bit. It is important to set the IT_ALARM = 0 to disable the alarm interrupt prior to configuring the ALARM registers to prevent the interrupt from mis-firing.
- TIMER interrupt. This interrupt is generated when the periodic time (day, hour, minute, second) set in the EVERY bits of the RTC_INTERRUPTS register is reached. The first of the periodic interrupt will occur when the RTC counter reaches the next day, hour, minute, or second counter value. For example, if a timer interrupt is set for every hour at 2:59 AM, the first interrupt will occur at 3:00 AM instead of 3:59 AM. This interrupt is enabled and disabled by setting the IT_TIMER bit. It is important to set the IT_TIMER = 0 to disable the timer interrupt prior to configuring the periodic time value to prevent the interrupt from mis-firing.

Both types of the RTC interrupts can be used to wake up the device from the STANDBY state or the LP_STANDBY state when they are not masked.

2.12.5 RTC 32-kHz Oscillator Drift Compensation

The RTC_COMP_MSB_REG and RTC_COMP_LSB_REG registers are used to compensate for any inaccuracy of the 32-kHz clock output from the 32-kHz crystal oscillator. To compensate for any inaccuracy, MCU must perform an external calibration of the oscillator frequency, calculate the drift compensation needed versus one time hour period, and load the compensation registers with the drift compensation value.

The compensation mechanism is enabled by the AUTO_COMP_EN bit in the RTC_CTRL_REG register. The process happens after the first second of each hour. The time between second 1 to second 2 (T_ADJ) is adjusted based on the settings of the two RTC_COMP_MSB_REG and RTC_COMP_LSB_REG registers. These two registers form a 16-bit, 2 s complement value COMP_REG (from –32767 to 32767) that is subtracted from the 32-kHz counter as per the following formula to adjust the length of T_ADJ: (32768 - COMP_REG) / 32768. It is therefore possible to adjust the compensation with a 1/32768-second time unit accuracy per hour and up to 1 second per hour.

Software must ensure that these registers are updated before each compensation process (there is no hardware protection). For example, software can load the compensation value into these registers after each hour event, during second 0 to second 1, just before the compensation period, happening from second 1 to second 2.

It is also possible to preload the internal 32-kHz counter with the content of the RTC_COMP_MSB_REG and RTC_COMP_LSB_REG registers when setting the SET_32_COUNTER bit in the RTC_CTRL_REG register. This must be done when the RTC is stopped.

Figure 2-24 shows the RTC compensation scheduling.

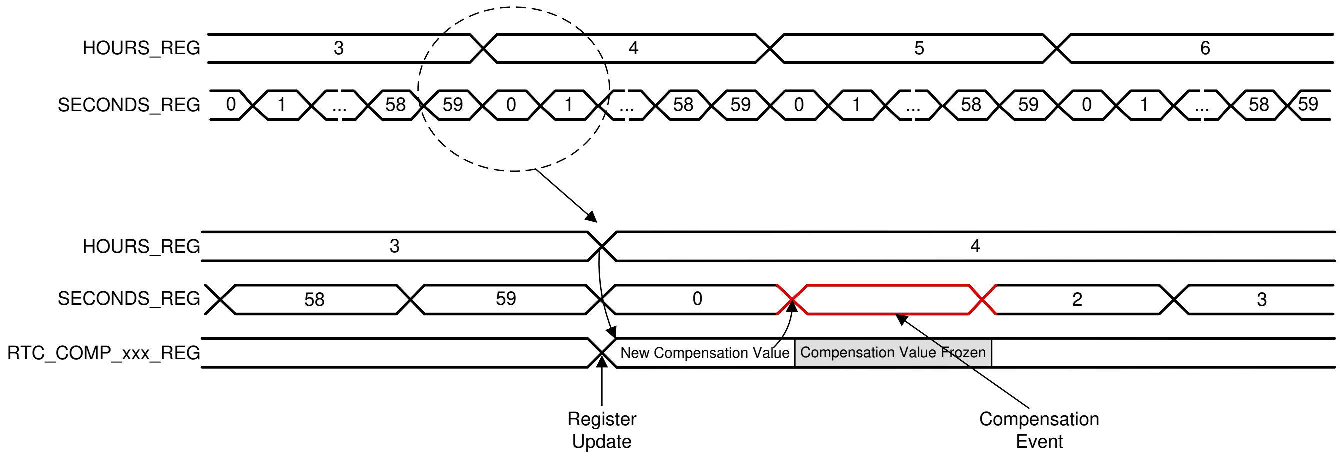


Figure 2-24. RTC Compensation Scheduling

2.13 Watchdog (WD)

The watchdog monitors the correct operation of the MCU. This watchdog requires specific messages from the MCU in specific time intervals to detect correct operation of the MCU. The MCU can control the logic-level of the EN_DRV pin when the watchdog detects correct operation of the MCU. When the watchdog detects incorrect operation of the MCU, the TPS6594-Q1 device pulls the EN_DRV pin low . This EN_DRV pin can be used in the application as a control-signal to deactivate the power output stages, for example a motor driver, in case of incorrect operation of the MCU.

The watchdog has two different modes which are defined as follows:

Trigger mode In trigger mode, the MCU applies a pulse signal with a minimum pulse width of t_{WD_pulse} on the pre-assigned GPIO input pin to send the required watchdog trigger. To select this mode, the MCU must clear bit WD_MODE_SELECT. More details are available in Section 2.13.6

Q&A (question and answer) mode In Q&A mode, the MCU sends watchdog answers through the I2C bus or SPI bus. To select this mode, the MCU must set bit WD_MODE_SELECT. More details are available in Section 2.13.7.1

2.13.1 Watchdog Fail Counter and Status

The watchdog includes a watchdog fail counter WD_FAIL_CNT[3:0] that increments because of *bad events* or decrements because of *good events*. Furthermore, the watchdog includes two configurable thresholds:

1. Fail-threshold (configurable through bits WD_FAIL_TH[2:0])
2. Reset-threshold (configurable through bits WD_RST_TH[2:0])

When the WD_FAIL_CNT[3:0] counter value is less than or equal to the configured Watchdog-Fail threshold (WD_FAIL_TH[2:0]) and bit WD_FIRST_OK=1, the MCU can set the ENABLE_DRV bit when no other error-flags are set.

When the WD_FAIL_CNT[3:0] counter value is greater than the configured Watchdog-Fail threshold (WD_FAIL_CNT[3:0] > WD_FAIL_TH[2:0]), the device clears the ENABLE_DRV bit, sets the error-flag WD_FAIL_INT, and pulls the nINT pin low.

When the WD_FAIL_CNT[3:0] counter value is greater than the configured Watchdog-Fail plus Watchdog-Reset threshold (WD_FAIL_CNT[3:0] > (WD_FAIL_TH[2:0] + WD_RST_TH[2:0])) and the watchdog-reset function is enabled (configuration bit WD_RST_EN=1), the device generates a WD_ERROR trigger in the state machine and sets the error-flag WD_RST_INT, and pulls the nINT pin low.

The device clears the WD_FAIL_CNT[3:0] each time the watchdog enters the Long Window. The status bits WD_FAIL_INT and WD_RST_INT are latched until the MCU writes a ‘1’ to these bits.

Table gives Table 2-15 an overview of the Watchdog Fail Counter value ranges and the corresponding device status.

Table 2-15. Overview of Watchdog Fail Counter Value Ranges and Corresponding Device Status

Watchdog Fail Counter value WD_FAIL_CNT[3:0]	Device Status
WD_FAIL_CNT[3:0] ≤ WD_FAIL_TH[2:0]	MCU can set the ENABLE_DRV bit if WD_FIRST_OK=1 and no other error-flags are set
WD_FAIL_TH[2:0] < WD_FAIL_CNT[3:0] ≤ (WD_FAIL_TH[2:0] + WD_RST_TH[2:0])	The device clears the ENABLE_DRV bit, sets error-flag WD_FAIL_INT and pulls the nINT pin low
WD_FAIL_CNT[3:0] > (WD_FAIL_TH[2:0] + WD_RST_TH[2:0])	If configuration bit WD_RST_EN=1, device generates WD_ERROR trigger in the state machine and reacts as defined in the PFSM, sets the error-flag WD_RST_INT, and pulls the nINT pin low. See Interrupt handling for WD_RTS in Table 2-13.

The WD_FAIL_CNT[3:0] counter responds as follows:

- When the Watchdog is in the Long-Window, the WD_FAIL_CNT[3:0] is cleared to 4’b0000
- A good event decrements the WD_FAIL_CNT[3:0] by one before the start of the next Window-1
- A bad event increments the WD_FAIL_CNT[3:0] by one before the start of the next Window-1

For definitions of good event and bad event, please refer to Section 2.13.6 and Section 2.13.7.1 respectively.

2.13.2 Watchdog Start-Up and Configuration

When the device releases the nRSTOUT pin, the watchdog starts with the Long Window. This Long Window has a time interval (t_{LONG_WINDOW}) with a default value set in bits WD_LONGWIN[7:0].

As long as the watchdog is in the Long Window, the MCU can configure the watchdog through the following register bits:

- WD_EN to enable/disable the watchdog
- WD_LONGWIN[7:0] to increase the duration of the Long-Window time-interval
- WD_MODE_SELECT to select the Watchdog mode (Trigger mode or Q&A Mode)
- WD_PWRHOLD to activate the Watchdog Disable function (more detail in Section 2.13.4)
- WD_RETURN_LONGWIN to configure wheter to return to Long-Window or continue to the next seqeunce after the completion of the current watchdog sequence (more detail in Section 2.13.4)
- WD_WIN1[6:0] to configure the duration of the Window-1 time-interval
- WD_WIN2[6:0] to configure the duration of the Window-2 time-interval
- WD_RST_EN to enable/disable the watchdog-reset function
- WD_FAIL_TH[2:0] to configure the Watchdog-Fail threshold
- WD_RST_TH[2:0] to configure the Watchdog-Reset threshold
- WD_QA_FDBK[1:0] to configure the settings for the reference answer-generation
- WD_QA_LFSR[1:0] to configure the settings for the question-generation
- WD_QUESTION_SEED[3:0] to configure the starting-point for the 1st question-generation
- WD_QA_CFG for watchdog in Q&A Mode

The device will keep the above register bit values configured by the MCU as long as the device is powered.

The MCU can configure the time interval of the Long Window (t_{LONG_WINDOW}) with the WD_LONGWIN[7:0] bits. The WD_LONGWIN[7:0] bits are defined as:

- 0x00: 80 ms
- 0x01 - 0x40: 125 ms to 8 sec, in 125 ms steps
- 0x41 - 0xFF: 12 sec to 772 sec, in 4 sec stepts

Use Equation 5 and Equation 6 to calculate the minimum and maximum values for the Long Window (t_{LONG_WINDOW}) time interval when WD_LONGWIN[7:0] > 0x00:

$$t_{LONG\_WINDOW\_MIN} = WD\_LONGWIN[7:0] \times 0.95 \tag{5}$$

$$t_{LONG\_WINDOW\_MAX} = WD\_LONGWIN[7:0] \times 1.05 \tag{6}$$

Note

If the MCU software changes the duration of the Long-Window to an interval shorter than the time in which the watchdog has been in the Long-Window, the time-out function of the Long-Window will no longer operate.

When the MCU clears bit WD_EN, the watchdog goes out of the Long Window and disables the watchdog. When the watchdog is disabled in this way, the MCU can set bit WD_EN back to ‘1’ to enable the watchdog again, and the MCU can control the ENABLE_DRV bit when no other error-flags are set. When the MCU sets bit WD_EN back to ‘1’, the watchdog starts with the Long Window.

The watchdog locks the following configuration register bits when it goes out of the Long Window and starts the first watchdog sequence:

- WD_WIN1[6:0]
- WD_WIN2[6:0]
- WD_LONGWIN[7:0]
- WD_MODE_SELECT
- WD_QA_FDBK[1:0], WD_QA_LFSR[1:0] and WD_QUESTION_SEED[3:0]
- WD_RST_EN, WD_EN, WD_FAIL_TH[2:0] and WD_RST_TH[2:0]

2.13.3 MCU to Watchdog Synchronization

In order to go out of the Long Window and start the first watchdog sequence, the MCU must do the following:

- Clear bits WD_PWRHOLD (more detail in Section 2.13.4)
- Apply a pulse signal with a minimum pulse-width t_{WD_pulse} on the pre-assigned GPIO pin in the case the watchdog is configured for Trigger mode, or
- Write four times to WD_ANSWER[7:0] in the case the watchdog is configured for Q&A mode

When the MCU fails to get the watchdog out of the Long Window before the configured Long Window time interval (t_{LONG_WINDOW}) elapses, the device goes through a warm reset, and sets the WD_LONGWIN_TIMEOUT_INT. This bit latched until the MCU writes a ‘0’ to it ‘1’ to clear it.

2.13.4 Watchdog Disable Function

The watchdog in the TPS6594-Q1 device has a Watchdog Disable function to prevent an unwanted MCU reset in case the MCU is un-programmed or needs to be reprogrammed. In order to activate this Watchdog Disable function for an un-programmed MCU, DISABLE_WDOG pin must be asserted to a logic-high level for a time-interval longer than t_{WD_DIS} prior to the moment the device releases the nRSTOUT pin. If the Watchdog Disable function is activated in this way, the device sets bit WD_PWRHOLD to keep the watchdog in the Long Window. The watchdog stays in the Long Window until the MCU clears the WD_PWRHOLD bit.

In case the MCU needs to be reprogrammed while the watchdog monitors the correct operation of the MCU, the MCU can set bit WD_RETURN_LONGWIN to put the watchdog back in the Long Window. When the MCU set this bit, the watchdog returns to the Long Window after the current Watchdog Sequence completes. In order to make the watchdog stay in the Long Window as long as needed the MCU can either re-configure the Long Window (t_{LONG_WINDOW}) time interval, or set the WD_PWRHOLD bit. Once the MCU starts the first watchdog sequence (as described in Section 2.13.3), the MCU must clear bit WD_RETURN_LONGWIN before the end of the first watchdog sequence in order to continue the watchdog sequence operation.

2.13.5 Watchdog Sequence

Once the watchdog is out of the Long Window, each watchdog sequence starts with a Window-1 followed by a Window-2. The watchdog ends the current sequence and starts a next sequence when one of the events below occurs:

- The configured Window-2 time period elapses
- The watchdog detects a pulse signal with a minimum pulse-width t_{WD_pulse} on the pre-assigned GPIO pinif the watchdog is used in Trigger mode
- The watchdog detects four times a write access to WD_ANSWER[7:0] in case the watchdog is used in Q&A mode

The MCU can configure the time periods of the Window-1 ($t_{WINDOW1}$) and Window-2 ($t_{WINDOW2}$) with the bits WD_WIN1[6:0] and WD_WIN2[6:0] respectively, before starting the sequence.

Use Equation 7 and Equation 8 to calculate the minimum and maximum values for the $t_{WINDOW1}$ time interval.

$$t_{WINDOW1\_MIN} = (WD\_WIN1[6:0] + 1) \times 0.55 \times 0.95 \text{ ms} \tag{7}$$

$$t_{WINDOW1\_MAX} = (WD\_WIN1[6:0] + 1) \times 0.55 \times 1.05 \text{ ms} \tag{8}$$

Use Equation 9 and Equation 10 to calculate the minimum and maximum values for the $t_{WINDOW-2}$ time interval.

$$t_{WINDOW2\_MIN} = (WD\_WIN2[6:0] + 1) \times 0.55 \times 0.95 \text{ ms} \tag{9}$$

$$t_{WINDOW2\_MAX} = (WD\_WIN2[6:0] + 1) \times 0.55 \times 1.05 \text{ ms} \tag{10}$$

2.13.6 Watchdog Trigger Mode (Default Mode)

When the TPS6594-Q1 device is configured to use the Watchdog Trigger Mode, the watchdog receives the watchdog-triggers from the MCU on the pre-assigned GPIO pin. A rising edge on this GPIO pin, followed by a stable logic-high level on that pin for more than the maximum pulse time, $t_{WD\_pulse(max)}$, is a watchdog-trigger. The watchdog uses a deglitch filter with a t_{WD_pulse} filter time and an internal system clock to create the internally-generated trigger pulse from the watchdog-trigger on the pre-assigned GPIO pin.

The watchdog detects a *good event* when the watchdog-trigger comes in Window-2. The rising edge of the watchdog-trigger on the pre-assigned GPIO pin must occur for at least the t_{WD_pulse} time before the end of Window-2 to generate such a good event.

The watchdog detects a *bad event* when one of the following events occurs:

- The watchdog-trigger comes in Window-1. The rising edge of the watchdog-trigger on the pre-assigned GPIO pin must occur for at least the t_{WD_pulse} time before the end of Window-1 to generate such a bad event. In case of this bad event, the device sets bits WD_TRIG_EARLY and WD_BAD_EVENT.
- No watchdog-trigger comes in Window-2. In case of this bad event (also referred to as time-out event), the device sets bits WD_TIMEOUT and WD_BAD_EVENT.

Please consider that the minimum WD-pulse duration needs to meet the maximum deglitch time $t_{WD\_pulse\ (max)}$.

The status bit WD_BAD_EVENT is read-only. The watchdog clears the WD_BAD_EVENT status bit at the end of the watchdog-sequence.

Figure 2-25 shows the flow-chart of the watchdog in Trigger mode.

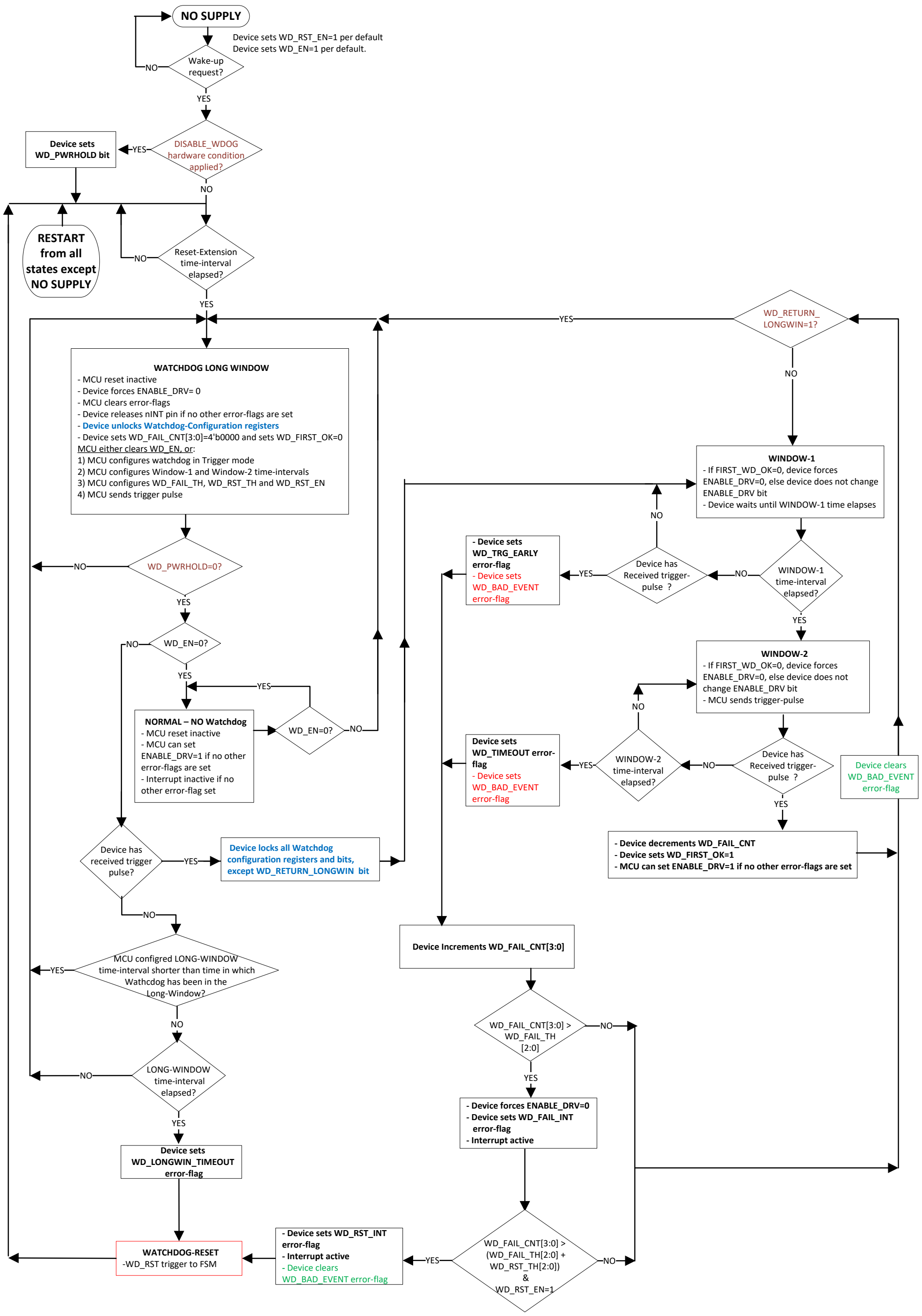


Figure 2-25. Flow Chart for WatchDog Monitor in Trigger Mode

Figure 2-26, Figure 2-27, Figure 2-28, Figure 2-29, and Figure 2-30 give examples of watchdog is trigger mode with good and bad events after device startup.

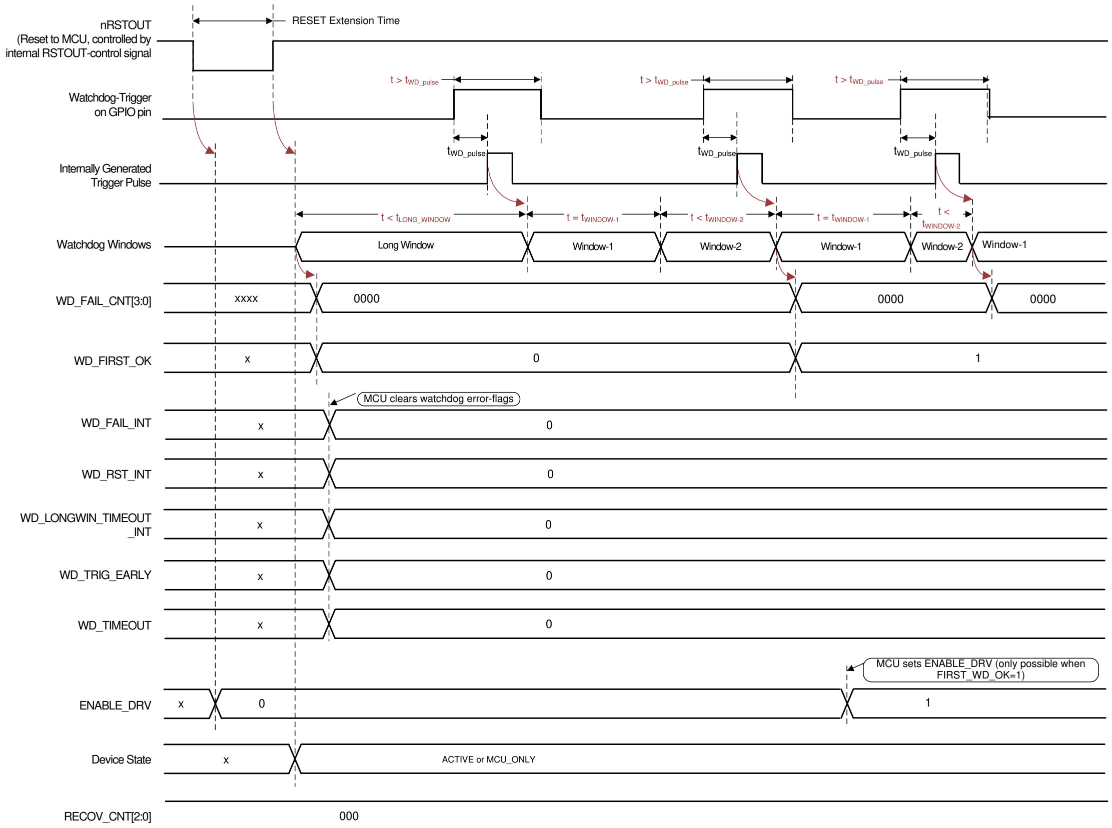


Figure 2-26. Watchdog in Trigger Mode – Normal MCU Startup with Correct Watchdog-Triggers

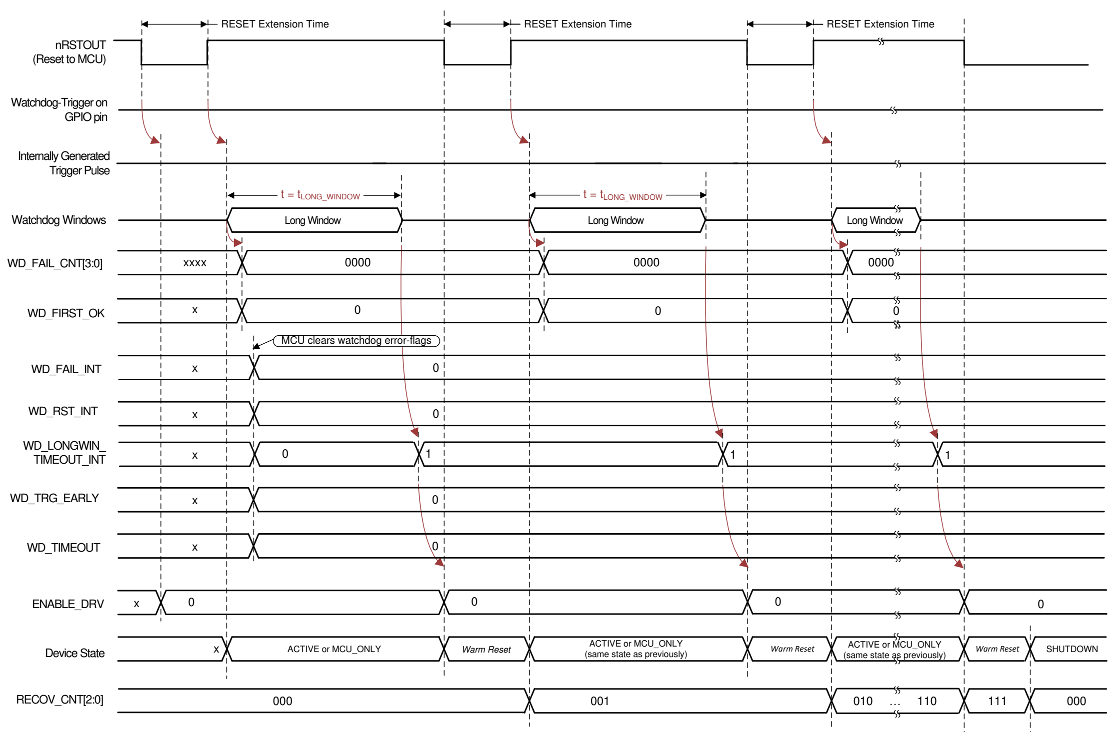


Figure 2-27. Watchdog in Trigger Mode – MCU Does Not Send Watchdog-Triggers After Startup

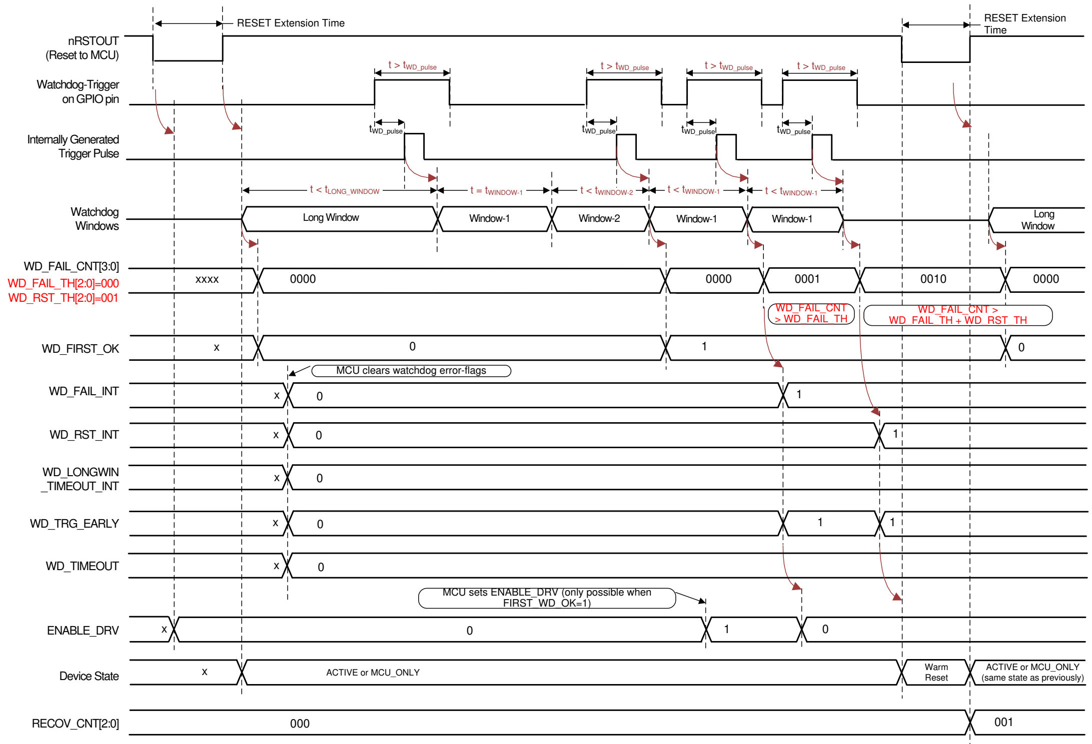


Figure 2-28. Watchdog in Trigger Mode – Bad Event (Watchdog-Triggers in Window-1) After Startup

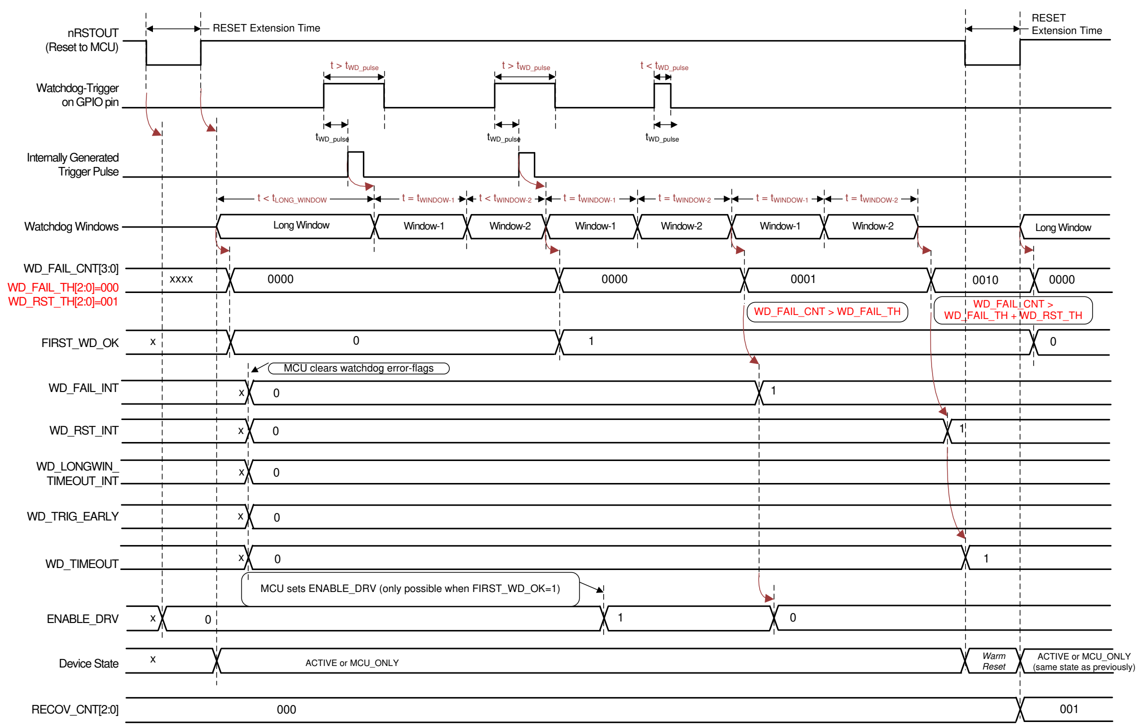


Figure 2-29. Watchdog in Trigger Mode – Bad Events (Too Short or no Trigger in Window-2) After Startup

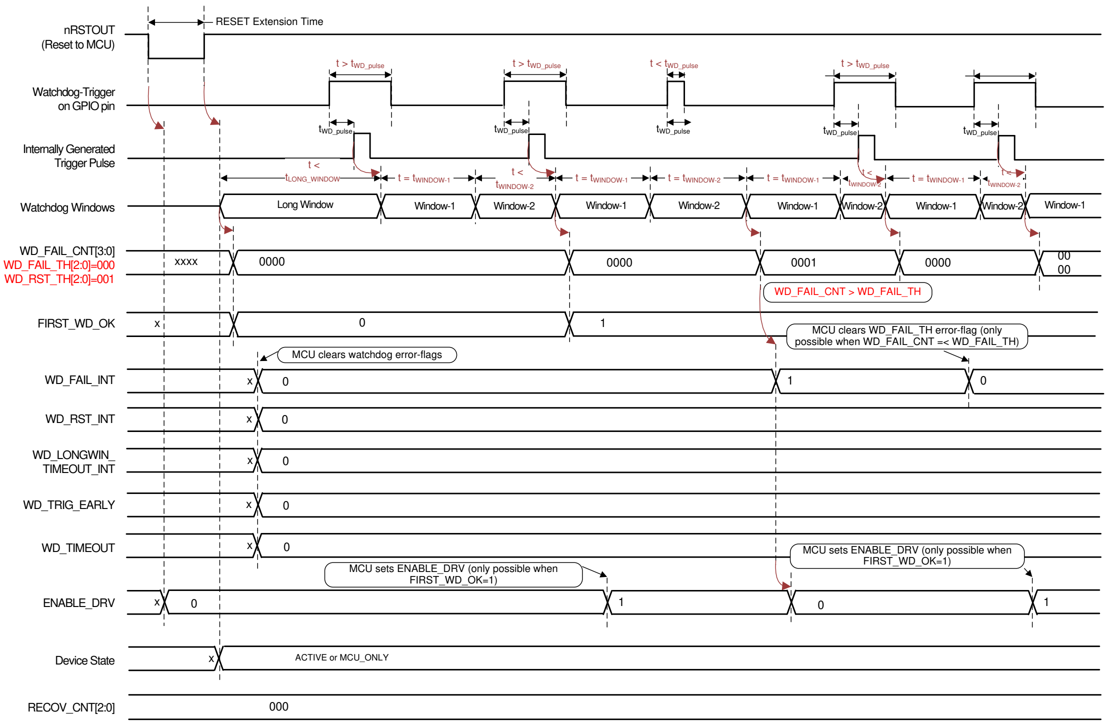


Figure 2-30. Watchdog in Trigger Mode – Good Events (Correct Watchdog-Triggers) After Startup, Followed by a Bad-Event (No Watchdog-Trigger in Window-2) and After That Followed by a Good Event.

2.13.7 Watchdog Question-Answer Mode

When the TPS6594-Q1 device is configured to use the Watchdog Question Answer mode, the watchdog requires specific messages from the MCU in specific time intervals to detect correct operation of the MCU.

During operation, the device provides a question for the MCU in WD_QUESTION[3:0]. The MCU performs a fixed series of arithmetic operations on this question to calculate the required 32-bit answer. This answer is split into four answer bytes: Answer-3, Answer-2, Answer-1, and Answer-0. The MCU writes these answer bytes one byte at a time into WD_ANSWER[7:0] from the SPI or the dedicated I^{2}C2 interface, mapped to GPIO1 and GPIO2 pins.

A good event occurs when the MCU sends the correct answer-bytes calculated for the current question in the correct watchdog window and in the correct sequence.

A bad event occurs when one of the events that follows occur:

- The MCU sends the correct answer-bytes, but not in the correct watchdog window.
- The MCU sends incorrect answer-bytes.
- The MCU returns correct answer-bytes, but in the incorrect sequence.

If the MCU stops providing answer-bytes for the duration of the watchdog time-period, the watchdog detects a time-out event. This time-out event sets the WD_TIMEOUT status bit, increments the WD_FAIL_CNT[3:0] counter, and starts a new watchdog sequence.

2.13.7.1 Watchdog Q&A Related Definitions

A question and answer are defined as follows:

Question A question is a 4-bit word (see Section 2.13.7.2).

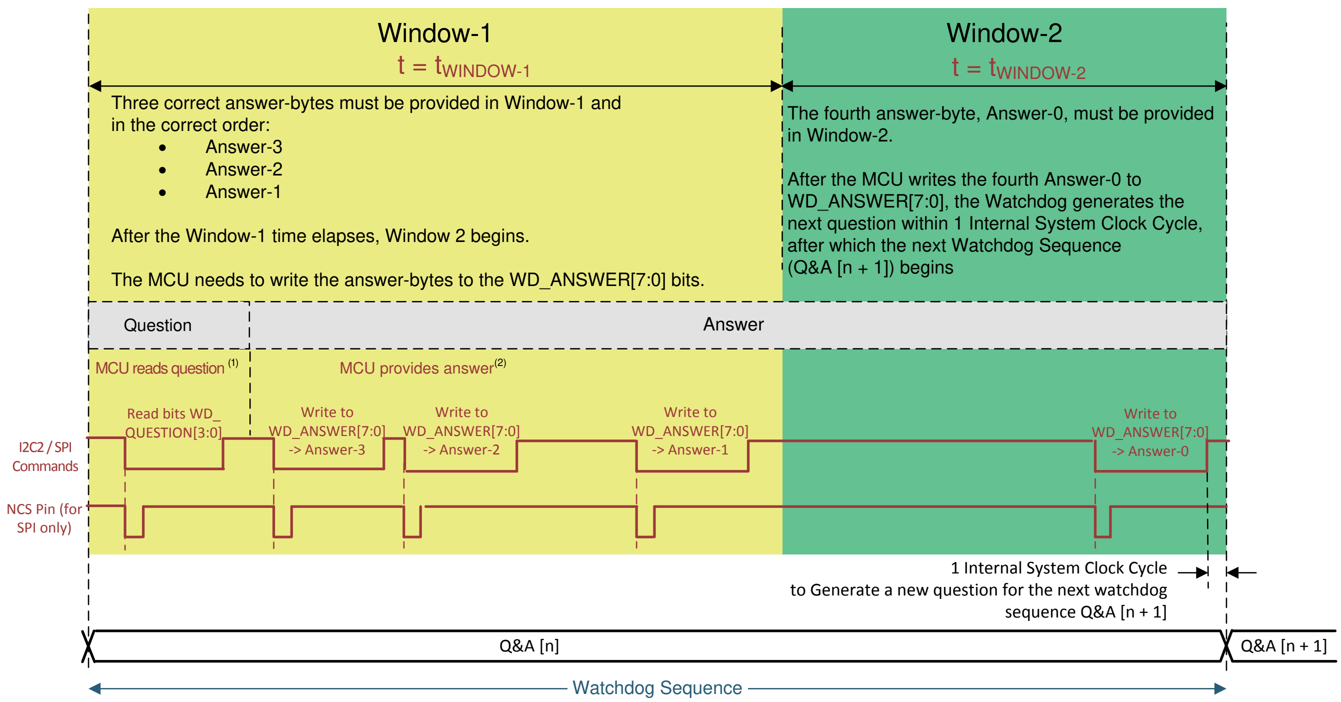
The watchdog provides the question to the MCU when the MCU reads the WD_QUESTION[3:0] bits.

The MCU can request each new question at the start of the watchdog sequence, but this is not required to calculate the answer. The MCU can also have a software implementation which generates the question according the circuit as shown in Figure 2-33. Nevertheless, the answer and therefore the answer-bytes are always based on the question generated inside the watchdog of the device. So if the MCU generates an incorrect question and gives answer-bytes calculated from this incorrect question, the watchdog detects a bad event

Answer An answer is a 32-bit word that is split into four answer bytes: Answer-3, Answer-2, Answer-1, and Answer-0.

The watchdog receives an answer-byte when the MCU writes to the WD_ANSWER[7:0] bits. For each question, the watchdog requires four correct answer-bytes from the MCU in the correct timing and order (Answer-3, Answer-2, and Answer-1 in Window 1 in the correct sequence, and Answer-0 in Window 2) to detect a good event.

The watchdog sequence in Q&A mode ends after the MCU writes the fourth answer byte (Answer-0), or after a time-out event when the Window-2 time-interval elapses.



(1) The MCU is not required to read the question. The MCU can give correct answer-bytes Answer-3, Answer-2, Answer-1 as soon as Window-1 starts. The next watchdog sequence always starts in 1 system clock cycle after the watchdog receives the final Answer-0.
(2) The MCU can put other I^2C or SPI commands in-between the write-commands to WD_ANSWER[7:0] (even re-requesting the question). This has no influence on the detection of a good event, as long as the three correct answer-bytes in Window-1 are in the correct sequence, and the fourth correct answer-byte is provided before the configured Window-2 time-interval elapses.

Figure 2-31. Watchdog Sequence in Q&A Mode

2.13.7.2 Question Generation

The watchdog uses a 4-bit *question counter* (QST_CNT[3:0] bits in Figure 2-32), and a 4-bit Markov chain to generate a 4-bit question. The MCU can read this question in the WD_QUESTION[3:0] bits. The watchdog generates a new question when the question counter increments, which only occurs when the watchdog detects a good event. The watchdog does not generate a new question when it detects a bad event or a time-out event.

The question-counter provides a clock pulse to the Markov chain when it transitions from 4’b1111 to 4’b0000. The question counter and the Markov chain are set to the default value of 4’b0000 when the watchdog goes out of the Long Window.

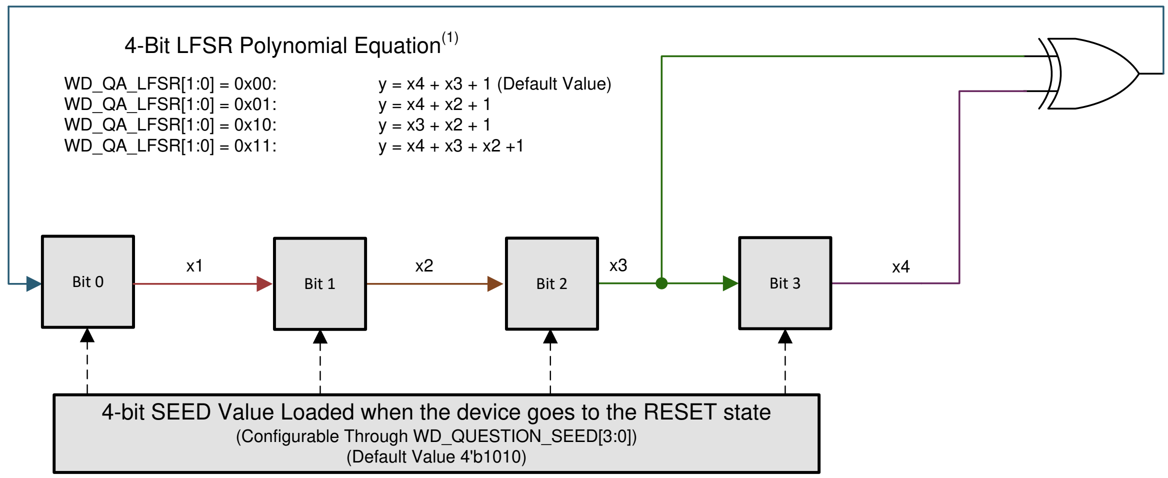
Note

The Question-Generator is only re-initialized (starting with question 0000) at device power-up. In following situations, the MCU software needs to read the current question in order to synchronize with the Question-Generator:

- after MCU re-boot from a warm-reset
- after MCU software sets bit WD_RETURN_LONGWIN=1 to put the Watchdog back into Long Window
- after MCU wrote WD_EN=0, then reenable Watchdog again with WD_EN=1

Figure 2-32 shows the logic combination for the WD_QUESTION[3:0] generation.

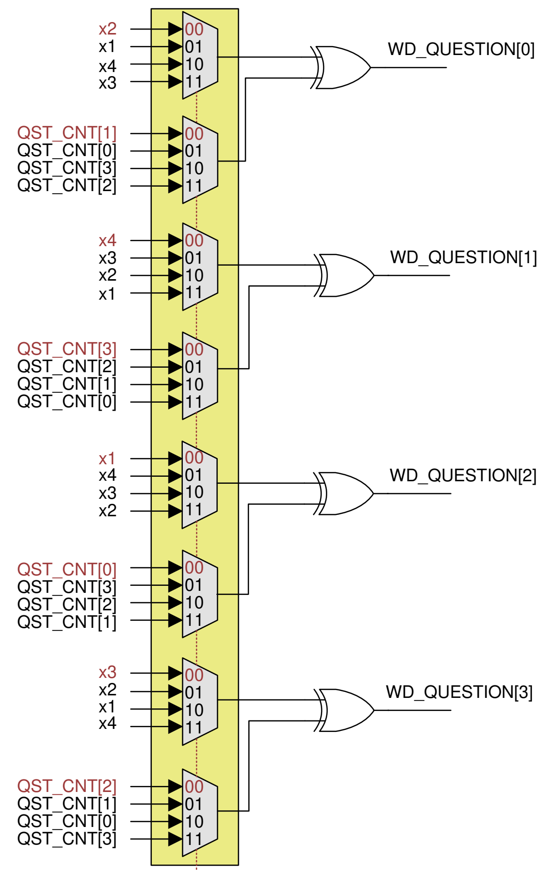
The logic combination of the question-counter with the WD_ANSW_CNT[1:0] status bits generates the reference answer-bytes as shown in Figure 2-33.

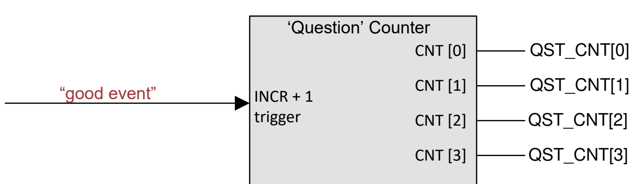


	x1	x2	x3	x4
SEED	**1**	**0**	**1**	**0**
1	1	1	0	1
2	1	1	1	0
3	1	1	1	1
4	0	1	1	1
5	0	0	1	1
6	0	0	0	1
7	1	0	0	0
8	0	1	0	0
9	0	0	1	0
10	1	0	0	1
11	1	1	0	0
12	0	1	1	0
13	1	0	1	1
14	0	1	0	1
15	1	0	1	0

Question Sequence Order 1 to 15

The default question-sequence order with the default WD_QUESTION_SEED[3:0] and WD_QA_LFSR[1:0] values





Feedback settings are controllable through the bits WD_QA_FDBK[1:0]
(Default value is 2'b00; the selected signals are in red)

A. If current, the y value is 0000, the next y value will be 0001, and any further question generation begins from this value.

Figure 2-32. Watchdog Question Generation

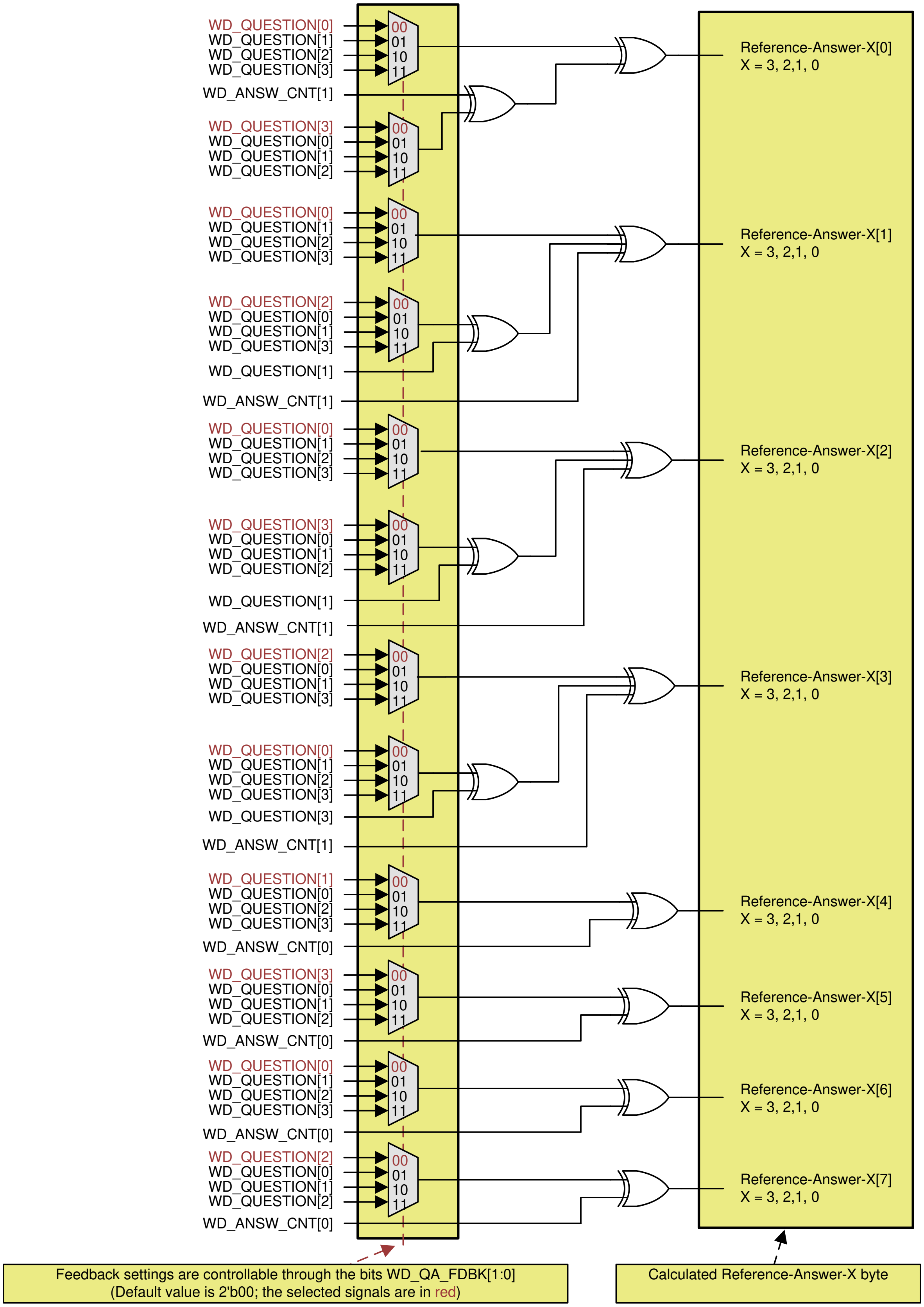


Figure 2-33. Watchdog Reference Answer Calculation

2.13.7.3 Answer Comparison

The 2-bit, watchdog-answer counter, WD_ANSW_CNT[1:0], counts the number of received answer-bytes and controls the generation of the reference answer-byte as shown in Figure 1-2. At the start of each watchdog sequence, the default value of the WD_ANSW_CNT[1:0] counter is 2’b11 to indicate that the watchdog expects the MCU to write the correct Answer-3 in WD_ANSWER[7:0].

The device sets the WD_ANSW_ERR status bit as soon as one answer byte is not correct. The device clears this status bit only if the MCU writes a ‘1’ to this bit.

2.13.7.3.1 Sequence of the 2-bit Watchdog Answer Counter

The sequence of the 2-bit, watchdog answer-counter is as follows for each counter value:

- WD_ANSW_CNT[1:0] = 2‘b11:
  1. The watchdog calculates the reference Answer-3.
  2. A write access occurs. The MCU writes the Answer-3 byte in WD_ANSWER[7:0].
  3. The watchdog compares the reference Answer-3 with the Answer-3 byte in WD_ANSWER[7:0].
  4. The watchdog decrements the WD_ANSW_CNT[1:0] bits to 2b‘10 and sets the WD_ANSW_ERR status bit to 1 if the Answer-3 byte was incorrect.
- WD_ANSW_CNT[1:0] = 2b‘10:
  1. The watchdog calculates the reference Answer-2.
  2. A write access occurs. The MCU writes the Answer-2 byte in WD_ANSWER[7:0].
  3. The watchdog compares the reference Answer-2 with the Answer-2 byte in WD_ANSWER[7:0]..
  4. The watchdog decrements the WD_ANSW_CNT[1:0] bits to 2b‘01 and sets the WD_ANSW_ERR status bit to 1 if the Answer-2 byte was incorrect.
- WD_ANSW_CNT[1:0] = 2b‘01:
  1. The watchdog calculates the reference Answer-1.
  2. A write access occurs. The MCU writes the Answer-1 byte in WD_ANSWER[7:0].
  3. The watchdog compares the reference Answer-1 with the Answer-1 byte in WD_ANSWER[7:0]..
  4. The watchdog decrements the WD_ANSW_CNT[1:0] bits to 2b‘00 and sets the WD_ANSW_ERR status bit to 1 if the Answer-1 byte was incorrect.
- WD_ANSW_CNT[1:0] = 2b‘00:
  1. The watchdog calculates the reference Answer-0.
  2. A write access occurs. The MCU writes the Answer-0 byte in WD_ANSWER[7:0].
  3. The watchdog compares the reference Answer-0 with the Answer-0 byte in WD_ANSWER[7:0].
  4. The watchdog sets the WD_ANSW_ERR status bit to 1 if the Answer-0 byte was incorrect.
  5. The watchdog starts a new watchdog sequence and sets the WD_ANSW_CNT[1:0] to 2‘b11’.

     The MCU needs to clear the bit by writing a '1' to the WD_ANSW_ERR bit.

Table 2-16. Set of Questions and Corresponding Answer-Bytes Using the Default Setting of WD_QA_CFG Register

WD QUESTION	ANSWER-BYTES (EACH BYTE TO BE WRITTEN INTO WD_ANSWER[7:0])			
	ANSWER-3	ANSWER-2	ANSWER-1	ANSWER-0
WD_QUESTION[3:0]	WD_ANSW_CNT [1:0] = 2’b11	WD_ANSW_CNT [1:0] = 2’b10	WD_ANSW_CNT [1:0] = 2’b01	WD_ANSW_CNT [1:0] = 2’b00
0x0	FF	0F	F0	00
0x1	B0	40	BF	4F
0x2	E9	19	E6	16
0x3	A6	56	A9	59
0x4	75	85	7A	8A
0x5	3A	CA	35	C5
0x6	63	93	6C	9C
0x7	2C	DC	23	D3
0x8	D2	22	DD	2D
0x9	9D	6D	92	62
0xA	C4	34	CB	3B
0xB	8B	7B	84	74
0xC	58	A8	57	A7
0xD	17	E7	18	E8
0xE	4E	BE	41	B1
0xF	01	F1	0E	FE

2.13.7.4 Watchdog Sequence Events and Status Updates

The watchdog sequence events are as follows for the different scenarios listed:

- A good event occurs when all answer bytes are correct in value and timing. After such a good event, following events will occur:
  1. The WD_FAIL_CNT[2:0] counter decrements by one at the end of the watchdog-sequence
  2. The question-counter increments by one and the watchdog generates a new question
- A bad event occurs when all answer-bytes are correct in value but not in correct timing. After such a bad event, following events will occur:
  1. The WD_SEQ_ERR and WD_BAD_EVENT status bits are set if Window-1 time-interval elapses before watchdog has received Answer-3, Answer-2 and Answer-1
  2. The WD_ANSW_EARLY and WD_BAD_EVENT status bits are set if watchdog receives all four answers in Window-1
  3. The WD_FAIL_CNT[2:0] counter increments by one at the end of the watchdog-sequence
  4. The question-counter does not change, and hence the watchdog does not generate a new question
- A bad event occurs when one or more of the answer-bytes are not correct in value but in correct timing. After such a bad event, following events will occur:
  1. The WD_ANSW_ERR and WD_BAD_EVENT status bits are set as soon as the watchdog detects an incorrect answer-byte
  2. The WD_FAIL_CNT[2:0] counter increments by one at the end of the watchdog-sequence
  3. The question-counter does not change, and hence the watchdog does not generate a new question
- A bad event occurs when one or more of the answer-bytes are not correct in value and not in correct timing. After such a bad event, following events will occur:
  1. The WD_ANSW_ERR and WD_BAD_EVENT status bits are set as soon as the watchdog detects an incorrect answer-byte
  2. The WD_SEQ_ERR and WD_BAD_EVENT status bits are set if Window-1 time-interval elapses before watchdog has received Answer-3, Answer-2 and Answer-1
  3. The WD_ANSW_EARLY and WD_BAD_EVENT status bits are set if watchdog receives all four answer-bytes in Window-1

4. The WD_FAIL_CNT[2:0] counter increments by one at the end of the watchdog-sequence
5. The question-counter does not change, and hence the watchdog does not generate a new question

- A time-out event occurs when the device receives less than 4 answer-bytes before Window-2 time-interval elapses. After a time-out event occurs, following events will occur:
  1. WD_SEQ_ERR and WD_BAD_EVENT status bits are set if Window-1 time-interval elapses before watchdog has received Answer-3, Answer-2 and Answer-1
  2. The WD_TIMEOUT and WD_BAD_EVENT status bits are set at the end of the watchdog-sequence
  3. The WD_FAIL_CNT[2:0] counter increments by one at the end of the watchdog-sequence
  4. The question-counter does not change, and hence the watchdog does not generate a new question

The status bit WD_BAD_EVENT is read-only. The watchdog clears the WD_BAD_EVENT status bit at the end of the watchdog-sequence.

The status bits WD_SEQ_ERR, WD_ANSW_EARLY, and WD_TIMEOUT are latched until the MCU writes a ‘1’ to these bits. If one or more of these status bits are set, the watchdog can still detect a good event in the next watchdog-sequence. These status bits are read-only. The watchdog clears the WD_BAD_EVENT status bit at the end of the watchdog-sequence.

Note

The WD_FIRST_OK bit is set after receiving 4 answers in the correct time frames, regardless of the correctness of the answers. In order to not clear the bit in case of incorrect answers, the following procedure is recommended:

- When WD_FIRST_OK bit is set, the MCU shall read the WD_FAIL_CNT (address 0x40).
- If WD_FAIL_CNT is zero, the MCU shall clear the WD_FIRST_OK bit.
- If WD_FAIL_CNT is not zero, the MCU shall continue sending frames until WD_FAIL_CNT decrements before clearing WD_FIRST_OK.

Figure 2-34 shows the flow-chart of the watchdog in Q&A mode.

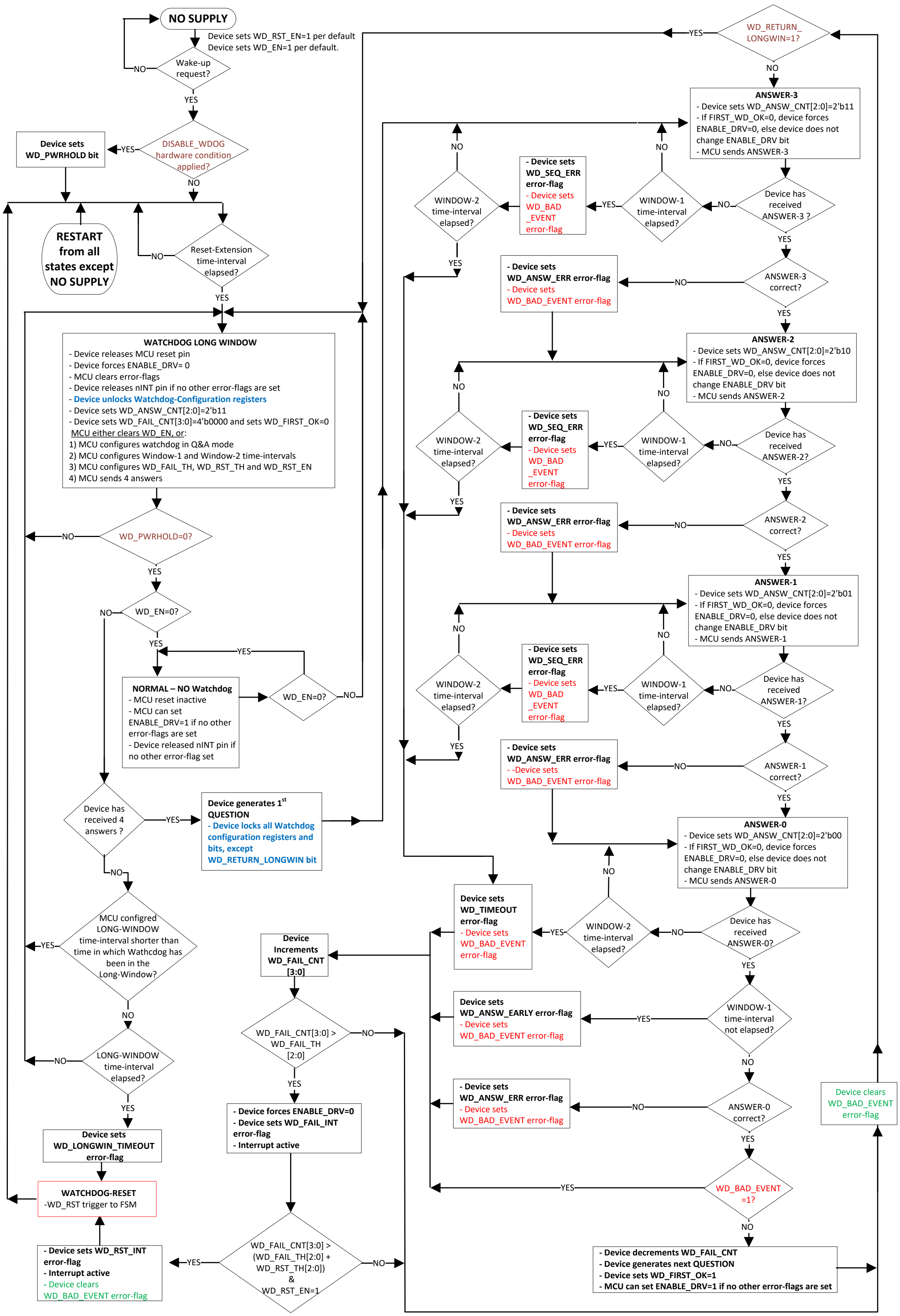


Figure 2-34. Flow Chart for WatchDog in Q&A Mode

2.14 Error Signal Monitor (ESM) Error

The TPS6594-Q1 device has two Error Signal Monitor (ESMs): one ESM_MCU to monitor the MCU error output signal at the nERR_MCU input pin, and one ESM_SoC to monitor the SoC error output signal at the nERR_SoC input pin.

By default, each ESM is disabled at start-up of the TPS6594-Q1 device. To start each ESM, the MCU sets the start bits ESM_MCU_START or ESM_SOC_START for the respective ESM through software after the system is powered up and the initial software configuration is completed. If the MCU clears a start bit, the prospective ESM stops monitoring its input pin. The MCU can set the ENABLE_DRV bit only when the MCU has either started or disabled the ESM. When the prospective ESM is started, the following configuration registers are write protected and can only be read:

Configuration registers write-protected by the ESM_MCU_START register bit:
- ESM_MCU_DELAY1_REG
- ESM_MCU_DELAY2_REG
- ESM_MCU_MODE_CFG
- ESM_MCU_HMAX_REG
- ESM_MCU_HMIN_REG
- ESM_MCU_LMAX_REG
- ESM_MCU_LMIN_REG

Configuration registers write-protected by the ESM_SOC_START register bit:
- ESM_SOC_DELAY1_REG
- ESM_SOC_DELAY2_REG
- ESM_SOC_MODE_CFG
- ESM_SOC_HMAX_REG
- ESM_SOC_HMIN_REG
- ESM_SOC_LMAX_REG
- ESM_SOC_LMIN_REG

ESM uses a deglitch-filter with deglitch-time t_{degl_ESMx} to monitor its related input pin.

The MCU can configure the ESM in two different modes which are defined as follows:

Level Mode the ESM detects an ESM-error when the input pin remains low for a time equal to or longer than the deglitch-time t_{degl_ESMx}.

To select this mode for the ESM_MCU, the MCU must clear bit ESM_MCU_MODE. To select this mode for the ESM_SoC, the MCU must clear bit ESM_SOC_MODE. See Section 2.14.2 for further detail

PWM Mode the ESM monitors a PWM signal at its input pin. The ESM detects a bad-event when the frequency or duty cycle of the PWM input signal deviates from the expected signal. The ESM detects a good-event when both frequency and duty cycle of the PWM signal match with the expected signal for one signal period.

The ESM has an error-counter (ESM_MCU_ERR_CNT[4:0] or ESM_SOC_ERR_CNT[4:0]), which increments with +2 after each bad-event, and decrements with -1 after each good-event. The ESM detects an ESM-error when the error-counter value is more than its related threshold value.

To select this mode for the ESM_MCU, the MCU must set bit ESM_MCU_MODE. To select this mode for the ESM_SoC, the MCU must set bit ESM_SOC_MODE. See Section 2.14.3 for further details.

The MCU can configure each ESM as long as its related start bit is cleared to 0 (bit ESM_MCU_START or ESM_SOC_START). As soon as the MCU sets a start bit, the device sets a write-protection on the configuration registers of the related ESM except the related start bits ESM_MCU_START and ESM_SOC_START.

2.14.1 ESM Error-Handling Procedure

Each ESM has two of its own configurable delay-timers, which are reset at when the device clears the respective ESM_x_START bit. When an ESM detects an ESM-error, the ESM starts the following procedure:

1. The device sets interrupt bit ESM_MCU_PIN_INT or ESM_SOC_PIN_INT, and pulls the nINT pin low.
2. The ESM starts the delay-1 timer (configurable through related ESM_MCU_DELAY1[7:0] or ESM_SOC_DELAY1[7:0] bits).
3. If the ESM-error is no longer present and MCU has cleared the related interrupt bit ESM_MCU_PIN_INT or ESM_SOC_PIN_INT before the delay-1 timer elapses, the device will release the nINTpin, the ESM will reset the delay-1 and delay-2 timers and continues to monitor its input pin.
4. If the ESM-error is still present and the delay-1 timer elapses, then the ESM clears the ENABLE_DRV bit if bit ESM_MCU_ENDRV=1 or if bit ESM_SOC_ENDRV=1.
5. If the delay-2 timer (configurable through related ESM_MCU_DELAY2[7:0] or ESM_SOC_DELAY2[7:0] bits) is set to 0, then the ESM skips steps 6 of this list, and performs step 7.
6. If the delay-2 timer is not set to 0, then:
   a. For ESM_MCU, the device sets interrupt bit ESM_MCU_FAIL_INT and pulls the nINT pin low and starts the delay-2 timer.
   b. For ESM_SOC: the device sets interrupt bit ESM_SOC_FAIL_INT, pulls the nINT pin low and starts the delay-2 timer.
7. If the ESM-error is no longer present and the MCU has cleared the related interrupt bits listed below before the delay-2 timer elapses, the device will release the nINTpin, the ESM will reset the delay-1 and delay-2 timers and continues to monitor its input pin:
   - ESM_MCU_PIN_INT (and ESM_MCU_FAIL_INT if set in step 6), or
   - ESM_SOC_PIN_INT (and ESM_SOC_FAIL_INT if set in step 6)
8. If the ESM-error is still present and the delay-2 timer elapses, then:
   a. For ESM_MCU, the device:
      i. clears the ESM_MCU_START BIT
      ii. sets interrupt bit ESM_MCU_RST_INT, which the device handles as an ESM_MCU_RST trigger for FSM, described in Table 2-13
      iii. After this trigger handling completes, the device re-initializes the ESM_MCU
   b. For ESM_SoC, the device:
      i. clears the ESM_SOC_START bit
      ii. sets interrupt bit ESM_SOC_RST_INT, which the device handles as an ESM_SOC_RST trigger for FSM, described in Table 2-13
      iii. After this trigger handling completes, the device re-initializes the ESM_SoC

ESM_MCU_DELAY1[7:0] and ESM_SOC_DELAY1[7:0] set the delay-1 time-interval ($t_{DELAY\text{-}1}$) for the related ESM_MCU or ESM_SoC. Use Equation 11 and Equation 12 to calculate the worst-case values for the $t_{DELAY\text{-}1}$:

$$\text{Min. } t_{DELAY\text{-}1} = (\text{ESM\_x\_DELAY1[7:0]} \times 2.048 \text{ ms}) \times 0.95 \tag{11}$$

$$\text{Max. } t_{DELAY\text{-}1} = (\text{ESM \_x\_DELAY1[7:0]} \times 2.048 \text{ ms}) \times 1.05 \tag{12}$$

, in which x stands for either MCU or SoC.

ESM_MCU_DELAY2[7:0] or ESM_SOC_DELAY2[7:0] bits set the delay-2 time-interval ($t_{DELAY\text{-}2}$) for the related ESM_MCU or ESM_SoC. Use Equation 13 and Equation 14 to calculate the worst-case values for the $t_{DELAY\text{-}2}$:

$$\text{Min. } t_{DELAY\text{-}2} = (\text{ESM\_x\_DELAY2[7:0]} \times 2.048 \text{ ms}) \times 0.95 \tag{13}$$

$$\text{Max. } t_{DELAY\text{-}2} = (\text{ESM \_x\_DELAY2[7:0]} \times 2.048 \text{ ms}) \times 1.05 \tag{14}$$

, in which x stands for either MCU or SoC.

2.14.2 Level Mode

In Level Mode, after MCU has set the start bit (bit ESM_MCU_START or bit ESM_SOC_START), the ESM monitors its nERR_MCU or nERR_SoC input pin. Each ESM detects an ESM-error when the voltage level on its input pin remains low for a time equal or longer than the deglitch-time t_{degli_ESMx}. When an ESM detects an ESM-error, it starts the ESM Error-Handling procedure as described in Section 2.14.1. If the voltage level on its input pin remains high for a time equal or longer than the deglitch-time t_{degli_ESMx} before the elapse of the configured delay-1 or delay-2 time-intervals, the ESM-error is no longer present and the ESM stops the Error-Handling Procedure as described in Section 2.14.1

For a complete overview on how the ESM works in Level Mode, please refer to the flow-chart in Figure 2-35. Figure 2-36, Figure 2-37, Figure 2-38, and Figure 2-39 show example wave forms for several error-cases for the ESM in Level Mode. In these examples, only the ESM_MCU is shown

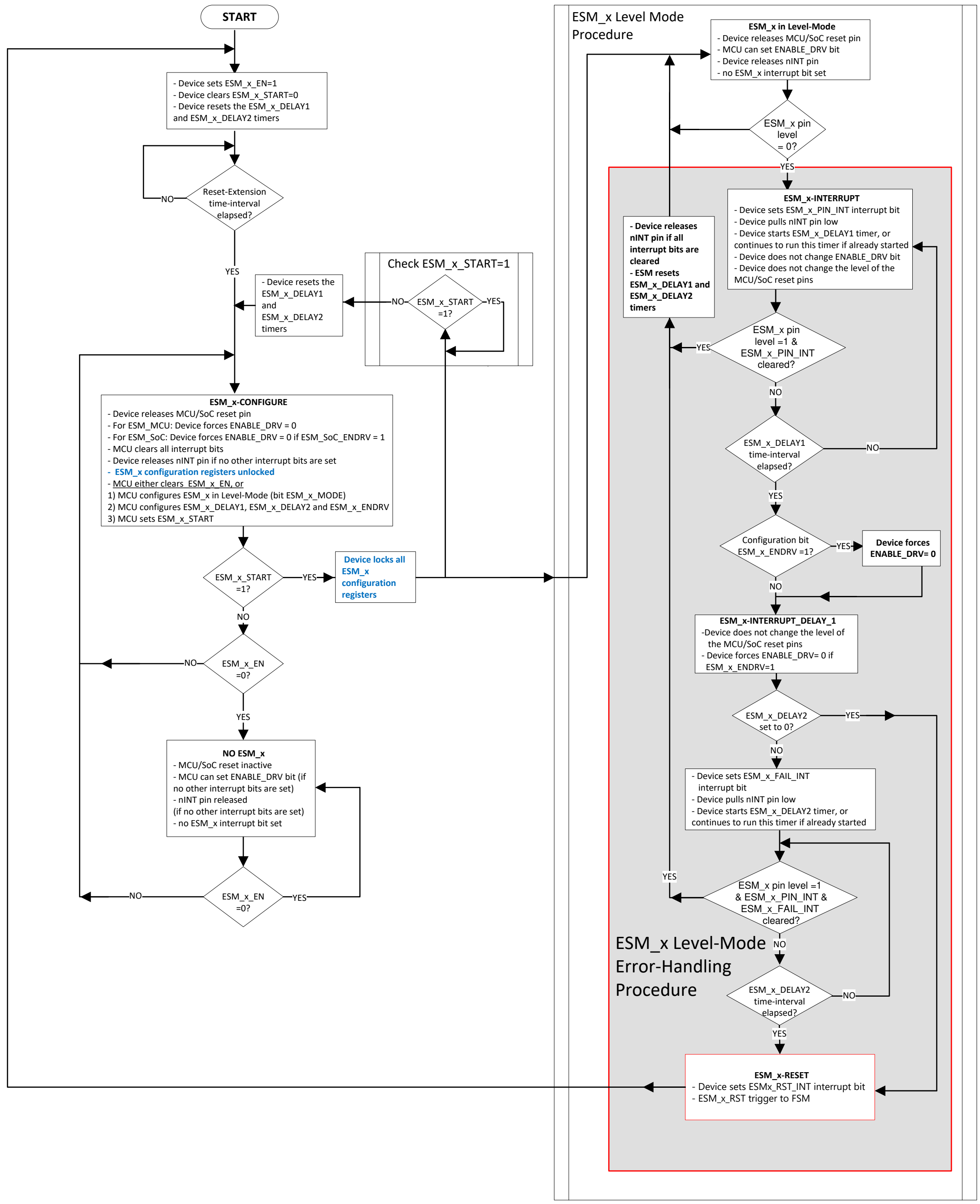


Figure 2-35. Flow Chart for Error Detection in Level Mode

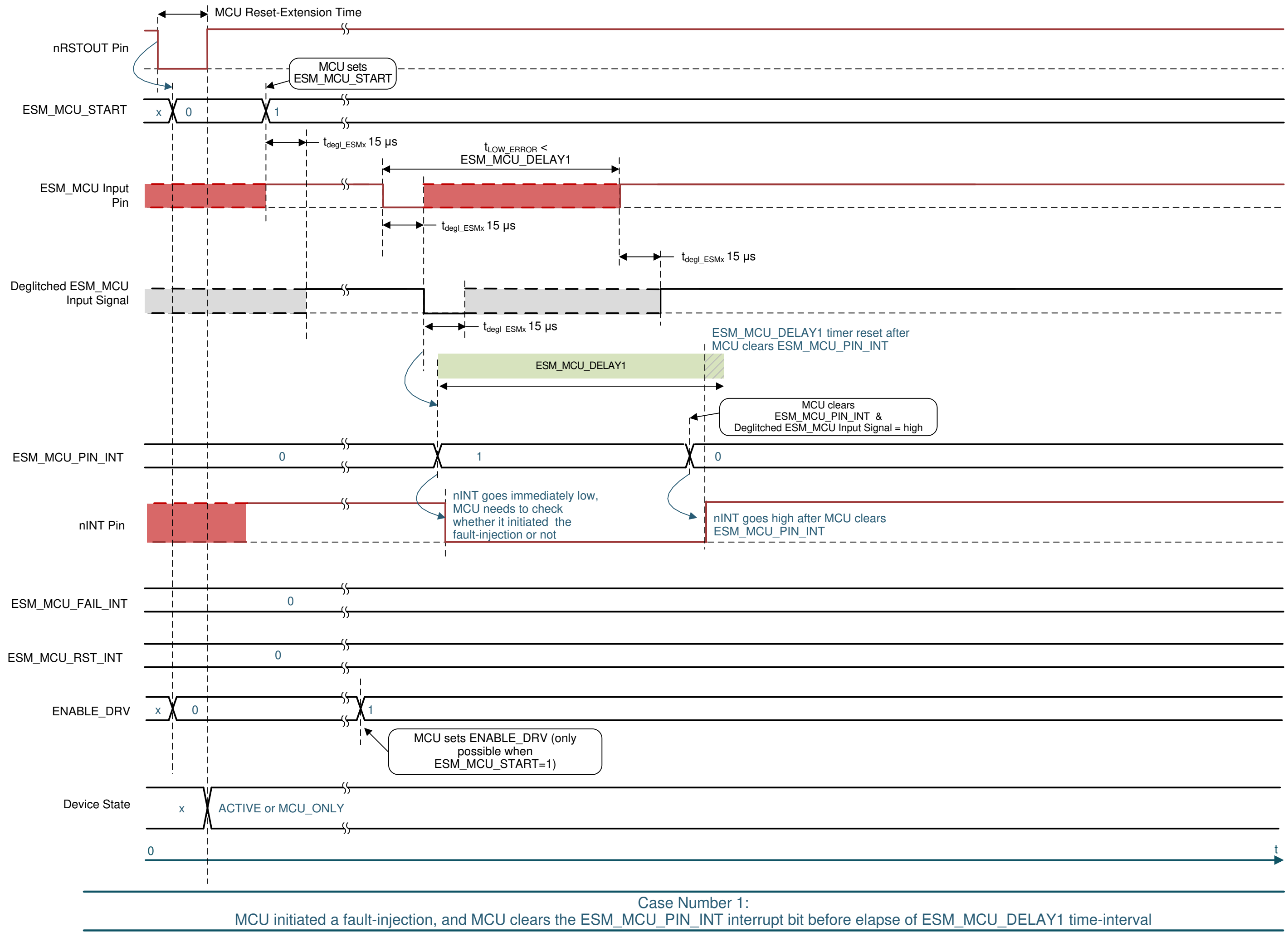


Figure 2-36. Example Waveform for ESMx in Level Mode - Case Number 1: ESM_MCU Signal Recovers Before Elapse of Delay-1 time-interval

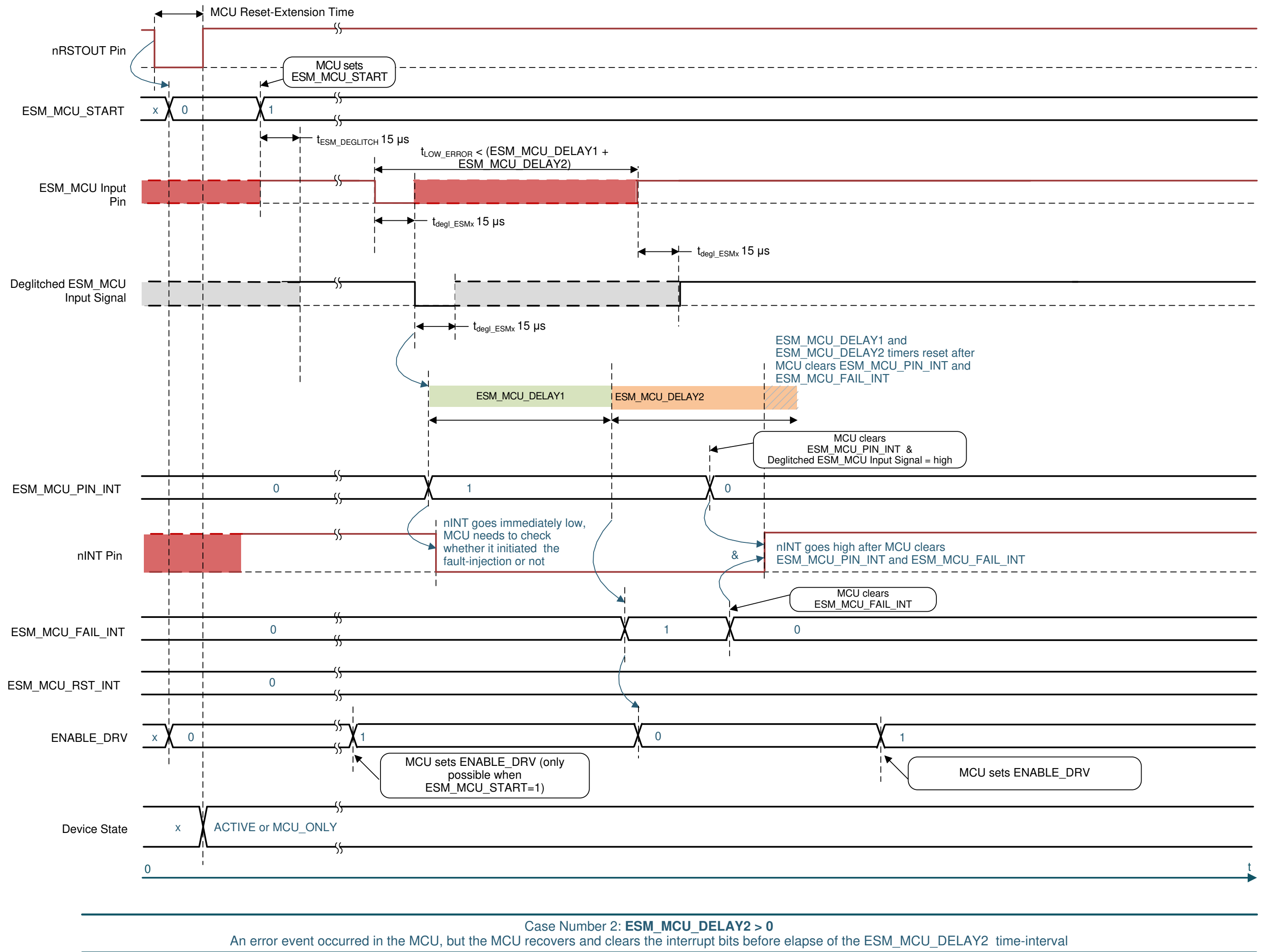


Figure 2-37. Example Waveform for ESM in Level Mode - Case Number 2: Delay-2 not set to 0 and ESM_MCU_ENDRV=1, ESM_MCU Signal Recovers Elapse of Delay-2 time-interval

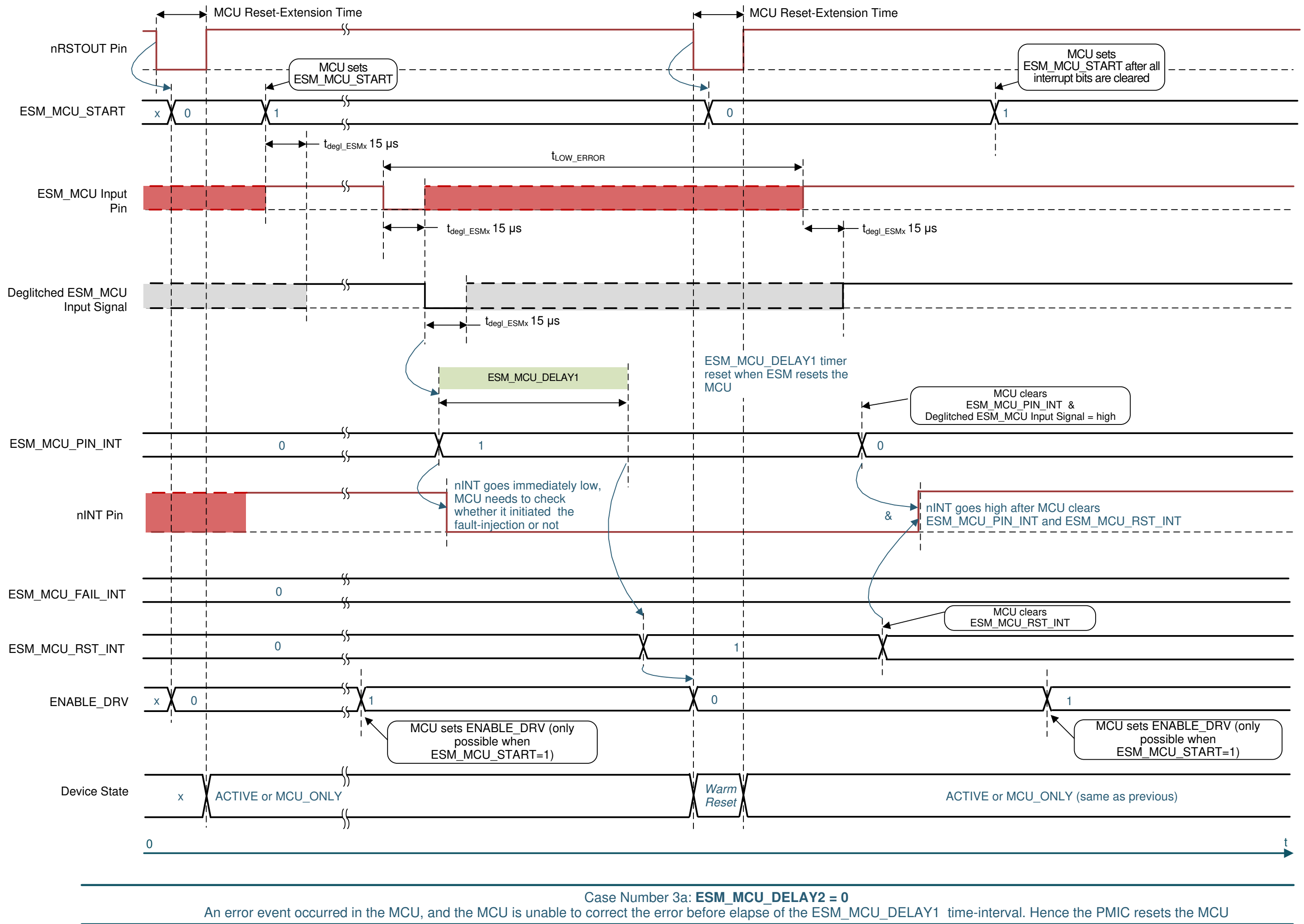


Figure 2-38. Example Waveform for ESM in Level Mode - Case Number 3a: Delay-2 set to 0 and ESM_MCU_ENDRV=1, ESM_MCU input signal recovers too late and MCU-reset occurs

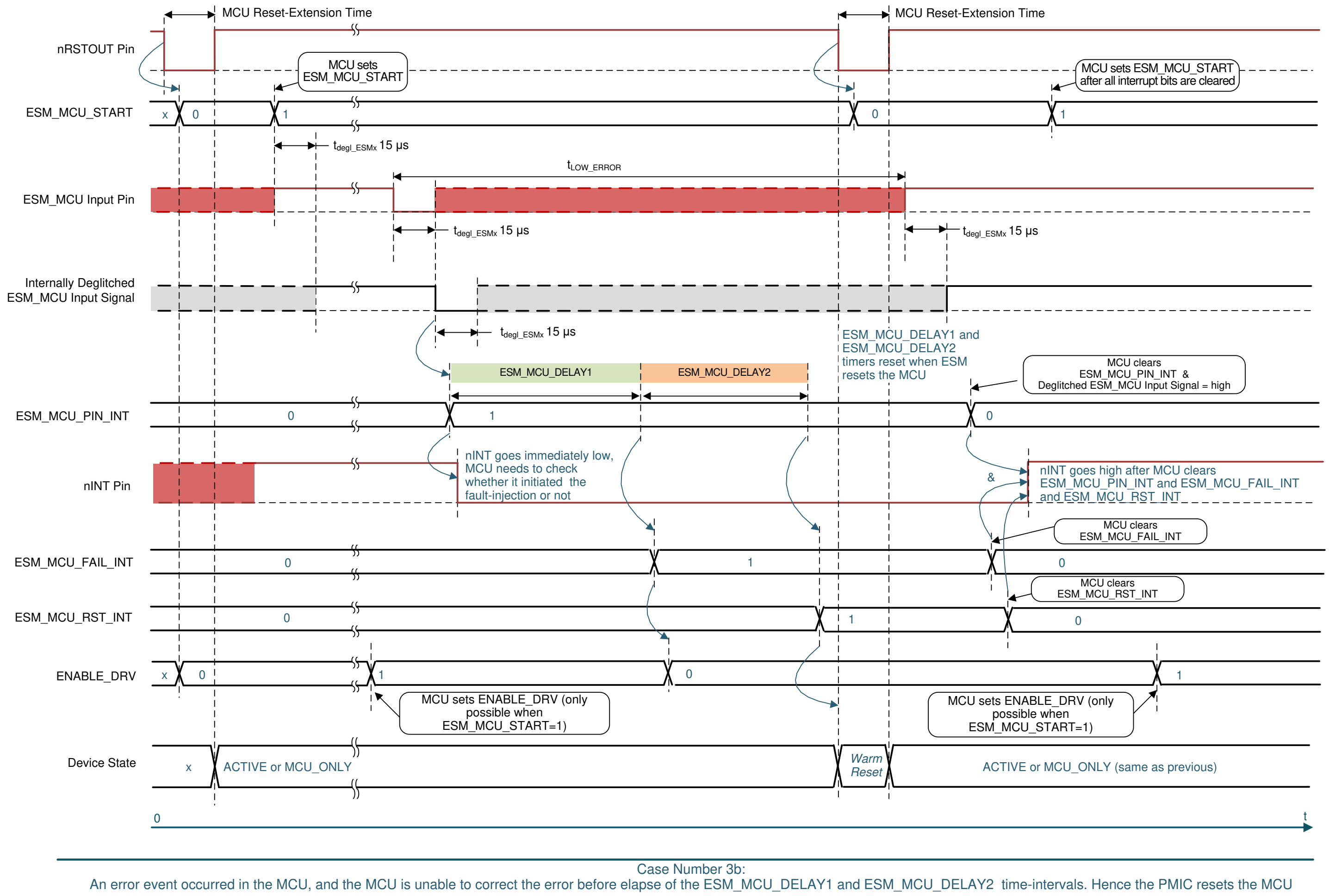


Figure 2-39. Example Waveform for ESM in Level Mode - Case Number 3b: Delay-2 not set to 0 and ESM_MCU_ENDRV=1, ESM_MCU input signal recovers too late and MCU-reset occurs

2.14.3 PWM Mode

2.14.3.1 Good-Events and Bad-Events

In PWM mode, each ESM monitors the high-pulse and low-pulse duration times its PWM inputs signal as follows:

- after a falling edge, the ESM starts monitoring the low-pulse time-duration. If the input signal remains low after exceeding the maximum low-pulse time-threshold ($t_{LOW\_MAX\_TH}$), the ESM detects a bad event and the low-pulse duration counter reinitializes. Each time the signal further exceeds the maximum threshold, the ESM detects a bad event. On the next rising edge on the input signal, the ESM starts the high-pulse time-duration monitoring
- after a rising edge, the ESM starts monitoring the high-pulse time-duration. If the input signal remains high after exceeding the maximum high-pulse time-threshold ($t_{HIGH\_MAX\_TH}$), the ESM detects a bad event and the high-pulse duration counter reinitializes. Each time the signal further exceeds the maximum threshold, the ESM detects a bad event. On the next falling edge on the input signal, the ESM starts the low-pulse time-duration monitoring.

In addition, each ESM detects a bad-event in PWM mode if one of the events that follow occurs on the deglitched signal of the related input pin nERR_MCU or nERR_SoC:

- A high-pulse time-duration which is longer than the maximum high-pulse time-threshold ($t_{HIGH\_MAX\_TH}$) that is configured in corresponding ESM_MCU_HMAX[7:0] or ESM_SOC_HMAX[7:0].
- A high-pulse time-duration which is shorter than the minimum high-pulse time-threshold ($t_{HIGH\_MIN\_TH}$) that is configured in corresponding ESM_MCU_HMIN[7:0] or ESM_SOC_HMIN[7:0].

- A low-pulse time-duration which is longer than the maximum low-pulse time-threshold ($t_{LOW\_MAX\_TH}$) that is configured in corresponding ESM_MCU_LMAX[7:0] or ESM_SOC_LMAX[7:0].
- A low-pulse time-duration which is less than the minimum low-pulse time-threshold ($t_{LOW\_MIN\_TH}$) that is configured in corresponding ESM_MCU_LMIN[7:0] or ESM_SOC_LMIN[7:0].

The ESM detects a good-event in PWM mode if one of the events that follow occurs on the deglitched signal of the related input pin nERR_MCU or nERR_SoC:

- a low-pulse time-duration within the minimum and maximum low-pulse time-thresholds is followed by a high-pulse time-duration within the minimum and maximum high-pulse time-thresholds, or
- a high-pulse duration within the minimum and maximum high-pulse time-thresholds is followed by a low-pulse duration within the minimum and maximum low-pulse time-thresholds

ESM_MCU_HMAX[7:0] and ESM_SOC_HMAX[7:0] set the maximum high-pulse time-threshold ($t_{HIGH\_MAX\_TH}$) for the related ESM. Use Equation 15 and Equation 16 to calculate the worst-case values for the $t_{HIGH\_MAX\_TH}$:

$$\text{Min. } t_{HIGH\_MAX\_TH} = (15\ \mu s + ESM\_x\_HMAX[7:0] \times 15\ \mu s) \times 0.95 \tag{15}$$

$$\text{Max. } t_{HIGH\_MAX\_TH} = (15\ \mu s + ESM\_x\_HMAX[7:0] \times 15\ \mu s) \times 1.05 \tag{16}$$

, in which x stands for either MCU or SoC.

ESM_MCU_HMIN[7:0] and ESM_SOC_HMIN[7:0] set the minimum high-pulse time-threshold ($t_{HIGH\_MIN\_TH}$) for the related ESM. Use Equation 17 and Equation 18 to calculate the worst-case values for the $t_{HIGH\_MIN\_TH}$:

$$\text{Min. } t_{HIGH\_MIN\_TH} = (15\ \mu s + ESM\_x\_HMIN[7:0] \times 15\ \mu s) \times 0.95 \tag{17}$$

$$\text{Max. } t_{HIGH\_MIN\_TH} = (15\ \mu s + ESM\_x\_HMIN[7:0] \times 15\ \mu s) \times 1.05 \tag{18}$$

, in which x stands for either MCU or SoC.

ESM_MCU_LMAX[7:0] and ESM_SOC_LMAX[7:0] set the maximum low-pulse time-threshold ($t_{LOW\_MAX\_TH}$) for the related ESM. Use Equation 19 and Equation 20 to calculate the worst-case values for the $t_{LOW\_MAX\_TH}$:

$$\text{Min. } t_{LOW\_MAX\_TH} = (15\ \mu s + ESM\_x\_LMAX[7:0] \times 15\ \mu s) \times 0.95 \tag{19}$$

$$\text{Max. } t_{LOW\_MAX\_TH} = (15\ \mu s + ESM\_x\_LMAX[7:0] \times 15\ \mu s) \times 1.05 \tag{20}$$

, in which x stands for either MCUor SoC.

ESM_MCU_LMIN[7:0] and ESM_SOC_LMIN[7:0] set the minimum low-pulse time-threshold ($t_{LOW\_MIN\_TH}$) for the related ESM. Use Equation 21 and Equation 22 to calculate the worst-case values for the $t_{LOW\_MIN\_TH}$:

$$\text{Min. } t_{LOW\_MIN\_TH} = (15\ \mu s + ESM\_x\_LMIN[7:0] \times 15\ \mu s) \times 0.95 \tag{21}$$

$$\text{Max. } t_{LOW\_MIN\_TH} = (15\ \mu s + ESM\_x\_LMIN[7:0] \times 15\ \mu s) \times 1.05 \tag{22}$$

, in which x stands for either MCU or SoC.

Please note that when setting up the minimum and the maximum low/high-pulse time-thresholds need to be configured such that clock tolerances from the TPS6594-Q1 and from the processor are incorporated. Equation 23, Equation 24, Equation 25, and Equation 26 are a guideline on how to incorporate these clock-tolerances:

$$ESM\_x\_HMIN[7:0] < 0.5 \times (ESM\_x\_HMAX[7:0] + ESM\_x\_HMIN[7:0]) \times 0.95 \times (1 - \text{MCU/SoC clock tolerance}) \tag{23}$$

$$ESM\_x\_HMAX[7:0] > 0.5 \times (ESM\_x\_HMAX[7:0] + ESM\_x\_HMIN[7:0]) \times 1.05 \times (1 + \text{MCU/SoC clock tolerance}) \tag{24}$$

$$ESM\_x\_LMIN[7:0] < 0.5 \times (ESM\_x\_LMAX[7:0] + ESM\_x\_LMIN[7:0]) \times 0.95 \times (1 - \text{MCU/SoC clock tolerance}) \tag{25}$$

ESM_x_LMAX[7:0] > 0.5 × (ESM_x_LMAX[7:0] + ESM_x_LMIN[7:0]) × 1.05 × (1 + MCU/SoC clock tolerance) (26)

2.14.3.1.1 ESM Error-Counter

If an ESM detects a bad-event, it increments its related error-counter (bits ESM_MCU_ERR_CNT[4:0] or bits ESM_SOC_ERR_CNT[4:0]) by 2. If an ESM detects a good-event, it decrements its related error-counter (bits ESM_MCU_ERR_CNT[4:0] or bits ESM_SOC_ERR_CNT[4:0]) by 1.

The device clears each error counter when ESM_x_START=0. Furthermore, the device clears the error-counter ESM_SOC_ERR[4:0] when it resets the SoC.

Each error-counter has a related threshold (bits ESM_MCU_ERR_CNT_TH[3:0] or bits ESM_SOC_ERR_CNT_TH[3:0]) which the MCU can configure if the related ESM start-bit is 0. If the error-counter value is above its configured threshold, the related ESM has detected a so-called ESM-error and starts the Error-Handling Procedure as described in Section 2.14.1. If the error-counter reached a value equal or less its configured threshold before the elapse of the configured delay-1 or delay-2 time-intervals, the ESM-error is no longer present and the ESM stops the Error-Handling Procedure as described in Section 2.14.1.

2.14.3.1.1.1 ESM Start-Up in PWM Mode

After MCU has set the start bit of an ESM (bit ESM_MCU_START or bit ESM_SOC_START), there are two possible scenarios:

1. The deglitched signal of the monitored input pin has a low level at the moment the MCU sets the start bit. In this scenario, the related ESM starts the following procedure:
   a. Start a timer with a time-length according the value configured in correspondig ESM_MCU_LMAX[7:0] or ESM_SOC_LMAX[7:0].
   b. Wait for a first rising edge on its deglitched input signal.
   c. If the rising edge comes before the configured time-length elapses, the ESM skips the next step and starts to monitor the high-pulse duration time. Hereafter, the ESM detects good-events or bad-events as described in Section 2.14.3.1. Figure 2-41 shows an example this scenario as Case Number 1.
   d. If the configured time-length (configured in corresponding ESM_MCU_LMAX[7:0] or ESM_SOC_LMAX[7:0]) elapses, the ESM detects a bad-event and increments the related error-counter with +2. Hereafter, the ESM detects good-events or bad-events as described in Section 2.14.3.1. Figure 2-43 shows an example this scenario as Case Number 3.
   e. If the error-counter value is above its configured threshold, the related ESM has detected a so-called ESM-error and starts the Error-Handling Procedure as described in Section 2.14.3.1.
   f. During this Error-Handling Procedure, the ESM continues to monitor its related input pin, and updates the error-counter accordingly when it detects good-events or bad-events, until the Error-Handling Procedure reaches the step in which the device causes an ESM ERROR trigger in the state machine, which may reset the MCU or SoC according to the PFSM definition. Figure 2-44 shows a scenario in which the device resets the MCU or SoC as Case Number 4.
   g. If the error-counter reaches a value equal or less its configured threshold before the elapse of the configured delay-1 or delay-2 time-intervals, the ESM-error is no longer present and the ESM stops the Error-Handling Procedure as described in Section 2.14.3.1.
2. The deglitched signal monitored input pin has a high level at the moment the MCU sets the start bit. In this scenario, the related ESM starts the following procedure:
   a. Start a timer with a time-length according the value configured in corresponding ESM_MCU_HMAX[7:0] or ESM_SOC_HMAX[7:0].
   b. Wait for a first falling edge on its deglitched input signal.
   c. If the falling edge comes before the configured time-length elapses, the ESM skips the next step and starts to monitor the low-pulse duration time. Hereafter, the ESM detects good-events or bad-events as described in Section 2.14.3.1. Figure 2-42 shows an example this scenario as Case Number 2.
   d. If the configured time-length (configured in corresponding ESM_MCU_HMAX[7:0] or ESM_SOC_HMAX[7:0]) elapses, the ESM detects a bad-event and increments the related error-counter with +2. Hereafter, the ESM detects good-events or bad-events as described in Section 2.14.3.1.

e. If the error-counter value is above its configured threshold, the related ESM has detected a so-called ESM-error and starts the Error-Handling Procedure as described in Section 2.14.3.1.
f. During this Error-Handling Procedure, the ESM continues to monitor its related input pin, and updates the error-counter accordingly when it detects good-events or bad-events, until the Error-Handling Procedure reaches the step in which the device causes an ESM ERROR trigger in the state machine, which may reset the MCU or SoC according to the PFSM definition, as Case Number 4.
g. If the error-counter reaches a value equal or less its configured threshold before the elapse of the configured delay-1 or delay-2 time-intervals, the ESM-error is no longer present and the ESM stops the Error-Handling Procedure as described in Section 2.14.3.1.

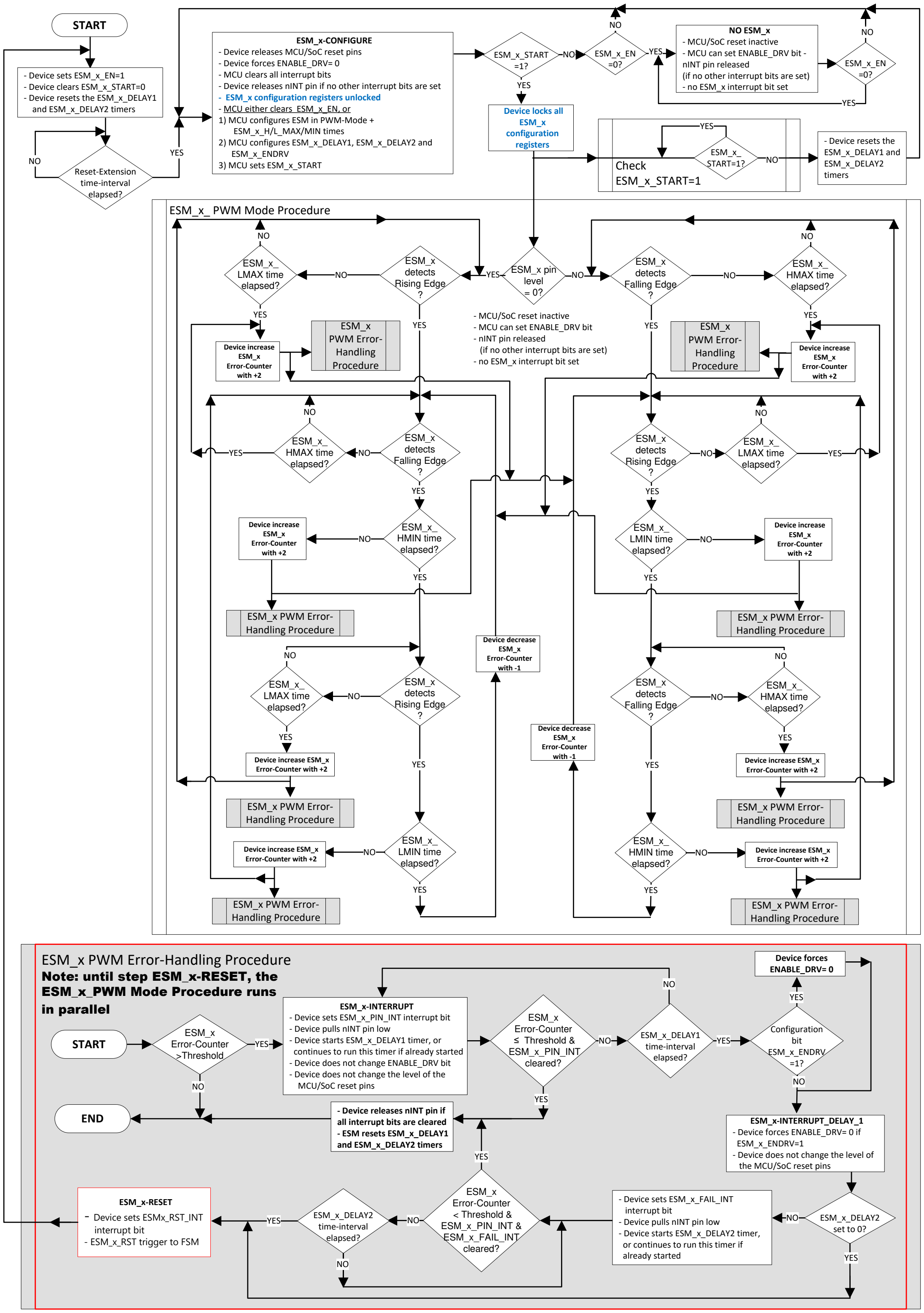


Figure 2-40. Flow-Chart for ESM_MCU and ESM_SoC in PWM Mode

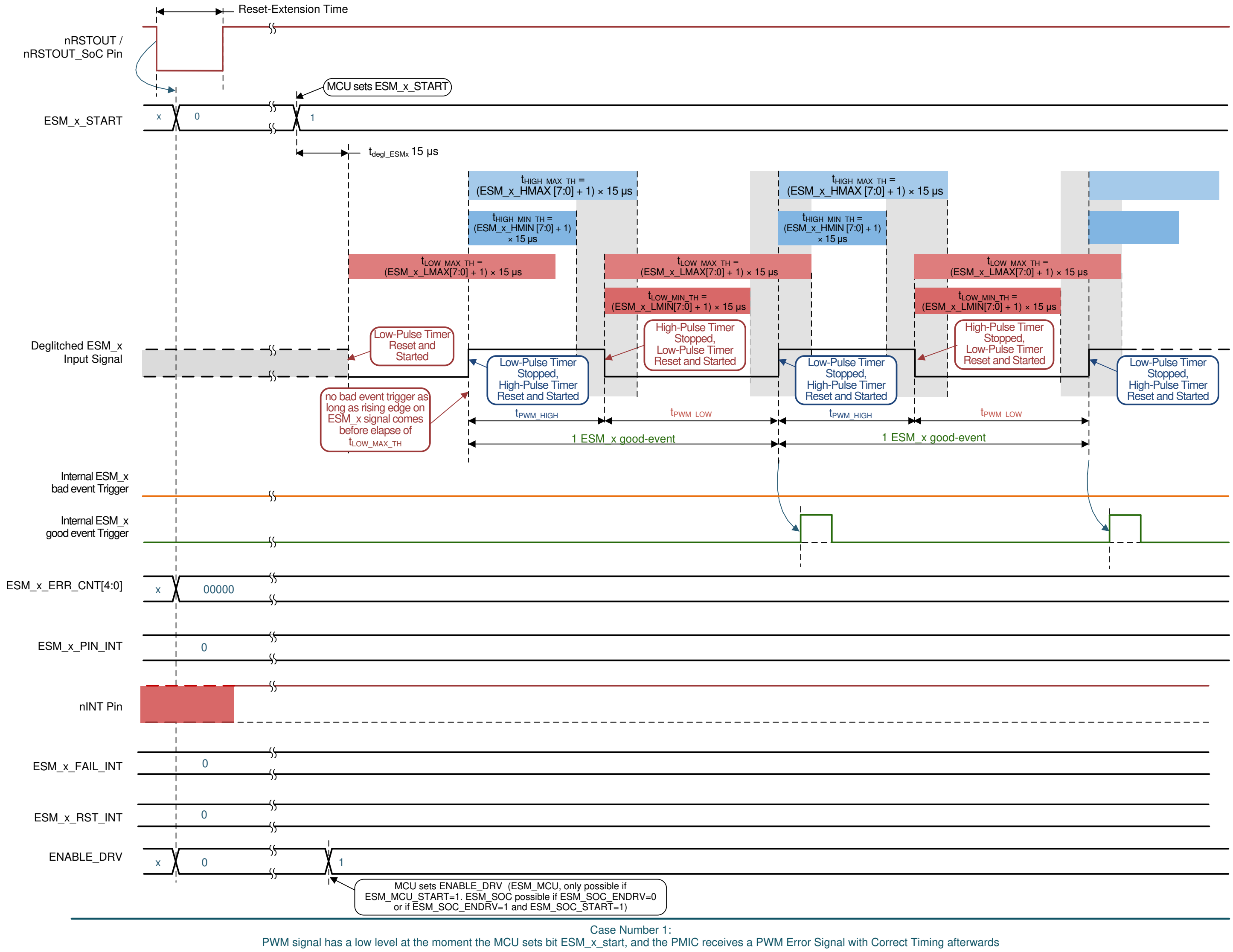


Figure 2-41. Example Waveform for ESM in PWM Mode - Case Number 1 ESM Starts with Low-Level at Deglitched Input Signal, and Receives Correct PWM Signal Afterwards

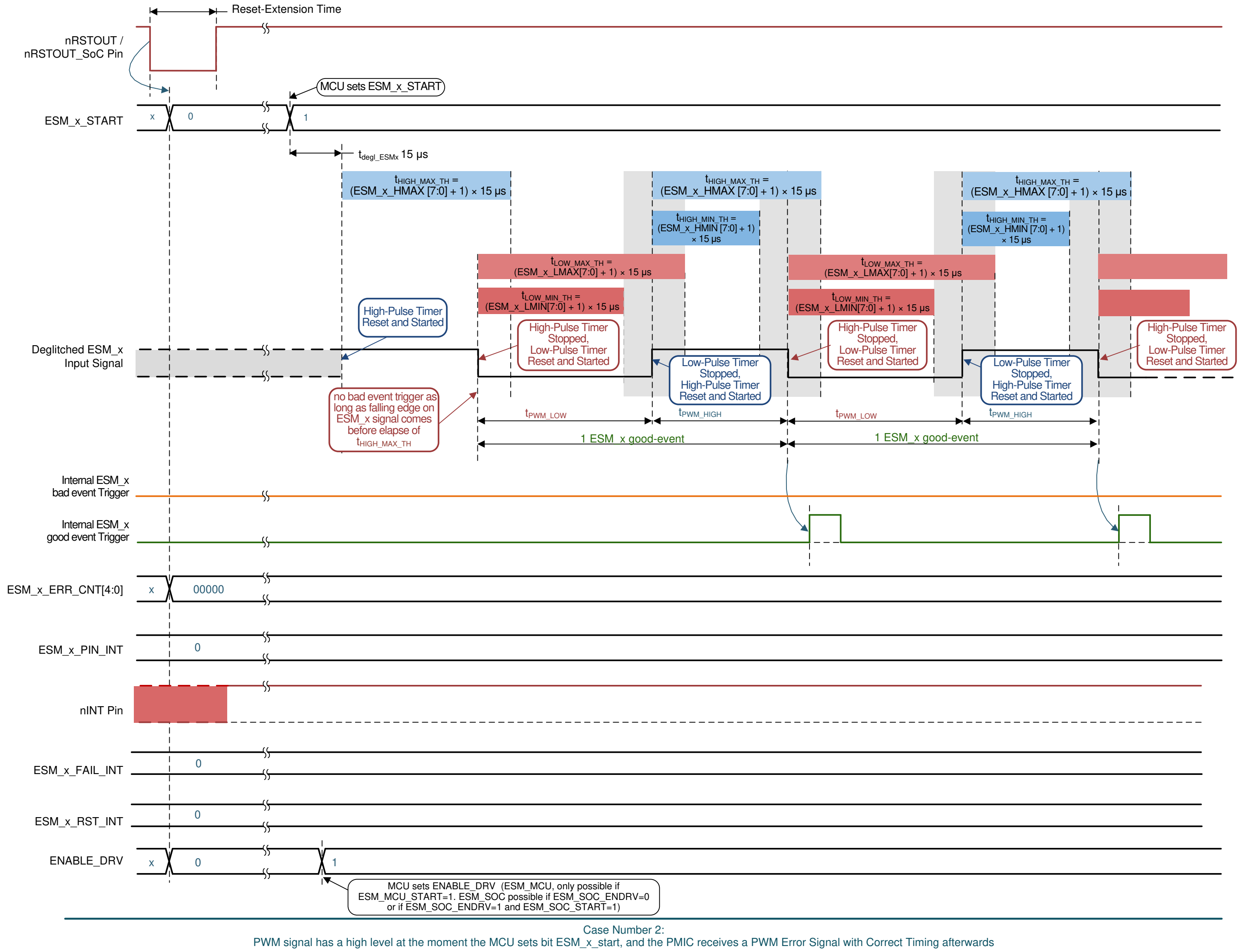
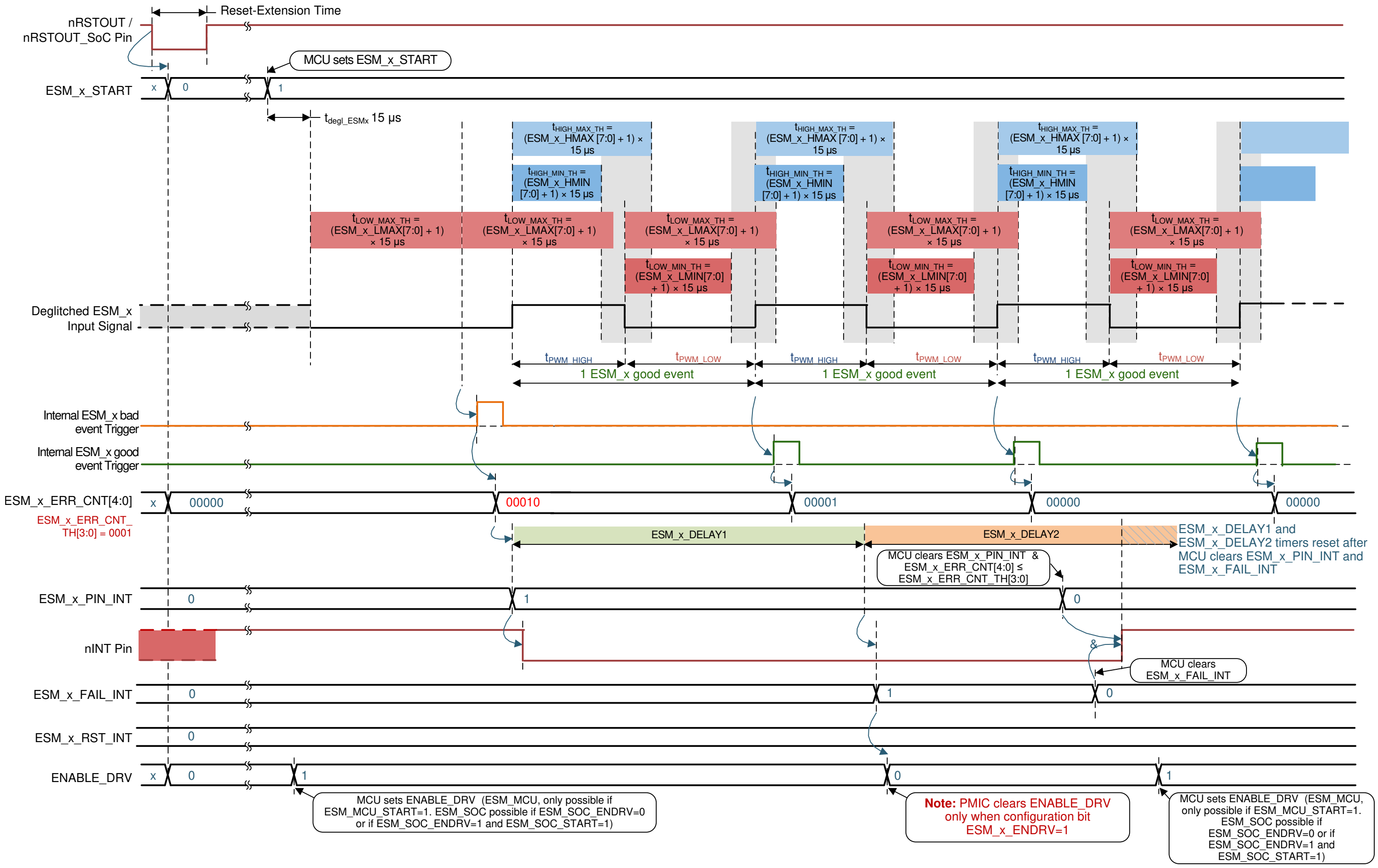
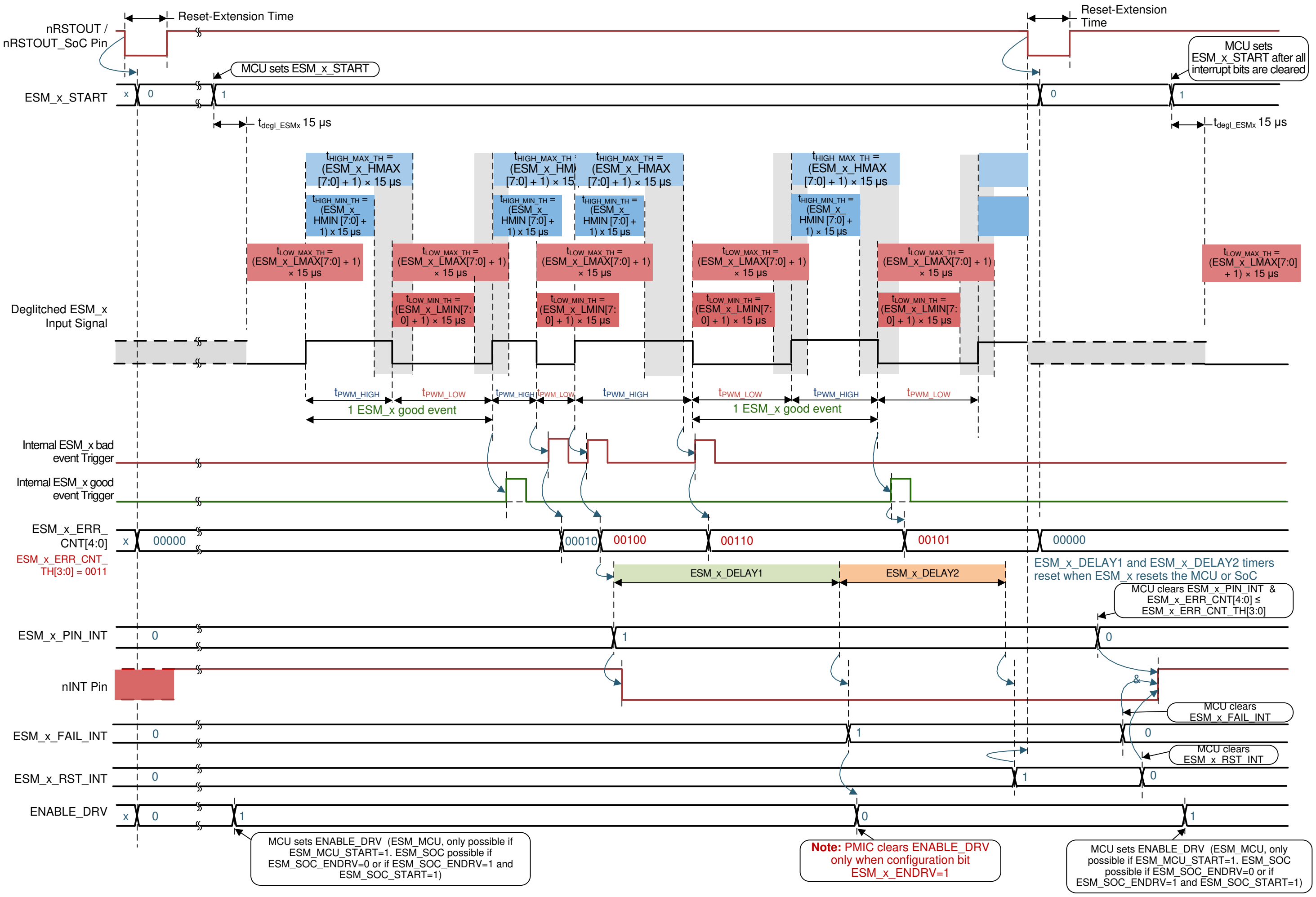


Figure 2-42. Example Waveform for ESM in PWM Mode - Case Number 2 ESM Starts with High-Level at Deglitched Input Signal, and Receives Correct PWM Signal Afterwards



Case Number 3: ESM_DELAY2 > 0
PWM signal has a low level at the moment the MCU sets bit ESM_x_start, but the PMIC receives the PWM Error Signal too late. Afterwards PWM Error Signal recovers with Correct Timing and ESM_x_ERR_CNT[4:0] reaches a value less than the configured ESM_x_ERR_CNT_TH[3:0] before elapse of the ESM_x_DELAY2 time-interval

Figure 2-43. Example Waveform for ESM in PWM Mode - Case Number 3 ESM Starts with Low-Level at Deglitched Input Signal, but Receives Too Late a Correct PWM Signal Afterwards



Case Number 4: _DELAY2 > 0
PWM signal has an error after start-up, and the ESM_x_ERR_CNT[4:0] > ESM_x_ERR_CNT_TH[3:0] during the elapse of ESM_x_DELAY1 and ESM_x_DELAY2. Hence the PMIC pulls the nRSTOUT / nRSOUT_SoC pin low, and releases this pin after the reset-extension time. After this, MCU clears all errors and restarts the ESM_x

Figure 2-44. Example Waveform for ESM in PWM Mode - Case Number 4 ESM Starts with Low-Level at Deglitched Input Signal and Receives a Correct PWM signal. Afterwards the ESM detects Bad Events, and the PWM Signal Recovers Too Late Which Leads to an ESM ERROR Trigger in the State Machine

2.15 Multi-PMIC Synchronization

A multi-PMIC synchronization scheme is implemented in the TPS6594-Q1 device to synchronize the power state changes with other PMIC devices. This feature consolidates and simplifies the IO control signals required between the application processor or the micro controller and multiple PMICs in the system. The control interface consists of an SPMI protocol which communicates the next power state information from the primary TPS6594-Q1 device to up to 5 secondary PMICs, and receives feedback signal from the secondary PMICs to indicate any error condition. Figure 2-45 is the block diagram of the power state synchronization scheme. The TPS6594-Q1 is represented as the primary PMIC in this block diagram, which is responsible for broadcasting the synchronous system power state data, and processing the error feedback signals from the secondary PMICs. It is the primary device in the SPMI interface bus.

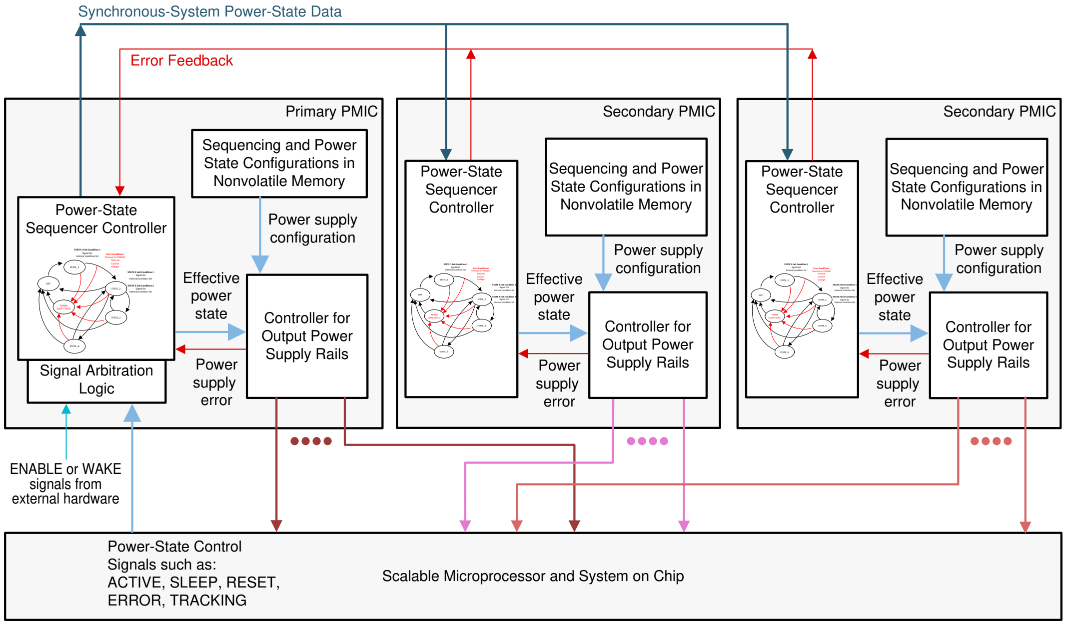


Figure 2-45. Multi-PMIC Power State Synchronization Block Diagram

In this scheme each primary and secondary PMIC runs on its own system clock, and maintains it's own I2C register set. Each PMIC will monitor it's own activities and pull down the open-drain output of nINT or PGOOD pin when errors are detected. The microprocessor will need to read the status bits on each PMIC device to find out the source of error being reported.

To synchronize the timing when entering and exiting from the LP_STANDBY state, the VOUT_LDOVINT of the TPS6594-Q1 device must be connected to the ENABLE input of the secondary PMICs, which are the slave devices in the SPMI interface bus. Figure 2-46 illustrates the pin connections between the primary, the secondary, and the application processor or the System on Chip.

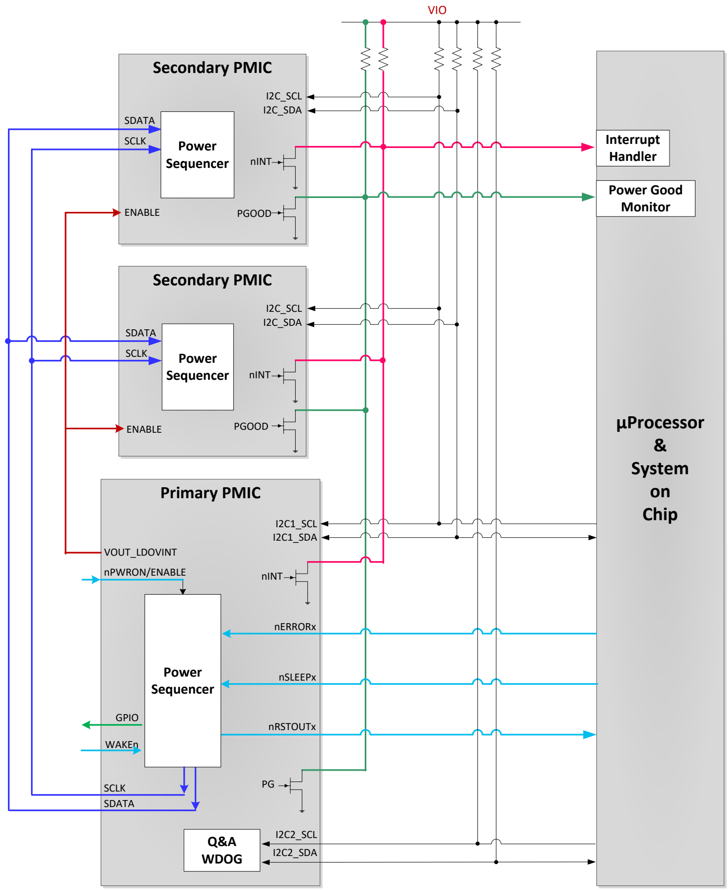


Figure 2-46. Multi-PIMC Pin Connections

The power sequencer of the multiple PMICs are synchronized at the beginning of each power up and power down sequence. However, due to the ±5% clock accuracy of the independent system clocks on the primary and secondary PMICs, a variation in the sequence timing is still possible. The worst case sequence timing variation from different PMIC rails is up to ±10% of the target delay time. Figure 2-47 illustrates the creation of this timing variation between PMICs.

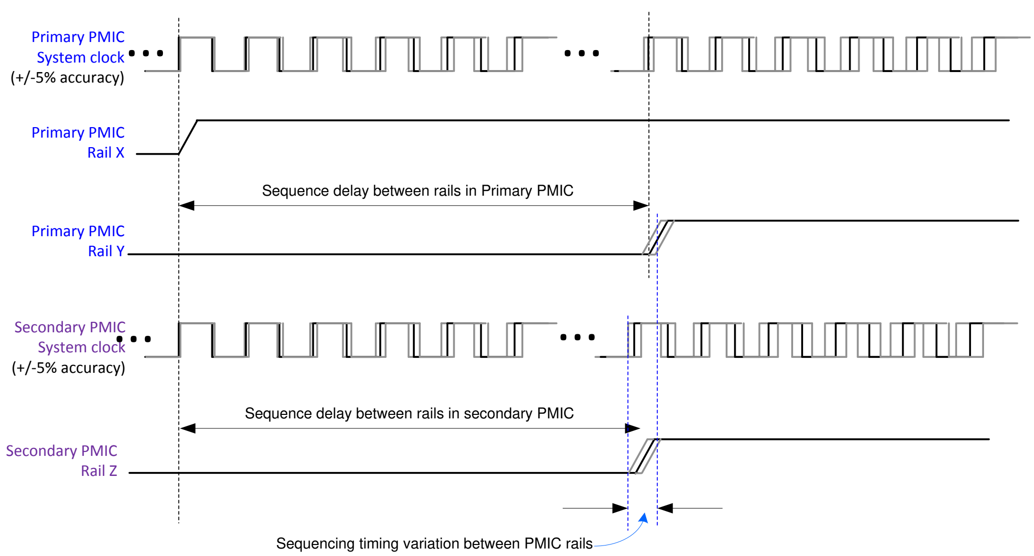


Figure 2-47. Multi-PMIC Rail Sequencing Timing Variation

2.15.1 SPMI Interface System Setup

An SPMI interface in the TPS6594-Q1 device is utilized to communicate the power state transition across multiple PMICs in the system. The interface block contains a SPMI master block and a SPMI slave block. There is only one SPMI master device in any given system. The TPS6594-Q1 device is generally the SPMI master in the system with both master and slave blocks enabled. As the SPMI master it initiates SPMI interface BIST and executes periodic checking of the SPMI bus health.

The SPMI master ID (MID) of the TPS6594-Q1 device is 1. TPS6594-Q1 will also contain a logical SPMI slave interface in order to receive SPMI communications from the SPMI slave devices. The TPS6594-Q1 as the SPMI master device will contain the slave (SID) = 0101.

All of the slave devices on this SPMI network will only have the slave interface enabled. There cannot be more than 5 slave devices in the system. The SIDs for the five slave devices are:

- 1st slave device: 0011
- 2nd slave device: 1100
- 3rd slave device: 1001
- 4th slave device: 0110
- 5th slave device: 1010

All devices in the SPMI network will listen to Group Slave ID (GSID): 1111. This address is used to communicate all power state transition information in broadcast mode to all connected devices in parallel.

2.15.2 Transmission Protocol and CRC

The communication between the devices on the network utilizes Extended Register Write command to GSID address *1111* with byte length of 2. Sequence format complies with MIPI SPMI 2.0 specification. First data frame carries the data payload of 5 bits and 3 filler bits.

Communication over the SPMI interface may contain information regarding the power state transition or the unique SID of one or more the slave devices. In the case of power state information the data payload contains 5 bits of Trigger ID information and 3 trigger state bits. In the case of SID information all 8 bits contain the SID of the slave device.

Second data frame carries 8 bits of CRC information. CRC polynomial used is $X^8 + X^2 + X + 1$. CRC is calculated over the SPMI command frame, the address frame and the first data frame which contains the payload, excluding the parity bits in these three frames.

Figure 2-48 shows the data format of the SPMI Extended Register Write Command.

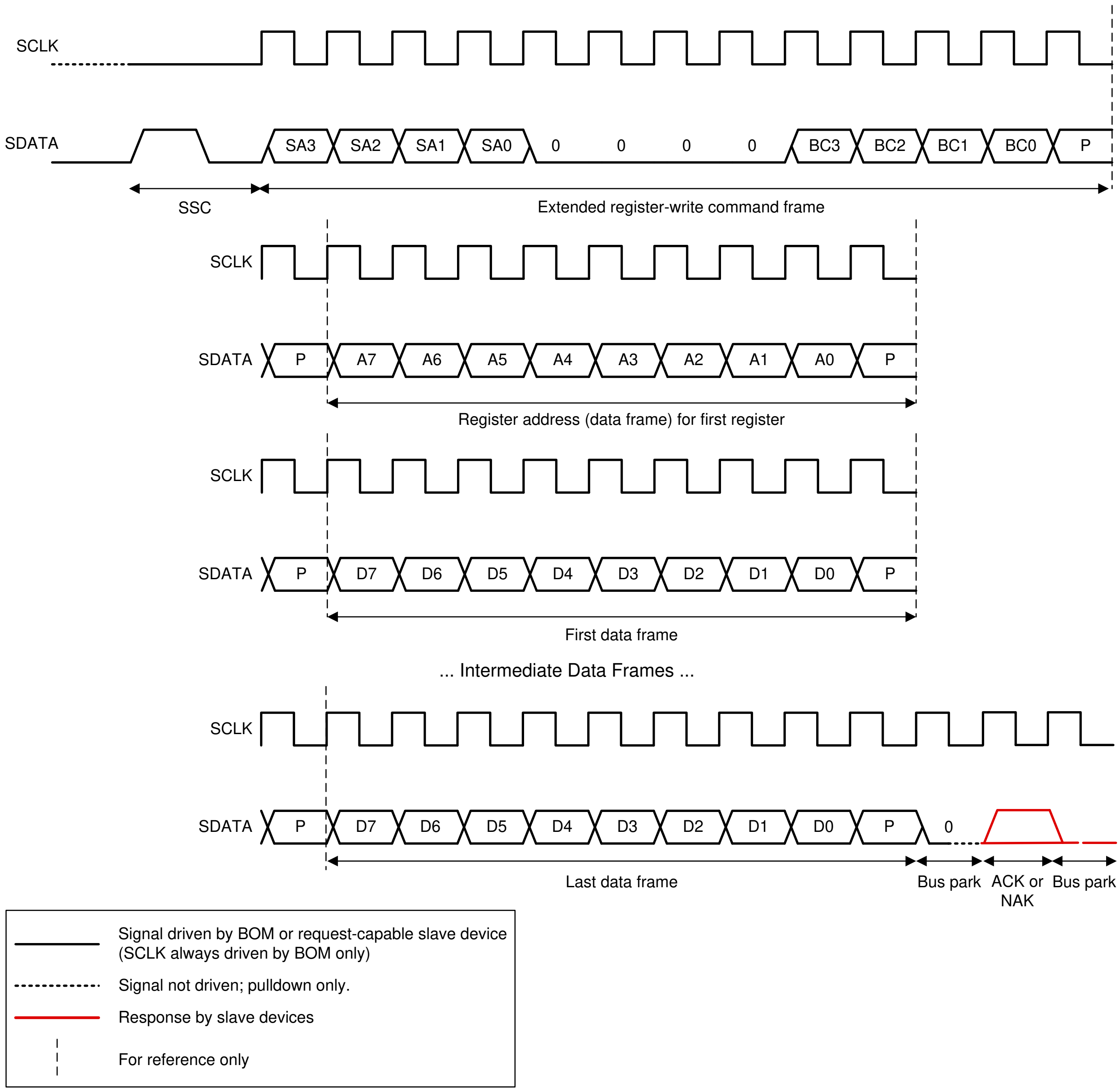


Figure 2-48. SPMI Extended Register Write Command

2.15.2.1 Operation with Transmission Errors

If the receiving device detects a parity or CRC error in the incoming sequence it will respond with negative ACK/NACK per SPMI standard.

If the transmitting device sees NACK response, it will try to resend the message as many times as indicated by SPMI_RETRY_LIMIT register bits. After that it will consider SPMI bus inoperable, sets SPMI_ERR_INT interrupt and goes to the safe recovery state and executes an orderly shutdown. Bus arbitration requests do not count as failed attempts if a slave device loses bus arbitration. SPMI_RETRY_LIMIT counter will be reset after each successful transmission by the device.

If a slave device has determined that SPMI does not work reliably it will not respond to any SPMI commands anymore until power-on-reset event has occurred. This is to prevent continued operation in a situation where SPMI is unreliable. This will force the TPS6594-Q1 device to detect a missing secondary device on the network during the periodic testing of SPMI bus. The slave device will then internally handle the SPMI error condition per error handling rules set for the device (in general executing an orderly shutdown). SPMI block signals to the device that SPMI bus error has occurred after the retry limit has been exceeded.

2.15.2.2 Transmitted Information

SPMI bus will be used to carry two types of information:

- PFSM Trigger ID between the SPMI master and slave devices
- SID from SPMI slave devices to SPMI master device

The SPMI master device reads the SID of the slave devices periodically to check the health of the interface. Exchanging Trigger IDs for the power state transition is sufficient to keep the PFSMs of all the devices on the SPMI network in synchronization. Device interrupts will provide insights to the reason which cause power state transitions.

2.15.3 SPMI Slave Communication to SPMI Master

An SPMI slave device communicates to the SPMI master device and any other SPMI slave devices only if there is an internal error which is not SPMI related. The slave device initiates the error communication using Slave Arbitration Request with A-bit as defined in the SPMI 2.0 specification. SPMI 2.0 protocol manages the situation with multiple slave devices requesting error communication at the same time. This is resolved using the slave arbitration process as described in SPMI 2.0 specification. Once the SPMI slave device wins the Slave Arbitration using A-bit protocol it will perform Extended Register Write command to Group Slave ID address *1111* with using the protocol described earlier in this document for communicating PFSM trigger ID.

2.15.3.1 Incomplete Communication from SPMI Slave to SPMI Master

In case the TPS6594-Q1 device as the SPMI master detects Slave Arbitration Request on the SPMI interface, but the received sequence has an error or is incomplete, it will immediately perform SPMI interface BIST. If this fails, SPMI master will execute error handling for the SPMI error. If the SPMI interface BIST is successfully executed, TPS6594-Q1 will resume normal operation.

2.16 SPMI BIST Overview

The BIST operation is implemented both during BIST state and regularly during runtime operation. Figure 2-49 below illustrates how SPMI BIST operates during device power-up.

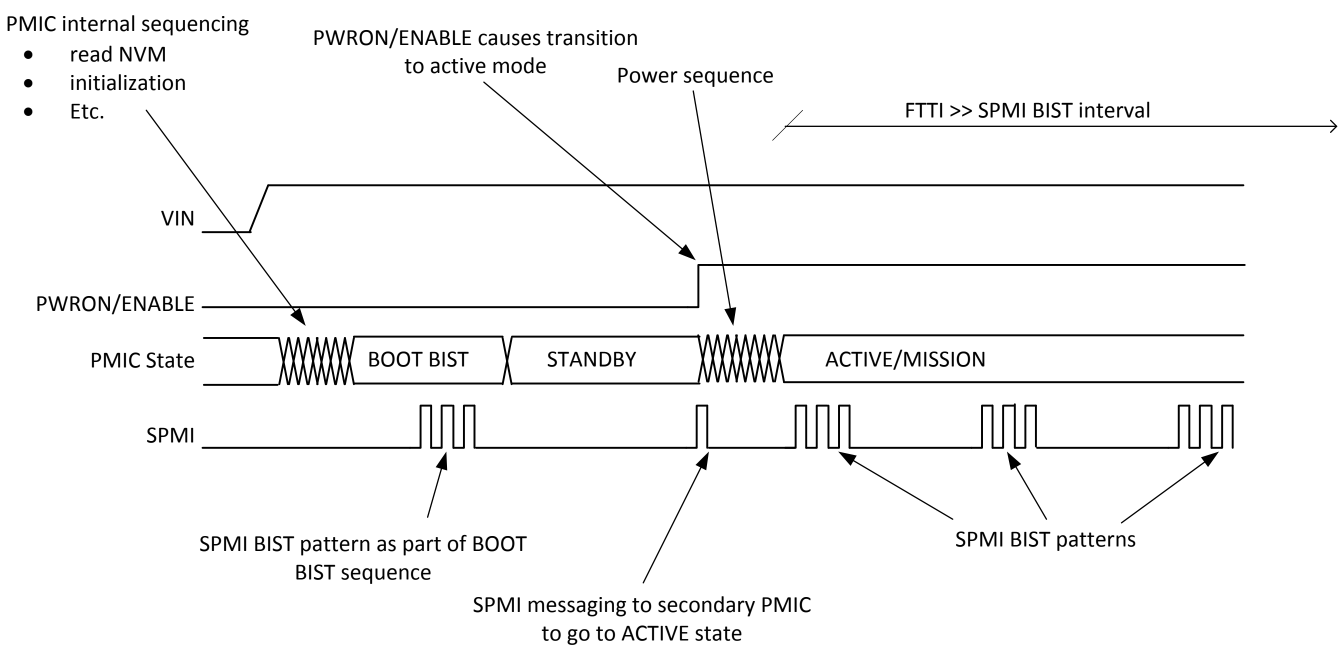


Figure 2-49. SPMI BIST Operation

After input power is detected and verified to be at the correct level, TPS6594-Q1 will initialize itself by reading NVM and performing any actions needed to prepare for operation. TPS6594-Q1 will then enter BOOT BIST state in which internal logic will sequence a series of tests to verify TPS6594-Q1 and the system are OK. As part of this test the SPMI interface BIST is performed. After it is successfully completed the device goes to standby state and wait for further signals from the system to initiate the power-up sequence of the processor.

A valid on request initiates the processor power-up sequence. TPS6594-Q1 as the SPMI master will communicate this event via SPMI to all of the slave devices in the system. The power-up sequence will then be executed and TPS6594-Q1 will then enter the active state or any mission states.

2.16.1 SPMI Bus Boot BIST

Boot BIST will include both LBIST of the SPMI logic and an interface BIST. LBIST is performed first before the interface BIST during BOOT BIST or RUNTIME BIST. Interface BIST is implemented by reading SID from each slave device into the TPS6594-Q1 device, and ensuring they are unique and match the expected slave count. This ensures that

- All slave device(s) are present in the system as expected
- Each slave device has the right NVM settings
- The SPMI logic blocks are working on the master and all of the slave devices
- The pins and wires on the ICs and PCB are in working order

The SPMI interface BIST is initiated by the SPMI master device by writing a request to the slave devices (using GSID) requesting the slave devices to send their SIDs to the master device. Upon receiving this command from SPMI master the slave devices will request SPMI bus arbitration using SR-bit protocol; and upon winning the bus arbitration the slave devices will transmit their SID into the logical slave of the SPMI master device.

The TPS6594-Q1 device contains a list of all SPMI slave devices on the SPMI bus and their SIDs in the register set. As the master device TPS6594-Q1 will read the NVM_ID register in each SID and compare the result with the stored SID for the corresponding slave ID. The master device has to ensure that every non-zero slave SID on its list is returned. This is important for use cases where there are two or more identical slave devices in the system. In these cases it is mandatory that correct number of the same SID is returned. If no identical devices are to be used, then return of the same SID multiple times is an error due to incorrect assembly of identical devices onto the PCB. An all-zero SID stored in the register indicates the slave device is not present in the system.

2.16.2 Periodic Checking of the SPMI

The TPS6594-Q1device as SPMI master automatically executes the SPMI interface BIST periodically while device is operating. Frequency of SPMI interface BIST is set by the SPMI_WD_BOOT_INTERVAL[3:0] register bits during the device boot time, and by SPMI_WD_RUNTIME_INTERVAL[3:0] after the device reaches mission states. The setting of the SPMI_WD_x_INTERVAL[3:0] bits should be the same for all the devices on the same SPMI network. Slave devices should expect that the master device polls SID within 1.5x the period of SPMI_WD_x_INTERVAL[3:0]. This provides enough margins for clock uncertainty.

During mission state operation, master device expects the slave devices to respond to SID request within the polling period set by the SPMI_WD_RESPONSE_TIMEOUT[3:0] register bits. In other words, from the polling start command the slaves must responded within the time interval set by SPMI_WD_RESPONSE_TIMEOUT[3:0].

During boot time or when the device enters Safe Recovery state, to prevent the SPMI master from polling the slave devices too often while the slave device is recovering from a system error such as a thermal shutdown event, the SPMI_WD_AUTO_BOOT_TIMEOUT[7:0] register bits sets a longer timeout period for the slave devices to respond to the master device before the master device reports an error.

Violating either the SPMI_WD_RESPONSE_TIMEOUT[3:0] period or the SPMI_WD_AUTO_BOOT_TIMEOUT[7:0] period will cause the triggering of SPMI error.

2.16.3 SPMI Message Priorities

SPMI bus will use the protocol priority levels listed in Table 2-17 for each message type of communications.

Table 2-17. SPMI Message Types and Priorities

SPMI protocol priority level	Name of priority level in SPMI standard	Message types
Highest	Slave A-bit arbitration	State transition messages from slave(s) to master
	Master priority arbitration	State transition messages from master to slave(s)
	Slave SR-bit arbitration	Slave SID to master
Lowest	Master secondary arbitration	Master request of SIDs from slave(s)

2.17 Control Interfaces

The device has two, exclusive selectable (from factory settings) interfaces. The first selection is up to two high-speed I^{2}C interfaces (I2C_SPI_SEL=0). The second selection is one SPI interface (I2C_SPI_SEL=1). Both the SPI and the I2C1 interfaces are used to fully control and configure the device and have access to all of the configuration registers, as well as the Watchdog registers. During normal operating mode, when the I^{2}C configuration is selected, and GPIO1 and GPIO2 pins can be configured as the SCL_I2C2 and SDA_I2C2 pins, I2C2 interface will become the dedicated interface for the Q&A Watchdog communication channel, while I2C1 interface will no longer have access to the Watchdog registers. When the device enters EEPROM programing mode , I2C2 interface is automatically disabled, and I2C1 interface will have access to all of the registers, including the Watchdog registers.

2.17.1 CRC Calculation for I^{2}C and SPI Interface Protocols

For safety applications, the TPS6594-Q1 device supports read and write protocols with embedded CRC data fields. Both the master and slave devices use a standard CRC-8 polynomial to calculate the checksum value: $X^8 + X^2 + X + 1$.The CRC algorithm details are as follows:

- Initial value for the remainder is all 1s
- Big-endian bit stream order
- Result inversion is enabled

For I^{2}C Interface, the TPS6594-Q1 device uses this polynomial to calculate the checksum value on every bit except the ACK and NACK bits it receives from the MCU during a write protocol. The device compares this calculated checksum with the MCRC checksum value which the device receives from the MCU. The device also uses this polynomial to calculate the SCRC checksum value based on every bit except the ACK and NACK bits which the device transmits to the MCU during a read protocol. The master device (MCU) must use this same polynomial to calculate the checksum value based on the bits which the MCU receives from the device. The MCU must compare this calculated checksum with the SCRC checksum value which it receives from the device.

For the SPI interface, the TPS6594-Q1 device uses this polynomial to calculate the checksum value on every bit it receives from the MCU during a write protocol. The device compares this calculated checksum with the MCRC checksum value which the device receives from the master device (MCU). During a read protocol, the device also uses this polynomial to calculate the SCRC checksum value based on the first 16 bits sent by the master device, and the next 8 bits the device transmits to the master device. The master device must use this same polynomial to calculate the checksum value based on the bits which the master device sents to and receives from the device, and compare it with the SCRC checksum value which it receives from the device.

Figure 2-50 and Figure 2-51 are examples for the 4-bit MCRC and the SCRC calculation from 16-bit databus.

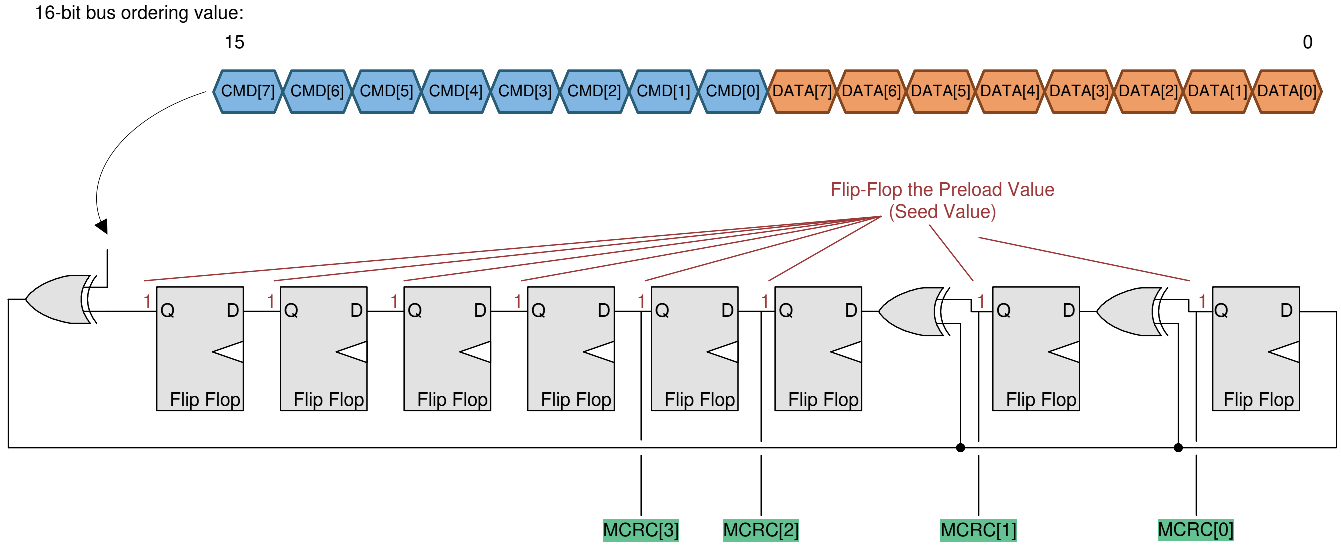


Figure 2-50. Calculation of 4-Bit Master CRC (MCRC) Output

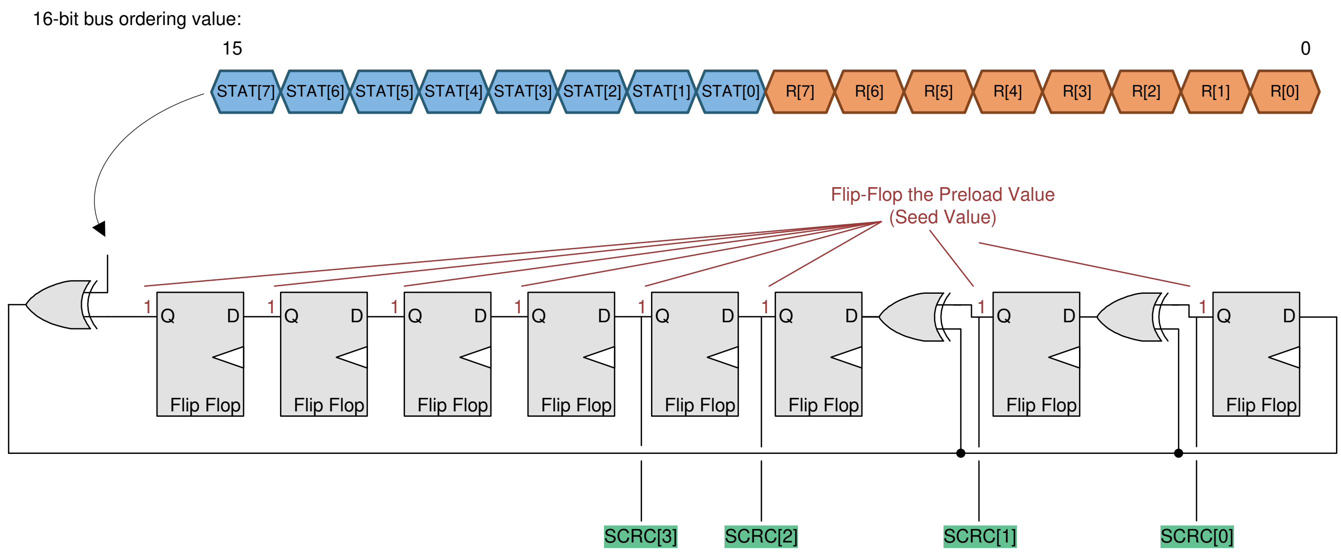


Figure 2-51. Calculation of 4-Bit Slave CRC (SCRC) Input

2.17.2 I²C-Compatible Interface

The default I²C1 7-bit slave device address of the TPS6594-Q1 device is set to 0x48 (0b1001000 in binary), while the two least-significant bits can be changed for alternative page selection listed under Section 2.18.1. The default 7-bit slave device address for the Q&A WatchDog I²C2 interface is set to 0x12. The I2C1_ID and I2C2_ID register bits can be used to reconfigure the 7-bit default slave address for the corresponding I²C interface.

The I²C-compatible synchronous serial interface provides access to the configurable functions and registers on the device. This protocol uses a two-wire interface for bidirectional communications between the devices connected to the bus. The two interface lines are the serial data line (SDA), and the serial clock line (SCL). Every device on the bus is assigned a unique address and acts as either a master or a slave depending on whether it generates or receives the serial clock SCL. The SCL and SDA lines must each have a pullup resistor placed somewhere on the line and remain HIGH even when the bus is idle. The device supports standard mode (100 kHz), fast mode (400 kHz), and fast mode plus (1 MHz) when VIO is 3.3 V or 1.8 V, and high-speed mode (3.4 MHz) only when VIO is 1.8 V.

2.17.2.1 Data Validity

The data on the SDA line must be stable during the HIGH period of the clock signal (SCL). In other words, the state of the data line can only be changed when clock signal is LOW.

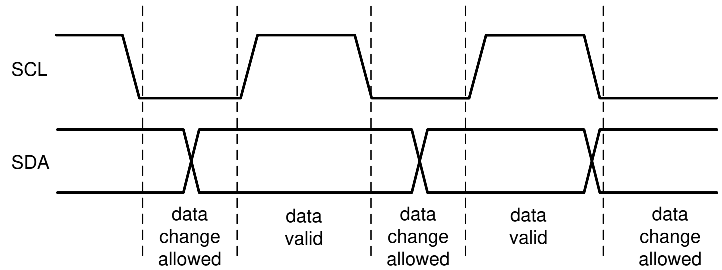


Figure 2-52. Data Validity Diagram

2.17.2.2 Start and Stop Conditions

The device is controlled through an I^{2}C-compatible interface. START and STOP conditions classify the beginning and end of the I^{2}C session. A START condition is defined as the SDA signal going from HIGH to LOW while the SCL signal is HIGH. A STOP condition is defined as the SDA signal going from LOW to HIGH while the SCL signal is HIGH. The I^{2}C master device always generates the START and STOP conditions.

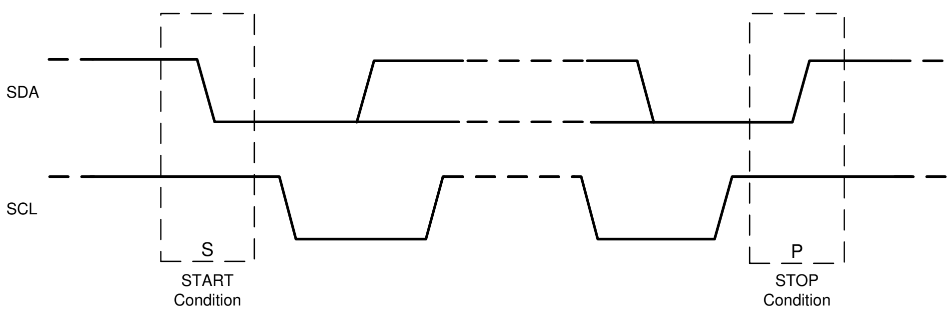


Figure 2-53. Start and Stop Sequences

The I^{2}C bus is considered busy after a START condition and free after a STOP condition. During data transmission the I^{2}C master can generate repeated START conditions. A START and a repeated START condition are equivalent function-wise. Figure 2-54 shows the SDA and SCL signal timing for the I^{2}C-compatible bus. For timing values, see the *Specification* section.

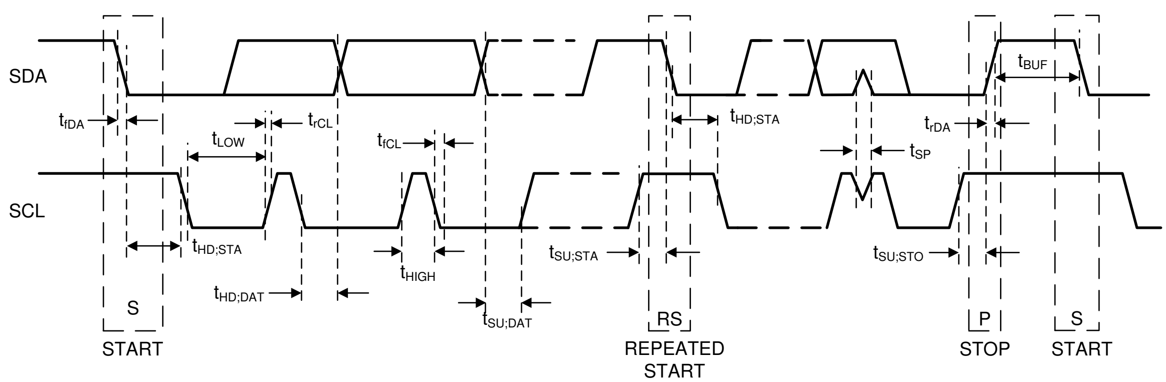


Figure 2-54. I^{2}C-Compatible Timing

2.17.2.3 Transferring Data

Every byte put on the SDA line must be eight bits long, with the most significant bit (MSB) being transferred first. Each byte of data has to be followed by an acknowledge bit. The acknowledge related clock pulse is generated by the master. The master releases the SDA line (HIGH) during the acknowledge clock pulse. The device pulls down the SDA line during the 9th clock pulse, signifying an acknowledge. The device generates an acknowledge after each byte has been received.

There is one exception to the *acknowledge after every byte* rule. When the master is the receiver, it must indicate to the transmitter an end of data by not-acknowledging (*negative acknowledge*) the last byte clocked out of the slave. This *negative acknowledge* still includes the acknowledge clock pulse (generated by the master), but the SDA line is not pulled down.

After the START condition, the bus master sends a chip address. This address is seven bits long followed by an eighth bit which is a data direction bit (READ or WRITE). For the eighth bit, a 0 indicates a WRITE and a 1 indicates a READ. The second byte selects the register to which the data will be written. The third byte contains data to write to the selected register. Figure 2-55 shows an example bit format of device address 110000-Bin = 60Hex.

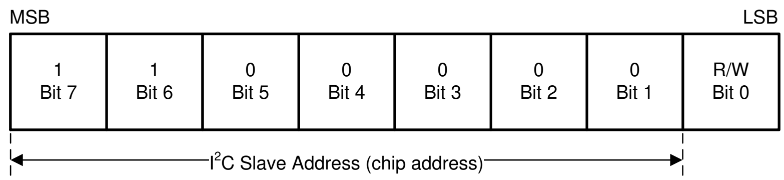


Figure 2-55. Example Device Address

In case the MCU attempts to write to a register-address that does not exist, the device sets the COMM_ADR_ERR_INT (for I2C1) or I2C2_ADR_ERR_INT (for I2C2) bit, unless the COMM_ADR_ERR_MASK or I2C2_ADR_ERR_MASK bit is set. The MCU must clear this bit by writing a ‘1’ to the COMM_ADR_ERR_INT (for I2C1) or I2C2_ADR_ERR_INT (for I2C2) bit.

For safety applications, the device supports read and write protocols with embedded CRC data fields. In a write cycle, the I^{2}C master should provide the 8-bit CRC value after sending the write data bits and receiving the ACK from the slave. The CRC value should be calculated from every bit included in the write protocol except the ACK bits from the slave. In a read cycle, the I^{2}C slave should provide the 8-bit CRC value after sending the read data bits and receiving the NACK from the master. The CRC value should be calculated from every bit included in the read protocol except the ACK and NACK bits.

The embedded CRC field can be enabled or disabled from the protocol by setting the I2C1_SPI_CRC_EN (for I2C1) or I2C2_CRC_EN (for I2C2) register bit to '1' - enabled, '0' - disabled. The default of this bit is configurable through the NVM.

In case the calculated CRC-value does not match the received CRC-check-sum, an I^{2}C-CRC-error is detected, the COMM_CRC_ERR_INT (for I2C1) or I2C2_CRC_ERR_INT (for I2C2) bit is set, unless it is masked by the COMM_CRC_ERR_MASK or I2C2_CRC_ERR_MASK bit. The MCU must clear this bit by writing a ‘1’ to the COMM_CRC_ERR_INT (for I2C1) or I2C2_CRC_ERR_INT (for I2C2) bit.

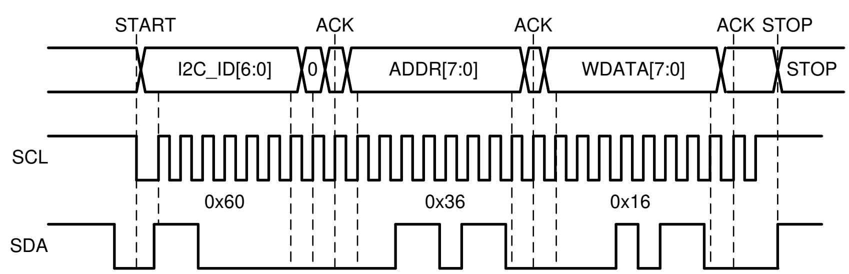
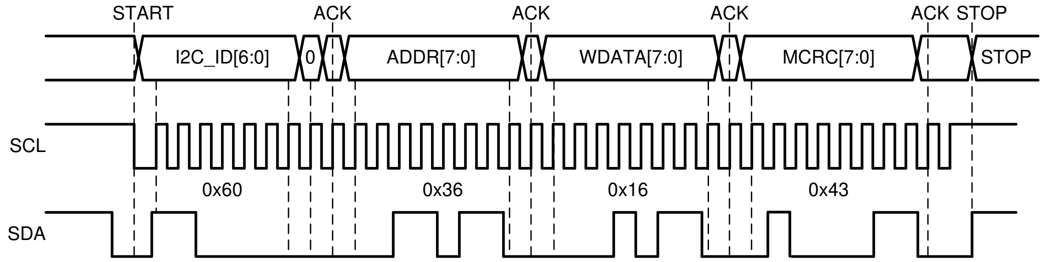
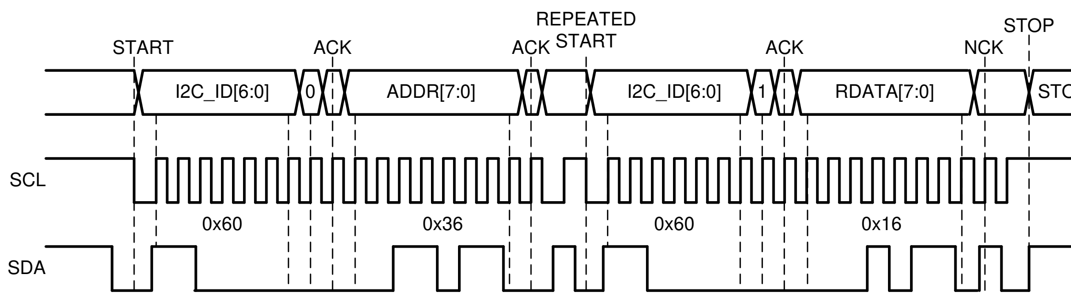


Figure 2-56. I^{2}C Write Cycle without CRC



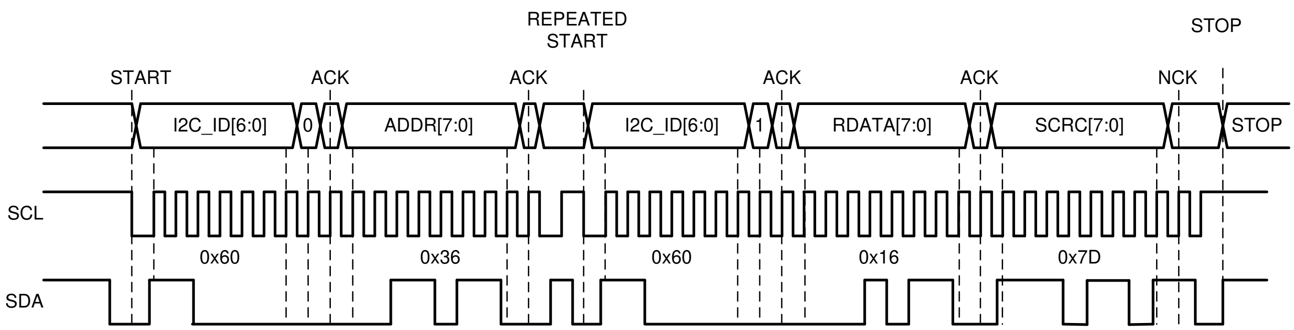
Master provides MCRC[7:0], which is calculated from the I2C_ID, R/W, ADDR, and the WDATA bits (24 bits).

Figure 2-57. I²C Write Cycle with CRC



When READ function is to be accomplished, a WRITE function must precede the READ function as shown above.

Figure 2-58. I²C Read Cycle without CRC



Slave provides SCRC[7:0], which is calculated from the I2C_ID, R/W, ADDR, I2C_ID, R/W, and the RDATA bits (32 bits).

Figure 2-59. I²C READ Cycle with CRC

2.17.2.4 Auto-Increment Feature

The auto-increment feature allows writing several consecutive registers within one transmission. Every time an 8-bit word is sent to the device, the internal address index counter is incremented by one and the next register is written. Table 2-18 lists the writing sequence to two consecutive registers. Note that auto increment feature does not support CRC protocol.

Table 2-18. Auto-Increment Example

MASTER ACTION	START	DEVICE ADDRESS = 0x60	WRITE		REGISTER ADDRESS		DATA		DATA		STOP
PMIC device				ACK		ACK		ACK		ACK	

2.17.3 Serial Peripheral Interface (SPI)

The device supports SPI serial-bus interface and it operates as a slave. A single read and write transmissions consist of 24-bit write and read cycles (32-bit if CRC is enabled) in the following order:

- Bits 1-8: ADDR[7:0], Register address
- Bits 9-11: PAGE[2:0], Page address for register
- Bit 12: Read/Write definition, 0 = WRITE, 1 = READ.
- Bits 13-16: RESERVED[4:0], Reserved, use all zeros.

- For Write: Bits 17-24: WDATA[7:0], write data
- For Write with CRC enabled: Bits 25-32: MCRC[7:0], CRC error code calculated from bits 1-24, sent by master
- For Read: Bits 17-24: RDATA[7:0], read data
- For Read with CRC enabled: Bits 25-32: SCRC[7:0], CRC error code calculated from bits 1-16, sent by master, and bits 17-24, sent by slave

The embedded CRC filed can be enabled or disabled from the protocol by setting the I2C1_SPI_CRC_EN register bit to '1' - enabled, '0' - disabled. The default of this bit is configurable through the NVM.

The SDO output is in a high-impedance state when the CS pin is high. When the CS pin is low, the SDO output is always driven low except when the RDATA or SCRC bits are sent. When the RDATA or SCRC bits are sent, the SDO output is driven accordingly.

The address, page, data, and CRC are transmitted MSB first. The slave-select signal, CS , must be low during the cycle transmission. The CS signal resets the interface when it is high, and must be taken high between successive cycles. Data is clocked in on the rising edge of the SCLK clock signal and it is clocked out on the falling edge of SCLK clock signal

The *SPI Timing* diagram shows the timing information for these signals.

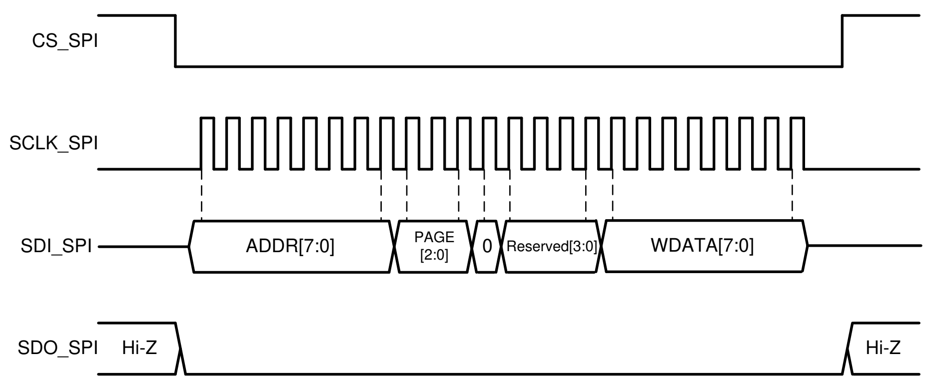


Figure 2-60. SPI Write Cycle

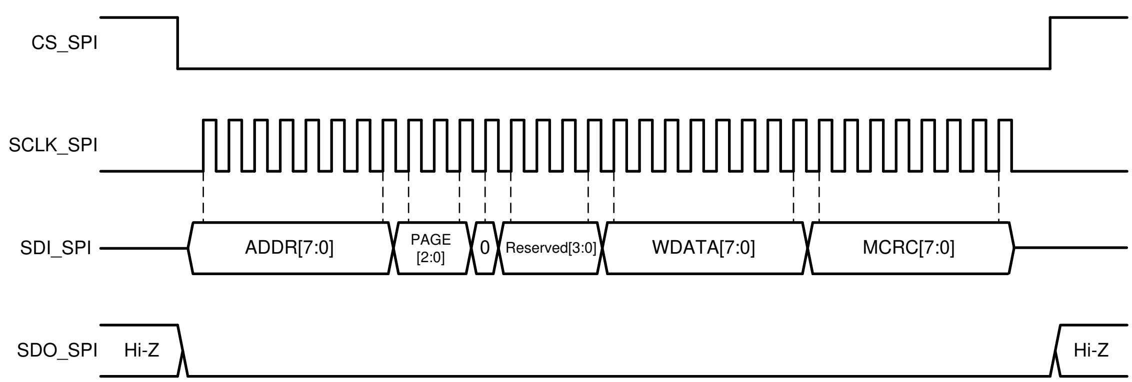


Figure 2-61. SPI Write Cycle with Master CRC

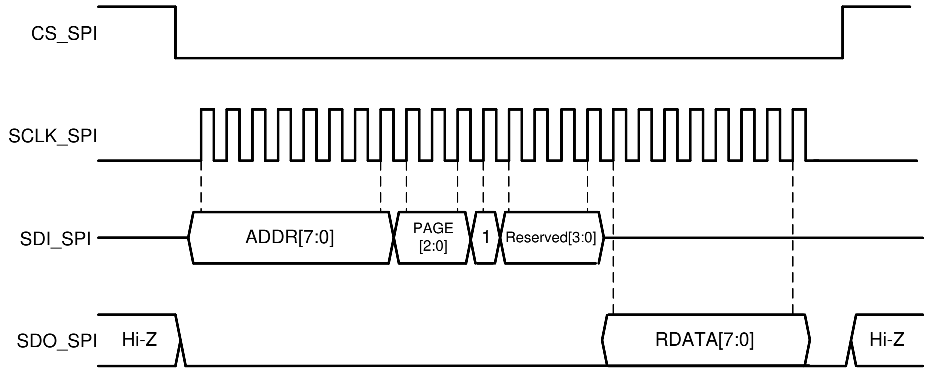


Figure 2-62. SPI Read Cycle

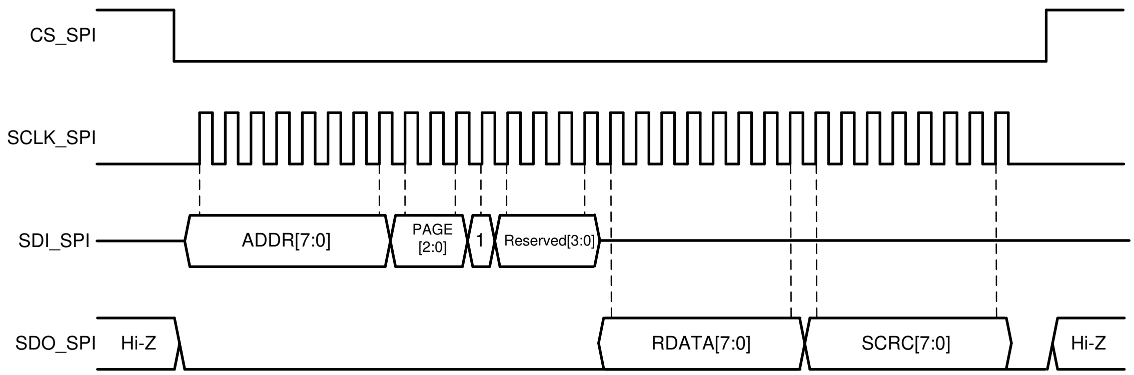


Figure 2-63. SPI Read Cycle with Slave CRC

2.18 Configurable Registers

2.18.1 Register Page Partitioning

The registers in the TPS6594-Q1 device are organized into five internal pages. Below is a list of the pages which the each type of the registers belongs to:

- Page 0: User Registers
- Page 1: NVM Control, Configuration, and Test Registers
- Page 2: Trim Registers
- Page 3: SRAM for PFSM Registers
- Page 4: WatchDog Registers

2.18.2 CRC Protection for Configuration, Control, and Test Registers

A static CRC-16 engine exist to protect all the static registers in the design. Static registers are registers in Page 1, 2, and 3, with values that do not change once loaded from NVM. The CRC-16 engine continuously checks the control registers on the device. The expected CRC-16 value is stored in the NVM. Anytime a mismatch between the calculated and expected CRC-16 values is detected, the interrupt bit REG_CRC_ERR_INT is set and the device will force a orderly shutdown sequence to return to the SAFE RECOVERY state. The device NVM control, configuration, and test registers in page 1 are protected against read or write access when the device is in normal functional mode. The CRC-16 protection for the NVM registers is configured and enabled only when the device is in DFT/DEBUG operating mode.

The CRC-16 engine uses a standard CRC-16 polynomial to calculate the internal known-good checksum-value which is $X^{16} + X^{14} + X^{13} + X^{12} + X^{10} + X^{8} + X^{6} + X^{4} + X^{3} + X + 1$.

The initial value for the remainder of the polynomial is all 1s and is in big-endian bit-stream order. The inversion of the calculated result is enabled.

Note

The CRC-16 engine assume the value of '0' for all undefined or reserved bits in all control registers. Therefore the software MUST NOT write the value of '1' to any of these undefined or reserved bits. If the value of '1 is written to any undefined or reserved bit of a writeable register, a mismatch between the calculated and expected CRC-16 values will be detected, a REG_CRC_ERR interrupt will be set, and the device will force a orderly shutdown sequence to return to the SAFE RECOVERY state.

2.18.3 CRC Protection for User Registers

A dynamic CRC-8 engine exists to protect registers that have values which can change during operation. These are registers in Page 1 and 4. When writes occur to these pages the dynamic CRC-8 is checked, computed, and updated. Continuously during operation the CRC-8 are evaluated and verified in a round-robin fashion.

The CRC-8 engine utilizes the Polynomial(0xA6) = $X^8 + X^6 + X^3 + X^2 + 1$, which provides a H4 hamming distance.

2.18.4 Register Write Protection

For safety application, in order to prevent unintentional writes to the control registers, the TPS6594-Q1 device implements locking and unlocking mechanisms to many of its configuration/control registers described in the following subsections.

2.18.4.1 ESM and WDOG Configuration Registers

The configuration registers for the watchdog and the ESM modules are locked when their monitoring functions are in operation. The timing and the list of the watchdog registers which will locked is described under Section 2.13.2. The list of ESM registers locked after the start of each ESM module is described under Section 2.14

2.18.4.2 User Registers

User registers in page 0, except the ESM and the WDOG configuration registers described in Section 2.18.4.1, and the interrupt registers (x_INT) at address 0x5a through 0x6c in page 0, can be write protected by a dedicated lock. User must write '0x9B' to the REGISTER_LOCK register to unlock the register. Writing any value other than '0x9B' will activate the lock again. To check the register lock status, user should read the REGISTER_LOCK_STATUS bit. When this bit is '0', it indicates the user registers are unlocked. When this bit is '1', the user registers are locked. During startup sequence such as powering up for the first time, waking up from LP_STANDBY, or recovering from SAFE_RECOVERY, the user registers will be unlocked automatically.

As an extra measure of protection to prevent the accidental change of the buck frequency while the buck is in operation, the BUCKn_FREQ_SEL register bits are locked by the REGISTER_LOCK register. User is advised against changing the buck frequency while the buck is in operation.

3 Register Maps

3.1 TPS6594-Q1 Registers

Table 3-1 lists the memory-mapped registers for the TPS6594-Q1 registers. All register offset addresses not listed in Table 3-1 should be considered as reserved locations and the register contents should not be modified.

Table 3-1. TPS6594-Q1 Registers

Offset	Acronym	Register Name	Section
0x1	DEV_REV		Section 3.1.1
0x2	NVM_CODE_1		Section 3.1.2
0x3	NVM_CODE_2		Section 3.1.3
0x4	BUCK1_CTRL		Section 3.1.4
0x5	BUCK1_CONF		Section 3.1.5
0x6	BUCK2_CTRL		Section 3.1.6
0x7	BUCK2_CONF		Section 3.1.7
0x8	BUCK3_CTRL		Section 3.1.8
0x9	BUCK3_CONF		Section 3.1.9
0xA	BUCK4_CTRL		Section 3.1.10
0xB	BUCK4_CONF		Section 3.1.11
0xC	BUCK5_CTRL		Section 3.1.12
0xD	BUCK5_CONF		Section 3.1.13
0xE	BUCK1_VOUT_1		Section 3.1.14
0xF	BUCK1_VOUT_2		Section 3.1.15
0x10	BUCK2_VOUT_1		Section 3.1.16
0x11	BUCK2_VOUT_2		Section 3.1.17
0x12	BUCK3_VOUT_1		Section 3.1.18
0x13	BUCK3_VOUT_2		Section 3.1.19
0x14	BUCK4_VOUT_1		Section 3.1.20
0x15	BUCK4_VOUT_2		Section 3.1.21
0x16	BUCK5_VOUT_1		Section 3.1.22
0x17	BUCK5_VOUT_2		Section 3.1.23
0x18	BUCK1_PG_WINDOW		Section 3.1.24
0x19	BUCK2_PG_WINDOW		Section 3.1.25
0x1A	BUCK3_PG_WINDOW		Section 3.1.26
0x1B	BUCK4_PG_WINDOW		Section 3.1.27
0x1C	BUCK5_PG_WINDOW		Section 3.1.28
0x1D	LDO1_CTRL		Section 3.1.29
0x1E	LDO2_CTRL		Section 3.1.30
0x1F	LDO3_CTRL		Section 3.1.31
0x20	LDO4_CTRL		Section 3.1.32
0x22	LDORTC_CTRL		Section 3.1.33
0x23	LDO1_VOUT		Section 3.1.34
0x24	LDO2_VOUT		Section 3.1.35
0x25	LDO3_VOUT		Section 3.1.36
0x26	LDO4_VOUT		Section 3.1.37
0x27	LDO1_PG_WINDOW		Section 3.1.38
0x28	LDO2_PG_WINDOW		Section 3.1.39
0x29	LDO3_PG_WINDOW		Section 3.1.40
0x2A	LDO4_PG_WINDOW		Section 3.1.41
0x2B	VCCA_VMON_CTRL		Section 3.1.42
0x2C	VCCA_PG_WINDOW		Section 3.1.43

Table 3-1. TPS6594-Q1 Registers (continued)

Offset	Acronym	Register Name	Section
0x31	GPIO1_CONF		Section 3.1.44
0x32	GPIO2_CONF		Section 3.1.45
0x33	GPIO3_CONF		Section 3.1.46
0x34	GPIO4_CONF		Section 3.1.47
0x35	GPIO5_CONF		Section 3.1.48
0x36	GPIO6_CONF		Section 3.1.49
0x37	GPIO7_CONF		Section 3.1.50
0x38	GPIO8_CONF		Section 3.1.51
0x39	GPIO9_CONF		Section 3.1.52
0x3A	GPIO10_CONF		Section 3.1.53
0x3B	GPIO11_CONF		Section 3.1.54
0x3C	NPWRON_CONF		Section 3.1.55
0x3D	GPIO_OUT_1		Section 3.1.56
0x3E	GPIO_OUT_2		Section 3.1.57
0x3F	GPIO_IN_1		Section 3.1.58
0x40	GPIO_IN_2		Section 3.1.59
0x41	RAIL_SEL_1		Section 3.1.60
0x42	RAIL_SEL_2		Section 3.1.61
0x43	RAIL_SEL_3		Section 3.1.62
0x44	FSM_TRIG_SEL_1		Section 3.1.63
0x45	FSM_TRIG_SEL_2		Section 3.1.64
0x46	FSM_TRIG_MASK_1		Section 3.1.65
0x47	FSM_TRIG_MASK_2		Section 3.1.66
0x48	FSM_TRIG_MASK_3		Section 3.1.67
0x49	MASK_BUCK1_2		Section 3.1.68
0x4A	MASK_BUCK3_4		Section 3.1.69
0x4B	MASK_BUCK5		Section 3.1.70
0x4C	MASK_LDO1_2		Section 3.1.71
0x4D	MASK_LDO3_4		Section 3.1.72
0x4E	MASK_VMON		Section 3.1.73
0x4F	MASK_GPIO1_8_FALL		Section 3.1.74
0x50	MASK_GPIO1_8_RISE		Section 3.1.75
0x51	MASK_GPIO9_11		Section 3.1.76
0x52	MASK_STARTUP		Section 3.1.77
0x53	MASK_MISC		Section 3.1.78
0x54	MASK_MODERATE_ERR		Section 3.1.79
0x56	MASK_FSM_ERR		Section 3.1.80
0x57	MASK_COMM_ERR		Section 3.1.81
0x58	MASK_READBACK_ERR		Section 3.1.82
0x59	MASK_ESM		Section 3.1.83
0x5A	INT_TOP		Section 3.1.84
0x5B	INT_BUCK		Section 3.1.85
0x5C	INT_BUCK1_2		Section 3.1.86
0x5D	INT_BUCK3_4		Section 3.1.87
0x5E	INT_BUCK5		Section 3.1.88
0x5F	INT_LDO_VMON		Section 3.1.89
0x60	INT_LDO1_2		Section 3.1.90

Table 3-1. TPS6594-Q1 Registers (continued)

Offset	Acronym	Register Name	Section
0x61	INT_LDO3_4		Section 3.1.91
0x62	INT_VMON		Section 3.1.92
0x63	INT_GPIO		Section 3.1.93
0x64	INT_GPIO1_8		Section 3.1.94
0x65	INT_STARTUP		Section 3.1.95
0x66	INT_MISC		Section 3.1.96
0x67	INT_MODERATE_ERR		Section 3.1.97
0x68	INT_SEVERE_ERR		Section 3.1.98
0x69	INT_FSM_ERR		Section 3.1.99
0x6A	INT_COMM_ERR		Section 3.1.100
0x6B	INT_READBACK_ERR		Section 3.1.101
0x6C	INT_ESM		Section 3.1.102
0x6D	STAT_BUCK1_2		Section 3.1.103
0x6E	STAT_BUCK3_4		Section 3.1.104
0x6F	STAT_BUCK5		Section 3.1.105
0x70	STAT_LDO1_2		Section 3.1.106
0x71	STAT_LDO3_4		Section 3.1.107
0x72	STAT_VMON		Section 3.1.108
0x73	STAT_STARTUP		Section 3.1.109
0x74	STAT_MISC		Section 3.1.110
0x75	STAT_MODERATE_ERR		Section 3.1.111
0x76	STAT_SEVERE_ERR		Section 3.1.112
0x77	STAT_READBACK_ERR		Section 3.1.113
0x78	PGOOD_SEL_1		Section 3.1.114
0x79	PGOOD_SEL_2		Section 3.1.115
0x7A	PGOOD_SEL_3		Section 3.1.116
0x7B	PGOOD_SEL_4		Section 3.1.117
0x7C	PLL_CTRL		Section 3.1.118
0x7D	CONFIG_1		Section 3.1.119
0x7E	CONFIG_2		Section 3.1.120
0x80	ENABLE_DRV_REG		Section 3.1.121
0x81	MISC_CTRL		Section 3.1.122
0x82	ENABLE_DRV_STAT		Section 3.1.123
0x83	RECOV_CNT_REG_1		Section 3.1.124
0x84	RECOV_CNT_REG_2		Section 3.1.125
0x85	FSM_I2C_TRIGGERS		Section 3.1.126
0x86	FSM_NSLEEP_TRIGGERS		Section 3.1.127
0x87	BUCK_RESET_REG		Section 3.1.128
0x88	SPREAD_SPECTRUM_1		Section 3.1.129
0x8A	FREQ_SEL		Section 3.1.130
0x8B	FSM_STEP_SIZE		Section 3.1.131
0x8C	LDO_RV_TIMEOUT_REG_1		Section 3.1.132
0x8D	LDO_RV_TIMEOUT_REG_2		Section 3.1.133
0x8E	USER_SPARE_REGS		Section 3.1.134
0x8F	ESM_MCU_START_REG		Section 3.1.135
0x90	ESM_MCU_DELAY1_REG		Section 3.1.136
0x91	ESM_MCU_DELAY2_REG		Section 3.1.137

Table 3-1. TPS6594-Q1 Registers (continued)

Offset	Acronym	Register Name	Section
0x92	ESM_MCU_MODE_CFG		Section 3.1.138
0x93	ESM_MCU_HMAX_REG		Section 3.1.139
0x94	ESM_MCU_HMIN_REG		Section 3.1.140
0x95	ESM_MCU_LMAX_REG		Section 3.1.141
0x96	ESM_MCU_LMIN_REG		Section 3.1.142
0x97	ESM_MCU_ERR_CNT_REG		Section 3.1.143
0x98	ESM_SOC_START_REG		Section 3.1.144
0x99	ESM_SOC_DELAY1_REG		Section 3.1.145
0x9A	ESM_SOC_DELAY2_REG		Section 3.1.146
0x9B	ESM_SOC_MODE_CFG		Section 3.1.147
0x9C	ESM_SOC_HMAX_REG		Section 3.1.148
0x9D	ESM_SOC_HMIN_REG		Section 3.1.149
0x9E	ESM_SOC_LMAX_REG		Section 3.1.150
0x9F	ESM_SOC_LMIN_REG		Section 3.1.151
0xA0	ESM_SOC_ERR_CNT_REG		Section 3.1.152
0xA1	REGISTER_LOCK		Section 3.1.153
0xA6	MANUFACTURING_VER		Section 3.1.154
0xA7	CUSTOMER_NVM_ID_REG		Section 3.1.155
0xAB	SOFT_REBOOT_REG		Section 3.1.156
0xB5	RTC_SECONDS		Section 3.1.157
0xB6	RTC_MINUTES		Section 3.1.158
0xB7	RTC_HOURS		Section 3.1.159
0xB8	RTC_DAYS		Section 3.1.160
0xB9	RTC_MONTHS		Section 3.1.161
0xBA	RTC_YEARS		Section 3.1.162
0xBB	RTC_WEEKS		Section 3.1.163
0xBC	ALARM_SECONDS		Section 3.1.164
0xBD	ALARM_MINUTES		Section 3.1.165
0xBE	ALARM_HOURS		Section 3.1.166
0xBF	ALARM_DAYS		Section 3.1.167
0xC0	ALARM_MONTHS		Section 3.1.168
0xC1	ALARM_YEARS		Section 3.1.169
0xC2	RTC_CTRL_1		Section 3.1.170
0xC3	RTC_CTRL_2		Section 3.1.171
0xC4	RTC_STATUS		Section 3.1.172
0xC5	RTC_INTERRUPTS		Section 3.1.173
0xC6	RTC_COMP_LSB		Section 3.1.174
0xC7	RTC_COMP_MSB		Section 3.1.175
0xC8	RTC_RESET_STATUS		Section 3.1.176
0xC9	SCRATCH_PAD_REG_1		Section 3.1.177
0xCA	SCRATCH_PAD_REG_2		Section 3.1.178
0xCB	SCRATCH_PAD_REG_3		Section 3.1.179
0xCC	SCRATCH_PAD_REG_4		Section 3.1.180
0xCD	PFSM_DELAY_REG_1		Section 3.1.181
0xCE	PFSM_DELAY_REG_2		Section 3.1.182
0xCF	PFSM_DELAY_REG_3		Section 3.1.183
0xD0	PFSM_DELAY_REG_4		Section 3.1.184

Table 3-1. TPS6594-Q1 Registers (continued)

Offset	Acronym	Register Name	Section
0x401	WD_ANSWER_REG		Section 3.1.185
0x402	WD_QUESTION_ANSW_CNT		Section 3.1.186
0x403	WD_WIN1_CFG		Section 3.1.187
0x404	WD_WIN2_CFG		Section 3.1.188
0x405	WD_LONGWIN_CFG		Section 3.1.189
0x406	WD_MODE_REG		Section 3.1.190
0x407	WD_QA_CFG		Section 3.1.191
0x408	WD_ERR_STATUS		Section 3.1.192
0x409	WD_THR_CFG		Section 3.1.193
0x40A	WD_FAIL_CNT_REG		Section 3.1.194

Complex bit access types are encoded to fit into small table cells. Table 3-2 shows the codes that are used for access types in this section.

Table 3-2. TPS6594-Q1 Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
W1C	W 1C	Write 1 to clear
WSelfClrF	W	Write
Reset or Default Value		
-n		Value after reset or the default value
Register Array Variables		
i,j,k,l,m,n		When these variables are used in a register name, an offset, or an address, they refer to the value of a register array where the register is part of a group of repeating registers. The register groups form a hierarchical structure and the array is represented with a formula.
y		When this variable is used in a register name, an offset, or an address it refers to the value of a register array.

3.1.1 DEV_REV Register (Offset = 0x1) [Reset = 0x0]

DEV_REV is shown in Figure 3-1 and described in Table 3-3.

Return to the Table 3-1.

Figure 3-1. DEV_REV Register

7	6	5	4	3	2	1	0
TI_DEVICE_ID							
R/W-0b							

Table 3-3. DEV_REV Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	TI_DEVICE_ID	R/W	0b	TI_DEVICE_ID[7]: 0 - Industrial 1 - Auto TI_DEVICE_ID[6:2] = Device GPN TI_DEVICE_ID[1]: 0 - QM 1 - ASIL TI_DEVICE_ID[0] = Reserved Note: This register can be programmed only by the manufacturer. (Default from NVM memory)

3.1.2 NVM_CODE_1 Register (Offset = 0x2) [Reset = 0x0]

NVM_CODE_1 is shown in Figure 3-2 and described in Table 3-4.

Return to the Table 3-1.

Figure 3-2. NVM_CODE_1 Register

7	6	5	4	3	2	1	0
TI_NVM_ID							
R/W-0b							

Table 3-4. NVM_CODE_1 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	TI_NVM_ID	R/W	0b	Note: This register can be programmed only by the manufacturer. (Default from NVM memory)

3.1.3 NVM_CODE_2 Register (Offset = 0x3) [Reset = 0x0]

NVM_CODE_2 is shown in Figure 3-3 and described in Table 3-5.

Return to the Table 3-1.

Figure 3-3. NVM_CODE_2 Register

7	6	5	4	3	2	1	0
TI_NVM_REV							
R/W-0b							

Table 3-5. NVM_CODE_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	TI_NVM_REV	R/W	0b	NVM revision of the IC Note: This register can be programmed only by the manufacturer. (Default from NVM memory)

3.1.4 BUCK1_CTRL Register (Offset = 0x4) [Reset = 0x22]

BUCK1_CTRL is shown in Figure 3-4 and described in Table 3-6.

Return to the Table 3-1.

Figure 3-4. BUCK1_CTRL Register

7	6	5	4	3	2	1	0
BUCK1_RV_SEL	RESERVED	BUCK1_PLDN	BUCK1_VMON_EN	BUCK1_VSEL	BUCK1_FPWM_MP	BUCK1_FPWM	BUCK1_EN
R/W-0b	R/W-0b	R/W-1b	R/W-0b	R/W-0b	R/W-0b	R/W-1b	R/W-0b

Table 3-6. BUCK1_CTRL Register Field Descriptions

Bit	Field	Type	Reset	Description
7	BUCK1_RV_SEL	R/W	0b	Select residual voltage checking for BUCK1 feedback pin. (Default from NVM memory) 0b = Disabled 1b = Enabled
6	RESERVED	R/W	0b	
5	BUCK1_PLDN	R/W	1b	Enable output pull-down resistor when BUCK1 is disabled: (Default from NVM memory) 0b = Pull-down resistor disabled 1b = Pull-down resistor enabled
4	BUCK1_VMON_EN	R/W	0b	Enable BUCK1 OV, UV, SC and ILIM comparators: (Default from NVM memory) 0b = OV, UV, SC and ILIM comparators are disabled 1b = OV, UV, SC and ILIM comparators are enabled
3	BUCK1_VSEL	R/W	0b	Select output voltage register for BUCK1: (Default from NVM memory) 0b = BUCK1_VOUT_1 1b = BUCK1_VOUT_2
2	BUCK1_FPWM_MP	R/W	0b	Forces the BUCK1 regulator to operate always in multi-phase and forced PWM operation mode: (Default from NVM memory) 0b = Automatic phase adding and shedding. 1b = Forced to multi-phase operation, all phases in the multi-phase configuration.
1	BUCK1_FPWM	R/W	1b	Forces the BUCK1 regulator to operate in PWM mode: (Default from NVM memory) 0b = Automatic transitions between PFM and PWM modes (AUTO mode). 1b = Forced to PWM operation.
0	BUCK1_EN	R/W	0b	Enable BUCK1 regulator: (Default from NVM memory) 0b = BUCK regulator is disabled 1b = BUCK regulator is enabled

3.1.5 BUCK1_CONF Register (Offset = 0x5) [Reset = 0x22]

BUCK1_CONF is shown in Figure 3-5 and described in Table 3-7.

Return to the Table 3-1.

Figure 3-5. BUCK1_CONF Register

7	6	5	4	3	2	1	0
RESERVED		BUCK1_ILIM			BUCK1_SLEW_RATE		
R/W-0b		R/W-100b			R/W-10b		

Table 3-7. BUCK1_CONF Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5:3	BUCK1_ILIM	R/W	100b	Sets the switch peak current limit of BUCK1. Can be programmed at any time during operation: (Default from NVM memory) 0b = Reserved 1b = Reserved 10b = 2.5 A 11b = 3.5 A 100b = 4.5 A 101b = 5.5 A 110b = Reserved 111b = Reserved
2:0	BUCK1_SLEW_RATE	R/W	10b	Sets the output voltage slew rate for BUCK1 regulator (rising and falling edges): (Default from NVM memory) 0b = 33 mV/µs 1b = 20 mV/µs 10b = 10 mV/µs 11b = 5.0 mV/µs 100b = 2.5 mV/µs 101b = 1.3 mV/µs 110b = 0.63 mV/µs 111b = 0.31 mV/µs

3.1.6 BUCK2_CTRL Register (Offset = 0x6) [Reset = 0x22]

BUCK2_CTRL is shown in Figure 3-6 and described in Table 3-8.

Return to the Table 3-1.

Figure 3-6. BUCK2_CTRL Register

7	6	5	4	3	2	1	0
BUCK2_RV_SEL	RESERVED	BUCK2_PLDN	BUCK2_VMON_EN	BUCK2_VSEL	RESERVED	BUCK2_FPWM	BUCK2_EN
R/W-0b	R/W-0b	R/W-1b	R/W-0b	R/W-0b	R/W-0b	R/W-1b	R/W-0b

Table 3-8. BUCK2_CTRL Register Field Descriptions

Bit	Field	Type	Reset	Description
7	BUCK2_RV_SEL	R/W	0b	Select residual voltage checking for BUCK2 feedback pin. (Default from NVM memory) 0b = Disabled 1b = Enabled
6	RESERVED	R/W	0b	
5	BUCK2_PLDN	R/W	1b	Enable output pull-down resistor when BUCK2 is disabled: (Default from NVM memory) 0b = Pull-down resistor disabled 1b = Pull-down resistor enabled
4	BUCK2_VMON_EN	R/W	0b	Enable BUCK2 OV, UV, SC and ILIM comparators: (Default from NVM memory) 0b = OV, UV, SC and ILIM comparators are disabled 1b = OV, UV, SC and ILIM comparators are enabled
3	BUCK2_VSEL	R/W	0b	Select output voltage register for BUCK2: (Default from NVM memory) 0b = BUCK2_VOUT_1 1b = BUCK2_VOUT_2
2	RESERVED	R/W	0b	
1	BUCK2_FPWM	R/W	1b	Forces the BUCK2 regulator to operate in PWM mode: (Default from NVM memory) 0b = Automatic transitions between PFM and PWM modes (AUTO mode). 1b = Forced to PWM operation.
0	BUCK2_EN	R/W	0b	Enable BUCK2 regulator: (Default from NVM memory) 0b = BUCK regulator is disabled 1b = BUCK regulator is enabled

3.1.7 BUCK2_CONF Register (Offset = 0x7) [Reset = 0x22]

BUCK2_CONF is shown in Figure 3-7 and described in Table 3-9.

Return to the Table 3-1.

Figure 3-7. BUCK2_CONF Register

7	6	5	4	3	2	1	0
RESERVED		BUCK2_ILIM			BUCK2_SLEW_RATE		
R/W-0b		R/W-100b			R/W-10b		

Table 3-9. BUCK2_CONF Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5:3	BUCK2_ILIM	R/W	100b	Sets the switch peak current limit of BUCK2. Can be programmed at any time during operation: (Default from NVM memory) 0b = Reserved 1b = Reserved 10b = 2.5 A 11b = 3.5 A 100b = 4.5 A 101b = 5.5 A 110b = Reserved 111b = Reserved
2:0	BUCK2_SLEW_RATE	R/W	10b	Sets the output voltage slew rate for BUCK2 regulator (rising and falling edges): (Default from NVM memory) 0b = 33 mV/μs 1b = 20 mV/μs 10b = 10 mV/μs 11b = 5.0 mV/μs 100b = 2.5 mV/μs 101b = 1.3 mV/μs 110b = 0.63 mV/μs 111b = 0.31 mV/μs

3.1.8 BUCK3_CTRL Register (Offset = 0x8) [Reset = 0x22]

BUCK3_CTRL is shown in Figure 3-8 and described in Table 3-10.

Return to the Table 3-1.

Figure 3-8. BUCK3_CTRL Register

7	6	5	4	3	2	1	0
BUCK3_RV_SEL	RESERVED	BUCK3_PLDN	BUCK3_VMON_EN	BUCK3_VSEL	BUCK3_FPWM_MP	BUCK3_FPWM	BUCK3_EN
R/W-0b	R/W-0b	R/W-1b	R/W-0b	R/W-0b	R/W-0b	R/W-1b	R/W-0b

Table 3-10. BUCK3_CTRL Register Field Descriptions

Bit	Field	Type	Reset	Description
7	BUCK3_RV_SEL	R/W	0b	Select residual voltage checking for BUCK3 feedback pin. (Default from NVM memory) 0b = Disabled 1b = Enabled
6	RESERVED	R/W	0b	
5	BUCK3_PLDN	R/W	1b	Enable output pull-down resistor when BUCK3 is disabled: (Default from NVM memory) 0b = Pull-down resistor disabled 1b = Pull-down resistor enabled
4	BUCK3_VMON_EN	R/W	0b	Enable BUCK3 OV, UV, SC and ILIM comparators: (Default from NVM memory) 0b = OV, UV, SC and ILIM comparators are disabled 1b = OV, UV, SC and ILIM comparators are enabled
3	BUCK3_VSEL	R/W	0b	Select output voltage register for BUCK3: (Default from NVM memory) 0b = BUCK3_VOUT_1 1b = BUCK3_VOUT_2
2	BUCK3_FPWM_MP	R/W	0b	Forces the BUCK3 regulator to operate always in multi-phase and forced PWM operation mode: (Default from NVM memory) 0b = Automatic phase adding and shedding. 1b = Forced to multi-phase operation, all phases in the multi-phase configuration.
1	BUCK3_FPWM	R/W	1b	Forces the BUCK3 regulator to operate in PWM mode: (Default from NVM memory) 0b = Automatic transitions between PFM and PWM modes (AUTO mode). 1b = Forced to PWM operation.
0	BUCK3_EN	R/W	0b	Enable BUCK3 regulator: (Default from NVM memory) 0b = BUCK regulator is disabled 1b = BUCK regulator is enabled

3.1.9 BUCK3_CONF Register (Offset = 0x9) [Reset = 0x22]

BUCK3_CONF is shown in Figure 3-9 and described in Table 3-11.

Return to the Table 3-1.

Figure 3-9. BUCK3_CONF Register

7	6	5	4	3	2	1	0
RESERVED		BUCK3_ILIM			BUCK3_SLEW_RATE		
R/W-0b		R/W-100b			R/W-10b		

Table 3-11. BUCK3_CONF Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5:3	BUCK3_ILIM	R/W	100b	Sets the switch peak current limit of BUCK3. Can be programmed at any time during operation: (Default from NVM memory) 0b = Reserved 1b = Reserved 10b = 2.5 A 11b = 3.5 A 100b = 4.5 A 101b = 5.5 A 110b = Reserved 111b = Reserved
2:0	BUCK3_SLEW_RATE	R/W	10b	Sets the output voltage slew rate for BUCK3 regulator (rising and falling edges): (Default from NVM memory) 0b = 33 mV/µs 1b = 20 mV/µs 10b = 10 mV/µs 11b = 5.0 mV/µs 100b = 2.5 mV/µs 101b = 1.3 mV/µs 110b = 0.63 mV/µs 111b = 0.31 mV/µs

3.1.10 BUCK4_CTRL Register (Offset = 0xA) [Reset = 0x22]

BUCK4_CTRL is shown in Figure 3-10 and described in Table 3-12.

Return to the Table 3-1.

Figure 3-10. BUCK4_CTRL Register

7	6	5	4	3	2	1	0
BUCK4_RV_SEL	RESERVED	BUCK4_PLDN	BUCK4_VMON_EN	BUCK4_VSEL	RESERVED	BUCK4_FPWM	BUCK4_EN
R/W-0b	R/W-0b	R/W-1b	R/W-0b	R/W-0b	R/W-0b	R/W-1b	R/W-0b

Table 3-12. BUCK4_CTRL Register Field Descriptions

Bit	Field	Type	Reset	Description
7	BUCK4_RV_SEL	R/W	0b	Select residual voltage checking for BUCK4 feedback pin. (Default from NVM memory) 0b = Disabled 1b = Enabled
6	RESERVED	R/W	0b	
5	BUCK4_PLDN	R/W	1b	Enable output pull-down resistor when BUCK4 is disabled: (Default from NVM memory) 0b = Pull-down resistor disabled 1b = Pull-down resistor enabled
4	BUCK4_VMON_EN	R/W	0b	Enable BUCK4 OV, UV, SC and ILIM comparators: (Default from NVM memory) 0b = OV, UV, SC and ILIM comparators are disabled 1b = OV, UV, SC and ILIM comparators are enabled
3	BUCK4_VSEL	R/W	0b	Select output voltage register for BUCK4: (Default from NVM memory) 0b = BUCK4_VOUT_1 1b = BUCK4_VOUT_2
2	RESERVED	R/W	0b	
1	BUCK4_FPWM	R/W	1b	Forces the BUCK4 regulator to operate in PWM mode: (Default from NVM memory) 0b = Automatic transitions between PFM and PWM modes (AUTO mode). 1b = Forced to PWM operation.
0	BUCK4_EN	R/W	0b	Enable BUCK4 regulator: (Default from NVM memory) 0b = BUCK regulator is disabled 1b = BUCK regulator is enabled

3.1.11 BUCK4_CONF Register (Offset = 0xB) [Reset = 0x22]

BUCK4_CONF is shown in Figure 3-11 and described in Table 3-13.

Return to the Table 3-1.

Figure 3-11. BUCK4_CONF Register

7	6	5	4	3	2	1	0
RESERVED		BUCK4_ILIM			BUCK4_SLEW_RATE		
R/W-0b		R/W-100b			R/W-10b		

Table 3-13. BUCK4_CONF Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5:3	BUCK4_ILIM	R/W	100b	Sets the switch peak current limit of BUCK4. Can be programmed at any time during operation: (Default from NVM memory) 0b = Reserved 1b = Reserved 10b = 2.5 A 11b = 3.5 A 100b = 4.5 A 101b = 5.5 A 110b = Reserved 111b = Reserved
2:0	BUCK4_SLEW_RATE	R/W	10b	Sets the output voltage slew rate for BUCK4 regulator (rising and falling edges): (Default from NVM memory) 0b = 33 mV/µs 1b = 20 mV/µs 10b = 10 mV/µs 11b = 5.0 mV/µs 100b = 2.5 mV/µs 101b = 1.3 mV/µs 110b = 0.63 mV/µs 111b = 0.31 mV/µs

3.1.12 BUCK5_CTRL Register (Offset = 0xC) [Reset = 0x22]

BUCK5_CTRL is shown in Figure 3-12 and described in Table 3-14.

Return to the Table 3-1.

Figure 3-12. BUCK5_CTRL Register

7	6	5	4	3	2	1	0
BUCK5_RV_SEL	RESERVED	BUCK5_PLDN	BUCK5_VMON_EN	BUCK5_VSEL	RESERVED	BUCK5_FPWM	BUCK5_EN
R/W-0b	R/W-0b	R/W-1b	R/W-0b	R/W-0b	R/W-0b	R/W-1b	R/W-0b

Table 3-14. BUCK5_CTRL Register Field Descriptions

Bit	Field	Type	Reset	Description
7	BUCK5_RV_SEL	R/W	0b	Select residual voltage checking for BUCK5 feedback pin. (Default from NVM memory) 0b = Disabled 1b = Enabled
6	RESERVED	R/W	0b	
5	BUCK5_PLDN	R/W	1b	Enable output pull-down resistor when BUCK5 is disabled: (Default from NVM memory) 0b = Pull-down resistor disabled 1b = Pull-down resistor enabled
4	BUCK5_VMON_EN	R/W	0b	Enable BUCK5 OV, UV, SC and ILIM comparators: (Default from NVM memory) 0b = OV, UV, SC and ILIM comparators are disabled 1b = OV, UV, SC and ILIM comparators are enabled
3	BUCK5_VSEL	R/W	0b	Select output voltage register for BUCK5: (Default from NVM memory) 0b = BUCK5_VOUT_1 1b = BUCK5_VOUT_2
2	RESERVED	R/W	0b	
1	BUCK5_FPWM	R/W	1b	Forces the BUCK5 regulator to operate in PWM mode: (Default from NVM memory) 0b = Automatic transitions between PFM and PWM modes (AUTO mode). 1b = Forced to PWM operation.
0	BUCK5_EN	R/W	0b	Enable BUCK5 regulator: (Default from NVM memory) 0b = BUCK regulator is disabled 1b = BUCK regulator is enabled

3.1.13 BUCK5_CONF Register (Offset = 0xD) [Reset = 0x22]

BUCK5_CONF is shown in Figure 3-13 and described in Table 3-15.

Return to the Table 3-1.

Figure 3-13. BUCK5_CONF Register

7	6	5	4	3	2	1	0
RESERVED		BUCK5_ILIM			BUCK5_SLEW_RATE		
R/W-0b		R/W-100b			R/W-10b		

Table 3-15. BUCK5_CONF Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5:3	BUCK5_ILIM	R/W	100b	Sets the switch peak current limit of BUCK5. Can be programmed at any time during operation: (Default from NVM memory) 0b = Reserved 1b = Reserved 10b = 2.5 A 11b = 3.5 A 100b = Reserved 101b = Reserved 110b = Reserved 111b = Reserved
2:0	BUCK5_SLEW_RATE	R/W	10b	Sets the output voltage slew rate for BUCK5 regulator (rising and falling edges): (Default from NVM memory) 0b = 33 mV/µs 1b = 20 mV/µs 10b = 10 mV/µs 11b = 5.0 mV/µs 100b = 2.5 mV/µs 101b = 1.3 mV/µs 110b = 0.63 mV/µs 111b = 0.31 mV/µs

3.1.14 BUCK1_VOUT_1 Register (Offset = 0xE) [Reset = 0x0]

BUCK1_VOUT_1 is shown in Figure 3-14 and described in Table 3-16.

Return to the Table 3-1.

Figure 3-14. BUCK1_VOUT_1 Register

7	6	5	4	3	2	1	0
BUCK1_VSET1							
R/W-0b							

Table 3-16. BUCK1_VOUT_1 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	BUCK1_VSET1	R/W	0b	Voltage selection for buck regulator. See Buck regulators chapter for voltage levels. (Default from NVM memory)

3.1.15 BUCK1_VOUT_2 Register (Offset = 0xF) [Reset = 0x0]

BUCK1_VOUT_2 is shown in Figure 3-15 and described in Table 3-17.

Return to the Table 3-1.

Figure 3-15. BUCK1_VOUT_2 Register

7	6	5	4	3	2	1	0
BUCK1_VSET2							
R/W-0b							

Table 3-17. BUCK1_VOUT_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	BUCK1_VSET2	R/W	0b	Voltage selection for buck regulator. See Buck regulators chapter for voltage levels. (Default from NVM memory)

3.1.16 BUCK2_VOUT_1 Register (Offset = 0x10) [Reset = 0x0]

BUCK2_VOUT_1 is shown in Figure 3-16 and described in Table 3-18.

Return to the Table 3-1.

Figure 3-16. BUCK2_VOUT_1 Register

7	6	5	4	3	2	1	0
BUCK2_VSET1							
R/W-0b							

Table 3-18. BUCK2_VOUT_1 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	BUCK2_VSET1	R/W	0b	Voltage selection for buck regulator. See Buck regulators chapter for voltage levels. (Default from NVM memory)

3.1.17 BUCK2_VOUT_2 Register (Offset = 0x11) [Reset = 0x0]

BUCK2_VOUT_2 is shown in Figure 3-17 and described in Table 3-19.

Return to the Table 3-1.

Figure 3-17. BUCK2_VOUT_2 Register

7	6	5	4	3	2	1	0
BUCK2_VSET2							
R/W-0b							

Table 3-19. BUCK2_VOUT_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	BUCK2_VSET2	R/W	0b	Voltage selection for buck regulator. See Buck regulators chapter for voltage levels. (Default from NVM memory)

3.1.18 BUCK3_VOUT_1 Register (Offset = 0x12) [Reset = 0x0]

BUCK3_VOUT_1 is shown in Figure 3-18 and described in Table 3-20.

Return to the Table 3-1.

Figure 3-18. BUCK3_VOUT_1 Register

7	6	5	4	3	2	1	0
BUCK3_VSET1							
R/W-0b							

Table 3-20. BUCK3_VOUT_1 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	BUCK3_VSET1	R/W	0b	Voltage selection for buck regulator. See Buck regulators chapter for voltage levels. (Default from NVM memory)

3.1.19 BUCK3_VOUT_2 Register (Offset = 0x13) [Reset = 0x0]

BUCK3_VOUT_2 is shown in Figure 3-19 and described in Table 3-21.

Return to the Table 3-1.

Figure 3-19. BUCK3_VOUT_2 Register

7	6	5	4	3	2	1	0
BUCK3_VSET2							
R/W-0b							

Table 3-21. BUCK3_VOUT_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	BUCK3_VSET2	R/W	0b	Voltage selection for buck regulator. See Buck regulators chapter for voltage levels. (Default from NVM memory)

3.1.20 BUCK4_VOUT_1 Register (Offset = 0x14) [Reset = 0x0]

BUCK4_VOUT_1 is shown in Figure 3-20 and described in Table 3-22.

Return to the Table 3-1.

Figure 3-20. BUCK4_VOUT_1 Register

7	6	5	4	3	2	1	0
BUCK4_VSET1							
R/W-0b							

Table 3-22. BUCK4_VOUT_1 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	BUCK4_VSET1	R/W	0b	Voltage selection for buck regulator. See Buck regulators chapter for voltage levels. (Default from NVM memory)

3.1.21 BUCK4_VOUT_2 Register (Offset = 0x15) [Reset = 0x0]

BUCK4_VOUT_2 is shown in Figure 3-21 and described in Table 3-23.

Return to the Table 3-1.

Figure 3-21. BUCK4_VOUT_2 Register

7	6	5	4	3	2	1	0
BUCK4_VSET2							
R/W-0b							

Table 3-23. BUCK4_VOUT_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	BUCK4_VSET2	R/W	0b	Voltage selection for buck regulator. See Buck regulators chapter for voltage levels. (Default from NVM memory)

3.1.22 BUCK5_VOUT_1 Register (Offset = 0x16) [Reset = 0x0]

BUCK5_VOUT_1 is shown in Figure 3-22 and described in Table 3-24.

Return to the Table 3-1.

Figure 3-22. BUCK5_VOUT_1 Register

7	6	5	4	3	2	1	0
BUCK5_VSET1							
R/W-0b							

Table 3-24. BUCK5_VOUT_1 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	BUCK5_VSET1	R/W	0b	Voltage selection for buck regulator. See Buck regulators chapter for voltage levels. (Default from NVM memory)

3.1.23 BUCK5_VOUT_2 Register (Offset = 0x17) [Reset = 0x0]

BUCK5_VOUT_2 is shown in Figure 3-23 and described in Table 3-25.

Return to the Table 3-1.

Figure 3-23. BUCK5_VOUT_2 Register

7	6	5	4	3	2	1	0
BUCK5_VSET2							
R/W-0b							

Table 3-25. BUCK5_VOUT_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	BUCK5_VSET2	R/W	0b	Voltage selection for buck regulator. See Buck regulators chapter for voltage levels. (Default from NVM memory)

3.1.24 BUCK1_PG_WINDOW Register (Offset = 0x18) [Reset = 0x0]

BUCK1_PG_WINDOW is shown in Figure 3-24 and described in Table 3-26.

Return to the Table 3-1.

Figure 3-24. BUCK1_PG_WINDOW Register

7	6	5	4	3	2	1	0
RESERVED		BUCK1_UV_THR			BUCK1_OV_THR		
R/W-0b		R/W-0b			R/W-0b		

Table 3-26. BUCK1_PG_WINDOW Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5:3	BUCK1_UV_THR	R/W	0b	Powergood low threshold level for BUCK1: (Default from NVM memory) 0b = -3% / -30mV 1b = -3.5% / -35 mV 10b = -4% / -40 mV 11b = -5% / -50 mV 100b = -6% / -60 mV 101b = -7% / -70 mV 110b = -8% / -80 mV 111b = -10% / -100mV
2:0	BUCK1_OV_THR	R/W	0b	Powergood high threshold level for BUCK1: (Default from NVM memory) 0b = +3% / +30mV 1b = +3.5% / +35 mV 10b = +4% / +40 mV 11b = +5% / +50 mV 100b = +6% / +60 mV 101b = +7% / +70 mV 110b = +8% / +80 mV 111b = +10% / +100mV

3.1.25 BUCK2_PG_WINDOW Register (Offset = 0x19) [Reset = 0x0]

BUCK2_PG_WINDOW is shown in Figure 3-25 and described in Table 3-27.

Return to the Table 3-1.

Figure 3-25. BUCK2_PG_WINDOW Register

7	6	5	4	3	2	1	0
RESERVED		BUCK2_UV_THR			BUCK2_OV_THR		
R/W-0b		R/W-0b			R/W-0b		

Table 3-27. BUCK2_PG_WINDOW Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5:3	BUCK2_UV_THR	R/W	0b	Powergood low threshold level for BUCK2: (Default from NVM memory) 0b = -3% / -30mV 1b = -3.5% / -35 mV 10b = -4% / -40 mV 11b = -5% / -50 mV 100b = -6% / -60 mV 101b = -7% / -70 mV 110b = -8% / -80 mV 111b = -10% / -100mV
2:0	BUCK2_OV_THR	R/W	0b	Powergood high threshold level for BUCK2: (Default from NVM memory) 0b = +3% / +30mV 1b = +3.5% / +35 mV 10b = +4% / +40 mV 11b = +5% / +50 mV 100b = +6% / +60 mV 101b = +7% / +70 mV 110b = +8% / +80 mV 111b = +10% / +100mV

3.1.26 BUCK3_PG_WINDOW Register (Offset = 0x1A) [Reset = 0x0]

BUCK3_PG_WINDOW is shown in Figure 3-26 and described in Table 3-28.

Return to the Table 3-1.

Figure 3-26. BUCK3_PG_WINDOW Register

7	6	5	4	3	2	1	0
RESERVED		BUCK3_UV_THR			BUCK3_OV_THR		
R/W-0b		R/W-0b			R/W-0b		

Table 3-28. BUCK3_PG_WINDOW Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5:3	BUCK3_UV_THR	R/W	0b	Powergood low threshold level for BUCK3: (Default from NVM memory) 0b = -3% / -30mV 1b = -3.5% / -35 mV 10b = -4% / -40 mV 11b = -5% / -50 mV 100b = -6% / -60 mV 101b = -7% / -70 mV 110b = -8% / -80 mV 111b = -10% / -100mV
2:0	BUCK3_OV_THR	R/W	0b	Powergood high threshold level for BUCK3: (Default from NVM memory) 0b = +3% / +30mV 1b = +3.5% / +35 mV 10b = +4% / +40 mV 11b = +5% / +50 mV 100b = +6% / +60 mV 101b = +7% / +70 mV 110b = +8% / +80 mV 111b = +10% / +100mV

3.1.27 BUCK4_PG_WINDOW Register (Offset = 0x1B) [Reset = 0x0]

BUCK4_PG_WINDOW is shown in Figure 3-27 and described in Table 3-29.

Return to the Table 3-1.

Figure 3-27. BUCK4_PG_WINDOW Register

7	6	5	4	3	2	1	0
RESERVED		BUCK4_UV_THR			BUCK4_OV_THR		
R/W-0b		R/W-0b			R/W-0b		

Table 3-29. BUCK4_PG_WINDOW Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5:3	BUCK4_UV_THR	R/W	0b	Powergood low threshold level for BUCK4: (Default from NVM memory) 0b = -3% / -30mV 1b = -3.5% / -35 mV 10b = -4% / -40 mV 11b = -5% / -50 mV 100b = -6% / -60 mV 101b = -7% / -70 mV 110b = -8% / -80 mV 111b = -10% / -100mV
2:0	BUCK4_OV_THR	R/W	0b	Powergood high threshold level for BUCK4: (Default from NVM memory) 0b = +3% / +30mV 1b = +3.5% / +35 mV 10b = +4% / +40 mV 11b = +5% / +50 mV 100b = +6% / +60 mV 101b = +7% / +70 mV 110b = +8% / +80 mV 111b = +10% / +100mV

3.1.28 BUCK5_PG_WINDOW Register (Offset = 0x1C) [Reset = 0x0]

BUCK5_PG_WINDOW is shown in Figure 3-28 and described in Table 3-30.

Return to the Table 3-1.

Figure 3-28. BUCK5_PG_WINDOW Register

7	6	5	4	3	2	1	0
RESERVED		BUCK5_UV_THR			BUCK5_OV_THR		
R/W-0b		R/W-0b			R/W-0b		

Table 3-30. BUCK5_PG_WINDOW Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5:3	BUCK5_UV_THR	R/W	0b	Powergood low threshold level for BUCK5: (Default from NVM memory) 0b = -3% / -30mV 1b = -3.5% / -35 mV 10b = -4% / -40 mV 11b = -5% / -50 mV 100b = -6% / -60 mV 101b = -7% / -70 mV 110b = -8% / -80 mV 111b = -10% / -100mV
2:0	BUCK5_OV_THR	R/W	0b	Powergood high threshold level for BUCK5: (Default from NVM memory) 0b = +3% / +30mV 1b = +3.5% / +35 mV 10b = +4% / +40 mV 11b = +5% / +50 mV 100b = +6% / +60 mV 101b = +7% / +70 mV 110b = +8% / +80 mV 111b = +10% / +100mV

3.1.29 LDO1_CTRL Register (Offset = 0x1D) [Reset = 0x60]

LDO1_CTRL is shown in Figure 3-29 and described in Table 3-31.

Return to the Table 3-1.

Figure 3-29. LDO1_CTRL Register

7	6	5	4	3	2	1	0
LDO1_RV_SEL	LDO1_PLDN		LDO1_VMON_EN	RESERVED		LDO1_SLOW_RAMP	LDO1_EN
R/W-0b	R/W-11b		R/W-0b	R/W-0b		R/W-0b	R/W-0b

Table 3-31. LDO1_CTRL Register Field Descriptions

Bit	Field	Type	Reset	Description
7	LDO1_RV_SEL	R/W	0b	Select residual voltage checking for LDO1 output pin. (Default from NVM memory) 0b = Disabled 1b = Enabled
6:5	LDO1_PLDN	R/W	11b	Enable output pull-down resistor when LDO1 is disabled: (Default from NVM memory) 0b = 50 kOhm 1b = 125 Ohm 10b = 250 Ohm 11b = 500 Ohm
4	LDO1_VMON_EN	R/W	0b	Enable LDO1 OV and UV comparators: (Default from NVM memory) 0b = OV and UV comparators are disabled 1b = OV and UV comparators are enabled.
3:2	RESERVED	R/W	0b	
1	LDO1_SLOW_RAMP	R/W	0b	LDO1 startup slew rate selection 0b = 25mV/us max ramp up slew rate for LDO output from 0.3V to 90% of LDOn_VSET 1b = 3mV/us max ramp up slew rate for LDO output from 0.3V to 90% of LDOn_VSET
0	LDO1_EN	R/W	0b	Enable LDO1 regulator: (Default from NVM memory) 0b = LDO1 regulator is disabled 1b = LDO1 regulator is enabled.

3.1.30 LDO2_CTRL Register (Offset = 0x1E) [Reset = 0x60]

LDO2_CTRL is shown in Figure 3-30 and described in Table 3-32.

Return to the Table 3-1.

Figure 3-30. LDO2_CTRL Register

7	6	5	4	3	2	1	0
LDO2_RV_SEL	LDO2_PLDN		LDO2_VMON_EN	RESERVED		LDO2_SLOW_RAMP	LDO2_EN
R/W-0b	R/W-11b		R/W-0b	R/W-0b		R/W-0b	R/W-0b

Table 3-32. LDO2_CTRL Register Field Descriptions

Bit	Field	Type	Reset	Description
7	LDO2_RV_SEL	R/W	0b	Select residual voltage checking for LDO2 output pin. (Default from NVM memory) 0b = Disabled 1b = Enabled
6:5	LDO2_PLDN	R/W	11b	Enable output pull-down resistor when LDO2 is disabled: (Default from NVM memory) 0b = 50 kOhm 1b = 125 Ohm 10b = 250 Ohm 11b = 500 Ohm
4	LDO2_VMON_EN	R/W	0b	Enable LDO2 OV and UV comparators: (Default from NVM memory) 0b = OV and UV comparators are disabled 1b = OV and UV comparators are enabled.
3:2	RESERVED	R/W	0b	
1	LDO2_SLOW_RAMP	R/W	0b	LDO2 startup slew rate selection 0b = 25mV/us max ramp up slew rate for LDO output from 0.3V to 90% of LDOn_VSET 1b = 3mV/us max ramp up slew rate for LDO output from 0.3V to 90% of LDOn_VSET
0	LDO2_EN	R/W	0b	Enable LDO2 regulator: (Default from NVM memory) 0b = LDO1 regulator is disabled 1b = LDO1 regulator is enabled.

3.1.31 LDO3_CTRL Register (Offset = 0x1F) [Reset = 0x60]

LDO3_CTRL is shown in Figure 3-31 and described in Table 3-33.

Return to the Table 3-1.

Figure 3-31. LDO3_CTRL Register

7	6	5	4	3	2	1	0
LDO3_RV_SEL	LDO3_PLDN		LDO3_VMON_EN	RESERVED		LDO3_SLOW_RAMP	LDO3_EN
R/W-0b	R/W-11b		R/W-0b	R/W-0b		R/W-0b	R/W-0b

Table 3-33. LDO3_CTRL Register Field Descriptions

Bit	Field	Type	Reset	Description
7	LDO3_RV_SEL	R/W	0b	Select residual voltage checking for LDO3 output pin. (Default from NVM memory) 0b = Disabled 1b = Enabled
6:5	LDO3_PLDN	R/W	11b	Enable output pull-down resistor when LDO3 is disabled: (Default from NVM memory) 0b = 50 kOhm 1b = 125 Ohm 10b = 250 Ohm 11b = 500 Ohm
4	LDO3_VMON_EN	R/W	0b	Enable LDO3 OV and UV comparators: (Default from NVM memory) 0b = OV and UV comparators are disabled 1b = OV and UV comparators are enabled.
3:2	RESERVED	R/W	0b	
1	LDO3_SLOW_RAMP	R/W	0b	LDO3 startup slew rate selection 0b = 25mV/us max ramp up slew rate for LDO output from 0.3V to 90% of LDOn_VSET 1b = 3mV/us max ramp up slew rate for LDO output from 0.3V to 90% of LDOn_VSET
0	LDO3_EN	R/W	0b	Enable LDO3 regulator: (Default from NVM memory) 0b = LDO1 regulator is disabled 1b = LDO1 regulator is enabled.

3.1.32 LDO4_CTRL Register (Offset = 0x20) [Reset = 0x60]

LDO4_CTRL is shown in Figure 3-32 and described in Table 3-34.

Return to the Table 3-1.

Figure 3-32. LDO4_CTRL Register

7	6	5	4	3	2	1	0
LDO4_RV_SEL	LDO4_PLDN		LDO4_VMON_EN	RESERVED		LDO4_SLOW_RAMP	LDO4_EN
R/W-0b	R/W-11b		R/W-0b	R/W-0b		R/W-0b	R/W-0b

Table 3-34. LDO4_CTRL Register Field Descriptions

Bit	Field	Type	Reset	Description
7	LDO4_RV_SEL	R/W	0b	Select residual voltage checking for LDO4 output pin. (Default from NVM memory) 0b = Disabled 1b = Enabled
6:5	LDO4_PLDN	R/W	11b	Enable output pull-down resistor when LDO4 is disabled: (Default from NVM memory) 0b = 50 kOhm 1b = 125 Ohm 10b = 250 Ohm 11b = 500 Ohm
4	LDO4_VMON_EN	R/W	0b	Enable LDO4 OV and UV comparators: (Default from NVM memory) 0b = OV and UV comparators are disabled 1b = OV and UV comparators are enabled.
3:2	RESERVED	R/W	0b	
1	LDO4_SLOW_RAMP	R/W	0b	LDO4 startup slew rate selection 0b = 25mV/us max ramp up slew rate for LDO output from 0.3V to 90% of LDOn_VSET 1b = 3mV/us max ramp up slew rate for LDO output from 0.3V to 90% of LDOn_VSET
0	LDO4_EN	R/W	0b	Enable LDO4 regulator: (Default from NVM memory) 0b = LDO1 regulator is disabled 1b = LDO1 regulator is enabled.

3.1.33 LDORTC_CTRL Register (Offset = 0x22) [Reset = 0x0]

LDORTC_CTRL is shown in Figure 3-33 and described in Table 3-35.

Return to the Table 3-1.

Figure 3-33. LDORTC_CTRL Register

7	6	5	4	3	2	1	0
RESERVED							LDORTC_DIS
R/W-0b							R/W-0b

Table 3-35. LDORTC_CTRL Register Field Descriptions

Bit	Field	Type	Reset	Description
7:1	RESERVED	R/W	0b	
0	LDORTC_DIS	R/W	0b	Disable LDORTC regulator: 0b = LDORTC regulator is enabled 1b = LDORTC regulator is disabled

3.1.34 LDO1_VOUT Register (Offset = 0x23) [Reset = 0x0]

LDO1_VOUT is shown in Figure 3-34 and described in Table 3-36.

Return to the Table 3-1.

Figure 3-34. LDO1_VOUT Register

7	6	5	4	3	2	1	0
LDO1_BYPASS	LDO1_VSET						RESERVED
R/W-0b	R/W-0b						R/W-0b

Table 3-36. LDO1_VOUT Register Field Descriptions

Bit	Field	Type	Reset	Description
7	LDO1_BYPASS	R/W	0b	Set LDO1 to bypass mode: (Default from NVM memory) 0b = LDO is set to linear regulator mode. 1b = LDO is set to bypass mode.
6:1	LDO1_VSET	R/W	0b	Voltage selection for LDO regulator. See LDO regulators chapter for voltage levels. (Default from NVM memory)
0	RESERVED	R/W	0b	

3.1.35 LDO2_VOUT Register (Offset = 0x24) [Reset = 0x0]

LDO2_VOUT is shown in Figure 3-35 and described in Table 3-37.

Return to the Table 3-1.

Figure 3-35. LDO2_VOUT Register

7	6	5	4	3	2	1	0
LDO2_BYPASS	LDO2_VSET						RESERVED
R/W-0b	R/W-0b						R/W-0b

Table 3-37. LDO2_VOUT Register Field Descriptions

Bit	Field	Type	Reset	Description
7	LDO2_BYPASS	R/W	0b	Set LDO2 to bypass mode: (Default from NVM memory) 0b = LDO is set to linear regulator mode. 1b = LDO is set to bypass mode.
6:1	LDO2_VSET	R/W	0b	Voltage selection for LDO regulator. See LDO regulators chapter for voltage levels. (Default from NVM memory)
0	RESERVED	R/W	0b	

3.1.36 LDO3_VOUT Register (Offset = 0x25) [Reset = 0x0]

LDO3_VOUT is shown in Figure 3-36 and described in Table 3-38.

Return to the Table 3-1.

Figure 3-36. LDO3_VOUT Register

7	6	5	4	3	2	1	0
LDO3_BYPASS	LDO3_VSET						RESERVED
R/W-0b	R/W-0b						R/W-0b

Table 3-38. LDO3_VOUT Register Field Descriptions

Bit	Field	Type	Reset	Description
7	LDO3_BYPASS	R/W	0b	Set LDO3 to bypass mode: (Default from NVM memory) 0b = LDO is set to linear regulator mode. 1b = LDO is set to bypass mode.
6:1	LDO3_VSET	R/W	0b	Voltage selection for LDO regulator. See LDO regulators chapter for voltage levels. (Default from NVM memory)
0	RESERVED	R/W	0b	

3.1.37 LDO4_VOUT Register (Offset = 0x26) [Reset = 0x0]

LDO4_VOUT is shown in Figure 3-37 and described in Table 3-39.

Return to the Table 3-1.

Figure 3-37. LDO4_VOUT Register

7	6	5	4	3	2	1	0
RESERVED	LDO4_VSET						
R/W-0b	R/W-0b						

Table 3-39. LDO4_VOUT Register Field Descriptions

Bit	Field	Type	Reset	Description
7	RESERVED	R/W	0b	
6:0	LDO4_VSET	R/W	0b	Voltage selection for LDO regulator. See LDO regulators chapter for voltage levels. (Default from NVM memory)

3.1.38 LDO1_PG_WINDOW Register (Offset = 0x27) [Reset = 0x0]

LDO1_PG_WINDOW is shown in Figure 3-38 and described in Table 3-40.

Return to the Table 3-1.

Figure 3-38. LDO1_PG_WINDOW Register

7	6	5	4	3	2	1	0
RESERVED		LDO1_UV_THR			LDO1_OV_THR		
R/W-0b		R/W-0b			R/W-0b		

Table 3-40. LDO1_PG_WINDOW Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5:3	LDO1_UV_THR	R/W	0b	Powergood low threshold level for LDO1: (Default from NVM memory) 0b = -3% / -30mV 1b = -3.5% / -35 mV 10b = -4% / -40 mV 11b = -5% / -50 mV 100b = -6% / -60 mV 101b = -7% / -70 mV 110b = -8% / -80 mV 111b = -10% / -100mV
2:0	LDO1_OV_THR	R/W	0b	Powergood high threshold level for LDO1: (Default from NVM memory) 0b = +3% / +30mV 1b = +3.5% / +35 mV 10b = +4% / +40 mV 11b = +5% / +50 mV 100b = +6% / +60 mV 101b = +7% / +70 mV 110b = +8% / +80 mV 111b = +10% / +100mV

3.1.39 LDO2_PG_WINDOW Register (Offset = 0x28) [Reset = 0x0]

LDO2_PG_WINDOW is shown in Figure 3-39 and described in Table 3-41.

Return to the Table 3-1.

Figure 3-39. LDO2_PG_WINDOW Register

7	6	5	4	3	2	1	0
RESERVED		LDO2_UV_THR			LDO2_OV_THR		
R/W-0b		R/W-0b			R/W-0b		

Table 3-41. LDO2_PG_WINDOW Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5:3	LDO2_UV_THR	R/W	0b	Powergood low threshold level for LDO2: (Default from NVM memory) 0b = -3% / -30mV 1b = -3.5% / -35 mV 10b = -4% / -40 mV 11b = -5% / -50 mV 100b = -6% / -60 mV 101b = -7% / -70 mV 110b = -8% / -80 mV 111b = -10% / -100mV
2:0	LDO2_OV_THR	R/W	0b	Powergood high threshold level for LDO2: (Default from NVM memory) 0b = +3% / +30mV 1b = +3.5% / +35 mV 10b = +4% / +40 mV 11b = +5% / +50 mV 100b = +6% / +60 mV 101b = +7% / +70 mV 110b = +8% / +80 mV 111b = +10% / +100mV

3.1.40 LDO3_PG_WINDOW Register (Offset = 0x29) [Reset = 0x0]

LDO3_PG_WINDOW is shown in Figure 3-40 and described in Table 3-42.

Return to the Table 3-1.

Figure 3-40. LDO3_PG_WINDOW Register

7	6	5	4	3	2	1	0
RESERVED		LDO3_UV_THR			LDO3_OV_THR		
R/W-0b		R/W-0b			R/W-0b		

Table 3-42. LDO3_PG_WINDOW Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5:3	LDO3_UV_THR	R/W	0b	Powergood low threshold level for LDO3: (Default from NVM memory) 0b = -3% / -30mV 1b = -3.5% / -35 mV 10b = -4% / -40 mV 11b = -5% / -50 mV 100b = -6% / -60 mV 101b = -7% / -70 mV 110b = -8% / -80 mV 111b = -10% / -100mV
2:0	LDO3_OV_THR	R/W	0b	Powergood high threshold level for LDO3: Default from NVM memory) 0b = +3% / +30mV 1b = +3.5% / +35 mV 10b = +4% / +40 mV 11b = +5% / +50 mV 100b = +6% / +60 mV 101b = +7% / +70 mV 110b = +8% / +80 mV 111b = +10% / +100mV

3.1.41 LDO4_PG_WINDOW Register (Offset = 0x2A) [Reset = 0x0]

LDO4_PG_WINDOW is shown in Figure 3-41 and described in Table 3-43.

Return to the Table 3-1.

Figure 3-41. LDO4_PG_WINDOW Register

7	6	5	4	3	2	1	0
RESERVED		LDO4_UV_THR			LDO4_OV_THR		
R/W-0b		R/W-0b			R/W-0b		

Table 3-43. LDO4_PG_WINDOW Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5:3	LDO4_UV_THR	R/W	0b	Powergood low threshold level for LDO4: (Default from NVM memory) 0b = -3% / -30mV 1b = -3.5% / -35 mV 10b = -4% / -40 mV 11b = -5% / -50 mV 100b = -6% / -60 mV 101b = -7% / -70 mV 110b = -8% / -80 mV 111b = -10% / -100mV
2:0	LDO4_OV_THR	R/W	0b	Powergood high threshold level for LDO4: (Default from NVM memory) 0b = +3% / +30mV 1b = +3.5% / +35 mV 10b = +4% / +40 mV 11b = +5% / +50 mV 100b = +6% / +60 mV 101b = +7% / +70 mV 110b = +8% / +80 mV 111b = +10% / +100mV

3.1.42 VCCA_VMON_CTRL Register (Offset = 0x2B) [Reset = 0x0]

VCCA_VMON_CTRL is shown in Figure 3-42 and described in Table 3-44.

Return to the Table 3-1.

Figure 3-42. VCCA_VMON_CTRL Register

<table>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="2">RESERVED</td><td>VMON_DEGLITCH_SEL</td><td colspan="4">RESERVED</td><td>VCCA_VMON_EN</td></tr>
<tr><td colspan="2">R/W-0b</td><td>R/W-0b</td><td colspan="4">R/W-0b</td><td>R/W-0b</td></tr>
</table>

Table 3-44. VCCA_VMON_CTRL Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5	VMON_DEGLITCH_SEL	R/W	0b	Deglitch time select for BUCKx_VMON, LDOx_VMON and VCCA_VMON (Default from NVM memory) 0b = 4 us 1b = 20 us
4:1	RESERVED	R/W	0b	
0	VCCA_VMON_EN	R/W	0b	Enable VCCA OV and UV comparators: (Default from NVM memory) 0b = OV and UV comparators are disabled 1b = OV and UV comparators are enabled.

3.1.43 VCCA_PG_WINDOW Register (Offset = 0x2C) [Reset = 0x40]

VCCA_PG_WINDOW is shown in Figure 3-43 and described in Table 3-45.

Return to the Table 3-1.

Figure 3-43. VCCA_PG_WINDOW Register

7	6	5	4	3	2	1	0
RESERVED	VCCA_PG_SET	VCCA_UV_THR			VCCA_OV_THR		
R/W-0b	R/W-1b	R/W-0b			R/W-0b		

Table 3-45. VCCA_PG_WINDOW Register Field Descriptions

Bit	Field	Type	Reset	Description
7	RESERVED	R/W	0b	
6	VCCA_PG_SET	R/W	1b	Powergood level for VCCA pin: (Default from NVM memory) 0b = 3.3 V 1b = 5.0 V
5:3	VCCA_UV_THR	R/W	0b	Powergood low threshold level for VCCA pin: (Default from NVM memory) 0b = -3% 1b = -3.5% 10b = -4% 11b = -5% 100b = -6% 101b = -7% 110b = -8% 111b = -10%
2:0	VCCA_OV_THR	R/W	0b	Powergood high threshold level for VCCA pin: (Default from NVM memory) 0b = +3% 1b = +3.5% 10b = +4% 11b = +5% 100b = +6% 101b = +7% 110b = +8% 111b = +10%

3.1.44 GPIO1_CONF Register (Offset = 0x31) [Reset = 0xA]

GPIO1_CONF is shown in Figure 3-44 and described in Table 3-46.

Return to the Table 3-1.

Figure 3-44. GPIO1_CONF Register

7	6	5	4	3	2	1	0
GPIO1_SEL			GPIO1_DEGLITCH_EN	GPIO1_PU_PD_EN	GPIO1_PU_SEL	GPIO1_OD	GPIO1_DIR
R/W-0b			R/W-0b	R/W-1b	R/W-0b	R/W-1b	R/W-0b

Table 3-46. GPIO1_CONF Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	GPIO1_SEL	R/W	0b	GPIO1 signal function: (Default from NVM memory) 0b = GPIO1 1b = TPS6594: SCL_I2C2/CS_SPI, TPS6593: CS_SPI 10b = NRSTOUT_SOC 11b = NRSTOUT_SOC 100b = NSLEEP1 101b = NSLEEP2 110b = WKUP1 111b = WKUP2
4	GPIO1_DEGLITCH_EN	R/W	0b	GPIO1 signal deglitch time when signal direction is input: (Default from NVM memory) 0b = No deglitch, only synchronization. 1b = 8 us deglitch time.
3	GPIO1_PU_PD_EN	R/W	1b	Control for GPIO1 pin pull-up/pull-down resistor: (Default from NVM memory) 0b = Pull-up/pull-down resistor disabled 1b = Pull-up/pull-down resistor enabled
2	GPIO1_PU_SEL	R/W	0b	Control for GPIO1 pin pull-up/pull-down resistor: GPIO1_PU_PD_EN must be 1 to select the resistor. (Default from NVM memory) 0b = Pull-down resistor selected 1b = Pull-up resistor selected
1	GPIO1_OD	R/W	1b	GPIO1 signal type when configured to output: (Default from NVM memory) 0b = Push-pull output 1b = Open-drain output
0	GPIO1_DIR	R/W	0b	GPIO1 signal direction: (Default from NVM memory) 0b = Input 1b = Output

3.1.45 GPIO2_CONF Register (Offset = 0x32) [Reset = 0xA]

GPIO2_CONF is shown in Figure 3-45 and described in Table 3-47.

Return to the Table 3-1.

Figure 3-45. GPIO2_CONF Register

7	6	5	4	3	2	1	0
GPIO2_SEL			GPIO2_DEGLITCH_EN	GPIO2_PU_PD_EN	GPIO2_PU_SEL	GPIO2_OD	GPIO2_DIR
R/W-0b			R/W-0b	R/W-1b	R/W-0b	R/W-1b	R/W-0b

Table 3-47. GPIO2_CONF Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	GPIO2_SEL	R/W	0b	GPIO2 signal function: (Default from NVM memory) 0b = GPIO2 1b = TRIG_WDOG 10b = TPS6594: SDA_I2C2/SDO_SPI, TPS6593: SDO_SPI 11b = TPS6594: SDA_I2C2/SDO_SPI, TPS6593: SDO_SPI 100b = NSLEEP1 101b = NSLEEP2 110b = WKUP1 111b = WKUP2
4	GPIO2_DEGLITCH_EN	R/W	0b	GPIO2 signal deglitch time when signal direction is input: (Default from NVM memory) 0b = No deglitch, only synchronization. 1b = 8 us deglitch time.
3	GPIO2_PU_PD_EN	R/W	1b	Control for GPIO2 pin pull-up/pull-down resistor: (Default from NVM memory) 0b = Pull-up/pull-down resistor disabled 1b = Pull-up/pull-down resistor enabled
2	GPIO2_PU_SEL	R/W	0b	Control for GPIO2 pin pull-up/pull-down resistor: GPIO2_PU_PD_EN must be 1 to select the resistor. (Default from NVM memory) 0b = Pull-down resistor selected 1b = Pull-up resistor selected
1	GPIO2_OD	R/W	1b	GPIO2 signal type when configured to output: (Default from NVM memory) 0b = Push-pull output 1b = Open-drain output
0	GPIO2_DIR	R/W	0b	GPIO2 signal direction: (Default from NVM memory) 0b = Input 1b = Output

3.1.46 GPIO3_CONF Register (Offset = 0x33) [Reset = 0xA]

GPIO3_CONF is shown in Figure 3-46 and described in Table 3-48.

Return to the Table 3-1.

Figure 3-46. GPIO3_CONF Register

7	6	5	4	3	2	1	0
GPIO3_SEL			GPIO3_DEGLITCH_EN	GPIO3_PU_PD_EN	GPIO3_PU_SEL	GPIO3_OD	GPIO3_DIR
R/W-0b			R/W-0b	R/W-1b	R/W-0b	R/W-1b	R/W-0b

Table 3-48. GPIO3_CONF Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	GPIO3_SEL	R/W	0b	GPIO3 signal function: (Default from NVM memory) 0b = GPIO3 1b = CLK32KOUT 10b = TPS6594: NERR_SOC, TPS6593: reserved 11b = TPS6594: NERR_SOC, TPS6593: reserved 100b = NSLEEP1 101b = NSLEEP2 110b = LP_WKUP1 111b = LP_WKUP2
4	GPIO3_DEGLITCH_EN	R/W	0b	GPIO3 signal deglitch time when signal direction is input: (Default from NVM memory) 0b = No deglitch, only synchronization. 1b = 8 us deglitch time.
3	GPIO3_PU_PD_EN	R/W	1b	Control for GPIO3 pin pull-up/pull-down resistor: (Default from NVM memory) 0b = Pull-up/pull-down resistor disabled 1b = Pull-up/pull-down resistor enabled
2	GPIO3_PU_SEL	R/W	0b	Control for GPIO3 pin pull-up/pull-down resistor: GPIO3_PU_PD_EN must be 1 to select the resistor. (Default from NVM memory) 0b = Pull-down resistor selected 1b = Pull-up resistor selected
1	GPIO3_OD	R/W	1b	GPIO3 signal type when configured to output: (Default from NVM memory) 0b = Push-pull output 1b = Open-drain output
0	GPIO3_DIR	R/W	0b	GPIO3 signal direction: (Default from NVM memory) 0b = Input 1b = Output

3.1.47 GPIO4_CONF Register (Offset = 0x34) [Reset = 0xA]

GPIO4_CONF is shown in Figure 3-47 and described in Table 3-49.

Return to the Table 3-1.

Figure 3-47. GPIO4_CONF Register

7	6	5	4	3	2	1	0
GPIO4_SEL			GPIO4_DEGLITCH_EN	GPIO4_PU_PD_EN	GPIO4_PU_SEL	GPIO4_OD	GPIO4_DIR
R/W-0b			R/W-0b	R/W-1b	R/W-0b	R/W-1b	R/W-0b

Table 3-49. GPIO4_CONF Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	GPIO4_SEL	R/W	0b	GPIO4 signal function: (Default from NVM memory) 0b = GPIO4 1b = CLK32KOUT 10b = CLK32KOUT 11b = CLK32KOUT 100b = NSLEEP1 101b = NSLEEP2 110b = LP_WKUP1 111b = LP_WKUP2
4	GPIO4_DEGLITCH_EN	R/W	0b	GPIO4 signal deglitch time when signal direction is input: (Default from NVM memory) 0b = No deglitch, only synchronization. 1b = 8 us deglitch time.
3	GPIO4_PU_PD_EN	R/W	1b	Control for GPIO4 pin pull-up/pull-down resistor: (Default from NVM memory) 0b = Pull-up/pull-down resistor disabled 1b = Pull-up/pull-down resistor enabled
2	GPIO4_PU_SEL	R/W	0b	Control for GPIO4 pin pull-up/pull-down resistor: GPIO4_PU_PD_EN must be 1 to select the resistor. (Default from NVM memory) 0b = Pull-down resistor selected 1b = Pull-up resistor selected
1	GPIO4_OD	R/W	1b	GPIO4 signal type when configured to output: (Default from NVM memory) 0b = Push-pull output 1b = Open-drain output
0	GPIO4_DIR	R/W	0b	GPIO4 signal direction: (Default from NVM memory) 0b = Input 1b = Output

3.1.48 GPIO5_CONF Register (Offset = 0x35) [Reset = 0xA]

GPIO5_CONF is shown in Figure 3-48 and described in Table 3-50.

Return to the Table 3-1.

Figure 3-48. GPIO5_CONF Register

7	6	5	4	3	2	1	0
GPIO5_SEL			GPIO5_DEGLITCH_EN	GPIO5_PU_PD_EN	GPIO5_PU_SEL	GPIO5_OD	GPIO5_DIR
R/W-0b			R/W-0b	R/W-1b	R/W-0b	R/W-1b	R/W-0b

Table 3-50. GPIO5_CONF Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	GPIO5_SEL	R/W	0b	GPIO5 signal function: (Default from NVM memory) 0b = GPIO5 1b = SCLK_SPMI 10b = SCLK_SPMI 11b = SCLK_SPMI 100b = NSLEEP1 101b = NSLEEP2 110b = WKUP1 111b = WKUP2
4	GPIO5_DEGLITCH_EN	R/W	0b	GPIO5 signal deglitch time when signal direction is input: (Default from NVM memory) 0b = No deglitch, only synchronization. 1b = 8 us deglitch time.
3	GPIO5_PU_PD_EN	R/W	1b	Control for GPIO5 pin pull-up/pull-down resistor: (Default from NVM memory) 0b = Pull-up/pull-down resistor disabled 1b = Pull-up/pull-down resistor enabled
2	GPIO5_PU_SEL	R/W	0b	Control for GPIO5 pin pull-up/pull-down resistor: GPIO5_PU_PD_EN must be 1 to select the resistor. (Default from NVM memory) 0b = Pull-down resistor selected 1b = Pull-up resistor selected
1	GPIO5_OD	R/W	1b	GPIO5 signal type when configured to output: (Default from NVM memory) 0b = Push-pull output 1b = Open-drain output
0	GPIO5_DIR	R/W	0b	GPIO5 signal direction: (Default from NVM memory) 0b = Input 1b = Output

3.1.49 GPIO6_CONF Register (Offset = 0x36) [Reset = 0xA]

GPIO6_CONF is shown in Figure 3-49 and described in Table 3-51.

Return to the Table 3-1.

Figure 3-49. GPIO6_CONF Register

7	6	5	4	3	2	1	0
GPIO6_SEL			GPIO6_DEGLITCH_EN	GPIO6_PU_PD_EN	GPIO6_PU_SEL	GPIO6_OD	GPIO6_DIR
R/W-0b			R/W-0b	R/W-1b	R/W-0b	R/W-1b	R/W-0b

Table 3-51. GPIO6_CONF Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	GPIO6_SEL	R/W	0b	GPIO6 signal function: (Default from NVM memory) 0b = GPIO6 1b = SDATA_SPMI 10b = SDATA_SPMI 11b = SDATA_SPMI 100b = NSLEEP1 101b = NSLEEP2 110b = WKUP1 111b = WKUP2
4	GPIO6_DEGLITCH_EN	R/W	0b	GPIO6 signal deglitch time when signal direction is input: (Default from NVM memory) 0b = No deglitch, only synchronization. 1b = 8 us deglitch time.
3	GPIO6_PU_PD_EN	R/W	1b	Control for GPIO6 pin pull-up/pull-down resistor: (Default from NVM memory) 0b = Pull-up/pull-down resistor disabled 1b = Pull-up/pull-down resistor enabled
2	GPIO6_PU_SEL	R/W	0b	Control for GPIO6 pin pull-up/pull-down resistor: GPIO6_PU_PD_EN must be 1 to select the resistor. (Default from NVM memory) 0b = Pull-down resistor selected 1b = Pull-up resistor selected
1	GPIO6_OD	R/W	1b	GPIO6 signal type when configured to output: (Default from NVM memory) 0b = Push-pull output 1b = Open-drain output
0	GPIO6_DIR	R/W	0b	GPIO6 signal direction: (Default from NVM memory) 0b = Input 1b = Output

3.1.50 GPIO7_CONF Register (Offset = 0x37) [Reset = 0xA]

GPIO7_CONF is shown in Figure 3-50 and described in Table 3-52.

Return to the Table 3-1.

Figure 3-50. GPIO7_CONF Register

7	6	5	4	3	2	1	0
GPIO7_SEL			GPIO7_DEGLITCH_EN	GPIO7_PU_PD_EN	GPIO7_PU_SEL	GPIO7_OD	GPIO7_DIR
R/W-0b			R/W-0b	R/W-1b	R/W-0b	R/W-1b	R/W-0b

Table 3-52. GPIO7_CONF Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	GPIO7_SEL	R/W	0b	GPIO7 signal function: (Default from NVM memory) 0b = GPIO7 1b = TPS6594: NERR_MCU, TPS6593: reserved 10b = TPS6594: NERR_MCU, TPS6593: reserved 11b = TPS6594: NERR_MCU, TPS6593: reserved 100b = NSLEEP1 101b = NSLEEP2 110b = WKUP1 111b = WKUP2
4	GPIO7_DEGLITCH_EN	R/W	0b	GPIO7 signal deglitch time when signal direction is input: (Default from NVM memory) 0b = No deglitch, only synchronization. 1b = 8 us deglitch time.
3	GPIO7_PU_PD_EN	R/W	1b	Control for GPIO7 pin pull-up/pull-down resistor: (Default from NVM memory) 0b = Pull-up/pull-down resistor disabled 1b = Pull-up/pull-down resistor enabled
2	GPIO7_PU_SEL	R/W	0b	Control for GPIO7 pin pull-up/pull-down resistor: GPIO7_PU_PD_EN must be 1 to select the resistor. (Default from NVM memory) 0b = Pull-down resistor selected 1b = Pull-up resistor selected
1	GPIO7_OD	R/W	1b	GPIO7 signal type when configured to output: (Default from NVM memory) 0b = Push-pull output 1b = Open-drain output
0	GPIO7_DIR	R/W	0b	GPIO7 signal direction: (Default from NVM memory) 0b = Input 1b = Output

3.1.51 GPIO8_CONF Register (Offset = 0x38) [Reset = 0xA]

GPIO8_CONF is shown in Figure 3-51 and described in Table 3-53.

Return to the Table 3-1.

Figure 3-51. GPIO8_CONF Register

7	6	5	4	3	2	1	0
GPIO8_SEL			GPIO8_DEGLITCH_EN	GPIO8_PU_PD_EN	GPIO8_PU_SEL	GPIO8_OD	GPIO8_DIR
R/W-0b			R/W-0b	R/W-1b	R/W-0b	R/W-1b	R/W-0b

Table 3-53. GPIO8_CONF Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	GPIO8_SEL	R/W	0b	GPIO8 signal function: (Default from NVM memory) 0b = GPIO8 1b = CLK32KOUT 10b = SYNCCLKOUT 11b = TPS6594: DISABLE_WDOG, TPS6593: reserved 100b = NSLEEP1 101b = NSLEEP2 110b = WKUP1 111b = WKUP2
4	GPIO8_DEGLITCH_EN	R/W	0b	GPIO8 signal deglitch time when signal direction is input: (Default from NVM memory) 0b = No deglitch, only synchronization. 1b = 8 us deglitch time.
3	GPIO8_PU_PD_EN	R/W	1b	Control for GPIO8 pin pull-up/pull-down resistor: (Default from NVM memory) 0b = Pull-up/pull-down resistor disabled 1b = Pull-up/pull-down resistor enabled
2	GPIO8_PU_SEL	R/W	0b	Control for GPIO8 pin pull-up/pull-down resistor: GPIO8_PU_PD_EN must be 1 to select the resistor. (Default from NVM memory) 0b = Pull-down resistor selected 1b = Pull-up resistor selected
1	GPIO8_OD	R/W	1b	GPIO8 signal type when configured to output: (Default from NVM memory) 0b = Push-pull output 1b = Open-drain output
0	GPIO8_DIR	R/W	0b	GPIO8 signal direction: (Default from NVM memory) 0b = Input 1b = Output

3.1.52 GPIO9_CONF Register (Offset = 0x39) [Reset = 0xA]

GPIO9_CONF is shown in Figure 3-52 and described in Table 3-54.

Return to the Table 3-1.

Figure 3-52. GPIO9_CONF Register

7	6	5	4	3	2	1	0
GPIO9_SEL			GPIO9_DEGLITCH_EN	GPIO9_PU_PD_EN	GPIO9_PU_SEL	GPIO9_OD	GPIO9_DIR
R/W-0b			R/W-0b	R/W-1b	R/W-0b	R/W-1b	R/W-0b

Table 3-54. GPIO9_CONF Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	GPIO9_SEL	R/W	0b	GPIO9 signal function: (Default from NVM memory) 0b = GPIO9 1b = PGOOD 10b = TPS6594: DISABLE_WDOG, TPS6593: reserved 11b = SYNCCLKOUT 100b = NSLEEP1 101b = NSLEEP2 110b = WKUP1 111b = WKUP2
4	GPIO9_DEGLITCH_EN	R/W	0b	GPIO9 signal deglitch time when signal direction is input: (Default from NVM memory) 0b = No deglitch, only synchronization. 1b = 8 us deglitch time.
3	GPIO9_PU_PD_EN	R/W	1b	Control for GPIO9 pin pull-up/pull-down resistor: (Default from NVM memory) 0b = Pull-up/pull-down resistor disabled 1b = Pull-up/pull-down resistor enabled
2	GPIO9_PU_SEL	R/W	0b	Control for GPIO9 pin pull-up/pull-down resistor: GPIO9_PU_PD_EN must be 1 to select the resistor. (Default from NVM memory) 0b = Pull-down resistor selected 1b = Pull-up resistor selected
1	GPIO9_OD	R/W	1b	GPIO9 signal type when configured to output: (Default from NVM memory) 0b = Push-pull output 1b = Open-drain output
0	GPIO9_DIR	R/W	0b	GPIO9 signal direction: (Default from NVM memory) 0b = Input 1b = Output

3.1.53 GPIO10_CONF Register (Offset = 0x3A) [Reset = 0xA]

GPIO10_CONF is shown in Figure 3-53 and described in Table 3-55.

Return to the Table 3-1.

Figure 3-53. GPIO10_CONF Register

7	6	5	4	3	2	1	0
GPIO10_SEL			GPIO10_DEGLITCH_EN	GPIO10_PU_PD_EN	GPIO10_PU_SEL	GPIO10_OD	GPIO10_DIR
R/W-0b			R/W-0b	R/W-1b	R/W-0b	R/W-1b	R/W-0b

Table 3-55. GPIO10_CONF Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	GPIO10_SEL	R/W	0b	GPIO10 signal function: (Default from NVM memory) 0b = GPIO10 1b = SYNCCLKIN 10b = SYNCCLKOUT 11b = CLK32KOUT 100b = NSLEEP1 101b = NSLEEP2 110b = WKUP1 111b = WKUP2
4	GPIO10_DEGLITCH_EN	R/W	0b	GPIO10 signal deglitch time when signal direction is input: (Default from NVM memory) 0b = No deglitch, only synchronization. 1b = 8 us deglitch time.
3	GPIO10_PU_PD_EN	R/W	1b	Control for GPIO10 pin pull-up/pull-down resistor: (Default from NVM memory) 0b = Pull-up/pull-down resistor disabled 1b = Pull-up/pull-down resistor enabled
2	GPIO10_PU_SEL	R/W	0b	Control for GPIO10 pin pull-up/pull-down resistor: GPIO10_PU_PD_EN must be 1 to select the resistor. (Default from NVM memory) 0b = Pull-down resistor selected 1b = Pull-up resistor selected
1	GPIO10_OD	R/W	1b	GPIO10 signal type when configured to output: (Default from NVM memory) 0b = Push-pull output 1b = Open-drain output
0	GPIO10_DIR	R/W	0b	GPIO10 signal direction: (Default from NVM memory) 0b = Input 1b = Output

3.1.54 GPIO11_CONF Register (Offset = 0x3B) [Reset = 0xA]

GPIO11_CONF is shown in Figure 3-54 and described in Table 3-56.

Return to the Table 3-1.

Figure 3-54. GPIO11_CONF Register

7	6	5	4	3	2	1	0
GPIO11_SEL			GPIO11_DEGLITCH_EN	GPIO11_PU_PD_EN	GPIO11_PU_SEL	GPIO11_OD	GPIO11_DIR
R/W-0b			R/W-0b	R/W-1b	R/W-0b	R/W-1b	R/W-0b

Table 3-56. GPIO11_CONF Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	GPIO11_SEL	R/W	0b	GPIO11 signal function: (Default from NVM memory) 0b = GPIO11 1b = TRIG_WDOG 10b = NRSTOUT_SOC 11b = NRSTOUT_SOC 100b = NSLEEP1 101b = NSLEEP2 110b = WKUP1 111b = WKUP2
4	GPIO11_DEGLITCH_EN	R/W	0b	GPIO11 signal deglitch time when signal direction is input: (Default from NVM memory) 0b = No deglitch, only synchronization. 1b = 8 us deglitch time.
3	GPIO11_PU_PD_EN	R/W	1b	Control for GPIO11 pin pull-up/pull-down resistor: (Default from NVM memory) 0b = Pull-up/pull-down resistor disabled 1b = Pull-up/pull-down resistor enabled
2	GPIO11_PU_SEL	R/W	0b	Control for GPIO11 pin pull-up/pull-down resistor: GPIO11_PU_PD_EN must be 1 to select the resistor. (Default from NVM memory) 0b = Pull-down resistor selected 1b = Pull-up resistor selected
1	GPIO11_OD	R/W	1b	GPIO11 signal type when configured to output: (Default from NVM memory) 0b = Push-pull output 1b = Open-drain output
0	GPIO11_DIR	R/W	0b	GPIO11 signal direction: (Default from NVM memory) 0b = Input 1b = Output

3.1.55 NPWRON_CONF Register (Offset = 0x3C) [Reset = 0x88]

NPWRON_CONF is shown in Figure 3-55 and described in Table 3-57.

Return to the Table 3-1.

Figure 3-55. NPWRON_CONF Register

7	6	5	4	3	2	1	0
NPWRON_SEL		ENABLE_POL	ENABLE_DEGLITCH_EN	ENABLE_PU_PD_EN	ENABLE_PU_SEL	RESERVED	NRSTOUT_OD
R/W-10b		R/W-0b	R/W-0b	R/W-1b	R/W-0b	R/W-0b	R/W-0b

Table 3-57. NPWRON_CONF Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	NPWRON_SEL	R/W	10b	NPWRON/ENABLE signal function: (Default from NVM memory) 0b = ENABLE 1b = NPWRON 10b = None 11b = None
5	ENABLE_POL	R/W	0b	Control for ENABLE pin polarity: (Default from NVM memory) 0b = Active high 1b = Active low
4	ENABLE_DEGLITCH_EN	R/W	0b	NPWRON/ENABLE signal deglitch time: (Default from NVM memory) 0b = No deglitch, only synchronization. 1b = 8 us deglitch time when ENABLE, 50 ms deglitch time when NPWRON.
3	ENABLE_PU_PD_EN	R/W	1b	Control for NPWRON/ENABLE pin pull-up resistor: (Default from NVM memory) 0b = Pull-up/pull-down resistor disabled 1b = Pull-up/pull-down resistor enabled
2	ENABLE_PU_SEL	R/W	0b	Control for NPWRON/ENABLE pin pull-down resistor: ENABLE_PU_PD_EN must be 1 to select the resistor. (Default from NVM memory) 0b = Pull-down resistor selected 1b = Pull-up resistor selected
1	RESERVED	R/W	0b	
0	NRSTOUT_OD	R/W	0b	NRSTOUT signal type: (Default from NVM memory) 0b = Push-pull output 1b = Open-drain output

3.1.56 GPIO_OUT_1 Register (Offset = 0x3D) [Reset = 0x0]

GPIO_OUT_1 is shown in Figure 3-56 and described in Table 3-58.

Return to the Table 3-1.

Figure 3-56. GPIO_OUT_1 Register

7	6	5	4	3	2	1	0
GPIO8_OUT	GPIO7_OUT	GPIO6_OUT	GPIO5_OUT	GPIO4_OUT	GPIO3_OUT	GPIO2_OUT	GPIO1_OUT
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-58. GPIO_OUT_1 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	GPIO8_OUT	R/W	0b	Control for GPIO8 signal when configured to GPIO Output: (Default from NVM memory) 0b = Low 1b = High
6	GPIO7_OUT	R/W	0b	Control for GPIO7 signal when configured to GPIO Output: (Default from NVM memory) 0b = Low 1b = High
5	GPIO6_OUT	R/W	0b	Control for GPIO6 signal when configured to GPIO Output: (Default from NVM memory) 0b = Low 1b = High
4	GPIO5_OUT	R/W	0b	Control for GPIO5 signal when configured to GPIO Output: (Default from NVM memory) 0b = Low 1b = High
3	GPIO4_OUT	R/W	0b	Control for GPIO4 signal when configured to GPIO Output: (Default from NVM memory) 0b = Low 1b = High
2	GPIO3_OUT	R/W	0b	Control for GPIO3 signal when configured to GPIO Output: (Default from NVM memory) 0b = Low 1b = High
1	GPIO2_OUT	R/W	0b	Control for GPIO2 signal when configured to GPIO Output: (Default from NVM memory) 0b = Low 1b = High
0	GPIO1_OUT	R/W	0b	Control for GPIO1 signal when configured to GPIO Output: (Default from NVM memory) 0b = Low 1b = High

3.1.57 GPIO_OUT_2 Register (Offset = 0x3E) [Reset = 0x0]

GPIO_OUT_2 is shown in Figure 3-57 and described in Table 3-59.

Return to the Table 3-1.

Figure 3-57. GPIO_OUT_2 Register

7	6	5	4	3	2	1	0
RESERVED					GPIO11_OUT	GPIO10_OUT	GPIO9_OUT
R/W-0b					R/W-0b	R/W-0b	R/W-0b

Table 3-59. GPIO_OUT_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:3	RESERVED	R/W	0b	
2	GPIO11_OUT	R/W	0b	Control for GPIO11 signal when configured to GPIO Output: (Default from NVM memory) 0b = Low 1b = High
1	GPIO10_OUT	R/W	0b	Control for GPIO10 signal when configured to GPIO Output: (Default from NVM memory) 0b = Low 1b = High
0	GPIO9_OUT	R/W	0b	Control for GPIO9 signal when configured to GPIO Output: (Default from NVM memory) 0b = Low 1b = High

3.1.58 GPIO_IN_1 Register (Offset = 0x3F) [Reset = 0x0]

GPIO_IN_1 is shown in Figure 3-58 and described in Table 3-60.

Return to the Table 3-1.

Figure 3-58. GPIO_IN_1 Register

7	6	5	4	3	2	1	0
GPIO8_IN	GPIO7_IN	GPIO6_IN	GPIO5_IN	GPIO4_IN	GPIO3_IN	GPIO2_IN	GPIO1_IN
R-0b	R-0b	R-0b	R-0b	R-0b	R-0b	R-0b	R-0b

Table 3-60. GPIO_IN_1 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	GPIO8_IN	R	0b	Level of GPIO8 signal: 0b = Low 1b = High
6	GPIO7_IN	R	0b	Level of GPIO7 signal: 0b = Low 1b = High
5	GPIO6_IN	R	0b	Level of GPIO6 signal: 0b = Low 1b = High
4	GPIO5_IN	R	0b	Level of GPIO5 signal: 0b = Low 1b = High
3	GPIO4_IN	R	0b	Level of GPIO4 signal: 0b = Low 1b = High
2	GPIO3_IN	R	0b	Level of GPIO3 signal: 0b = Low 1b = High
1	GPIO2_IN	R	0b	Level of GPIO2 signal: 0b = Low 1b = High
0	GPIO1_IN	R	0b	Level of GPIO1 signal: 0b = Low 1b = High

3.1.59 GPIO_IN_2 Register (Offset = 0x40) [Reset = 0x0]

GPIO_IN_2 is shown in Figure 3-59 and described in Table 3-61.

Return to the Table 3-1.

Figure 3-59. GPIO_IN_2 Register

7	6	5	4	3	2	1	0
RESERVED				NPWRON_IN	GPIO11_IN	GPIO10_IN	GPIO9_IN
R-0b				R-0b	R-0b	R-0b	R-0b

Table 3-61. GPIO_IN_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	RESERVED	R	0b	
3	NPWRON_IN	R	0b	Level of NPWRON/ENABLE signal: 0b = Low 1b = High
2	GPIO11_IN	R	0b	Level of GPIO11 signal: 0b = Low 1b = High
1	GPIO10_IN	R	0b	Level of GPIO10 signal: 0b = Low 1b = High
0	GPIO9_IN	R	0b	Level of GPIO9 signal: 0b = Low 1b = High

3.1.60 RAIL_SEL_1 Register (Offset = 0x41) [Reset = 0x0]

RAIL_SEL_1 is shown in Figure 3-60 and described in Table 3-62.

Return to the Table 3-1.

Figure 3-60. RAIL_SEL_1 Register

7	6	5	4	3	2	1	0
BUCK4_GRP_SEL		BUCK3_GRP_SEL		BUCK2_GRP_SEL		BUCK1_GRP_SEL	
R/W-0b		R/W-0b		R/W-0b		R/W-0b	

Table 3-62. RAIL_SEL_1 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	BUCK4_GRP_SEL	R/W	0b	Rail group selection for BUCK4: (Default from NVM memory) 0b = No group assigned 1b = MCU rail group 10b = SOC rail group 11b = OTHER rail group
5:4	BUCK3_GRP_SEL	R/W	0b	Rail group selection for BUCK3: (Default from NVM memory) 0b = No group assigned 1b = MCU rail group 10b = SOC rail group 11b = OTHER rail group
3:2	BUCK2_GRP_SEL	R/W	0b	Rail group selection for BUCK2: (Default from NVM memory) 0b = No group assigned 1b = MCU rail group 10b = SOC rail group 11b = OTHER rail group
1:0	BUCK1_GRP_SEL	R/W	0b	Rail group selection for BUCK1: (Default from NVM memory) 0b = No group assigned 1b = MCU rail group 10b = SOC rail group 11b = OTHER rail group

3.1.61 RAIL_SEL_2 Register (Offset = 0x42) [Reset = 0x0]

RAIL_SEL_2 is shown in Figure 3-61 and described in Table 3-63.

Return to the Table 3-1.

Figure 3-61. RAIL_SEL_2 Register

7	6	5	4	3	2	1	0
LDO3_GRP_SEL		LDO2_GRP_SEL		LDO1_GRP_SEL		BUCK5_GRP_SEL	
R/W-0b		R/W-0b		R/W-0b		R/W-0b	

Table 3-63. RAIL_SEL_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	LDO3_GRP_SEL	R/W	0b	Rail group selection for LDO3: (Default from NVM memory) 0b = No group assigned 1b = MCU rail group 10b = SOC rail group 11b = OTHER rail group
5:4	LDO2_GRP_SEL	R/W	0b	Rail group selection for LDO2: (Default from NVM memory) 0b = No group assigned 1b = MCU rail group 10b = SOC rail group 11b = OTHER rail group
3:2	LDO1_GRP_SEL	R/W	0b	Rail group selection for LDO1: (Default from NVM memory) 0b = No group assigned 1b = MCU rail group 10b = SOC rail group 11b = OTHER rail group
1:0	BUCK5_GRP_SEL	R/W	0b	Rail group selection for BUCK5: (Default from NVM memory) 0b = No group assigned 1b = MCU rail group 10b = SOC rail group 11b = OTHER rail group

3.1.62 RAIL_SEL_3 Register (Offset = 0x43) [Reset = 0x0]

RAIL_SEL_3 is shown in Figure 3-62 and described in Table 3-64.

Return to the Table 3-1.

Figure 3-62. RAIL_SEL_3 Register

7	6	5	4	3	2	1	0
RESERVED				VCCA_GRP_SEL		LDO4_GRP_SEL	
R/W-0b				R/W-0b		R/W-0b	

Table 3-64. RAIL_SEL_3 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	RESERVED	R/W	0b	
3:2	VCCA_GRP_SEL	R/W	0b	Rail group selection for VCCA monitoring: (Default from NVM memory) 0b = No group assigned 1b = MCU rail group 10b = SOC rail group 11b = OTHER rail group
1:0	LDO4_GRP_SEL	R/W	0b	Rail group selection for LDO4: (Default from NVM memory) 0b = No group assigned 1b = MCU rail group 10b = SOC rail group 11b = OTHER rail group

3.1.63 FSM_TRIG_SEL_1 Register (Offset = 0x44) [Reset = 0x0]

FSM_TRIG_SEL_1 is shown in Figure 3-63 and described in Table 3-65.

Return to the Table 3-1.

Figure 3-63. FSM_TRIG_SEL_1 Register

7	6	5	4	3	2	1	0
SEVERE_ERR_TRIG		OTHER_RAIL_TRIG		SOC_RAIL_TRIG		MCU_RAIL_TRIG	
R/W-0b		R/W-0b		R/W-0b		R/W-0b	

Table 3-65. FSM_TRIG_SEL_1 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	SEVERE_ERR_TRIG	R/W	0b	Trigger selection for Severe Error: (Default from NVM memory) 0b = Immediate shutdown 1b = Orderly shutdown 10b = MCU power error 11b = SOC power error
5:4	OTHER_RAIL_TRIG	R/W	0b	Trigger selection for OTHER rail group: (Default from NVM memory) 0b = Immediate shutdown 1b = Orderly shutdown 10b = MCU power error 11b = SOC power error
3:2	SOC_RAIL_TRIG	R/W	0b	Trigger selection for SOC rail group: (Default from NVM memory) 0b = Immediate shutdown 1b = Orderly shutdown 10b = MCU power error 11b = SOC power error
1:0	MCU_RAIL_TRIG	R/W	0b	Trigger selection for MCU rail group: (Default from NVM memory) 0b = Immediate shutdown 1b = Orderly shutdown 10b = MCU power error 11b = SOC power error

3.1.64 FSM_TRIG_SEL_2 Register (Offset = 0x45) [Reset = 0x0]

FSM_TRIG_SEL_2 is shown in Figure 3-64 and described in Table 3-66.

Return to the Table 3-1.

Figure 3-64. FSM_TRIG_SEL_2 Register

7	6	5	4	3	2	1	0
RESERVED						MODERATE_ERR_TRIG	
R/W-0b						R/W-0b	

Table 3-66. FSM_TRIG_SEL_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:2	RESERVED	R/W	0b	
1:0	MODERATE_ERR_TRIG	R/W	0b	Trigger selection for Moderate Error: (Default from NVM memory) 0b = Immediate shutdown 1b = Orderly shutdown 10b = MCU power error 11b = SOC power error

3.1.65 FSM_TRIG_MASK_1 Register (Offset = 0x46) [Reset = 0x0]

FSM_TRIG_MASK_1 is shown in Figure 3-65 and described in Table 3-67.

Return to the Table 3-1.

Figure 3-65. FSM_TRIG_MASK_1 Register

7	6	5	4	3	2	1	0
GPIO4_FSM_MASK_POL	GPIO4_FSM_MASK	GPIO3_FSM_MASK_POL	GPIO3_FSM_MASK	GPIO2_FSM_MASK_POL	GPIO2_FSM_MASK	GPIO1_FSM_MASK_POL	GPIO1_FSM_MASK
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-67. FSM_TRIG_MASK_1 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	GPIO4_FSM_MASK_POL	R/W	0b	FSM trigger masking polarity select for GPIOx: (Default from NVM memory) 0b = Masking sets signal value to '0' 1b = Masking sets signal value to '1'
6	GPIO4_FSM_MASK	R/W	0b	FSM trigger mask for GPIOx: (Default from NVM memory) 0b = Not masked 1b = Masked
5	GPIO3_FSM_MASK_POL	R/W	0b	FSM trigger masking polarity select for GPIOx: (Default from NVM memory) 0b = Masking sets signal value to '0' 1b = Masking sets signal value to '1'
4	GPIO3_FSM_MASK	R/W	0b	FSM trigger mask for GPIOx: (Default from NVM memory) 0b = Not masked 1b = Masked
3	GPIO2_FSM_MASK_POL	R/W	0b	FSM trigger masking polarity select for GPIOx: (Default from NVM memory) 0b = Masking sets signal value to '0' 1b = Masking sets signal value to '1'
2	GPIO2_FSM_MASK	R/W	0b	FSM trigger mask for GPIOx: (Default from NVM memory) 0b = Not masked 1b = Masked
1	GPIO1_FSM_MASK_POL	R/W	0b	FSM trigger masking polarity select for GPIOx: (Default from NVM memory) 0b = Masking sets signal value to '0' 1b = Masking sets signal value to '1'
0	GPIO1_FSM_MASK	R/W	0b	FSM trigger mask for GPIOx: (Default from NVM memory) 0b = Not masked 1b = Masked

3.1.66 FSM_TRIG_MASK_2 Register (Offset = 0x47) [Reset = 0x0]

FSM_TRIG_MASK_2 is shown in Figure 3-66 and described in Table 3-68.

Return to the Table 3-1.

Figure 3-66. FSM_TRIG_MASK_2 Register

7	6	5	4	3	2	1	0
GPIO8_FSM_MASK_POL	GPIO8_FSM_MASK	GPIO7_FSM_MASK_POL	GPIO7_FSM_MASK	GPIO6_FSM_MASK_POL	GPIO6_FSM_MASK	GPIO5_FSM_MASK_POL	GPIO5_FSM_MASK
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-68. FSM_TRIG_MASK_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	GPIO8_FSM_MASK_POL	R/W	0b	FSM trigger masking polarity select for GPIOx: (Default from NVM memory) 0b = Masking sets signal value to '0' 1b = Masking sets signal value to '1'
6	GPIO8_FSM_MASK	R/W	0b	FSM trigger mask for GPIOx: (Default from NVM memory) 0b = Not masked 1b = Masked
5	GPIO7_FSM_MASK_POL	R/W	0b	FSM trigger masking polarity select for GPIOx: (Default from NVM memory) 0b = Masking sets signal value to '0' 1b = Masking sets signal value to '1'
4	GPIO7_FSM_MASK	R/W	0b	FSM trigger mask for GPIOx: (Default from NVM memory) 0b = Not masked 1b = Masked
3	GPIO6_FSM_MASK_POL	R/W	0b	FSM trigger masking polarity select for GPIOx: (Default from NVM memory) 0b = Masking sets signal value to '0' 1b = Masking sets signal value to '1'
2	GPIO6_FSM_MASK	R/W	0b	FSM trigger mask for GPIOx: (Default from NVM memory) 0b = Not masked 1b = Masked
1	GPIO5_FSM_MASK_POL	R/W	0b	FSM trigger masking polarity select for GPIOx: (Default from NVM memory) 0b = Masking sets signal value to '0' 1b = Masking sets signal value to '1'
0	GPIO5_FSM_MASK	R/W	0b	FSM trigger mask for GPIOx: (Default from NVM memory) 0b = Not masked 1b = Masked

3.1.67 FSM_TRIG_MASK_3 Register (Offset = 0x48) [Reset = 0x0]

FSM_TRIG_MASK_3 is shown in Figure 3-67 and described in Table 3-69.

Return to the Table 3-1.

Figure 3-67. FSM_TRIG_MASK_3 Register

7	6	5	4	3	2	1	0
RESERVED		GPIO11_FSM_MASK_POL	GPIO11_FSM_MASK	GPIO10_FSM_MASK_POL	GPIO10_FSM_MASK	GPIO9_FSM_MASK_POL	GPIO9_FSM_MASK
R/W-0b		R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-69. FSM_TRIG_MASK_3 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5	GPIO11_FSM_MASK_POL	R/W	0b	FSM trigger masking polarity select for GPIOx: (Default from NVM memory) 0b = Masking sets signal value to '0' 1b = Masking sets signal value to '1'
4	GPIO11_FSM_MASK	R/W	0b	FSM trigger mask for GPIOx: (Default from NVM memory) 0b = Not masked 1b = Masked
3	GPIO10_FSM_MASK_POL	R/W	0b	FSM trigger masking polarity select for GPIOx: (Default from NVM memory) 0b = Masking sets signal value to '0' 1b = Masking sets signal value to '1'
2	GPIO10_FSM_MASK	R/W	0b	FSM trigger mask for GPIOx: (Default from NVM memory) 0b = Not masked 1b = Masked
1	GPIO9_FSM_MASK_POL	R/W	0b	FSM trigger masking polarity select for GPIOx: (Default from NVM memory) 0b = Masking sets signal value to '0' 1b = Masking sets signal value to '1'
0	GPIO9_FSM_MASK	R/W	0b	FSM trigger mask for GPIOx: (Default from NVM memory) 0b = Not masked 1b = Masked

3.1.68 MASK_BUCK1_2 Register (Offset = 0x49) [Reset = 0x0]

MASK_BUCK1_2 is shown in Figure 3-68 and described in Table 3-70.

Return to the Table 3-1.

Figure 3-68. MASK_BUCK1_2 Register

7	6	5	4	3	2	1	0
BUCK2_ILIM_MASK	RESERVED	BUCK2_UV_MASK	BUCK2_OV_MASK	BUCK1_ILIM_MASK	RESERVED	BUCK1_UV_MASK	BUCK1_OV_MASK
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-70. MASK_BUCK1_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	BUCK2_ILIM_MASK	R/W	0b	Masking for BUCK2 current monitoring interrupt BUCK2_ILIM_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
6	RESERVED	R/W	0b	
5	BUCK2_UV_MASK	R/W	0b	Masking of BUCK2 under-voltage detection interrupt BUCK2_UV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
4	BUCK2_OV_MASK	R/W	0b	Masking of BUCK2 over-voltage detection interrupt BUCK2_OV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
3	BUCK1_ILIM_MASK	R/W	0b	Masking for BUCK1 current monitoring interrupt BUCK1_ILIM_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
2	RESERVED	R/W	0b	
1	BUCK1_UV_MASK	R/W	0b	Masking of BUCK1 under-voltage detection interrupt BUCK1_UV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
0	BUCK1_OV_MASK	R/W	0b	Masking of BUCK1 over-voltage detection interrupt BUCK1_OV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.

3.1.69 MASK_BUCK3_4 Register (Offset = 0x4A) [Reset = 0x0]

MASK_BUCK3_4 is shown in Figure 3-69 and described in Table 3-71.

Return to the Table 3-1.

Figure 3-69. MASK_BUCK3_4 Register

7	6	5	4	3	2	1	0
BUCK4_ILIM_MASK	RESERVED	BUCK4_UV_MASK	BUCK4_OV_MASK	BUCK3_ILIM_MASK	RESERVED	BUCK3_UV_MASK	BUCK3_OV_MASK
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-71. MASK_BUCK3_4 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	BUCK4_ILIM_MASK	R/W	0b	Masking for BUCK4 current monitoring interrupt BUCK4_ILIM_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
6	RESERVED	R/W	0b	
5	BUCK4_UV_MASK	R/W	0b	Masking of BUCK4 under-voltage detection interrupt BUCK4_UV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
4	BUCK4_OV_MASK	R/W	0b	Masking of BUCK4 over-voltage detection interrupt BUCK4_OV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
3	BUCK3_ILIM_MASK	R/W	0b	Masking for BUCK3 current monitoring interrupt BUCK3_ILIM_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
2	RESERVED	R/W	0b	
1	BUCK3_UV_MASK	R/W	0b	Masking of BUCK3 under-voltage detection interrupt BUCK3_UV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
0	BUCK3_OV_MASK	R/W	0b	Masking of BUCK3 over-voltage detection interrupt BUCK3_OV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.

3.1.70 MASK_BUCK5 Register (Offset = 0x4B) [Reset = 0x0]

MASK_BUCK5 is shown in Figure 3-70 and described in Table 3-72.

Return to the Table 3-1.

Figure 3-70. MASK_BUCK5 Register

7	6	5	4	3	2	1	0
RESERVED				BUCK5_ILIM_MASK	RESERVED	BUCK5_UV_MASK	BUCK5_OV_MASK
R/W-0b				R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-72. MASK_BUCK5 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	RESERVED	R/W	0b	
3	BUCK5_ILIM_MASK	R/W	0b	Masking for BUCK5 current monitoring interrupt BUCK5_ILIM_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
2	RESERVED	R/W	0b	
1	BUCK5_UV_MASK	R/W	0b	Masking of BUCK5 under-voltage detection interrupt BUCK5_UV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
0	BUCK5_OV_MASK	R/W	0b	Masking of BUCK5 over-voltage detection interrupt BUCK5_OV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.

3.1.71 MASK_LDO1_2 Register (Offset = 0x4C) [Reset = 0x0]

MASK_LDO1_2 is shown in Figure 3-71 and described in Table 3-73.

Return to the Table 3-1.

Figure 3-71. MASK_LDO1_2 Register

7	6	5	4	3	2	1	0
LDO2_ILIM_MASK	RESERVED	LDO2_UV_MASK	LDO2_OV_MASK	LDO1_ILIM_MASK	RESERVED	LDO1_UV_MASK	LDO1_OV_MASK
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-73. MASK_LDO1_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	LDO2_ILIM_MASK	R/W	0b	Masking for LDO2 current monitoring interrupt LDO2_ILIM_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
6	RESERVED	R/W	0b	
5	LDO2_UV_MASK	R/W	0b	Masking of LDO2 under-voltage detection interrupt LDO2_UV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
4	LDO2_OV_MASK	R/W	0b	Masking of LDO2 over-voltage detection interrupt LDO2_OV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
3	LDO1_ILIM_MASK	R/W	0b	Masking for LDO1 current monitoring interrupt LDO1_ILIM_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
2	RESERVED	R/W	0b	
1	LDO1_UV_MASK	R/W	0b	Masking of LDO1 under-voltage detection interrupt LDO1_UV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
0	LDO1_OV_MASK	R/W	0b	Masking of LDO1 over-voltage detection interrupt LDO1_OV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.

3.1.72 MASK_LDO3_4 Register (Offset = 0x4D) [Reset = 0x0]

MASK_LDO3_4 is shown in Figure 3-72 and described in Table 3-74.

Return to the Table 3-1.

Figure 3-72. MASK_LDO3_4 Register

7	6	5	4	3	2	1	0
LDO4_ILIM_MASK	RESERVED	LDO4_UV_MASK	LDO4_OV_MASK	LDO3_ILIM_MASK	RESERVED	LDO3_UV_MASK	LDO3_OV_MASK
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-74. MASK_LDO3_4 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	LDO4_ILIM_MASK	R/W	0b	Masking for LDO4 current monitoring interrupt LDO4_ILIM_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
6	RESERVED	R/W	0b	
5	LDO4_UV_MASK	R/W	0b	Masking of LDO4 under-voltage detection interrupt LDO4_UV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
4	LDO4_OV_MASK	R/W	0b	Masking of LDO4 over-voltage detection interrupt LDO4_OV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
3	LDO3_ILIM_MASK	R/W	0b	Masking for LDO3 current monitoring interrupt LDO3_ILIM_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
2	RESERVED	R/W	0b	
1	LDO3_UV_MASK	R/W	0b	Masking of LDO3 under-voltage detection interrupt LDO3_UV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
0	LDO3_OV_MASK	R/W	0b	Masking of LDO3 over-voltage detection interrupt LDO3_OV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.

3.1.73 MASK_VMON Register (Offset = 0x4E) [Reset = 0x0]

MASK_VMON is shown in Figure 3-73 and described in Table 3-75.

Return to the Table 3-1.

Figure 3-73. MASK_VMON Register

7	6	5	4	3	2	1	0
RESERVED						VCCA_UV_MASK	VCCA_OV_MASK
R/W-0b						R/W-0b	R/W-0b

Table 3-75. MASK_VMON Register Field Descriptions

Bit	Field	Type	Reset	Description
7:2	RESERVED	R/W	0b	
1	VCCA_UV_MASK	R/W	0b	Masking of VCCA under-voltage detection interrupt VCCA_UV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
0	VCCA_OV_MASK	R/W	0b	Masking of VCCA over-voltage detection interrupt VCCA_OV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.

3.1.74 MASK_GPIO1_8_FALL Register (Offset = 0x4F) [Reset = 0x0]

MASK_GPIO1_8_FALL is shown in Figure 3-74 and described in Table 3-76.

Return to the Table 3-1.

Figure 3-74. MASK_GPIO1_8_FALL Register

7	6	5	4	3	2	1	0
GPIO8_FALL_MASK	GPIO7_FALL_MASK	GPIO6_FALL_MASK	GPIO5_FALL_MASK	GPIO4_FALL_MASK	GPIO3_FALL_MASK	GPIO2_FALL_MASK	GPIO1_FALL_MASK
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-76. MASK_GPIO1_8_FALL Register Field Descriptions

Bit	Field	Type	Reset	Description
7	GPIO8_FALL_MASK	R/W	0b	Masking of interrupt for GPIO8 low state transition: This bit does not affect GPIO8_IN status bit in GPIO_IN_1 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
6	GPIO7_FALL_MASK	R/W	0b	Masking of interrupt for GPIO7 low state transition: This bit does not affect GPIO7_IN status bit in GPIO_IN_1 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
5	GPIO6_FALL_MASK	R/W	0b	Masking of interrupt for GPIO6 low state transition: This bit does not affect GPIO6_IN status bit in GPIO_IN_1 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
4	GPIO5_FALL_MASK	R/W	0b	Masking of interrupt for GPIO5 low state transition: This bit does not affect GPIO5_IN status bit in GPIO_IN_1 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
3	GPIO4_FALL_MASK	R/W	0b	Masking of interrupt for GPIO4 low state transition: This bit does not affect GPIO4_IN status bit in GPIO_IN_1 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
2	GPIO3_FALL_MASK	R/W	0b	Masking of interrupt for GPIO3 low state transition: This bit does not affect GPIO3_IN status bit in GPIO_IN_1 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
1	GPIO2_FALL_MASK	R/W	0b	Masking of interrupt for GPIO2 low state transition: This bit does not affect GPIO2_IN status bit in GPIO_IN_1 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
0	GPIO1_FALL_MASK	R/W	0b	Masking of interrupt for GPIO1 low state transition: This bit does not affect GPIO1_IN status bit in GPIO_IN_1 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.

3.1.75 MASK_GPIO1_8_RISE Register (Offset = 0x50) [Reset = 0x0]

MASK_GPIO1_8_RISE is shown in Figure 3-75 and described in Table 3-77.

Return to the Table 3-1.

Figure 3-75. MASK_GPIO1_8_RISE Register

7	6	5	4	3	2	1	0
GPIO8_RISE_MASK	GPIO7_RISE_MASK	GPIO6_RISE_MASK	GPIO5_RISE_MASK	GPIO4_RISE_MASK	GPIO3_RISE_MASK	GPIO2_RISE_MASK	GPIO1_RISE_MASK
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-77. MASK_GPIO1_8_RISE Register Field Descriptions

Bit	Field	Type	Reset	Description
7	GPIO8_RISE_MASK	R/W	0b	Masking of interrupt for GPIO8 high state transition: This bit does not affect GPIO8_IN status bit in GPIO_IN_1 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
6	GPIO7_RISE_MASK	R/W	0b	Masking of interrupt for GPIO7 high state transition: This bit does not affect GPIO7_IN status bit in GPIO_IN_1 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
5	GPIO6_RISE_MASK	R/W	0b	Masking of interrupt for GPIO6 high state transition: This bit does not affect GPIO6_IN status bit in GPIO_IN_1 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
4	GPIO5_RISE_MASK	R/W	0b	Masking of interrupt for GPIO5 high state transition: This bit does not affect GPIO5_IN status bit in GPIO_IN_1 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
3	GPIO4_RISE_MASK	R/W	0b	Masking of interrupt for GPIO4 high state transition: This bit does not affect GPIO4_IN status bit in GPIO_IN_1 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
2	GPIO3_RISE_MASK	R/W	0b	Masking of interrupt for GPIO3 high state transition: This bit does not affect GPIO3_IN status bit in GPIO_IN_1 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
1	GPIO2_RISE_MASK	R/W	0b	Masking of interrupt for GPIO2 high state transition: This bit does not affect GPIO2_IN status bit in GPIO_IN_1 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
0	GPIO1_RISE_MASK	R/W	0b	Masking of interrupt for GPIO1 high state transition: This bit does not affect GPIO1_IN status bit in GPIO_IN_1 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.

3.1.76 MASK_GPIO9_11 Register (Offset = 0x51) [Reset = 0x0]

MASK_GPIO9_11 is shown in Figure 3-76 and described in Table 3-78.

Return to the Table 3-1.

Figure 3-76. MASK_GPIO9_11 Register

7	6	5	4	3	2	1	0
RESERVED		GPIO11_RISE_MASK	GPIO10_RISE_MASK	GPIO9_RISE_MASK	GPIO11_FALL_MASK	GPIO10_FALL_MASK	GPIO9_FALL_MASK
R/W-0b		R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-78. MASK_GPIO9_11 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5	GPIO11_RISE_MASK	R/W	0b	Masking of interrupt for GPIO11 high state transition: This bit does not affect GPIO11_IN status bit in GPIO_IN_2 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
4	GPIO10_RISE_MASK	R/W	0b	Masking of interrupt for GPIO10 high state transition: This bit does not affect GPIO10_IN status bit in GPIO_IN_2 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
3	GPIO9_RISE_MASK	R/W	0b	Masking of interrupt for GPIO9 high state transition: This bit does not affect GPIO9_IN status bit in GPIO_IN_2 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
2	GPIO11_FALL_MASK	R/W	0b	Masking of interrupt for GPIO11 low state transition: This bit does not affect GPIO11_IN status bit in GPIO_IN_2 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
1	GPIO10_FALL_MASK	R/W	0b	Masking of interrupt for GPIO10 low state transition: This bit does not affect GPIO10_IN status bit in GPIO_IN_2 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
0	GPIO9_FALL_MASK	R/W	0b	Masking of interrupt for GPIO9 low state transition: This bit does not affect GPIO9_IN status bit in GPIO_IN_2 register. (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.

3.1.77 MASK_STARTUP Register (Offset = 0x52) [Reset = 0x0]

MASK_STARTUP is shown in Figure 3-77 and described in Table 3-79.

Return to the Table 3-1.

Figure 3-77. MASK_STARTUP Register

7	6	5	4	3	2	1	0
RESERVED		SOFT_REBOOT_MASK	FSD_MASK	RESERVED		ENABLE_MASK	NPWRON_START_MASK
R/W-0b		R/W-0b	R/W-0b	R/W-0b		R/W-0b	R/W-0b

Table 3-79. MASK_STARTUP Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5	SOFT_REBOOT_MASK	R/W	0b	Masking of SOFT_REBOOT_MASK interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
4	FSD_MASK	R/W	0b	Masking of FSD_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
3:2	RESERVED	R/W	0b	
1	ENABLE_MASK	R/W	0b	Masking of ENABLE_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
0	NPWRON_START_MASK	R/W	0b	Masking of NPWRON_START_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.

3.1.78 MASK_MISC Register (Offset = 0x53) [Reset = 0x0]

MASK_MISC is shown in Figure 3-78 and described in Table 3-80.

Return to the Table 3-1.

Figure 3-78. MASK_MISC Register

7	6	5	4	3	2	1	0
RESERVED				TWARN_MASK	RESERVED	EXT_CLK_MASK	BIST_PASS_MASK
R/W-0b				R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-80. MASK_MISC Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	RESERVED	R/W	0b	
3	TWARN_MASK	R/W	0b	Masking of TWARN_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
2	RESERVED	R/W	0b	
1	EXT_CLK_MASK	R/W	0b	Masking of EXT_CLK_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
0	BIST_PASS_MASK	R/W	0b	Masking of BIST_PASS_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.

3.1.79 MASK_MODERATE_ERR Register (Offset = 0x54) [Reset = 0x0]

MASK_MODERATE_ERR is shown in Figure 3-79 and described in Table 3-81.

Return to the Table 3-1.

Figure 3-79. MASK_MODERATE_ERR Register

7	6	5	4	3	2	1	0
NRSTOUT_READBACK_MASK	NINT_READBACK_MASK	NPWRON_LONG_MASK	SPMI_ERR_MASK	RESERVED	REG_CRC_ERR_MASK	BIST_FAIL_MASK	RESERVED
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-81. MASK_MODERATE_ERR Register Field Descriptions

Bit	Field	Type	Reset	Description
7	NRSTOUT_READBACK_MASK	R/W	0b	Masking of NRSTOUT_READBACK_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
6	NINT_READBACK_MASK	R/W	0b	Masking of NINT_READBACK_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
5	NPWRON_LONG_MASK	R/W	0b	Masking of NPWRON_LONG_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
4	SPMI_ERR_MASK	R/W	0b	Masking of SPMI_ERR_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
3	RESERVED	R/W	0b	
2	REG_CRC_ERR_MASK	R/W	0b	Masking of REG_CRC_ERR_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
1	BIST_FAIL_MASK	R/W	0b	Masking of BIST_FAIL_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
0	RESERVED	R/W	0b	

3.1.80 MASK_FSM_ERR Register (Offset = 0x56) [Reset = 0x0]

MASK_FSM_ERR is shown in Figure 3-80 and described in Table 3-82.

Return to the Table 3-1.

Figure 3-80. MASK_FSM_ERR Register

7	6	5	4	3	2	1	0
RESERVED				SOC_PWR_ERR_MASK	MCU_PWR_ERR_MASK	ORD_SHUTDOWN_MASK	IMM_SHUTDOWN_MASK
R/W-0b				R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-82. MASK_FSM_ERR Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	RESERVED	R/W	0b	
3	SOC_PWR_ERR_MASK	R/W	0b	Masking of SOC_PWR_ERR_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
2	MCU_PWR_ERR_MASK	R/W	0b	Masking of MCU_PWR_ERR_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
1	ORD_SHUTDOWN_MASK	R/W	0b	Masking of ORD_SHUTDOWN_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
0	IMM_SHUTDOWN_MASK	R/W	0b	Masking of IMM_SHUTDOWN_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.

3.1.81 MASK_COMM_ERR Register (Offset = 0x57) [Reset = 0x0]

MASK_COMM_ERR is shown in Figure 3-81 and described in Table 3-83.

Return to the Table 3-1.

Figure 3-81. MASK_COMM_ERR Register

7	6	5	4	3	2	1	0
I2C2_ADR_ERR_MASK	RESERVED	I2C2_CRC_ERR_MASK	RESERVED	COMM_ADR_ERR_MASK	RESERVED	COMM_CRC_ERR_MASK	COMM_FRM_ERR_MASK
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-83. MASK_COMM_ERR Register Field Descriptions

Bit	Field	Type	Reset	Description
7	I2C2_ADR_ERR_MASK	R/W	0b	Masking of I2C2_ADR_ERR_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
6	RESERVED	R/W	0b	
5	I2C2_CRC_ERR_MASK	R/W	0b	Masking of I2C2_CRC_ERR_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
4	RESERVED	R/W	0b	
3	COMM_ADR_ERR_MASK	R/W	0b	Masking of COMM_ADR_ERR_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
2	RESERVED	R/W	0b	
1	COMM_CRC_ERR_MASK	R/W	0b	Masking of COMM_CRC_ERR_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
0	COMM_FRM_ERR_MASK	R/W	0b	Masking of COMM_FRM_ERR_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.

3.1.82 MASK_READBACK_ERR Register (Offset = 0x58) [Reset = 0x0]

MASK_READBACK_ERR is shown in Figure 3-82 and described in Table 3-84.

Return to the Table 3-1.

Figure 3-82. MASK_READBACK_ERR Register

7	6	5	4	3	2	1	0
RESERVED				NRSTOUT_SOC_READBACK_MASK	RESERVED		EN_DRV_READBACK_MASK
R/W-0b				R/W-0b	R/W-0b		R/W-0b

Table 3-84. MASK_READBACK_ERR Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	RESERVED	R/W	0b	
3	NRSTOUT_SOC_READBACK_MASK	R/W	0b	Masking of NRSTOUT_SOC_READBACK_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
2:1	RESERVED	R/W	0b	
0	EN_DRV_READBACK_MASK	R/W	0b	Masking of EN_DRV_READBACK_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.

3.1.83 MASK_ESM Register (Offset = 0x59) [Reset = 0x0]

MASK_ESM is shown in Figure 3-83 and described in Table 3-85.

Return to the Table 3-1.

Figure 3-83. MASK_ESM Register

7	6	5	4	3	2	1	0
RESERVED		ESM_MCU_RST_MASK	ESM_MCU_FAIL_MASK	ESM_MCU_PIN_MASK	ESM_SOC_RST_MASK	ESM_SOC_FAIL_MASK	ESM_SOC_PIN_MASK
R/W-0b		R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-85. MASK_ESM Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5	ESM_MCU_RST_MASK	R/W	0b	Masking of ESM_MCU_RST_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
4	ESM_MCU_FAIL_MASK	R/W	0b	Masking of ESM_MCU_FAIL_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
3	ESM_MCU_PIN_MASK	R/W	0b	Masking of ESM_MCU_PIN_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
2	ESM_SOC_RST_MASK	R/W	0b	Masking of ESM_SOC_RST_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
1	ESM_SOC_FAIL_MASK	R/W	0b	Masking of ESM_SOC_FAIL_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.
0	ESM_SOC_PIN_MASK	R/W	0b	Masking of ESM_SOC_PIN_INT interrupt: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated.

3.1.84 INT_TOP Register (Offset = 0x5A) [Reset = 0x0]

INT_TOP is shown in Figure 3-84 and described in Table 3-86.

Return to the Table 3-1.

Figure 3-84. INT_TOP Register

7	6	5	4	3	2	1	0
FSM_ERR_INT	SEVERE_ERR_INT	MODERATE_ERR_INT	MISC_INT	STARTUP_INT	GPIO_INT	LDO_VMON_INT	BUCK_INT
R-0b	R-0b	R-0b	R-0b	R-0b	R-0b	R-0b	R-0b

Table 3-86. INT_TOP Register Field Descriptions

Bit	Field	Type	Reset	Description
7	FSM_ERR_INT	R	0b	Interrupt indicating that INT_FSM_ERR register has pending interrupt. The reason for the interrupt is indicated in INT_FSM_ERR register. This bit is cleared automatically when INT_FSM_ERR register is cleared to 0x00.
6	SEVERE_ERR_INT	R	0b	Interrupt indicating that INT_SEVERE_ERR register has pending interrupt. The reason for the interrupt is indicated in INT_SEVERE_ERR register. This bit is cleared automatically when INT_SEVERE_ERR register is cleared to 0x00.
5	MODERATE_ERR_INT	R	0b	Interrupt indicating that INT_MODERATE_ERR register has pending interrupt. The reason for the interrupt is indicated in INT_MODERATE_ERR register. This bit is cleared automatically when INT_MODERATE_ERR register is cleared to 0x00.
4	MISC_INT	R	0b	Interrupt indicating that INT_MISC register has pending interrupt. The reason for the interrupt is indicated in INT_MISC register. This bit is cleared automatically when INT_MISC register is cleared to 0x00.
3	STARTUP_INT	R	0b	Interrupt indicating that INT_STARTUP register has pending interrupt. The reason for the interrupt is indicated in INT_STARTUP register. This bit is cleared automatically when INT_STARTUP register is cleared to 0x00.
2	GPIO_INT	R	0b	Interrupt indicating that INT_GPIO register has pending interrupt. The reason for the interrupt is indicated in INT_GPIO register. This bit is cleared automatically when INT_GPIO register is cleared to 0x00.
1	LDO_VMON_INT	R	0b	Interrupt indicating that INT_LDO_VMON register has pending interrupt. The reason for the interrupt is indicated in INT_LDO_VMON register. This bit is cleared automatically when INT_LDO_VMON register is cleared to 0x00.
0	BUCK_INT	R	0b	Interrupt indicating that INT_BUCK register has pending interrupt. The reason for the interrupt is indicated in INT_BUCK register. This bit is cleared automatically when INT_BUCK register is cleared to 0x00.

3.1.85 INT_BUCK Register (Offset = 0x5B) [Reset = 0x0]

INT_BUCK is shown in Figure 3-85 and described in Table 3-87.

Return to the Table 3-1.

Figure 3-85. INT_BUCK Register

7	6	5	4	3	2	1	0
RESERVED					BUCK5_INT	BUCK3_4_INT	BUCK1_2_INT
R-0b					R-0b	R-0b	R-0b

Table 3-87. INT_BUCK Register Field Descriptions

Bit	Field	Type	Reset	Description
7:3	RESERVED	R	0b	
2	BUCK5_INT	R	0b	Interrupt indicating that INT_BUCK5 register has pending interrupt. The reason for the interrupt is indicated in INT_BUCK5 register. This bit is cleared automatically when INT_BUCK5 register is cleared to 0x00.
1	BUCK3_4_INT	R	0b	Interrupt indicating that INT_BUCK3_4 register has pending interrupt. This bit is cleared automatically when INT_BUCK3_4 register is cleared to 0x00.
0	BUCK1_2_INT	R	0b	Interrupt indicating that INT_BUCK1_2 register has pending interrupt. This bit is cleared automatically when INT_BUCK1_2 register is cleared to 0x00.

3.1.86 INT_BUCK1_2 Register (Offset = 0x5C) [Reset = 0x0]

INT_BUCK1_2 is shown in Figure 3-86 and described in Table 3-88.

Return to the Table 3-1.

Figure 3-86. INT_BUCK1_2 Register

7	6	5	4	3	2	1	0
BUCK2_ILIM_INT	BUCK2_SC_INT	BUCK2_UV_INT	BUCK2_OV_INT	BUCK1_ILIM_INT	BUCK1_SC_INT	BUCK1_UV_INT	BUCK1_OV_INT
R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b

Table 3-88. INT_BUCK1_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	BUCK2_ILIM_INT	R/W1C	0b	Latched status bit indicating that BUCK2 output current limit has been triggered. Write 1 to clear.
6	BUCK2_SC_INT	R/W1C	0b	Latched status bit indicating that the BUCK2 output voltage has fallen below 150 mV level during operation or BUCK2 output didn't reach 150 mV level in TBD us from enable. Write 1 to clear.
5	BUCK2_UV_INT	R/W1C	0b	Latched status bit indicating that BUCK2 output under-voltage has been detected. Write 1 to clear.
4	BUCK2_OV_INT	R/W1C	0b	Latched status bit indicating that BUCK2 output over-voltage has been detected. Write 1 to clear.
3	BUCK1_ILIM_INT	R/W1C	0b	Latched status bit indicating that BUCK1 output current limit has been triggered. Write 1 to clear.
2	BUCK1_SC_INT	R/W1C	0b	Latched status bit indicating that the BUCK1 output voltage has fallen below 150 mV level during operation or BUCK1 output didn't reach 150 mV level in TBD us from enable. Write 1 to clear.
1	BUCK1_UV_INT	R/W1C	0b	Latched status bit indicating that BUCK1 output under-voltage has been detected. Write 1 to clear.
0	BUCK1_OV_INT	R/W1C	0b	Latched status bit indicating that BUCK1 output over-voltage has been detected. Write 1 to clear.

3.1.87 INT_BUCK3_4 Register (Offset = 0x5D) [Reset = 0x0]

INT_BUCK3_4 is shown in Figure 3-87 and described in Table 3-89.

Return to the Table 3-1.

Figure 3-87. INT_BUCK3_4 Register

7	6	5	4	3	2	1	0
BUCK4_ILIM_INT	BUCK4_SC_INT	BUCK4_UV_INT	BUCK4_OV_INT	BUCK3_ILIM_INT	BUCK3_SC_INT	BUCK3_UV_INT	BUCK3_OV_INT
R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b

Table 3-89. INT_BUCK3_4 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	BUCK4_ILIM_INT	R/W1C	0b	Latched status bit indicating that BUCK4 output current limit has been triggered. Write 1 to clear.
6	BUCK4_SC_INT	R/W1C	0b	Latched status bit indicating that the BUCK4 output voltage has fallen below 150 mV level during operation or BUCK4 output didn't reach 150 mV level in TBD us from enable. Write 1 to clear.
5	BUCK4_UV_INT	R/W1C	0b	Latched status bit indicating that BUCK4 output under-voltage has been detected. Write 1 to clear.
4	BUCK4_OV_INT	R/W1C	0b	Latched status bit indicating that BUCK4 output over-voltage has been detected. Write 1 to clear.
3	BUCK3_ILIM_INT	R/W1C	0b	Latched status bit indicating that BUCK3 output current limit has been triggered. Write 1 to clear.
2	BUCK3_SC_INT	R/W1C	0b	Latched status bit indicating that the BUCK3 output voltage has fallen below 150 mV level during operation or BUCK3 output didn't reach 150 mV level in TBD us from enable. Write 1 to clear.
1	BUCK3_UV_INT	R/W1C	0b	Latched status bit indicating that BUCK3 output under-voltage has been detected. Write 1 to clear.
0	BUCK3_OV_INT	R/W1C	0b	Latched status bit indicating that BUCK3 output over-voltage has been detected. Write 1 to clear.

3.1.88 INT_BUCK5 Register (Offset = 0x5E) [Reset = 0x0]

INT_BUCK5 is shown in Figure 3-88 and described in Table 3-90.

Return to the Table 3-1.

Figure 3-88. INT_BUCK5 Register

7	6	5	4	3	2	1	0
RESERVED				BUCK5_ILIM_INT	BUCK5_SC_INT	BUCK5_UV_INT	BUCK5_OV_INT
R/W-0b				R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b

Table 3-90. INT_BUCK5 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	RESERVED	R/W	0b	
3	BUCK5_ILIM_INT	R/W1C	0b	Latched status bit indicating that BUCK5 output current limit has been triggered. Write 1 to clear.
2	BUCK5_SC_INT	R/W1C	0b	Latched status bit indicating that the BUCK5 output voltage has fallen below 150 mV level during operation or BUCK5 output didn't reach 150 mV level in TBD us from enable. Write 1 to clear.
1	BUCK5_UV_INT	R/W1C	0b	Latched status bit indicating that BUCK5 output under-voltage has been detected. Write 1 to clear.
0	BUCK5_OV_INT	R/W1C	0b	Latched status bit indicating that BUCK5 output over-voltage has been detected. Write 1 to clear.

3.1.89 INT_LDO_VMON Register (Offset = 0x5F) [Reset = 0x0]

INT_LDO_VMON is shown in Figure 3-89 and described in Table 3-91.

Return to the Table 3-1.

Figure 3-89. INT_LDO_VMON Register

7	6	5	4	3	2	1	0
RESERVED			VCCA_INT	RESERVED		LDO3_4_INT	LDO1_2_INT
R-0b			R-0b	R-0b		R-0b	R-0b

Table 3-91. INT_LDO_VMON Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	RESERVED	R	0b	
4	VCCA_INT	R	0b	Interrupt indicating that INT_VMON register has pending interrupt. The reason for the interrupt is indicated in INT_VMON register. This bit is cleared automatically when INT_VMON register is cleared to 0x00.
3:2	RESERVED	R	0b	
1	LDO3_4_INT	R	0b	Interrupt indicating that INT_LDO3_4 register has pending interrupt. This bit is cleared automatically when INT_LDO3_4 register is cleared to 0x00.
0	LDO1_2_INT	R	0b	Interrupt indicating that INT_LDO1_2 register has pending interrupt. This bit is cleared automatically when INT_LDO1_2 register is cleared to 0x00.

3.1.90 INT_LDO1_2 Register (Offset = 0x60) [Reset = 0x0]

INT_LDO1_2 is shown in Figure 3-90 and described in Table 3-92.

Return to the Table 3-1.

Figure 3-90. INT_LDO1_2 Register

7	6	5	4	3	2	1	0
LDO2_ILIM_INT	LDO2_SC_INT	LDO2_UV_INT	LDO2_OV_INT	LDO1_ILIM_INT	LDO1_SC_INT	LDO1_UV_INT	LDO1_OV_INT
R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b

Table 3-92. INT_LDO1_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	LDO2_ILIM_INT	R/W1C	0b	Latched status bit indicating that LDO2 output current limit has been triggered. Write 1 to clear.
6	LDO2_SC_INT	R/W1C	0b	Latched status bit indicating that LDO2 output voltage has fallen below 150 mV level during operation or LDO2 output didn't reach 150 mV level in TBD us from enable. Write 1 to clear.
5	LDO2_UV_INT	R/W1C	0b	Latched status bit indicating that LDO2 output under-voltage has been detected. Write 1 to clear.
4	LDO2_OV_INT	R/W1C	0b	Latched status bit indicating that LDO2 output over-voltage has been detected. Write 1 to clear.
3	LDO1_ILIM_INT	R/W1C	0b	Latched status bit indicating that LDO1 output current limit has been triggered. Write 1 to clear.
2	LDO1_SC_INT	R/W1C	0b	Latched status bit indicating that LDO1 output voltage has fallen below 150 mV level during operation or LDO1 output didn't reach 150 mV level in TBD us from enable. Write 1 to clear.
1	LDO1_UV_INT	R/W1C	0b	Latched status bit indicating that LDO1 output under-voltage has been detected. Write 1 to clear.
0	LDO1_OV_INT	R/W1C	0b	Latched status bit indicating that LDO1 output over-voltage has been detected. Write 1 to clear.

3.1.91 INT_LDO3_4 Register (Offset = 0x61) [Reset = 0x0]

INT_LDO3_4 is shown in Figure 3-91 and described in Table 3-93.

Return to the Table 3-1.

Figure 3-91. INT_LDO3_4 Register

7	6	5	4	3	2	1	0
LDO4_ILIM_INT	LDO4_SC_INT	LDO4_UV_INT	LDO4_OV_INT	LDO3_ILIM_INT	LDO3_SC_INT	LDO3_UV_INT	LDO3_OV_INT
R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b

Table 3-93. INT_LDO3_4 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	LDO4_ILIM_INT	R/W1C	0b	Latched status bit indicating that LDO4 output current limit has been triggered. Write 1 to clear.
6	LDO4_SC_INT	R/W1C	0b	Latched status bit indicating that LDO4 output voltage has fallen below 150 mV level during operation or LDO4 output didn't reach 150 mV level in TBD us from enable. Write 1 to clear.
5	LDO4_UV_INT	R/W1C	0b	Latched status bit indicating that LDO4 output under-voltage has been detected. Write 1 to clear.
4	LDO4_OV_INT	R/W1C	0b	Latched status bit indicating that LDO4 output over-voltage has been detected. Write 1 to clear.
3	LDO3_ILIM_INT	R/W1C	0b	Latched status bit indicating that LDO3 output current limit has been triggered. Write 1 to clear.
2	LDO3_SC_INT	R/W1C	0b	Latched status bit indicating that LDO3 output voltage has fallen below 150 mV level during operation or LDO3 output didn't reach 150 mV level in TBD us from enable. Write 1 to clear.
1	LDO3_UV_INT	R/W1C	0b	Latched status bit indicating that LDO3 output under-voltage has been detected. Write 1 to clear.
0	LDO3_OV_INT	R/W1C	0b	Latched status bit indicating that LDO3 output over-voltage has been detected. Write 1 to clear.

3.1.92 INT_VMON Register (Offset = 0x62) [Reset = 0x0]

INT_VMON is shown in Figure 3-92 and described in Table 3-94.

Return to the Table 3-1.

Figure 3-92. INT_VMON Register

7	6	5	4	3	2	1	0
RESERVED						VCCA_UV_INT	VCCA_OV_INT
R/W-0b						R/W1C-0b	R/W1C-0b

Table 3-94. INT_VMON Register Field Descriptions

Bit	Field	Type	Reset	Description
7:2	RESERVED	R/W	0b	
1	VCCA_UV_INT	R/W1C	0b	Latched status bit indicating that the VCCA input voltage has decreased below the under-voltage monitoring level. The actual status of the VCCA under-voltage monitoring is indicated by VCCA_UV_STAT bit. Write 1 to clear interrupt.
0	VCCA_OV_INT	R/W1C	0b	Latched status bit indicating that the VCCA input voltage has exceeded the over-voltage detection level. The actual status of the over-voltage is indicated by VCCA_OV_STAT bit. Write 1 to clear interrupt.

3.1.93 INT_GPIO Register (Offset = 0x63) [Reset = 0x0]

INT_GPIO is shown in Figure 3-93 and described in Table 3-95.

Return to the Table 3-1.

Figure 3-93. INT_GPIO Register

7	6	5	4	3	2	1	0
RESERVED				GPIO1_8_INT	GPIO11_INT	GPIO10_INT	GPIO9_INT
R/W-0b				R-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b

Table 3-95. INT_GPIO Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	RESERVED	R/W	0b	
3	GPIO1_8_INT	R	0b	Interrupt indicating that INT_GPIO1_8 has pending interrupt. The reason for the interrupt is indicated in INT_GPIO1_8 register. This bit is cleared automatically when INT_GPIO1_8 register is cleared to 0x00.
2	GPIO11_INT	R/W1C	0b	Latched status bit indicating that GPIO11 has pending interrupt. GPIO11_IN bit in GPIO_IN_2 register shows the status of the GPIO11 signal. Write 1 to clear interrupt.
1	GPIO10_INT	R/W1C	0b	Latched status bit indicating that GPIO10 has pending interrupt. GPIO10_IN bit in GPIO_IN_2 register shows the status of the GPIO10 signal. Write 1 to clear interrupt.
0	GPIO9_INT	R/W1C	0b	Latched status bit indicating that GPIO9 has pending interrupt. GPIO9_IN bit in GPIO_IN_2 register shows the status of the GPIO9 signal. Write 1 to clear interrupt.

3.1.94 INT_GPIO1_8 Register (Offset = 0x64) [Reset = 0x0]

INT_GPIO1_8 is shown in Figure 3-94 and described in Table 3-96.

Return to the Table 3-1.

Figure 3-94. INT_GPIO1_8 Register

7	6	5	4	3	2	1	0
GPIO8_INT	GPIO7_INT	GPIO6_INT	GPIO5_INT	GPIO4_INT	GPIO3_INT	GPIO2_INT	GPIO1_INT
R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b

Table 3-96. INT_GPIO1_8 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	GPIO8_INT	R/W1C	0b	Latched status bit indicating that GPIO8 has has pending interrupt. GPIO8_IN bit in GPIO_IN_1 register shows the status of the GPIO8 signal. Write 1 to clear interrupt.
6	GPIO7_INT	R/W1C	0b	Latched status bit indicating that GPIO7 has has pending interrupt. GPIO7_IN bit in GPIO_IN_1 register shows the status of the GPIO7 signal. Write 1 to clear interrupt.
5	GPIO6_INT	R/W1C	0b	Latched status bit indicating that GPIO6 has has pending interrupt. GPIO6_IN bit in GPIO_IN_1 register shows the status of the GPIO6 signal. Write 1 to clear interrupt.
4	GPIO5_INT	R/W1C	0b	Latched status bit indicating that GPIO5 has has pending interrupt. GPIO5_IN bit in GPIO_IN_1 register shows the status of the GPIO5 signal. Write 1 to clear interrupt.
3	GPIO4_INT	R/W1C	0b	Latched status bit indicating that GPIO4 has has pending interrupt. GPIO4_IN bit in GPIO_IN_1 register shows the status of the GPIO4 signal. Write 1 to clear interrupt.
2	GPIO3_INT	R/W1C	0b	Latched status bit indicating that GPIO3 has has pending interrupt. GPIO3_IN bit in GPIO_IN_1 register shows the status of the GPIO3 signal. Write 1 to clear interrupt.
1	GPIO2_INT	R/W1C	0b	Latched status bit indicating that GPIO2 has pending interrupt. GPIO2_IN bit in GPIO_IN_1 register shows the status of the GPIO2 signal. Write 1 to clear interrupt.
0	GPIO1_INT	R/W1C	0b	Latched status bit indicating that GPIO1 has pending interrupt. GPIO1_IN bit in GPIO_IN_1 register shows the status of the GPIO1 signal. Write 1 to clear interrupt.

3.1.95 INT_STARTUP Register (Offset = 0x65) [Reset = 0x0]

INT_STARTUP is shown in Figure 3-95 and described in Table 3-97.

Return to the Table 3-1.

Figure 3-95. INT_STARTUP Register

7	6	5	4	3	2	1	0
RESERVED		SOFT_REBOOT_INT	FSD_INT	RESERVED	RTC_INT	ENABLE_INT	NPWRON_START_INT
R/W-0b		R/W1C-0b	R/W1C-0b	R/W-0b	R-0b	R/W1C-0b	R/W1C-0b

Table 3-97. INT_STARTUP Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5	SOFT_REBOOT_INT	R/W1C	0b	Latched status bit indicating that soft reboot event has been detected. Write 1 to clear.
4	FSD_INT	R/W1C	0b	Latched status bit indicating that PMIC has started from NO_SUPPLY or BACKUP state (first supply dectection). Write 1 to clear.
3	RESERVED	R/W	0b	
2	RTC_INT	R	0b	Latched status bit indicating that RTC_STATUS register has pending interrupt. This bit is cleared automatically when ALARM and TIMER interrupts are cleared.
1	ENABLE_INT	R/W1C	0b	Latched status bit indicating that ENABLE pin active event has been detected. Write 1 to clear.
0	NPWRON_START_INT	R/W1C	0b	Latched status bit indicating that NPWRON startup event has been detected. Write 1 to clear.

3.1.96 INT_MISC Register (Offset = 0x66) [Reset = 0x0]

INT_MISC is shown in Figure 3-96 and described in Table 3-98.

Return to the Table 3-1.

Figure 3-96. INT_MISC Register

7	6	5	4	3	2	1	0
RESERVED				TWARN_INT	RESERVED	EXT_CLK_INT	BIST_PASS_INT
R/W-0b				R/W1C-0b	R/W-0b	R/W1C-0b	R/W1C-0b

Table 3-98. INT_MISC Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	RESERVED	R/W	0b	
3	TWARN_INT	R/W1C	0b	Latched status bit indicating that the die junction temperature has exceeded the thermal warning level. The actual status of the thermal warning is indicated by TWARN_STAT bit in STAT_MISC register. Write 1 to clear interrupt.
2	RESERVED	R/W	0b	
1	EXT_CLK_INT	R/W1C	0b	Latched status bit indicating that external clock is not valid. Internal clock is automatically taken into use. Write 1 to clear.
0	BIST_PASS_INT	R/W1C	0b	Latched status bit indicating that BIST has been completed. Write 1 to clear interrupt.

3.1.97 INT_MODERATE_ERR Register (Offset = 0x67) [Reset = 0x0]

INT_MODERATE_ERR is shown in Figure 3-97 and described in Table 3-99.

Return to the Table 3-1.

Figure 3-97. INT_MODERATE_ERR Register

7	6	5	4	3	2	1	0
NRSTOUT_READBACK_INT	NINT_READBACK_INT	NPWRON_LONG_INT	SPMI_ERR_INT	RECOV_CNT_INT	REG_CRC_ERR_INT	BIST_FAIL_INT	TSD_ORD_INT
R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b

Table 3-99. INT_MODERATE_ERR Register Field Descriptions

Bit	Field	Type	Reset	Description
7	NRSTOUT_READBACK_INT	R/W1C	0b	Latched status bit indicating that NRSTOUT readback error has been detected. Write 1 to clear interrupt.
6	NINT_READBACK_INT	R/W1C	0b	Latched status bit indicating that NINT readback error has been detected. Write 1 to clear interrupt.
5	NPWRON_LONG_INT	R/W1C	0b	Latched status bit indicating that NPWRON long press has been detected. Write 1 to clear.
4	SPMI_ERR_INT	R/W1C	0b	Latched status bit indicating that the SPMI communication interface has detected an error. Write 1 to clear interrupt.
3	RECOV_CNT_INT	R/W1C	0b	Latched status bit indicating that RECOV_CNT has reached the limit (RECOV_CNT_THR). Write 1 to clear.
2	REG_CRC_ERR_INT	R/W1C	0b	Latched status bit indicating that the register CRC checking has detected an error. Write 1 to clear interrupt.
1	BIST_FAIL_INT	R/W1C	0b	Latched status bit indicating that the LBIST or ABIST has detected an error. Write 1 to clear interrupt.
0	TSD_ORD_INT	R/W1C	0b	Latched status bit indicating that the die junction temperature has exceeded the thermal level causing a sequenced shutdown. The regulators have been disabled. The regulators cannot be enabled if this bit is active. The actual status of the temperature is indicated by TSD_ORD_STAT bit in STAT_MODERATE_ERR register. Write 1 to clear interrupt.

3.1.98 INT_SEVERE_ERR Register (Offset = 0x68) [Reset = 0x0]

INT_SEVERE_ERR is shown in Figure 3-98 and described in Table 3-100.

Return to the Table 3-1.

Figure 3-98. INT_SEVERE_ERR Register

7	6	5	4	3	2	1	0
RESERVED					PFSM_ERR_INT	VCCA_OVP_INT	TSD_IMM_INT
R/W-0b					R/W1C-0b	R/W1C-0b	R/W1C-0b

Table 3-100. INT_SEVERE_ERR Register Field Descriptions

Bit	Field	Type	Reset	Description
7:3	RESERVED	R/W	0b	
2	PFSM_ERR_INT	R/W1C	0b	Latched status bit indicating that the PFSM sequencer has detected an error. Write 1 to clear interrupt.
1	VCCA_OVP_INT	R/W1C	0b	Latched status bit indicating that the VCCA input voltage has exceeded the over-voltage threshold level causing an immediate shutdown. The regulators have been disabled. Write 1 to clear interrupt.
0	TSD_IMM_INT	R/W1C	0b	Latched status bit indicating that the die junction temperature has exceeded the thermal level causing an immediate shutdown. The regulators have been disabled. The regulators cannot be enabled if this bit is active. The actual status of the temperature is indicated by TSD_IMM_STAT bit in THER_CLK_STATUS register. Write 1 to clear interrupt.

3.1.99 INT_FSM_ERR Register (Offset = 0x69) [Reset = 0x0]

INT_FSM_ERR is shown in Figure 3-99 and described in Table 3-101.

Return to the Table 3-1.

Figure 3-99. INT_FSM_ERR Register

7	6	5	4	3	2	1	0
WD_INT	ESM_INT	READBACK_ERR_INT	COMM_ERR_INT	SOC_PWR_ERR_INT	MCU_PWR_ERR_INT	ORD_SHUTDOWN_INT	IMM_SHUTDOWN_INT
R-0b	R-0b	R-0b	R-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b

Table 3-101. INT_FSM_ERR Register Field Descriptions

Bit	Field	Type	Reset	Description
7	WD_INT	R	0b	Interrupt indicating that WD_ERR_STATUS register has pending interrupt. This bit is cleared automatically when WD_RST_INT, WD_FAIL_INT and WD_LONGWIN_TIMEOUT_INT are cleared.
6	ESM_INT	R	0b	Interrupt indicating that INT_ESM has pending interrupt. This bit is cleared automatically when INT_ESM register is cleared to 0x00.
5	READBACK_ERR_INT	R	0b	Interrupt indicating that INT_READBACK_ERR has pending interrupt. This bit is cleared automatically when INT_READBACK_ERR register is cleared to 0x00.
4	COMM_ERR_INT	R	0b	Interrupt indicating that INT_COMM_ERR has pending interrupt. The reason for the interrupt is indicated in INT_COMM_ERR register. This bit is cleared automatically when INT_COMM_ERR register is cleared to 0x00.
3	SOC_PWR_ERR_INT	R/W1C	0b	Latched status bit indicating that SOC power error has been detected. Write 1 to clear.
2	MCU_PWR_ERR_INT	R/W1C	0b	Latched status bit indicating that MCU power error has been detected. Write 1 to clear.
1	ORD_SHUTDOWN_INT	R/W1C	0b	Latched status bit indicating that orderly shutdown has been detected. Write 1 to clear.
0	IMM_SHUTDOWN_INT	R/W1C	0b	Latched status bit indicating that immediate shutdown has been detected. Write 1 to clear.

3.1.100 INT_COMM_ERR Register (Offset = 0x6A) [Reset = 0x0]

INT_COMM_ERR is shown in Figure 3-100 and described in Table 3-102.

Return to the Table 3-1.

Figure 3-100. INT_COMM_ERR Register

7	6	5	4	3	2	1	0
I2C2_ADR_ERR_INT	RESERVED	I2C2_CRC_ERR_INT	RESERVED	COMM_ADR_ERR_INT	RESERVED	COMM_CRC_ERR_INT	COMM_FRM_ERR_INT
R/W1C-0b	R/W-0b	R/W1C-0b	R/W-0b	R/W1C-0b	R/W-0b	R/W1C-0b	R/W1C-0b

Table 3-102. INT_COMM_ERR Register Field Descriptions

Bit	Field	Type	Reset	Description
7	I2C2_ADR_ERR_INT	R/W1C	0b	Latched status bit indicating that I2C2 write to non-existing, protected or read-only register address has been detected. Write 1 to clear interrupt.
6	RESERVED	R/W	0b	
5	I2C2_CRC_ERR_INT	R/W1C	0b	Latched status bit indicating that I2C2 CRC error has been detected. Write 1 to clear interrupt.
4	RESERVED	R/W	0b	
3	COMM_ADR_ERR_INT	R/W1C	0b	Latched status bit indicating that I2C1/SPI write to non-existing, protected or read-only register address has been detected. Write 1 to clear interrupt.
2	RESERVED	R/W	0b	
1	COMM_CRC_ERR_INT	R/W1C	0b	Latched status bit indicating that I2C1/SPI CRC error has been detected. Write 1 to clear interrupt.
0	COMM_FRM_ERR_INT	R/W1C	0b	Latched status bit indicating that SPI frame error has been detected. Write 1 to clear interrupt.

3.1.101 INT_READBACK_ERR Register (Offset = 0x6B) [Reset = 0x0]

INT_READBACK_ERR is shown in Figure 3-101 and described in Table 3-103.

Return to the Table 3-1.

Figure 3-101. INT_READBACK_ERR Register

7	6	5	4	3	2	1	0
RESERVED				NRSTOUT_SOC_READBACK_INT	RESERVED		EN_DRV_READBACK_INT
R/W-0b				R/W1C-0b	R/W-0b		R/W1C-0b

Table 3-103. INT_READBACK_ERR Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	RESERVED	R/W	0b	
3	NRSTOUT_SOC_READBACK_INT	R/W1C	0b	Latched status bit indicating that NRSTOUT_SOC readback error has been detected. Write 1 to clear interrupt.
2:1	RESERVED	R/W	0b	
0	EN_DRV_READBACK_INT	R/W1C	0b	Latched status bit indicating that EN_DRV readback error has been detected. Write 1 to clear interrupt.

3.1.102 INT_ESM Register (Offset = 0x6C) [Reset = 0x0]

INT_ESM is shown in Figure 3-102 and described in Table 3-104.

Return to the Table 3-1.

Figure 3-102. INT_ESM Register

7	6	5	4	3	2	1	0
RESERVED		ESM_MCU_RST_INT	ESM_MCU_FAIL_INT	ESM_MCU_PIN_INT	ESM_SOC_RST_INT	ESM_SOC_FAIL_INT	ESM_SOC_PIN_INT
R/W-0b		R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b

Table 3-104. INT_ESM Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5	ESM_MCU_RST_INT	R/W1C	0b	Latched status bit indicating that MCU ESM reset has been detected. Write 1 to clear interrupt.
4	ESM_MCU_FAIL_INT	R/W1C	0b	Latched status bit indicating that MCU ESM fail has been detected. Write 1 to clear interrupt.
3	ESM_MCU_PIN_INT	R/W1C	0b	Latched status bit indicating that MCU ESM fault has been detected. Write 1 to clear interrupt.
2	ESM_SOC_RST_INT	R/W1C	0b	Latched status bit indicating that SOC ESM reset has been detected. Write 1 to clear interrupt.
1	ESM_SOC_FAIL_INT	R/W1C	0b	Latched status bit indicating that SOC ESM fail has been detected. Write 1 to clear interrupt.
0	ESM_SOC_PIN_INT	R/W1C	0b	Latched status bit indicating that SOC ESM fault has been detected. Write 1 to clear interrupt.

3.1.103 STAT_BUCK1_2 Register (Offset = 0x6D) [Reset = 0x0]

STAT_BUCK1_2 is shown in Figure 3-103 and described in Table 3-105.

Return to the Table 3-1.

Figure 3-103. STAT_BUCK1_2 Register

7	6	5	4	3	2	1	0
BUCK2_ILIM_STAT	RESERVED	BUCK2_UV_STAT	BUCK2_OV_STAT	BUCK1_ILIM_STAT	RESERVED	BUCK1_UV_STAT	BUCK1_OV_STAT
R-0b	R-0b	R-0b	R-0b	R-0b	R-0b	R-0b	R-0b

Table 3-105. STAT_BUCK1_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	BUCK2_ILIM_STAT	R	0b	Status bit indicating that BUCK2 output current is above current limit level.
6	RESERVED	R	0b	
5	BUCK2_UV_STAT	R	0b	Status bit indicating that BUCK2 output voltage is below under-voltage threshold.
4	BUCK2_OV_STAT	R	0b	Status bit indicating that BUCK2 output voltage is above over-voltage threshold.
3	BUCK1_ILIM_STAT	R	0b	Status bit indicating that BUCK1 output current is above current limit level.
2	RESERVED	R	0b	
1	BUCK1_UV_STAT	R	0b	Status bit indicating that BUCK1 output voltage is below under-voltage threshold.
0	BUCK1_OV_STAT	R	0b	Status bit indicating that BUCK1 output voltage is above over-voltage threshold.

3.1.104 STAT_BUCK3_4 Register (Offset = 0x6E) [Reset = 0x0]

STAT_BUCK3_4 is shown in Figure 3-104 and described in Table 3-106.

Return to the Table 3-1.

Figure 3-104. STAT_BUCK3_4 Register

7	6	5	4	3	2	1	0
BUCK4_ILIM_STAT	RESERVED	BUCK4_UV_STAT	BUCK4_OV_STAT	BUCK3_ILIM_STAT	RESERVED	BUCK3_UV_STAT	BUCK3_OV_STAT
R-0b	R-0b	R-0b	R-0b	R-0b	R-0b	R-0b	R-0b

Table 3-106. STAT_BUCK3_4 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	BUCK4_ILIM_STAT	R	0b	Status bit indicating that BUCK4 output current is above current limit level.
6	RESERVED	R	0b	
5	BUCK4_UV_STAT	R	0b	Status bit indicating that BUCK4 output voltage is below under-voltage threshold.
4	BUCK4_OV_STAT	R	0b	Status bit indicating that BUCK4 output voltage is above over-voltage threshold.
3	BUCK3_ILIM_STAT	R	0b	Status bit indicating that BUCK3 output current is above current limit level.
2	RESERVED	R	0b	
1	BUCK3_UV_STAT	R	0b	Status bit indicating that BUCK3 output voltage is below under-voltage threshold.
0	BUCK3_OV_STAT	R	0b	Status bit indicating that BUCK3 output voltage is above over-voltage threshold.

3.1.105 STAT_BUCK5 Register (Offset = 0x6F) [Reset = 0x0]

STAT_BUCK5 is shown in Figure 3-105 and described in Table 3-107.

Return to the Table 3-1.

Figure 3-105. STAT_BUCK5 Register

7	6	5	4	3	2	1	0
RESERVED				BUCK5_ILIM_STAT	RESERVED	BUCK5_UV_STAT	BUCK5_OV_STAT
R-0b				R-0b	R-0b	R-0b	R-0b

Table 3-107. STAT_BUCK5 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	RESERVED	R	0b	
3	BUCK5_ILIM_STAT	R	0b	Status bit indicating that BUCK5 output current is above current limit level.
2	RESERVED	R	0b	
1	BUCK5_UV_STAT	R	0b	Status bit indicating that BUCK5 output voltage is below under-voltage threshold.
0	BUCK5_OV_STAT	R	0b	Status bit indicating that BUCK5 output voltage is above over-voltage threshold.

3.1.106 STAT_LDO1_2 Register (Offset = 0x70) [Reset = 0x0]

STAT_LDO1_2 is shown in Figure 3-106 and described in Table 3-108.

Return to the Table 3-1.

Figure 3-106. STAT_LDO1_2 Register

7	6	5	4	3	2	1	0
LDO2_ILIM_STAT	RESERVED	LDO2_UV_STAT	LDO2_OV_STAT	LDO1_ILIM_STAT	RESERVED	LDO1_UV_STAT	LDO1_OV_STAT
R-0b	R-0b	R-0b	R-0b	R-0b	R-0b	R-0b	R-0b

Table 3-108. STAT_LDO1_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	LDO2_ILIM_STAT	R	0b	Status bit indicating that LDO2 output current is above current limit level.
6	RESERVED	R	0b	
5	LDO2_UV_STAT	R	0b	Status bit indicating that LDO2 output voltage is below under-voltage threshold.
4	LDO2_OV_STAT	R	0b	Status bit indicating that LDO2 output voltage is above over-voltage threshold.
3	LDO1_ILIM_STAT	R	0b	Status bit indicating that LDO1 output current is above current limit level.
2	RESERVED	R	0b	
1	LDO1_UV_STAT	R	0b	Status bit indicating that LDO1 output voltage is below under-voltage threshold.
0	LDO1_OV_STAT	R	0b	Status bit indicating that LDO1 output voltage is above over-voltage threshold.

3.1.107 STAT_LDO3_4 Register (Offset = 0x71) [Reset = 0x0]

STAT_LDO3_4 is shown in Figure 3-107 and described in Table 3-109.

Return to the Table 3-1.

Figure 3-107. STAT_LDO3_4 Register

7	6	5	4	3	2	1	0
LDO4_ILIM_STAT	RESERVED	LDO4_UV_STAT	LDO4_OV_STAT	LDO3_ILIM_STAT	RESERVED	LDO3_UV_STAT	LDO3_OV_STAT
R-0b	R-0b	R-0b	R-0b	R-0b	R-0b	R-0b	R-0b

Table 3-109. STAT_LDO3_4 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	LDO4_ILIM_STAT	R	0b	Status bit indicating that LDO4 output current is above current limit level.
6	RESERVED	R	0b	
5	LDO4_UV_STAT	R	0b	Status bit indicating that LDO4 output voltage is below under-voltage threshold.
4	LDO4_OV_STAT	R	0b	Status bit indicating that LDO4 output voltage is above over-voltage threshold.
3	LDO3_ILIM_STAT	R	0b	Status bit indicating that LDO3 output current is above current limit level.
2	RESERVED	R	0b	
1	LDO3_UV_STAT	R	0b	Status bit indicating that LDO3 output voltage is below under-voltage threshold.
0	LDO3_OV_STAT	R	0b	Status bit indicating that LDO3 output voltage is above over-voltage threshold.

3.1.108 STAT_VMON Register (Offset = 0x72) [Reset = 0x0]

STAT_VMON is shown in Figure 3-108 and described in Table 3-110.

Return to the Table 3-1.

Figure 3-108. STAT_VMON Register

7	6	5	4	3	2	1	0
RESERVED						VCCA_UV_STAT	VCCA_OV_STAT
R-0b						R-0b	R-0b

Table 3-110. STAT_VMON Register Field Descriptions

Bit	Field	Type	Reset	Description
7:2	RESERVED	R	0b	
1	VCCA_UV_STAT	R	0b	Status bit indicating that VCCA input voltage is below under-voltage level.
0	VCCA_OV_STAT	R	0b	Status bit indicating that VCCA input voltage is above over-voltage level.

3.1.109 STAT_STARTUP Register (Offset = 0x73) [Reset = 0x0]

STAT_STARTUP is shown in Figure 3-109 and described in Table 3-111.

Return to the Table 3-1.

Figure 3-109. STAT_STARTUP Register

7	6	5	4	3	2	1	0
RESERVED						ENABLE_STAT	RESERVED
R-0b						R-0b	R-0b

Table 3-111. STAT_STARTUP Register Field Descriptions

Bit	Field	Type	Reset	Description
7:2	RESERVED	R	0b	
1	ENABLE_STAT	R	0b	Status bit indicating nPWRON / EN pin status
0	RESERVED	R	0b	

3.1.110 STAT_MISC Register (Offset = 0x74) [Reset = 0x0]

STAT_MISC is shown in Figure 3-110 and described in Table 3-112.

Return to the Table 3-1.

Figure 3-110. STAT_MISC Register

7	6	5	4	3	2	1	0
RESERVED				TWARN_STAT	RESERVED	EXT_CLK_STAT	RESERVED
R-0b				R-0b	R-0b	R-0b	R-0b

Table 3-112. STAT_MISC Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	RESERVED	R	0b	
3	TWARN_STAT	R	0b	Status bit indicating that die junction temperature is above the thermal warning level.
2	RESERVED	R	0b	
1	EXT_CLK_STAT	R	0b	Status bit indicating that external clock is not valid.
0	RESERVED	R	0b	

3.1.111 STAT_MODERATE_ERR Register (Offset = 0x75) [Reset = 0x0]

STAT_MODERATE_ERR is shown in Figure 3-111 and described in Table 3-113.

Return to the Table 3-1.

Figure 3-111. STAT_MODERATE_ERR Register

7	6	5	4	3	2	1	0
RESERVED							TSD_ORD_STAT
R-0b							R-0b

Table 3-113. STAT_MODERATE_ERR Register Field Descriptions

Bit	Field	Type	Reset	Description
7:1	RESERVED	R	0b	
0	TSD_ORD_STAT	R	0b	Status bit indicating that the die junction temperature is above the thermal level causing a sequenced shutdown.

3.1.112 STAT_SEVERE_ERR Register (Offset = 0x76) [Reset = 0x0]

STAT_SEVERE_ERR is shown in Figure 3-112 and described in Table 3-114.

Return to the Table 3-1.

Figure 3-112. STAT_SEVERE_ERR Register

7	6	5	4	3	2	1	0
RESERVED						VCCA_OVP_STAT	TSD_IMM_STAT
R-0b						R-0b	R-0b

Table 3-114. STAT_SEVERE_ERR Register Field Descriptions

Bit	Field	Type	Reset	Description
7:2	RESERVED	R	0b	
1	VCCA_OVP_STAT	R	0b	Status bit indicating that the VCCA voltage is above overvoltage protection level.
0	TSD_IMM_STAT	R	0b	Status bit indicating that the die junction temperature is above the thermal level causing an immediate shutdown.

3.1.113 STAT_READBACK_ERR Register (Offset = 0x77) [Reset = 0x0]

STAT_READBACK_ERR is shown in Figure 3-113 and described in Table 3-115.

Return to the Table 3-1.

Figure 3-113. STAT_READBACK_ERR Register

7	6	5	4	3	2	1	0
RESERVED				NRSTOUT_SOC_READBACK_STAT	NRSTOUT_READBACK_STAT	NINT_READBACK_STAT	EN_DRV_READBACK_STAT
R-0b				R-0b	R-0b	R-0b	R-0b

Table 3-115. STAT_READBACK_ERR Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	RESERVED	R	0b	
3	NRSTOUT_SOC_READBACK_STAT	R	0b	Status bit indicating that NRSTOUT_SOC pin output is high and device is driving it low.
2	NRSTOUT_READBACK_STAT	R	0b	Status bit indicating that NRSTOUT pin output is high and device is driving it low.
1	NINT_READBACK_STAT	R	0b	Status bit indicating that NINT pin output is high and device is driving it low.
0	EN_DRV_READBACK_STAT	R	0b	Status bit indicating that EN_DRV pin output is different than driven.

3.1.114 PGOOD_SEL_1 Register (Offset = 0x78) [Reset = 0x0]

PGOOD_SEL_1 is shown in Figure 3-114 and described in Table 3-116.

Return to the Table 3-1.

Figure 3-114. PGOOD_SEL_1 Register

7	6	5	4	3	2	1	0
PGOOD_SEL_BUCK4		PGOOD_SEL_BUCK3		PGOOD_SEL_BUCK2		PGOOD_SEL_BUCK1	
R/W-0b		R/W-0b		R/W-0b		R/W-0b	

Table 3-116. PGOOD_SEL_1 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	PGOOD_SEL_BUCK4	R/W	0b	PGOOD signal source control from BUCK4 (Default from NVM memory) 0b = Masked 1b = Powergood threshold voltage 10b = Powergood threshold voltage AND current limit 11b = Powergood threshold voltage AND current limit
5:4	PGOOD_SEL_BUCK3	R/W	0b	PGOOD signal source control from BUCK3 (Default from NVM memory) 0b = Masked 1b = Powergood threshold voltage 10b = Powergood threshold voltage AND current limit 11b = Powergood threshold voltage AND current limit
3:2	PGOOD_SEL_BUCK2	R/W	0b	PGOOD signal source control from BUCK2 (Default from NVM memory) 0b = Masked 1b = Powergood threshold voltage 10b = Powergood threshold voltage AND current limit 11b = Powergood threshold voltage AND current limit
1:0	PGOOD_SEL_BUCK1	R/W	0b	PGOOD signal source control from BUCK1 (Default from NVM memory) 0b = Masked 1b = Powergood threshold voltage 10b = Powergood threshold voltage AND current limit 11b = Powergood threshold voltage AND current limit

3.1.115 PGOOD_SEL_2 Register (Offset = 0x79) [Reset = 0x0]

PGOOD_SEL_2 is shown in Figure 3-115 and described in Table 3-117.

Return to the Table 3-1.

Figure 3-115. PGOOD_SEL_2 Register

7	6	5	4	3	2	1	0
RESERVED						PGOOD_SEL_BUCK5	
R/W-0b						R/W-0b	

Table 3-117. PGOOD_SEL_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:2	RESERVED	R/W	0b	
1:0	PGOOD_SEL_BUCK5	R/W	0b	PGOOD signal source control from BUCK5 (Default from NVM memory) 0b = Masked 1b = Powergood threshold voltage 10b = Powergood threshold voltage AND current limit 11b = Powergood threshold voltage AND current limit

3.1.116 PGOOD_SEL_3 Register (Offset = 0x7A) [Reset = 0x0]

PGOOD_SEL_3 is shown in Figure 3-116 and described in Table 3-118.

Return to the Table 3-1.

Figure 3-116. PGOOD_SEL_3 Register

7	6	5	4	3	2	1	0
PGOOD_SEL_LDO4		PGOOD_SEL_LDO3		PGOOD_SEL_LDO2		PGOOD_SEL_LDO1	
R/W-0b		R/W-0b		R/W-0b		R/W-0b	

Table 3-118. PGOOD_SEL_3 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	PGOOD_SEL_LDO4	R/W	0b	PGOOD signal source control from LDO4 (Default from NVM memory) 0b = Masked 1b = Powergood threshold voltage 10b = Powergood threshold voltage AND current limit 11b = Powergood threshold voltage AND current limit
5:4	PGOOD_SEL_LDO3	R/W	0b	PGOOD signal source control from LDO3 (Default from NVM memory) 0b = Masked 1b = Powergood threshold voltage 10b = Powergood threshold voltage AND current limit 11b = Powergood threshold voltage AND current limit
3:2	PGOOD_SEL_LDO2	R/W	0b	PGOOD signal source control from LDO2 (Default from NVM memory) 0b = Masked 1b = Powergood threshold voltage 10b = Powergood threshold voltage AND current limit 11b = Powergood threshold voltage AND current limit
1:0	PGOOD_SEL_LDO1	R/W	0b	PGOOD signal source control from LDO1 (Default from NVM memory) 0b = Masked 1b = Powergood threshold voltage 10b = Powergood threshold voltage AND current limit 11b = Powergood threshold voltage AND current limit

3.1.117 PGOOD_SEL_4 Register (Offset = 0x7B) [Reset = 0x0]

PGOOD_SEL_4 is shown in Figure 3-117 and described in Table 3-119.

Return to the Table 3-1.

Figure 3-117. PGOOD_SEL_4 Register

7	6	5	4	3	2	1	0
PGOOD_WINDOW	PGOOD_POL	PGOOD_SEL_NRSTOUT_SOC	PGOOD_SEL_NRSTOUT	PGOOD_SEL_TDIE_WARN	RESERVED		PGOOD_SEL_VCCA
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b		R/W-0b

Table 3-119. PGOOD_SEL_4 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	PGOOD_WINDOW	R/W	0b	Type of voltage monitoring for PGOOD signal: (Default from NVM memory) 0b = Only undervoltage is monitored 1b = Both undervoltage and overvoltage are monitored
6	PGOOD_POL	R/W	0b	PGOOD signal polarity select: (Default from NVM memory) 0b = PGOOD signal is high when monitored inputs are valid 1b = PGOOD signal is low when monitored inputs are valid
5	PGOOD_SEL_NRSTOUT_SOC	R/W	0b	PGOOD signal source control from nRSTOUT_SOC pin: (Default from NVM memory) 0b = Masked 1b = nRSTOUT_SOC pin low state forces PGOOD signal to low
4	PGOOD_SEL_NRSTOUT	R/W	0b	PGOOD signal source control from nRSTOUT pin: (Default from NVM memory) 0b = Masked 1b = nRSTOUT pin low state forces PGOOD signal to low
3	PGOOD_SEL_TDIE_WARN	R/W	0b	PGOOD signal source control from thermal warning (Default from NVM memory) 0b = Masked 1b = Thermal warning affecting to PGOOD signal
2:1	RESERVED	R/W	0b	
0	PGOOD_SEL_VCCA	R/W	0b	PGOOD signal source control from VCCA monitoring (Default from NVM memory) 0b = Masked 1b = VCCA OV/UV threshold affecting PGOOD signal

3.1.118 PLL_CTRL Register (Offset = 0x7C) [Reset = 0x0]

PLL_CTRL is shown in Figure 3-118 and described in Table 3-120.

Return to the Table 3-1.

Figure 3-118. PLL_CTRL Register

7	6	5	4	3	2	1	0
RESERVED						EXT_CLK_FREQ	
R/W-0b						R/W-0b	

Table 3-120. PLL_CTRL Register Field Descriptions

Bit	Field	Type	Reset	Description
7:2	RESERVED	R/W	0b	
1:0	EXT_CLK_FREQ	R/W	0b	Frequency of the external clock (SYNCCLKIN): See electrical specification for input clock frequency tolerance. (Default from NVM memory) 0b = 1.1 MHz 1b = 2.2 MHz 10b = 4.4 MHz 11b = Reserved

3.1.119 CONFIG_1 Register (Offset = 0x7D) [Reset = 0xC0]

CONFIG_1 is shown in Figure 3-119 and described in Table 3-121.

Return to the Table 3-1.

Figure 3-119. CONFIG_1 Register

7	6	5	4	3	2	1	0
NSLEEP2_MASK	NSLEEP1_MASK	EN_ILIM_FSM_CTRL	I2C2_HS	I2C1_HS	RESERVED	TSD_ORD_LEVEL	TWARN_LEVEL
R/W-1b	R/W-1b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-121. CONFIG_1 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	NSLEEP2_MASK	R/W	1b	Masking for NSLEEP2 pin(s) and NSLEEP2B bit: (Default from NVM memory) 0b = NSLEEP2(B) affects FSM state transitions. 1b = NSLEEP2(B) does not affect FSM state transitions.
6	NSLEEP1_MASK	R/W	1b	Masking for NSLEEP1 pin(s) and NSLEEP1B bit: (Default from NVM memory) 0b = NSLEEP1(B) affects FSM state transitions. 1b = NSLEEP1(B) does not affect FSM state transitions.
5	EN_ILIM_FSM_CTRL	R/W	0b	(Default from NVM memory) 0b = Buck/LDO regulators ILIM interrupts do not affect FSM triggers. 1b = Buck/LDO regulators ILIM interrupts affect FSM triggers.
4	I2C2_HS	R/W	0b	Select I2C2 speed (input filter) (Default from NVM memory) 0b = Standard, fast or fast+ by default, can be set to Hs-mode by Hs-mode master code. 1b = Forced to Hs-mode
3	I2C1_HS	R/W	0b	Select I2C1 speed (input filter) (Default from NVM memory) 0b = Standard, fast or fast+ by default, can be set to Hs-mode by Hs-mode master code. 1b = Forced to Hs-mode
2	RESERVED	R/W	0b	
1	TSD_ORD_LEVEL	R/W	0b	Thermal shutdown threshold level. (Default from NVM memory) 0b = 140C 1b = 145C
0	TWARN_LEVEL	R/W	0b	Thermal warning threshold level. (Default from NVM memory) 0b = 130C 1b = 140C

3.1.120 CONFIG_2 Register (Offset = 0x7E) [Reset = 0x0]

CONFIG_2 is shown in Figure 3-120 and described in Table 3-122.

Return to the Table 3-1.

Figure 3-120. CONFIG_2 Register

7	6	5	4	3	2	1	0
BB_EOC_RDY	RESERVED			BB_VEOC		BB_ICHR	BB_CHARGER_EN
R-0b	R/W-0b			R/W-0b		R/W-0b	R/W-0b

Table 3-122. CONFIG_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	BB_EOC_RDY	R	0b	Backup end of charge indication 0b = Charging active or not enabled 1b = Charger has reached termination voltage set by BB_VEOC register
6:4	RESERVED	R/W	0b	
3:2	BB_VEOC	R/W	0b	End of charge voltage for backup battery charger: (Default from NVM memory) 0b = 2.5V 1b = 2.8V 10b = 3.0V 11b = 3.3V
1	BB_ICHR	R/W	0b	Backup battery charging current: (Default from NVM memory) 0b = 100uA 1b = 500uA
0	BB_CHARGER_EN	R/W	0b	Backup battery charging: 0b = Disabled 1b = Enabled

3.1.121 ENABLE_DRV_REG Register (Offset = 0x80) [Reset = 0x0]

ENABLE_DRV_REG is shown in Figure 3-121 and described in Table 3-123.

Return to the Table 3-1.

Figure 3-121. ENABLE_DRV_REG Register

7	6	5	4	3	2	1	0
RESERVED							ENABLE_DRV
R/W-0b							R/W-0b

Table 3-123. ENABLE_DRV_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:1	RESERVED	R/W	0b	
0	ENABLE_DRV	R/W	0b	Control for EN_DRV pin: FORCE_EN_DRV_LOW must be 0 to control EN_DRV pin. Otherwise EN_DRV pin is low. 0b = Low 1b = High

3.1.122 MISC_CTRL Register (Offset = 0x81) [Reset = 0x0]

MISC_CTRL is shown in Figure 3-122 and described in Table 3-124.

Return to the Table 3-1.

Figure 3-122. MISC_CTRL Register

7	6	5	4	3	2	1	0
SYNCCLKOUT_FREQ_SEL		SEL_EXT_CLK	AMUXOUT_EN	CLKMON_EN	LPM_EN	NRSTOUT_SOC	NRSTOUT
R/W-0b		R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-124. MISC_CTRL Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	SYNCCLKOUT_FREQ_SEL	R/W	0b	SYNCCLKOUT enable/frequency select: 0b = SYNCCLKOUT off 1b = 1.1 MHz 10b = 2.2 MHz 11b = 4.4 MHz
5	SEL_EXT_CLK	R/W	0b	Selection of external clock: 0b = Forced to internal RC oscillator. 1b = Automatic external clock used when available, interrupt is generated if the external clock is expected (SEL_EXT_CLK = 1), but it is not available or the clock frequency is not within the valid range.
4	AMUXOUT_EN	R/W	0b	Control bandgap voltage to AMUXOUT pin. 0b = Disabled 1b = Enabled
3	CLKMON_EN	R/W	0b	Control of internal clock monitoring. 0b = Disabled 1b = Enabled
2	LPM_EN	R/W	0b	Low power mode control. LPM_EN sets device in a low power mode. Intended use case is for the PFSM to set LPM_EN upon entering a deep sleep state. The end objective is to disable the digital oscillator to reduce power consumption. The following functions are disabled when LPM_EN=1. -TSD cycling of all sensors/thresholds -regmap/SRAM CRC continuous checking -SPMI WD NVM_ID request/response polling -Disable clock monitoring 0b = Low power mode disabled 1b = Low power mode enabled
1	NRSTOUT_SOC	R/W	0b	Control for nRSTOUT_SOC signal: 0b = Low 1b = High
0	NRSTOUT	R/W	0b	Control for nRSTOUT signal: 0b = Low 1b = High

3.1.123 ENABLE_DRV_STAT Register (Offset = 0x82) [Reset = 0x8]

ENABLE_DRV_STAT is shown in Figure 3-123 and described in Table 3-125.

Return to the Table 3-1.

Figure 3-123. ENABLE_DRV_STAT Register

7	6	5	4	3	2	1	0
RESERVED			SPMI_LPM_EN	FORCE_EN_DRV_LOW	NRSTOUT_SOC_IN	NRSTOUT_IN	EN_DRV_IN
R/W-0b			R/W-0b	R/W-1b	R-0b	R-0b	R-0b

Table 3-125. ENABLE_DRV_STAT Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	RESERVED	R/W	0b	
4	SPMI_LPM_EN	R/W	0b	This bit is read/write for PFSM and read-only for I2C/SPI SPMI low power mode control. SPMI_LPM_EN sets SPMI in a low power mode which stops SPMI WD (Bus heartbeat). PMICs should enter SPMI_LPM_EN=1 at similar times to prevent SPMI WD failures. Therefore to mitigate clock variations SPMI_LPM_EN=1 should be done early in the sequence. The following functions are disabled when SPMI_LPM_EN=1. -SPMI WD NVM_ID request/response polling 0b = SPMI low power mode disabled 1b = SPMI low power mode enabled
3	FORCE_EN_DRV_LOW	R/W	1b	This bit is read/write for PFSM and read-only for I2C/SPI 0b = ENABLE_DRV bit can be written by I2C/SPI 1b = ENABLE_DRV bit is forced low and cannot be written high by I2C/SPI
2	NRSTOUT_SOC_IN	R	0b	Level of NRSTOUT_SOC pin: 0b = Low 1b = High
1	NRSTOUT_IN	R	0b	Level of NRSTOUT pin: 0b = Low 1b = High
0	EN_DRV_IN	R	0b	Level of EN_DRV pin: 0b = Low 1b = High

3.1.124 RECOV_CNT_REG_1 Register (Offset = 0x83) [Reset = 0x0]

RECOV_CNT_REG_1 is shown in Figure 3-124 and described in Table 3-126.

Return to the Table 3-1.

Figure 3-124. RECOV_CNT_REG_1 Register

7	6	5	4	3	2	1	0
RESERVED				RECOV_CNT			
R-0b				R-0b			

Table 3-126. RECOV_CNT_REG_1 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	RESERVED	R	0b	
3:0	RECOV_CNT	R	0b	Recovery counter status. Counter value is incremented each time PMIC goes through warm reset.

3.1.125 RECOV_CNT_REG_2 Register (Offset = 0x84) [Reset = 0x0]

RECOV_CNT_REG_2 is shown in Figure 3-125 and described in Table 3-127.

Return to the Table 3-1.

Figure 3-125. RECOV_CNT_REG_2 Register

7	6	5	4	3	2	1	0
RESERVED			RECOV_CNT_CLR	RECOV_CNT_THR			
R/W-0b			R/WSelfClrF-0b	R/W-0b			

Table 3-127. RECOV_CNT_REG_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	RESERVED	R/W	0b	
4	RECOV_CNT_CLR	R/WSelfClrF	0b	Recovery counter clear. Write 1 to clear the counter. This bit is automatically set back to 0.
3:0	RECOV_CNT_THR	R/W	0b	Recovery counter threshold value for immediate power-down of all supply rails. (Default from NVM memory)

3.1.126 FSM_I2C_TRIGGERS Register (Offset = 0x85) [Reset = 0x0]

FSM_I2C_TRIGGERS is shown in Figure 3-126 and described in Table 3-128.

Return to the Table 3-1.

Figure 3-126. FSM_I2C_TRIGGERS Register

7	6	5	4	3	2	1	0
TRIGGER_I2C_7	TRIGGER_I2C_6	TRIGGER_I2C_5	TRIGGER_I2C_4	TRIGGER_I2C_3	TRIGGER_I2C_2	TRIGGER_I2C_1	TRIGGER_I2C_0
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/WSelfClrF-0b	R/WSelfClrF-0b	R/WSelfClrF-0b	R/WSelfClrF-0b

Table 3-128. FSM_I2C_TRIGGERS Register Field Descriptions

Bit	Field	Type	Reset	Description
7	TRIGGER_I2C_7	R/W	0b	Trigger for PFSM program.
6	TRIGGER_I2C_6	R/W	0b	Trigger for PFSM program.
5	TRIGGER_I2C_5	R/W	0b	Trigger for PFSM program.
4	TRIGGER_I2C_4	R/W	0b	Trigger for PFSM program.
3	TRIGGER_I2C_3	R/WSelfClrF	0b	Trigger for PFSM program. This bit is automatically cleared. Writing this bit 1 creates PFSM trigger pulse.
2	TRIGGER_I2C_2	R/WSelfClrF	0b	Trigger for PFSM program. This bit is automatically cleared. Writing this bit 1 creates PFSM trigger pulse.
1	TRIGGER_I2C_1	R/WSelfClrF	0b	Trigger for PFSM program. This bit is automatically cleared. Writing this bit 1 creates PFSM trigger pulse.
0	TRIGGER_I2C_0	R/WSelfClrF	0b	Trigger for PFSM program. This bit is automatically cleared. Writing this bit 1 creates PFSM trigger pulse.

3.1.127 FSM_NSLEEP_TRIGGERS Register (Offset = 0x86) [Reset = 0x0]

FSM_NSLEEP_TRIGGERS is shown in Figure 3-127 and described in Table 3-129.

Return to the Table 3-1.

Figure 3-127. FSM_NSLEEP_TRIGGERS Register

7	6	5	4	3	2	1	0
RESERVED						NSLEEP2B	NSLEEP1B
R/W-0b						R/W-0b	R/W-0b

Table 3-129. FSM_NSLEEP_TRIGGERS Register Field Descriptions

Bit	Field	Type	Reset	Description
7:2	RESERVED	R/W	0b	
1	NSLEEP2B	R/W	0b	Parallel register bit for NSLEEP2 function: 0b = NSLEEP2 low 1b = NSLEEP2 high
0	NSLEEP1B	R/W	0b	Parallel register bit for NSLEEP1 function: 0b = NSLEEP1 low 1b = NSLEEP1 high

3.1.128 BUCK_RESET_REG Register (Offset = 0x87) [Reset = 0x0]

BUCK_RESET_REG is shown in Figure 3-128 and described in Table 3-130.

Return to the Table 3-1.

Figure 3-128. BUCK_RESET_REG Register

7	6	5	4	3	2	1	0
RESERVED			BUCK5_RESET	BUCK4_RESET	BUCK3_RESET	BUCK2_RESET	BUCK1_RESET
R/W-0b			R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-130. BUCK_RESET_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	RESERVED	R/W	0b	
4	BUCK5_RESET	R/W	0b	Reset signal for Buck logic. Warning: This bit is for debug only. DO NOT SET THIS BIT TO "1" DURING DEVICE OPERATION.
3	BUCK4_RESET	R/W	0b	Reset signal for Buck logic. Warning: This bit is for debug only. DO NOT SET THIS BIT TO "1" DURING DEVICE OPERATION.
2	BUCK3_RESET	R/W	0b	Reset signal for Buck logic. Warning: This bit is for debug only. DO NOT SET THIS BIT TO "1" DURING DEVICE OPERATION.
1	BUCK2_RESET	R/W	0b	Reset signal for Buck logic. Warning: This bit is for debug only. DO NOT SET THIS BIT TO "1" DURING DEVICE OPERATION.
0	BUCK1_RESET	R/W	0b	Reset signal for Buck logic. Warning: This bit is for debug only. DO NOT SET THIS BIT TO "1" DURING DEVICE OPERATION.

3.1.129 SPREAD_SPECTRUM_1 Register (Offset = 0x88) [Reset = 0x0]

SPREAD_SPECTRUM_1 is shown in Figure 3-129 and described in Table 3-131.

Return to the Table 3-1.

Figure 3-129. SPREAD_SPECTRUM_1 Register

7	6	5	4	3	2	1	0
RESERVED					SS_EN	SS_DEPTH	
R/W-0b					R/W-0b	R/W-0b	

Table 3-131. SPREAD_SPECTRUM_1 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:3	RESERVED	R/W	0b	
2	SS_EN	R/W	0b	Spread spectrum enable. (Default from NVM memory) 0b = Spread spectrum disabled 1b = Spread spectrum enabled
1:0	SS_DEPTH	R/W	0b	Spread spectrum modulation depth. (Default from NVM memory) 0b = No modulation 1b = +/- 6.3% 10b = +/- 8.4% 11b = RESERVED

3.1.130 FREQ_SEL Register (Offset = 0x8A) [Reset = 0x0]

FREQ_SEL is shown in Figure 3-130 and described in Table 3-132.

Return to the Table 3-1.

Figure 3-130. FREQ_SEL Register

7	6	5	4	3	2	1	0
RESERVED			BUCK5_FREQ_SEL	BUCK4_FREQ_SEL	BUCK3_FREQ_SEL	BUCK2_FREQ_SEL	BUCK1_FREQ_SEL
R/W-0b			R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-132. FREQ_SEL Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	RESERVED	R/W	0b	
4	BUCK5_FREQ_SEL	R/W	0b	Buck5 switching frequency: This bit is Read/Write or Read-Only for I2C/SPI access depending on EEPROM configuration. See TRM for details. (Default from NVM memory) 0b = 2.2 MHz 1b = 4.4 MHz
3	BUCK4_FREQ_SEL	R/W	0b	Buck4 switching frequency: This bit is Read/Write or Read-Only for I2C/SPI access depending on EEPROM configuration. See TRM for details. (Default from NVM memory) 0b = 2.2 MHz 1b = 4.4 MHz
2	BUCK3_FREQ_SEL	R/W	0b	Buck3 switching frequency: This bit is Read/Write or Read-Only for I2C/SPI access depending on EEPROM configuration. See TRM for details. (Default from NVM memory) 0b = 2.2 MHz 1b = 4.4 MHz
1	BUCK2_FREQ_SEL	R/W	0b	Buck2 switching frequency: This bit is Read/Write or Read-Only for I2C/SPI access depending on EEPROM configuration. See TRM for details. (Default from NVM memory) 0b = 2.2 MHz 1b = 4.4 MHz
0	BUCK1_FREQ_SEL	R/W	0b	Buck1 switching frequency: This bit is Read/Write or Read-Only for I2C/SPI access depending on EEPROM configuration. See TRM for details. (Default from NVM memory) 0b = 2.2 MHz 1b = 4.4 MHz

3.1.131 FSM_STEP_SIZE Register (Offset = 0x8B) [Reset = 0x0]

FSM_STEP_SIZE is shown in Figure 3-131 and described in Table 3-133.

Return to the Table 3-1.

Figure 3-131. FSM_STEP_SIZE Register

7	6	5	4	3	2	1	0
RESERVED			PFSM_DELAY_STEP				
R/W-0b			R/W-0b				

Table 3-133. FSM_STEP_SIZE Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	RESERVED	R/W	0b	
4:0	PFSM_DELAY_STEP	R/W	0b	Step size for PFSM sequence counter. Step size is 50ns * $2^{PFSM\_DELAY\_STEP}$. (Default from NVM memory)

3.1.132 LDO_RV_TIMEOUT_REG_1 Register (Offset = 0x8C) [Reset = 0x0]

LDO_RV_TIMEOUT_REG_1 is shown in Figure 3-132 and described in Table 3-134.

Return to the Table 3-1.

Figure 3-132. LDO_RV_TIMEOUT_REG_1 Register

7	6	5	4	3	2	1	0
LDO2_RV_TIMEOUT				LDO1_RV_TIMEOUT			
R/W-0b				R/W-0b			

Table 3-134. LDO_RV_TIMEOUT_REG_1 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	LDO2_RV_TIMEOUT	R/W	0b	LDO residual voltage check timeout select. (Default from NVM memory) 0b = 0.5ms 1b = 1ms 10b = 1.5ms 11b = 2ms 100b = 2.5ms 101b = 3ms 110b = 3.5ms 111b = 4ms 1000b = 2ms 1001b = 4ms 1010b = 6ms 1011b = 8ms 1100b = 10ms 1101b = 12ms 1110b = 14ms 1111b = 16ms
3:0	LDO1_RV_TIMEOUT	R/W	0b	LDO residual voltage check timeout select. (Default from NVM memory) 0b = 0.5ms 1b = 1ms 10b = 1.5ms 11b = 2ms 100b = 2.5ms 101b = 3ms 110b = 3.5ms 111b = 4ms 1000b = 2ms 1001b = 4ms 1010b = 6ms 1011b = 8ms 1100b = 10ms 1101b = 12ms 1110b = 14ms 1111b = 16ms

3.1.133 LDO_RV_TIMEOUT_REG_2 Register (Offset = 0x8D) [Reset = 0x0]

LDO_RV_TIMEOUT_REG_2 is shown in Figure 3-133 and described in Table 3-135.

Return to the Table 3-1.

Figure 3-133. LDO_RV_TIMEOUT_REG_2 Register

7	6	5	4	3	2	1	0
LDO4_RV_TIMEOUT				LDO3_RV_TIMEOUT			
R/W-0b				R/W-0b			

Table 3-135. LDO_RV_TIMEOUT_REG_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	LDO4_RV_TIMEOUT	R/W	0b	LDO residual voltage check timeout select. (Default from NVM memory) 0b = 0.5ms 1b = 1ms 10b = 1.5ms 11b = 2ms 100b = 2.5ms 101b = 3ms 110b = 3.5ms 111b = 4ms 1000b = 2ms 1001b = 4ms 1010b = 6ms 1011b = 8ms 1100b = 10ms 1101b = 12ms 1110b = 14ms 1111b = 16ms
3:0	LDO3_RV_TIMEOUT	R/W	0b	LDO residual voltage check timeout select. (Default from NVM memory) 0b = 0.5ms 1b = 1ms 10b = 1.5ms 11b = 2ms 100b = 2.5ms 101b = 3ms 110b = 3.5ms 111b = 4ms 1000b = 2ms 1001b = 4ms 1010b = 6ms 1011b = 8ms 1100b = 10ms 1101b = 12ms 1110b = 14ms 1111b = 16ms

3.1.134 USER_SPARE_REGS Register (Offset = 0x8E) [Reset = 0x0]

USER_SPARE_REGS is shown in Figure 3-134 and described in Table 3-136.

Return to the Table 3-1.

Figure 3-134. USER_SPARE_REGS Register

7	6	5	4	3	2	1	0
RESERVED				USER_SPARE_4	USER_SPARE_3	USER_SPARE_2	USER_SPARE_1
R/W-0b				R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-136. USER_SPARE_REGS Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	RESERVED	R/W	0b	
3	USER_SPARE_4	R/W	0b	(Default from NVM memory)
2	USER_SPARE_3	R/W	0b	(Default from NVM memory)
1	USER_SPARE_2	R/W	0b	(Default from NVM memory)
0	USER_SPARE_1	R/W	0b	(Default from NVM memory)

3.1.135 ESM_MCU_START_REG Register (Offset = 0x8F) [Reset = 0x0]

ESM_MCU_START_REG is shown in Figure 3-135 and described in Table 3-137.

Return to the Table 3-1.

Figure 3-135. ESM_MCU_START_REG Register

7	6	5	4	3	2	1	0
RESERVED							ESM_MCU_START
R/W-0b							R/W-0b

Table 3-137. ESM_MCU_START_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:1	RESERVED	R/W	0b	
0	ESM_MCU_START	R/W	0b	Control bit to start the ESM_MCU: 0b = ESM_MCU not started. Device clears ENABLE_DRV bit when bit ESM_MCU_EN=1 1b = ESM_MCU started.

3.1.136 ESM_MCU_DELAY1_REG Register (Offset = 0x90) [Reset = 0x0]

ESM_MCU_DELAY1_REG is shown in Figure 3-136 and described in Table 3-138.

Return to the Table 3-1.

Figure 3-136. ESM_MCU_DELAY1_REG Register

7	6	5	4	3	2	1	0
ESM_MCU_DELAY1							
R/W-0b							

Table 3-138. ESM_MCU_DELAY1_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	ESM_MCU_DELAY1	R/W	0b	These bits configure the duration of the ESM_MCU delay-1 time-interval (see Error Signal Monitor chapter). These bits can be only be written when control bit ESM_MCU_START=0.

3.1.137 ESM_MCU_DELAY2_REG Register (Offset = 0x91) [Reset = 0x0]

ESM_MCU_DELAY2_REG is shown in Figure 3-137 and described in Table 3-139.

Return to the Table 3-1.

Figure 3-137. ESM_MCU_DELAY2_REG Register

7	6	5	4	3	2	1	0
ESM_MCU_DELAY2							
R/W-0b							

Table 3-139. ESM_MCU_DELAY2_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	ESM_MCU_DELAY2	R/W	0b	These bits configure the duration of the ESM_MCU delay-2 time-interval (see Error Signal Monitor chapter). These bits can be only be written when control bit ESM_MCU_START=0.

3.1.138 ESM_MCU_MODE_CFG Register (Offset = 0x92) [Reset = 0x0]

ESM_MCU_MODE_CFG is shown in Figure 3-138 and described in Table 3-140.

Return to the Table 3-1.

Figure 3-138. ESM_MCU_MODE_CFG Register

7	6	5	4	3	2	1	0
ESM_MCU_MODE	ESM_MCU_EN	ESM_MCU_ENDRV	RESERVED	ESM_MCU_ERR_CNT_TH			
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b			

Table 3-140. ESM_MCU_MODE_CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
7	ESM_MCU_MODE	R/W	0b	This bit selects the mode for the ESM_MCU: These bits can be only be written when control bit ESM_MCU_START=0. 0b = Level Mode 1b = PWM Mode
6	ESM_MCU_EN	R/W	0b	ESM_MCU enable configuration bit: These bits can be only be written when control bit ESM_MCU_START=0. 0b = ESM_MCU disabled. MCU can set ENABLE_DRV bit to 1 if all other interrupt bits are cleared 1b = ESM_MCU enabled. MCU can set ENABLE_DRV bit to 1 if: - bit ESM_MCU_START=1, and - (ESM_MCU_FAIL_INT=0 or ESM_MCU_ENDRV=0), and - ESM_MCU_RST_INT=0, and - all other interrupt bits are cleared
5	ESM_MCU_ENDRV	R/W	0b	Configuration bit to select ENABLE_DRV clear on ESM-error for ESM_MCU: These bits can be only be written when control bit ESM_MCU_START=0. 0b = ENABLE_DRV not cleared when ESM_MCU_FAIL_INT=1 1b = ENABLE_DRV cleared when ESM_MCU_FAIL_INT=1
4	RESERVED	R/W	0b	
3:0	ESM_MCU_ERR_CNT_TH	R/W	0b	Configuration bits for the threshold of the ESM_MCU error-counter. The ESM_MCU starts the Error Handling Procedure (see Error Signal Monitor chapter) if ESM_MCU_ERR_CNT[4:0] > ESM_MCU_ERR_CNT_TH[3:0]. These bits can be only be written when control bit ESM_MCU_START=0.

3.1.139 ESM_MCU_HMAX_REG Register (Offset = 0x93) [Reset = 0x0]

ESM_MCU_HMAX_REG is shown in Figure 3-139 and described in Table 3-141.

Return to the Table 3-1.

Figure 3-139. ESM_MCU_HMAX_REG Register

7	6	5	4	3	2	1	0
ESM_MCU_HMAX							
R/W-0b							

Table 3-141. ESM_MCU_HMAX_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	ESM_MCU_HMAX	R/W	0b	These bits configure the the maximum high-pulse time-threshold (tHIGH_MAX_TH) for ESM_MCU (see Error Signal Monitor chapter). These bits can be only be written when control bit ESM_MCU_START=0.

3.1.140 ESM_MCU_HMIN_REG Register (Offset = 0x94) [Reset = 0x0]

ESM_MCU_HMIN_REG is shown in Figure 3-140 and described in Table 3-142.

Return to the Table 3-1.

Figure 3-140. ESM_MCU_HMIN_REG Register

7	6	5	4	3	2	1	0
ESM_MCU_HMIN							
R/W-0b							

Table 3-142. ESM_MCU_HMIN_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	ESM_MCU_HMIN	R/W	0b	These bits configure the the minimum high-pulse time-threshold (tHIGH_MIN_TH) for ESM_MCU (see Error Signal Monitor chapter). These bits can be only be written when control bit ESM_MCU_START=0.

3.1.141 ESM_MCU_LMAX_REG Register (Offset = 0x95) [Reset = 0x0]

ESM_MCU_LMAX_REG is shown in Figure 3-141 and described in Table 3-143.

Return to the Table 3-1.

Figure 3-141. ESM_MCU_LMAX_REG Register

7	6	5	4	3	2	1	0
ESM_MCU_LMAX							
R/W-0b							

Table 3-143. ESM_MCU_LMAX_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	ESM_MCU_LMAX	R/W	0b	These bits configure the the maximum low-pulse time-threshold (tLOW_MAX_TH) for ESM_MCU (see Error Signal Monitor chapter). These bits can be only be written when control bit ESM_MCU_START=0.

3.1.142 ESM_MCU_LMIN_REG Register (Offset = 0x96) [Reset = 0x0]

ESM_MCU_LMIN_REG is shown in Figure 3-142 and described in Table 3-144.

Return to the Table 3-1.

Figure 3-142. ESM_MCU_LMIN_REG Register

7	6	5	4	3	2	1	0
ESM_MCU_LMIN							
R/W-0b							

Table 3-144. ESM_MCU_LMIN_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	ESM_MCU_LMIN	R/W	0b	These bits configure the the minimum low-pulse time-threshold (tLOW_MAX_TH) for ESM_MCU (see Error Signal Monitor chapter). These bits can be only be written when control bit ESM_MCU_START=0.

3.1.143 ESM_MCU_ERR_CNT_REG Register (Offset = 0x97) [Reset = 0x0]

ESM_MCU_ERR_CNT_REG is shown in Figure 3-143 and described in Table 3-145.

Return to the Table 3-1.

Figure 3-143. ESM_MCU_ERR_CNT_REG Register

7	6	5	4	3	2	1	0
RESERVED			ESM_MCU_ERR_CNT				
R-0b			R-0b				

Table 3-145. ESM_MCU_ERR_CNT_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	RESERVED	R	0b	
4:0	ESM_MCU_ERR_CNT	R	0b	Status bits to indicate the value of the ESM_MCU Error-Counter. The device clears these bits when ESM_MCU_START bit is 0, or when the device resets the MCU.

3.1.144 ESM_SOC_START_REG Register (Offset = 0x98) [Reset = 0x0]

ESM_SOC_START_REG is shown in Figure 3-144 and described in Table 3-146.

Return to the Table 3-1.

Figure 3-144. ESM_SOC_START_REG Register

7	6	5	4	3	2	1	0
RESERVED							ESM_SOC_START
R/W-0b							R/W-0b

Table 3-146. ESM_SOC_START_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:1	RESERVED	R/W	0b	
0	ESM_SOC_START	R/W	0b	Control bit to start the ESM_SoC: 0b = ESM_SoC not started. Device clears ENABLE_DRV bit when bit ESM_SOC_EN=1 1b = ESM_SoC started

3.1.145 ESM_SOC_DELAY1_REG Register (Offset = 0x99) [Reset = 0x0]

ESM_SOC_DELAY1_REG is shown in Figure 3-145 and described in Table 3-147.

Return to the Table 3-1.

Figure 3-145. ESM_SOC_DELAY1_REG Register

7	6	5	4	3	2	1	0
ESM_SOC_DELAY1							
R/W-0b							

Table 3-147. ESM_SOC_DELAY1_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	ESM_SOC_DELAY1	R/W	0b	These bits configure the duration of the ESM_SoC delay-1 time-interval (see Error Signal Monitor chapter). These bits can be only be written when control bit ESM_SOC_START=0.

3.1.146 ESM_SOC_DELAY2_REG Register (Offset = 0x9A) [Reset = 0x0]

ESM_SOC_DELAY2_REG is shown in Figure 3-146 and described in Table 3-148.

Return to the Table 3-1.

Figure 3-146. ESM_SOC_DELAY2_REG Register

7	6	5	4	3	2	1	0
ESM_SOC_DELAY2							
R/W-0b							

Table 3-148. ESM_SOC_DELAY2_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	ESM_SOC_DELAY2	R/W	0b	These bits configure the duration of the ESM_SoC delay-2 time-interval (see Error Signal Monitor chapter). These bits can be only be written when control bit ESM_SOC_START=0.

3.1.147 ESM_SOC_MODE_CFG Register (Offset = 0x9B) [Reset = 0x0]

ESM_SOC_MODE_CFG is shown in Figure 3-147 and described in Table 3-149.

Return to the Table 3-1.

Figure 3-147. ESM_SOC_MODE_CFG Register

7	6	5	4	3	2	1	0
ESM_SOC_MODE	ESM_SOC_EN	ESM_SOC_ENDRV	RESERVED	ESM_SOC_ERR_CNT_TH			
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b			

Table 3-149. ESM_SOC_MODE_CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
7	ESM_SOC_MODE	R/W	0b	This bit selects the mode for the ESM_SoC: These bits can be only be written when control bit ESM_SOC_START=0. 0b = Level Mode 1b = PWM Mode
6	ESM_SOC_EN	R/W	0b	ESM_SoC enable configuration bit: These bits can be only be written when control bit ESM_SOC_START=0. 0b = ESM_SoC disabled. MCU can set ENABLE_DRV bit to 1 if all other interrupt bits are cleared 1b = ESM_SoC enabled. MCU can set ENABLE_DRV bit to 1 if: - bit ESM_SOC_START=1, and - (ESM_SOC_FAIL_INT=0 or ESM_SOC_ENDRV=0), and - ESM_SOC_RST_INT=0, and - all other interrupt bits are cleared.
5	ESM_SOC_ENDRV	R/W	0b	Configuration bit to select ENABLE_DRV clear on ESM-error for ESM_SoC: These bits can be only be written when control bit ESM_SOC_START=0 0b = ENABLE_DRV not cleared when ESM_SOC_FAIL_INT=1 1b = ENABLE_DRV cleared when ESM_SOC_FAIL_INT=1.
4	RESERVED	R/W	0b	
3:0	ESM_SOC_ERR_CNT_TH	R/W	0b	Configuration bits for the threshold of the ESM_SoC error-counter The ESM_SoC starts the Error Handling Procedure (see Error Signal Monitor chapter) if ESM_SOC_ERR_CNT[4:0] > ESM_SOC_ERR_CNT_TH[3:0]. These bits can be only be written when control bit ESM_SOC_START=0.

3.1.148 ESM_SOC_HMAX_REG Register (Offset = 0x9C) [Reset = 0x0]

ESM_SOC_HMAX_REG is shown in Figure 3-148 and described in Table 3-150.

Return to the Table 3-1.

Figure 3-148. ESM_SOC_HMAX_REG Register

7	6	5	4	3	2	1	0
ESM_SOC_HMAX							
R/W-0b							

Table 3-150. ESM_SOC_HMAX_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	ESM_SOC_HMAX	R/W	0b	These bits configure the the maximum high-pulse time-threshold (tHIGH_MAX_TH) for ESM_SoC (see Error Signal Monitor chapter). These bits can be only be written when control bit ESM_SOC_START=0.

3.1.149 ESM_SOC_HMIN_REG Register (Offset = 0x9D) [Reset = 0x0]

ESM_SOC_HMIN_REG is shown in Figure 3-149 and described in Table 3-151.

Return to the Table 3-1.

Figure 3-149. ESM_SOC_HMIN_REG Register

7	6	5	4	3	2	1	0
ESM_SOC_HMIN							
R/W-0b							

Table 3-151. ESM_SOC_HMIN_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	ESM_SOC_HMIN	R/W	0b	These bits configure the the minimum high-pulse time-threshold (tHIGH_MIN_TH) for ESM_SoC (see Error Signal Monitor chapter). These bits can be only be written when control bit ESM_SOC_START=0.

3.1.150 ESM_SOC_LMAX_REG Register (Offset = 0x9E) [Reset = 0x0]

ESM_SOC_LMAX_REG is shown in Figure 3-150 and described in Table 3-152.

Return to the Table 3-1.

Figure 3-150. ESM_SOC_LMAX_REG Register

7	6	5	4	3	2	1	0
ESM_SOC_LMAX							
R/W-0b							

Table 3-152. ESM_SOC_LMAX_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	ESM_SOC_LMAX	R/W	0b	These bits configure the the maximum low-pulse time-threshold (tLOW_MAX_TH) for ESM_SoC (see Error Signal Monitor chapter). These bits can be only be written when control bit ESM_SOC_START=0.

3.1.151 ESM_SOC_LMIN_REG Register (Offset = 0x9F) [Reset = 0x0]

ESM_SOC_LMIN_REG is shown in Figure 3-151 and described in Table 3-153.

Return to the Table 3-1.

Figure 3-151. ESM_SOC_LMIN_REG Register

7	6	5	4	3	2	1	0
ESM_SOC_LMIN							
R/W-0b							

Table 3-153. ESM_SOC_LMIN_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	ESM_SOC_LMIN	R/W	0b	These bits configure the the minimum low-pulse time-threshold (tLOW_MAX_TH) for ESM_SoC (see Error Signal Monitor chapter). These bits can be only be written when control bit ESM_SOC_START=0.

3.1.152 ESM_SOC_ERR_CNT_REG Register (Offset = 0xA0) [Reset = 0x0]

ESM_SOC_ERR_CNT_REG is shown in Figure 3-152 and described in Table 3-154.

Return to the Table 3-1.

Figure 3-152. ESM_SOC_ERR_CNT_REG Register

7	6	5	4	3	2	1	0
RESERVED			ESM_SOC_ERR_CNT				
R-0b			R-0b				

Table 3-154. ESM_SOC_ERR_CNT_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	RESERVED	R	0b	
4:0	ESM_SOC_ERR_CNT	R	0b	Status bits to indicate the value of the ESM_SoC Error-Counter. The device clears these bits when ESM_SOC_START bit is 0, or when the device resets the SoC.

3.1.153 REGISTER_LOCK Register (Offset = 0xA1) [Reset = 0x0]

REGISTER_LOCK is shown in Figure 3-153 and described in Table 3-155.

Return to the Table 3-1.

Figure 3-153. REGISTER_LOCK Register

7	6	5	4	3	2	1	0
RESERVED							REGISTER_LOCK_STATUS
R/W-0b							R/W-0b

Table 3-155. REGISTER_LOCK Register Field Descriptions

Bit	Field	Type	Reset	Description
7:1	RESERVED	R/W	0b	
0	REGISTER_LOCK_STATUS	R/W	0b	Unlocking registers: write 0x9B to this address. Locking registers: write anything else than 0x9B to this address. Written 8 bit data to this address will not be stored, only lock status can be read. REGISTER_LOCK_STATUS bit shows the lock status: 0b = Registers are unlocked 1b = Registers are locked

3.1.154 MANUFACTURING_VER Register (Offset = 0xA6) [Reset = 0x0]

MANUFACTURING_VER is shown in Figure 3-154 and described in Table 3-156.

Return to the Table 3-1.

Figure 3-154. MANUFACTURING_VER Register

7	6	5	4	3	2	1	0
SILICON_REV							
R-0b							

Table 3-156. MANUFACTURING_VER Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	SILICON_REV	R	0b	SILICON_REV[7:6] - Reserved SILICON_REV[5:3] - ALR SILICON_REV[2:0] - Metal

3.1.155 CUSTOMER_NVM_ID_REG Register (Offset = 0xA7) [Reset = 0x0]

CUSTOMER_NVM_ID_REG is shown in Figure 3-155 and described in Table 3-157.

Return to the Table 3-1.

Figure 3-155. CUSTOMER_NVM_ID_REG Register

7	6	5	4	3	2	1	0
CUSTOMER_NVM_ID							
R/W-0b							

Table 3-157. CUSTOMER_NVM_ID_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	CUSTOMER_NVM_ID	R/W	0b	Customer defined value if customer programmed part Same value as in TI_NVM_ID register if TI programmed part

3.1.156 SOFT_REBOOT_REG Register (Offset = 0xAB) [Reset = 0x0]

SOFT_REBOOT_REG is shown in Figure 3-156 and described in Table 3-158.

Return to the Table 3-1.

Figure 3-156. SOFT_REBOOT_REG Register

7	6	5	4	3	2	1	0
RESERVED							SOFT_REBOOT
R/W-0b							R/WSelfClrF-0b

Table 3-158. SOFT_REBOOT_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:1	RESERVED	R/W	0b	
0	SOFT_REBOOT	R/WSelfClrF	0b	Write 1 to request a soft reboot. This bit is automatically cleared.

3.1.157 RTC_SECONDS Register (Offset = 0xB5) [Reset = 0x0]

RTC_SECONDS is shown in Figure 3-157 and described in Table 3-159.

Return to the Table 3-1.

Figure 3-157. RTC_SECONDS Register

7	6	5	4	3	2	1	0
RESERVED	SECOND_1			SECOND_0			
R/W-0b	R/W-0b			R/W-0b			

Table 3-159. RTC_SECONDS Register Field Descriptions

Bit	Field	Type	Reset	Description
7	RESERVED	R/W	0b	
6:4	SECOND_1	R/W	0b	Second digit of seconds (range is 0 up to 5)
3:0	SECOND_0	R/W	0b	First digit of seconds (range is 0 up to 9)

3.1.158 RTC_MINUTES Register (Offset = 0xB6) [Reset = 0x0]

RTC_MINUTES is shown in Figure 3-158 and described in Table 3-160.

Return to the Table 3-1.

Figure 3-158. RTC_MINUTES Register

7	6	5	4	3	2	1	0
RESERVED	MINUTE_1			MINUTE_0			
R/W-0b	R/W-0b			R/W-0b			

Table 3-160. RTC_MINUTES Register Field Descriptions

Bit	Field	Type	Reset	Description
7	RESERVED	R/W	0b	
6:4	MINUTE_1	R/W	0b	Second digit of minutes (range is 0 up to 5)
3:0	MINUTE_0	R/W	0b	First digit of minutes (range is 0 up to 9)

3.1.159 RTC_HOURS Register (Offset = 0xB7) [Reset = 0x0]

RTC_HOURS is shown in Figure 3-159 and described in Table 3-161.

Return to the Table 3-1.

Figure 3-159. RTC_HOURS Register

7	6	5	4	3	2	1	0
PM_NAM	RESERVED	HOUR_1		HOUR_0			
R/W-0b	R/W-0b	R/W-0b		R/W-0b			

Table 3-161. RTC_HOURS Register Field Descriptions

Bit	Field	Type	Reset	Description
7	PM_NAM	R/W	0b	Only used in PM_AM mode (otherwise it is set to 0) 0b = AM 1b = PM
6	RESERVED	R/W	0b	
5:4	HOUR_1	R/W	0b	Second digit of hours(range is 0 up to 2)
3:0	HOUR_0	R/W	0b	First digit of hours (range is 0 up to 9)

3.1.160 RTC_DAYS Register (Offset = 0xB8) [Reset = 0x0]

RTC_DAYS is shown in Figure 3-160 and described in Table 3-162.

Return to the Table 3-1.

Figure 3-160. RTC_DAYS Register

7	6	5	4	3	2	1	0
RESERVED		DAY_1		DAY_0			
R/W-0b		R/W-0b		R/W-0b			

Table 3-162. RTC_DAYS Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5:4	DAY_1	R/W	0b	Second digit of days (range is 0 up to 3)
3:0	DAY_0	R/W	0b	First digit of days (range is 0 up to 9)

3.1.161 RTC_MONTHS Register (Offset = 0xB9) [Reset = 0x0]

RTC_MONTHS is shown in Figure 3-161 and described in Table 3-163.

Return to the Table 3-1.

Figure 3-161. RTC_MONTHS Register

7	6	5	4	3	2	1	0
RESERVED			MONTH_1	MONTH_0			
R/W-0b			R/W-0b	R/W-0b			

Table 3-163. RTC_MONTHS Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	RESERVED	R/W	0b	
4	MONTH_1	R/W	0b	Second digit of months (range is 0 up to 1)
3:0	MONTH_0	R/W	0b	First digit of months (range is 0 up to 9)

3.1.162 RTC_YEARS Register (Offset = 0xBA) [Reset = 0x0]

RTC_YEARS is shown in Figure 3-162 and described in Table 3-164.

Return to the Table 3-1.

Figure 3-162. RTC_YEARS Register

7	6	5	4	3	2	1	0
YEAR_1				YEAR_0			
R/W-0b				R/W-0b			

Table 3-164. RTC_YEARS Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	YEAR_1	R/W	0b	Second digit of years (range is 0 up to 9)
3:0	YEAR_0	R/W	0b	First digit of years (range is 0 up to 9)

3.1.163 RTC_WEEKS Register (Offset = 0xBB) [Reset = 0x0]

RTC_WEEKS is shown in Figure 3-163 and described in Table 3-165.

Return to the Table 3-1.

Figure 3-163. RTC_WEEKS Register

7	6	5	4	3	2	1	0
RESERVED					WEEK		
R/W-0b					R/W-0b		

Table 3-165. RTC_WEEKS Register Field Descriptions

Bit	Field	Type	Reset	Description
7:3	RESERVED	R/W	0b	
2:0	WEEK	R/W	0b	First digit of day of the week (range is 0 up to 6)

3.1.164 ALARM_SECONDS Register (Offset = 0xBC) [Reset = 0x0]

ALARM_SECONDS is shown in Figure 3-164 and described in Table 3-166.

Return to the Table 3-1.

Figure 3-164. ALARM_SECONDS Register

7	6	5	4	3	2	1	0
RESERVED	ALR_SECOND_1			ALR_SECOND_0			
R/W-0b	R/W-0b			R/W-0b			

Table 3-166. ALARM_SECONDS Register Field Descriptions

Bit	Field	Type	Reset	Description
7	RESERVED	R/W	0b	
6:4	ALR_SECOND_1	R/W	0b	Second digit of alarm programmation for seconds (range is 0 up to 5)
3:0	ALR_SECOND_0	R/W	0b	First digit of alarm programmation for seconds (range is 0 up to 9)

3.1.165 ALARM_MINUTES Register (Offset = 0xBD) [Reset = 0x0]

ALARM_MINUTES is shown in Figure 3-165 and described in Table 3-167.

Return to the Table 3-1.

Figure 3-165. ALARM_MINUTES Register

7	6	5	4	3	2	1	0
RESERVED	ALR_MINUTE_1			ALR_MINUTE_0			
R/W-0b	R/W-0b			R/W-0b			

Table 3-167. ALARM_MINUTES Register Field Descriptions

Bit	Field	Type	Reset	Description
7	RESERVED	R/W	0b	
6:4	ALR_MINUTE_1	R/W	0b	Second digit of alarm programmation for minutes (range is 0 up to 5)
3:0	ALR_MINUTE_0	R/W	0b	First digit of alarm programmation for minutes (range is 0 up to 9)

3.1.166 ALARM_HOURS Register (Offset = 0xBE) [Reset = 0x0]

ALARM_HOURS is shown in Figure 3-166 and described in Table 3-168.

Return to the Table 3-1.

Figure 3-166. ALARM_HOURS Register

7	6	5	4	3	2	1	0
ALR_PM_NAM	RESERVED	ALR_HOUR_1		ALR_HOUR_0			
R/W-0b	R/W-0b	R/W-0b		R/W-0b			

Table 3-168. ALARM_HOURS Register Field Descriptions

Bit	Field	Type	Reset	Description
7	ALR_PM_NAM	R/W	0b	Only used in PM_AM mode for alarm programmation (otherwise it is set to 0) 0b = AM 1b = PM
6	RESERVED	R/W	0b	
5:4	ALR_HOUR_1	R/W	0b	Second digit of alarm programmation for hours(range is 0 up to 2)
3:0	ALR_HOUR_0	R/W	0b	First digit of alarm programmation for hours (range is 0 up to 9)

3.1.167 ALARM_DAYS Register (Offset = 0xBF) [Reset = 0x0]

ALARM_DAYS is shown in Figure 3-167 and described in Table 3-169.

Return to the Table 3-1.

Figure 3-167. ALARM_DAYS Register

7	6	5	4	3	2	1	0
RESERVED		ALR_DAY_1		ALR_DAY_0			
R/W-0b		R/W-0b		R/W-0b			

Table 3-169. ALARM_DAYS Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R/W	0b	
5:4	ALR_DAY_1	R/W	0b	Second digit of alarm programmation for days (range is 0 up to 3)
3:0	ALR_DAY_0	R/W	0b	First digit of alarm programmation for days (range is 0 up to 9)

3.1.168 ALARM_MONTHS Register (Offset = 0xC0) [Reset = 0x0]

ALARM_MONTHS is shown in Figure 3-168 and described in Table 3-170.

Return to the Table 3-1.

Figure 3-168. ALARM_MONTHS Register

7	6	5	4	3	2	1	0
RESERVED			ALR_MONTH_1	ALR_MONTH_0			
R/W-0b			R/W-0b	R/W-0b			

Table 3-170. ALARM_MONTHS Register Field Descriptions

Bit	Field	Type	Reset	Description
7:5	RESERVED	R/W	0b	
4	ALR_MONTH_1	R/W	0b	Second digit of alarm programmation for months (range is 0 up to 1)
3:0	ALR_MONTH_0	R/W	0b	First digit of alarm programmation for months (range is 0 up to 9)

3.1.169 ALARM_YEARS Register (Offset = 0xC1) [Reset = 0x0]

ALARM_YEARS is shown in Figure 3-169 and described in Table 3-171.

Return to the Table 3-1.

Figure 3-169. ALARM_YEARS Register

7	6	5	4	3	2	1	0
ALR_YEAR_1				ALR_YEAR_0			
R/W-0b				R/W-0b			

Table 3-171. ALARM_YEARS Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	ALR_YEAR_1	R/W	0b	Second digit of alarm programmation for years (range is 0 up to 9)
3:0	ALR_YEAR_0	R/W	0b	First digit of alarm programmation for years (range is 0 up to 9)

3.1.170 RTC_CTRL_1 Register (Offset = 0xC2) [Reset = 0x0]

RTC_CTRL_1 is shown in Figure 3-170 and described in Table 3-172.

Return to the Table 3-1.

Figure 3-170. RTC_CTRL_1 Register

7	6	5	4	3	2	1	0
RTC_V_OPT	GET_TIME	SET_32_COUNTER	RESERVED	MODE_12_24	AUTO_COMP	ROUND_30S	STOP_RTC
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-172. RTC_CTRL_1 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	RTC_V_OPT	R/W	0b	RTC date / time register selection: 0b = Read access directly to dynamic registers (RTC_SECONDS, RTC_MINUTES, RTC_HOURS, RTC_DAYS, RTC_MONTHS, RTC_YEAR, RTC_WEEKS) 1b = Read access to static shadowed registers: (see GET_TIME bit).
6	GET_TIME	R/W	0b	When writing a 1 into this register, the content of the dynamic registers (RTC_SECONDS, RTC_MINUTES, RTC_HOURS, RTC_DAYS, RTC_MONTHS, RTC_YEARS_ and RTC_WEEKS) is transferred into static shadowed registers. Each update of the shadowed registers needs to be done by re-asserting GET_TIME bit to 1 (i.e.: reset it to 0 and then rewrite it to 1) Note: Shadowed registers, linked to the GET_TIME feature, are a parallel set of calendar static registers, at the same I2C addresses as the dynamic registers. Note: The GET_TIME feature loads the RTC counter in the shadow registers and make the content of the shadow registers available and stable for reading. Note: The GET_TIME bit has to be set to 0 and again to 1 to get a new timing value. Note: If the time reading is done without GET_TIME, the read value comes directly from the RTC counter and software has to manage the counter change during the reading. Time reading remains always at the same address, with or without using the GET_TIME feature.
5	SET_32_COUNTER	R/W	0b	Note: This bit must only be used when the RTC is frozen. 0b = No action 1b = Set the 32kHz counter with RTC_COMP_MSB_REG/ RTC_COMP_LSB_REG value
4	RESERVED	R/W	0b	
3	MODE_12_24	R/W	0b	Note: It is possible to switch between the two modes at any time without disturbed the RTC, read or write are always performed with the current mode. 0b = 24 hours mode 1b = 12 hours mode (PM-AM mode)
2	AUTO_COMP	R/W	0b	AUTO_COMP 0b = No auto compensation 1b = Auto compensation enabled
1	ROUND_30S	R/W	0b	Note: This bit is a toggle bit, the micro-controller can only write one and RTC clears it. If the micro-controller sets the ROUND_30S bit and then read it, the micro-controller will read one until the rounding to the closest minute is performed at the next second. 0b = No update 1b = When a one is written, the time is rounded to the closest minute
0	STOP_RTC	R/W	0b	STOP_RTC 0b = RTC is frozen 1b = RTC is running

3.1.171 RTC_CTRL_2 Register (Offset = 0xC3) [Reset = 0x0]

RTC_CTRL_2 is shown in Figure 3-171 and described in Table 3-173.

Return to the Table 3-1.

Figure 3-171. RTC_CTRL_2 Register

7	6	5	4	3	2	1	0
FIRST_START UP_DONE	STARTUP_DEST		FAST_BIST	LP_STANDBY_ SEL	XTAL_SEL		XTAL_EN
R/W-0b	R/W-0b		R/W-0b	R/W-0b	R/W-0b		R/W-0b

Table 3-173. RTC_CTRL_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7	FIRST_STARTUP_DONE	R/W	0b	This bit controls if EEPROM defaults are loaded to RTC domain reg bits during EEPROM read 0b = EEPROM defaults are loaded to RTC domain bits 1b = EEPROM defaults are not loaded to RTC domain bits
6:5	STARTUP_DEST	R/W	0b	FSM startup destination select. (Default from NVM memory) 0b = STANDBY/LP_STANDBY based on LP_STANDBY_SEL 1b = Reserved 10b = MCU_ONLY 11b = ACTIVE
4	FAST_BIST	R/W	0b	FAST_BIST (Default from NVM memory) 0b = Logic and analog BIST is run at BOOT BIST. 1b = Only analog BIST is run at BOOT BIST.
3	LP_STANDBY_SEL	R/W	0b	Control to enter low power standby state: (Default from NVM memory) 0b = LDOINT is enabled in standby state. 1b = Low power standby state is used as standby state (LDOINT is disabled).
2:1	XTAL_SEL	R/W	0b	Crystal oscillator type select (Default from NVM memory) 0b = 6 pF 1b = 9 pF 10b = 12.5 pF 11b = Reserved
0	XTAL_EN	R/W	0b	Crystal oscillator enable. (Default from NVM memory) 0b = Crystal oscillator is disabled 1b = Crystal oscillator is enabled

3.1.172 RTC_STATUS Register (Offset = 0xC4) [Reset = 0x80]

RTC_STATUS is shown in Figure 3-172 and described in Table 3-174.

Return to the Table 3-1.

Figure 3-172. RTC_STATUS Register

7	6	5	4	3	2	1	0
POWER_UP	ALARM	TIMER	RESERVED			RUN	RESERVED
R/W1C-1b	R/W1C-0b	R/W1C-0b	R/W-0b			R-0b	R/W-0b

Table 3-174. RTC_STATUS Register Field Descriptions

Bit	Field	Type	Reset	Description
7	POWER_UP	R/W1C	1b	Indicates that a reset occurred (bit cleared to 0 by writing 1) and that RTC data are not valid anymore. Note: POWER_UP is set by a reset, is cleared by writing one in this bit. Note: The POWER_UP (RTC_STATUS) and RESET_STATUS (RTC_RESET_STATUS) register bits indicate the same information.
6	ALARM	R/W1C	0b	Indicates that an alarm interrupt has been generated (bit clear by writing 1).
5	TIMER	R/W1C	0b	Indicates that an timer interrupt has been generated (bit clear by writing 1).
4:2	RESERVED	R/W	0b	
1	RUN	R	0b	Note: This bit shows the real state of the RTC, indeed because of STOP_RTC (RTC_CTRL) signal was resynchronized on 32kHz clock, the action of this bit is delayed. 0b = RTC is frozen 1b = RTC is running
0	RESERVED	R/W	0b	

3.1.173 RTC_INTERRUPTS Register (Offset = 0xC5) [Reset = 0x0]

RTC_INTERRUPTS is shown in Figure 3-173 and described in Table 3-175.

Return to the Table 3-1.

Figure 3-173. RTC_INTERRUPTS Register

7	6	5	4	3	2	1	0
RESERVED				IT_ALARM	IT_TIMER	EVERY	
R/W-0b				R/W-0b	R/W-0b	R/W-0b	

Table 3-175. RTC_INTERRUPTS Register Field Descriptions

Bit	Field	Type	Reset	Description
7:4	RESERVED	R/W	0b	
3	IT_ALARM	R/W	0b	Enable one interrupt when the alarm value is reached (TC ALARM registers: ALARM_SECONDS, ALARM_MINUTES, ALARM_HOURS, ALARM_DAYS, ALARM_MONTHS, ALARM_YEARS) by the TC registers NOTE: To prevent mis-firing of the ALARM interrupt, set the IT_ALARM = 0 prior to configuring the ALARM registers 0b = interrupt disabled 1b = interrupt enabled
2	IT_TIMER	R/W	0b	Enable periodic interrupt NOTE: To prevent mis-firing of the TIMER interrupt, set the IT_TIMER = 0 prior to configuring the periodic time value 0b = interrupt disabled 1b = interrupt enabled
1:0	EVERY	R/W	0b	Interrupt period 0b = every second 1b = every minute 10b = every hour 11b = every day

3.1.174 RTC_COMP_LSB Register (Offset = 0xC6) [Reset = 0x0]

RTC_COMP_LSB is shown in Figure 3-174 and described in Table 3-176.

Return to the Table 3-1.

Figure 3-174. RTC_COMP_LSB Register

7	6	5	4	3	2	1	0
COMP_LSB_RTC							
R/W-0b							

Table 3-176. RTC_COMP_LSB Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	COMP_LSB_RTC	R/W	0b	This register contains the number of 32kHz periods to be added into the 32kHz counter every hour [LSB]

3.1.175 RTC_COMP_MSB Register (Offset = 0xC7) [Reset = 0x0]

RTC_COMP_MSB is shown in Figure 3-175 and described in Table 3-177.

Return to the Table 3-1.

Figure 3-175. RTC_COMP_MSB Register

7	6	5	4	3	2	1	0
COMP_MSB_RTC							
R/W-0b							

Table 3-177. RTC_COMP_MSB Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	COMP_MSB_RTC	R/W	0b	This register contains the number of 32kHz periods to be added into the 32kHz counter every hour [MSB]

3.1.176 RTC_RESET_STATUS Register (Offset = 0xC8) [Reset = 0x0]

RTC_RESET_STATUS is shown in Figure 3-176 and described in Table 3-178.

Return to the Table 3-1.

Figure 3-176. RTC_RESET_STATUS Register

7	6	5	4	3	2	1	0
RESERVED							RESET_STATUS_RTC
R/W-0b							R/W-0b

Table 3-178. RTC_RESET_STATUS Register Field Descriptions

Bit	Field	Type	Reset	Description
7:1	RESERVED	R/W	0b	
0	RESET_STATUS_RTC	R/W	0b	This bit can only be set to one and is cleared when a manual reset or a POR (case of VOUT_LDO_RTC below the LDO_RTC POR level) occur. If this bit is reset it means that the RTC has lost its configuration. Note: The RESET_STATUS (RTC_RESET_STATUS) and POWER_UP (RTC_STATUS) register bits indicate the same information.

3.1.177 SCRATCH_PAD_REG_1 Register (Offset = 0xC9) [Reset = 0x0]

SCRATCH_PAD_REG_1 is shown in Figure 3-177 and described in Table 3-179.

Return to the Table 3-1.

Figure 3-177. SCRATCH_PAD_REG_1 Register

7	6	5	4	3	2	1	0
SCRATCH_PAD_1							
R/W-0b							

Table 3-179. SCRATCH_PAD_REG_1 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	SCRATCH_PAD_1	R/W	0b	Scratchpad for temporary data storage. The register is reset only when VRTC is disabled. The data is maintained when VINT regulator is disabled, for example during LP_STANDBY state.

3.1.178 SCRATCH_PAD_REG_2 Register (Offset = 0xCA) [Reset = 0x0]

SCRATCH_PAD_REG_2 is shown in Figure 3-178 and described in Table 3-180.

Return to the Table 3-1.

Figure 3-178. SCRATCH_PAD_REG_2 Register

7	6	5	4	3	2	1	0
SCRATCH_PAD_2							
R/W-0b							

Table 3-180. SCRATCH_PAD_REG_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	SCRATCH_PAD_2	R/W	0b	Scratchpad for temporary data storage. The register is reset only when VRTC is disabled. The data is maintained when VINT regulator is disabled, for example during LP_STANDBY state.

3.1.179 SCRATCH_PAD_REG_3 Register (Offset = 0xCB) [Reset = 0x0]

SCRATCH_PAD_REG_3 is shown in Figure 3-179 and described in Table 3-181.

Return to the Table 3-1.

Figure 3-179. SCRATCH_PAD_REG_3 Register

7	6	5	4	3	2	1	0
SCRATCH_PAD_3							
R/W-0b							

Table 3-181. SCRATCH_PAD_REG_3 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	SCRATCH_PAD_3	R/W	0b	Scratchpad for temporary data storage. The register is reset only when VRTC is disabled. The data is maintained when VINT regulator is disabled, for example during LP_STANDBY state.

3.1.180 SCRATCH_PAD_REG_4 Register (Offset = 0xCC) [Reset = 0x0]

SCRATCH_PAD_REG_4 is shown in Figure 3-180 and described in Table 3-182.

Return to the Table 3-1.

Figure 3-180. SCRATCH_PAD_REG_4 Register

7	6	5	4	3	2	1	0
SCRATCH_PAD_4							
R/W-0b							

Table 3-182. SCRATCH_PAD_REG_4 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	SCRATCH_PAD_4	R/W	0b	Scratchpad for temporary data storage. The register is reset only when VRTC is disabled. The data is maintained when VINT regulator is disabled, for example during LP_STANDBY state.

3.1.181 PFSM_DELAY_REG_1 Register (Offset = 0xCD) [Reset = 0x0]

PFSM_DELAY_REG_1 is shown in Figure 3-181 and described in Table 3-183.

Return to the Table 3-1.

Figure 3-181. PFSM_DELAY_REG_1 Register

7	6	5	4	3	2	1	0
PFSM_DELAY1							
R/W-0b							

Table 3-183. PFSM_DELAY_REG_1 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	PFSM_DELAY1	R/W	0b	Generic delay1 for PFSM use. The step size is defined by PFSM_DELAY_STEP bits. (Default from NVM memory)

3.1.182 PFSM_DELAY_REG_2 Register (Offset = 0xCE) [Reset = 0x0]

PFSM_DELAY_REG_2 is shown in Figure 3-182 and described in Table 3-184.

Return to the Table 3-1.

Figure 3-182. PFSM_DELAY_REG_2 Register

7	6	5	4	3	2	1	0
PFSM_DELAY2							
R/W-0b							

Table 3-184. PFSM_DELAY_REG_2 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	PFSM_DELAY2	R/W	0b	Generic delay2 for PFSM use. The step size is defined by PFSM_DELAY_STEP bits. (Default from NVM memory)

3.1.183 PFSM_DELAY_REG_3 Register (Offset = 0xCF) [Reset = 0x0]

PFSM_DELAY_REG_3 is shown in Figure 3-183 and described in Table 3-185.

Return to the Table 3-1.

Figure 3-183. PFSM_DELAY_REG_3 Register

7	6	5	4	3	2	1	0
PFSM_DELAY3							
R/W-0b							

Table 3-185. PFSM_DELAY_REG_3 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	PFSM_DELAY3	R/W	0b	Generic delay3 for PFSM use. The step size is defined by PFSM_DELAY_STEP bits. (Default from NVM memory)

3.1.184 PFSM_DELAY_REG_4 Register (Offset = 0xD0) [Reset = 0x0]

PFSM_DELAY_REG_4 is shown in Figure 3-184 and described in Table 3-186.

Return to the Table 3-1.

Figure 3-184. PFSM_DELAY_REG_4 Register

7	6	5	4	3	2	1	0
PFSM_DELAY4							
R/W-0b							

Table 3-186. PFSM_DELAY_REG_4 Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	PFSM_DELAY4	R/W	0b	Generic delay4 for PFSM use. The step size is defined by PFSM_DELAY_STEP bits. (Default from NVM memory)

3.1.185 WD_ANSWER_REG Register (Offset = 0x401) [Reset = 0x0]

WD_ANSWER_REG is shown in Figure 3-185 and described in Table 3-187.

Return to the Table 3-1.

Figure 3-185. WD_ANSWER_REG Register

7	6	5	4	3	2	1	0
WD_ANSWER							
R/W-0b							

Table 3-187. WD_ANSWER_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	WD_ANSWER	R/W	0b	MCU answer byte. The MCU must write the expected reference Answer-x into this register. Each watchdog question requires four answer bytes: - Three answer bytes (Answer-3, Answer-2, Answer-1) must be written in Window-1. - The fourth (final) answer-byte (Answer-0) must be written in Window-2. The number of written answer bytes is tracked with the WD_ANSW_CNT counter in the WD_QUESTION_ANSW_CNT register. These bits only apply for Watchdog in Q&A mode.

3.1.186 WD_QUESTION_ANSW_CNT Register (Offset = 0x402) [Reset = 0x30]

WD_QUESTION_ANSW_CNT is shown in Figure 3-186 and described in Table 3-188.

Return to the Table 3-1.

Figure 3-186. WD_QUESTION_ANSW_CNT Register

7	6	5	4	3	2	1	0
RESERVED		WD_ANSW_CNT		WD_QUESTION			
R-0b		R-11b		R-0b			

Table 3-188. WD_QUESTION_ANSW_CNT Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	RESERVED	R	0b	
5:4	WD_ANSW_CNT	R	11b	Current, received watchdog-answer count state. These bits only apply for Watchdog in Q&A mode.
3:0	WD_QUESTION	R	0b	Watchdog question. The MCU must read (or calculate) the current watchdog question value to generate correct answers. These bits only apply for Watchdog in Q&A mode.

3.1.187 WD_WIN1_CFG Register (Offset = 0x403) [Reset = 0x7F]

WD_WIN1_CFG is shown in Figure 3-187 and described in Table 3-189.

Return to the Table 3-1.

Figure 3-187. WD_WIN1_CFG Register

7	6	5	4	3	2	1	0
RESERVED	WD_WIN1						
R/W-0b	R/W-1111111b						

Table 3-189. WD_WIN1_CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
7	RESERVED	R/W	0b	
6:0	WD_WIN1	R/W	1111111b	These bits are for programming the duration of Watchdog Window-1 (see Watchdoc chapter). These bits can be only be written when the watchdog is in the Long Window.

3.1.188 WD_WIN2_CFG Register (Offset = 0x404) [Reset = 0x7F]

WD_WIN2_CFG is shown in Figure 3-188 and described in Table 3-190.

Return to the Table 3-1.

Figure 3-188. WD_WIN2_CFG Register

7	6	5	4	3	2	1	0
RESERVED	WD_WIN2						
R/W-0b	R/W-1111111b						

Table 3-190. WD_WIN2_CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
7	RESERVED	R/W	0b	
6:0	WD_WIN2	R/W	1111111b	These bits are for programming the duration of Watchdog Window-2 (see Watchdog chapter). These bits can be only be written when the watchdog is in the Long Window.

3.1.189 WD_LONGWIN_CFG Register (Offset = 0x405) [Reset = 0xFF]

WD_LONGWIN_CFG is shown in Figure 3-189 and described in Table 3-191.

Return to the Table 3-1.

Figure 3-189. WD_LONGWIN_CFG Register

7	6	5	4	3	2	1	0
WD_LONGWIN							
R/W-11111111b							

Table 3-191. WD_LONGWIN_CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:0	WD_LONGWIN	R/W	11111111b	These bits are for programming the duration of Watchdog Long Window (see Watchdog chapter). These bits can be only be written when the watchdog is in the Long Window. (Default from NVM memory)

3.1.190 WD_MODE_REG Register (Offset = 0x406) [Reset = 0x2]

WD_MODE_REG is shown in Figure 3-190 and described in Table 3-192.

Return to the Table 3-1.

Figure 3-190. WD_MODE_REG Register

7	6	5	4	3	2	1	0
RESERVED					WD_PWRHOLD	WD_MODE_SELECT	WD_RETURN_LONGWIN
R/W-0b					R/W-0b	R/W-1b	R/W-0b

Table 3-192. WD_MODE_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:3	RESERVED	R/W	0b	
2	WD_PWRHOLD	R/W	0b	Device sets WD_PWRHOLD if hardware condition on pin DISABLE_WDOG (mapped to GPIO8 pin) is applied at startup (see Watchdog chapter). MCU can write this bit to 1. MCU needs to clear this bit to get out of the Long Window: 0b = watchdog goes out of the Long Window and starts the first watchdog-sequence when the configured Long Window time-interval elapses 1b = watchdog stays in Long Window
1	WD_MODE_SELECT	R/W	1b	Watchdog mode-select: MCU can set this to required value only when watchdog is in the Long Window. 0b = Trigger Mode 1b = Q&A mode.
0	WD_RETURN_LONGWIN	R/W	0b	MCU can set this bit to put the watchdog from operating back to the Long Window (see Watchdog chapter): 0b = Watchdog continues operating 1b = Watchdog returns to Long-Window after completion of the current watchdog-sequence.

3.1.191 WD_QA_CFG Register (Offset = 0x407) [Reset = 0xA]

WD_QA_CFG is shown in Figure 3-191 and described in Table 3-193.

Return to the Table 3-1.

Figure 3-191. WD_QA_CFG Register

7	6	5	4	3	2	1	0
WD_QA_FDBK		WD_QA_LFSR		WD_QUESTION_SEED			
R/W-0b		R/W-0b		R/W-1010b			

Table 3-193. WD_QA_CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
7:6	WD_QA_FDBK	R/W	0b	Feedback configuration bits for the watchdog question. These bits control the sequence of the generated questions and respective reference answers (see Watchdog chapter). These bits are only used for the watchdog in Q&A mode. These bits can be only be written when the watchdog is in the Long Window.
5:4	WD_QA_LFSR	R/W	0b	LFSR-equation configuration bits for the watchdog question (see Watchdog chapter). These bits are only used for the watchdog in Q&A mode. These bits can be only be written when the watchdog is in the Long Window.
3:0	WD_QUESTION_SEED	R/W	1010b	The watchdog question-seed value (see Watchdog chapter). The MCU updates the question-seed value to generate a set of new questions. These bits can be only be written when the watchdog is in the Long Window.

3.1.192 WD_ERR_STATUS Register (Offset = 0x408) [Reset = 0x0]

WD_ERR_STATUS is shown in Figure 3-192 and described in Table 3-194.

Return to the Table 3-1.

Figure 3-192. WD_ERR_STATUS Register

7	6	5	4	3	2	1	0
WD_RST_INT	WD_FAIL_INT	WD_ANSW_ERR	WD_SEQ_ERR	WD_ANSW_EARLY	WD_TRIG_EARLY	WD_TIMEOUT	WD_LONGWIN_TIMEOUT_INT
R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b	R/W1C-0b

Table 3-194. WD_ERR_STATUS Register Field Descriptions

Bit	Field	Type	Reset	Description
7	WD_RST_INT	R/W1C	0b	Latched status bit to indicate that the device went through warm reset due to WD_FAIL_CNT[3:0] > (WD_FAIL_TH[2:0] + WD_RST_TH[2:0]). Write 1 to clear.
6	WD_FAIL_INT	R/W1C	0b	Latched status bit to indicate that the watchdog has cleared the ENABLE_DRV bit due to WD_FAIL_CNT[3:0] > WD_FAIL_TH[2:0]. Write 1 to clear.
5	WD_ANSW_ERR	R/W1C	0b	Latched status bit to indicate that the watchdog has detected an incorrect answer-byte. Write 1 to clear. This bit only applies for Watchdog in Q&A mode.
4	WD_SEQ_ERR	R/W1C	0b	Latched status bit to indicate that the watchdog has detected an incorrect sequence of the answer-bytes. Write 1 to clear. This bit only applies for Watchdog in Q&A mode.
3	WD_ANSW_EARLY	R/W1C	0b	Latched status bit to indicate that the watchdog has received the final answer-byte in Window-1. Write 1 to clear. This bit only applies for Watchdog in Q&A mode.
2	WD_TRIG_EARLY	R/W1C	0b	Latched status bit to indicate that the watchdog has received the watchdog-trigger in Window-1. Write 1 to clear. This bit only applies for Watchdog in Trigger mode.
1	WD_TIMEOUT	R/W1C	0b	Latched status bit to indicate that the watchdog has detected a time-out event in the started watchdog sequence. Write 1 to clear.
0	WD_LONGWIN_TIMEOUT_INT	R/W1C	0b	Latched status bit to indicate that device went through warm reset due to elapse of Long Window time-interval. Write 1 to clear interrupt.

3.1.193 WD_THR_CFG Register (Offset = 0x409) [Reset = 0xFF]

WD_THR_CFG is shown in Figure 3-193 and described in Table 3-195.

Return to the Table 3-1.

Figure 3-193. WD_THR_CFG Register

7	6	5	4	3	2	1	0
WD_RST_EN	WD_EN	WD_FAIL_TH			WD_RST_TH		
R/W-1b	R/W-1b	R/W-111b			R/W-111b		

Table 3-195. WD_THR_CFG Register Field Descriptions

Bit	Field	Type	Reset	Description
7	WD_RST_EN	R/W	1b	Watchdog reset configuration bit: This bit can be only be written when the watchdog is in the Long Window. 0b = No warm reset when WD_FAIL_CNT[3:0] > (WD_FAIL_TH[2:0] + WD_RST_TH[2:0]) 1b = Warm reset when WD_FAIL_CNT[3:0] > (WD_FAIL_TH[2:0] + WD_RST_TH[2:0]).
6	WD_EN	R/W	1b	Watchdog enable configuration bit: This bit can be only be written when the watchdog is in the Long Window. (Default from NVM memory) 0b = watchdog disabled. MCU can set ENABLE_DRV bit to 1 if all other interrupt status bits are cleared 1b = watchdog enabled. MCU can set ENABLE_DRV bit to 1 if: - watchdog is out of the Long Window - WD_FAIL_CNT[3:0] =< WD_FAIL_TH[2:0] - WD_FIRST_OK=1 - all other interrupt status bits are cleared.
5:3	WD_FAIL_TH	R/W	111b	Configuration bits for the 1st threshold of the watchdog fail counter: Device clears ENABLE_DRV bit when WD_FAIL_CNT[3:0] > WD_FAIL_TH[2:0]. These bits can be only be written when the watchdog is in the Long Window.
2:0	WD_RST_TH	R/W	111b	Configuration bits for the 2nd threshold of the watchdog fail counter: Device goes through warm reset when WD_FAIL_CNT[3:0] > (WD_FAIL_TH[2:0] + WD_RST_TH[2:0]). These bits can be only be written when the watchdog is in the Long Window.

3.1.194 WD_FAIL_CNT_REG Register (Offset = 0x40A) [Reset = 0x20]

WD_FAIL_CNT_REG is shown in Figure 3-194 and described in Table 3-196.

Return to the Table 3-1.

Figure 3-194. WD_FAIL_CNT_REG Register

7	6	5	4	3	2	1	0
RESERVED	WD_BAD_EVENT	WD_FIRST_OK	RESERVED	WD_FAIL_CNT			
R-0b	R-0b	R-1b	R-0b	R-0b			

Table 3-196. WD_FAIL_CNT_REG Register Field Descriptions

Bit	Field	Type	Reset	Description
7	RESERVED	R	0b	
6	WD_BAD_EVENT	R	0b	Status bit to indicate that the watchdog has detected a bad event in the current watchdog sequence. The device clears this bit at the end of the watchdog sequence.
5	WD_FIRST_OK	R	1b	Status bit to indicate that the watchdog has detected a good event. The device clears this bit when the watchdog goes to the Long Window.
4	RESERVED	R	0b	
3:0	WD_FAIL_CNT	R	0b	Status bits to indicate the value of the Watchdog Fail Counter. The device clears these bits when the watchdog goes to the Long Window.

3.2 EEPROM_map Registers

Table 3-197 lists the memory-mapped registers for the EEPROM_map registers. All register offset addresses not listed in Table 3-197 should be considered as reserved locations and the register contents should not be modified.

Table 3-197. EEPROM_MAP Registers

Offset	Acronym	Register Name	Section
0x0	EEPROM_0_00		Section 3.2.1
0x1	EEPROM_0_01		Section 3.2.2
0x2	EEPROM_0_02		Section 3.2.3
0x3	EEPROM_0_03		Section 3.2.4
0x4	EEPROM_0_04		Section 3.2.5
0x5	EEPROM_0_05		Section 3.2.6
0x6	EEPROM_0_06		Section 3.2.7
0x7	EEPROM_0_07		Section 3.2.8
0x8	EEPROM_0_08		Section 3.2.9
0x9	EEPROM_0_09		Section 3.2.10
0xA	EEPROM_0_10		Section 3.2.11
0xB	EEPROM_0_11		Section 3.2.12
0xC	EEPROM_0_12		Section 3.2.13
0xD	EEPROM_0_13		Section 3.2.14
0xE	EEPROM_0_14		Section 3.2.15
0xF	EEPROM_0_15		Section 3.2.16
0x10	EEPROM_0_16		Section 3.2.17
0x11	EEPROM_0_17		Section 3.2.18
0x12	EEPROM_0_18		Section 3.2.19
0x13	EEPROM_0_19		Section 3.2.20
0x14	EEPROM_0_20		Section 3.2.21
0x15	EEPROM_0_21		Section 3.2.22
0x16	EEPROM_0_22		Section 3.2.23
0x17	EEPROM_0_23		Section 3.2.24
0x18	EEPROM_0_24		Section 3.2.25
0x19	EEPROM_0_25		Section 3.2.26
0x1A	EEPROM_0_26		Section 3.2.27
0x1B	EEPROM_0_27		Section 3.2.28
0x1C	EEPROM_0_28		Section 3.2.29
0x1D	EEPROM_0_29		Section 3.2.30
0x1E	EEPROM_0_30		Section 3.2.31
0x1F	EEPROM_0_31		Section 3.2.32
0x20	EEPROM_0_32		Section 3.2.33
0x21	EEPROM_0_33		Section 3.2.34
0x22	EEPROM_0_34		Section 3.2.35
0x23	EEPROM_0_35		Section 3.2.36
0x24	EEPROM_0_36		Section 3.2.37
0x25	EEPROM_0_37		Section 3.2.38
0x26	EEPROM_0_38		Section 3.2.39
0x27	EEPROM_0_39		Section 3.2.40
0x28	EEPROM_0_40		Section 3.2.41
0x29	EEPROM_0_41		Section 3.2.42

Table 3-197. EEPROM_MAP Registers (continued)

Offset	Acronym	Register Name	Section
0x2A	EEPROM_0_42		Section 3.2.43
0x2B	EEPROM_0_43		Section 3.2.44
0x2C	EEPROM_0_44		Section 3.2.45
0x2D	EEPROM_0_45		Section 3.2.46
0x2E	EEPROM_0_46		Section 3.2.47
0x2F	EEPROM_0_47		Section 3.2.48
0x30	EEPROM_0_48		Section 3.2.49
0x31	EEPROM_0_49		Section 3.2.50
0x32	EEPROM_0_50		Section 3.2.51
0x33	EEPROM_0_51		Section 3.2.52
0x34	EEPROM_0_52		Section 3.2.53
0x35	EEPROM_0_53		Section 3.2.54
0x36	EEPROM_0_54		Section 3.2.55
0x37	EEPROM_0_55		Section 3.2.56
0x38	EEPROM_0_56		Section 3.2.57
0x39	EEPROM_0_57		Section 3.2.58
0x3A	EEPROM_0_58		Section 3.2.59
0x3B	EEPROM_0_59		Section 3.2.60
0x3C	EEPROM_0_60		Section 3.2.61
0x3D	EEPROM_0_61		Section 3.2.62
0x3E	EEPROM_0_62		Section 3.2.63
0x3F	EEPROM_0_63		Section 3.2.64

Complex bit access types are encoded to fit into small table cells. Table 3-198 shows the codes that are used for access types in this section.

Table 3-198. EEPROM_map Access Type Codes

Access Type	Code	Description
Read Type		
R	R	Read
Write Type		
W	W	Write
Reset or Default Value		
-n		Value after reset or the default value
Register Array Variables		
i,j,k,l,m,n		When these variables are used in a register name, an offset, or an address, they refer to the value of a register array where the register is part of a group of repeating registers. The register groups form a hierarchical structure and the array is represented with a formula.
y		When this variable is used in a register name, an offset, or an address it refers to the value of a register array.

3.2.1 EEPROM_0_00 Register (Offset = 0x0) [Reset = 0xABEB57AB]

EEPROM_0_00 is shown in Figure 3-195 and described in Table 3-199.

Return to the Table 3-197.

Figure 3-195. EEPROM_0_00 Register

31	30	29	28	27	26	25	24
trim_vmpvin_rdiv_1v2					trim_vmpvin_bg		
R/W-10101b					R/W-1111b		

23	22	21	20	19	18	17	16
trim_vmpvin_bg		trim_vmvcca_rdiv_0v6					trim_vmvcca_rdiv_1v2
R/W-1111b		R/W-10101b					R/W-10101b

15	14	13	12	11	10	9	8
trim_vmvcca_rdiv_1v2				trim_vmvcca_bg			
R/W-10101b				R/W-1111b			

7	6	5	4	3	2	1	0
trim_vmvcca_bg	trim_osc_20mhz						
R/W-1111b	R/W-101011b						

Table 3-199. EEPROM_0_00 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:27	trim_vmpvin_rdiv_1v2	R/W	10101b	
26:22	trim_vmpvin_bg	R/W	1111b	
21:17	trim_vmvcca_rdiv_0v6	R/W	10101b	
16:12	trim_vmvcca_rdiv_1v2	R/W	10101b	
11:7	trim_vmvcca_bg	R/W	1111b	
6:0	trim_osc_20mhz	R/W	101011b	

3.2.2 EEPROM_0_01 Register (Offset = 0x1) [Reset = 0x225846AF]

EEPROM_0_01 is shown in Figure 3-196 and described in Table 3-200.

Return to the Table 3-197.

Figure 3-196. EEPROM_0_01 Register

31	30	29	28	27	26	25	24
RESERVED		trim_amux_buffer				trim_osc_128khz	
R-		R/W-1000b				R/W-100101b	
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
trim_osc_128khz				trim_ldo_int			
R/W-100101b				R/W-10000b			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
trim_ldo_int	trim_refsys_ibias					trim_refsys_rdiv_1v2	
R/W-10000b	R/W-10001b					R/W-10101b	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
trim_refsys_rdiv_1v2			trim_refsys_bg				
R/W-10101b			R/W-1111b				

Table 3-200. EEPROM_0_01 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:30	RESERVED	R	0b	
29:26	trim_amux_buffer	R/W	1000b	
25:20	trim_osc_128khz	R/W	100101b	
19:15	trim_ldo_int	R/W	10000b	
14:10	trim_refsys_ibias	R/W	10001b	
9:5	trim_refsys_rdiv_1v2	R/W	10101b	
4:0	trim_refsys_bg	R/W	1111b	

3.2.3 EEPROM_0_02 Register (Offset = 0x2) [Reset = 0x8C1EF8AB]

EEPROM_0_02 is shown in Figure 3-197 and described in Table 3-201.

Return to the Table 3-197.

Figure 3-197. EEPROM_0_02 Register

31	30	29	28	27	26	25	24
trim_vmon_buck1_rdiv_uv					trim_vmon_buck1_vref_dac		
R/W-10001b					R/W-10000b		
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
trim_vmon_buck1_vref_dac		trim_vmon_buck1_lshift_uv					trim_vmon_buck1_lshift_ov
R/W-10000b		R/W-1111b					R/W-1111b
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
trim_vmon_buck1_lshift_ov				trim_safety_ibias			
R/W-1111b				R/W-10001b			
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
trim_safety_ibias	trim_osc_52mhz						
R/W-10001b	R/W-101011b						

Table 3-201. EEPROM_0_02 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:27	trim_vmon_buck1_rdiv_uv	R/W	10001b	
26:22	trim_vmon_buck1_vref_dac	R/W	10000b	
21:17	trim_vmon_buck1_lshift_uv	R/W	1111b	
16:12	trim_vmon_buck1_lshift_ov	R/W	1111b	
11:7	trim_safety_ibias	R/W	10001b	
6:0	trim_osc_52mhz	R/W	101011b	

3.2.4 EEPROM_0_03 Register (Offset = 0x3) [Reset = 0x1EF8C1EF]

EEPROM_0_03 is shown in Figure 3-198 and described in Table 3-202.

Return to the Table 3-197.

Figure 3-198. EEPROM_0_03 Register

31	30	29	28	27	26	25	24
swc_buck_hiz_2	swc_buck_hiz_1	trim_vmon_buck3_lshift_uv					trim_vmon_buck3_lshift_ov
R/W-0b	R/W-0b	R/W-1111b					R/W-1111b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
trim_vmon_buck3_lshift_ov				trim_vmon_buck2_rdiv_uv			
R/W-1111b				R/W-10001b			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
trim_vmon_buck2_rdiv_uv	trim_vmon_buck2_vref_dac					trim_vmon_buck2_lshift_uv	
R/W-10001b	R/W-10000b					R/W-1111b	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
trim_vmon_buck2_lshift_uv			trim_vmon_buck2_lshift_ov				
R/W-1111b			R/W-1111b				

Table 3-202. EEPROM_0_03 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	swc_buck_hiz_2	R/W	0b	
30	swc_buck_hiz_1	R/W	0b	
29:25	trim_vmon_buck3_lshift_uv	R/W	1111b	
24:20	trim_vmon_buck3_lshift_ov	R/W	1111b	
19:15	trim_vmon_buck2_rdiv_uv	R/W	10001b	
14:10	trim_vmon_buck2_vref_dac	R/W	10000b	
9:5	trim_vmon_buck2_lshift_uv	R/W	1111b	
4:0	trim_vmon_buck2_lshift_ov	R/W	1111b	

3.2.5 EEPROM_0_04 Register (Offset = 0x4) [Reset = 0x2307BE30]

EEPROM_0_04 is shown in Figure 3-199 and described in Table 3-203.

Return to the Table 3-197.

Figure 3-199. EEPROM_0_04 Register

31	30	29	28	27	26	25	24
swc_buck_hiz_4	swc_buck_hiz_3	trim_vmon_buck4_rdiv_uv					trim_vmon_buck4_vref_dac
R/W-0b	R/W-0b	R/W-10001b					R/W-10000b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
trim_vmon_buck4_vref_dac				trim_vmon_buck4_lshift_uv			
R/W-10000b				R/W-1111b			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
trim_vmon_buck4_lshift_uv	trim_vmon_buck4_lshift_ov					trim_vmon_buck3_rdiv_uv	
R/W-1111b	R/W-1111b					R/W-10001b	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
trim_vmon_buck3_rdiv_uv			trim_vmon_buck3_vref_dac				
R/W-10001b			R/W-10000b				

Table 3-203. EEPROM_0_04 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	swc_buck_hiz_4	R/W	0b	
30	swc_buck_hiz_3	R/W	0b	
29:25	trim_vmon_buck4_rdiv_uv	R/W	10001b	
24:20	trim_vmon_buck4_vref_dac	R/W	10000b	
19:15	trim_vmon_buck4_lshift_uv	R/W	1111b	
14:10	trim_vmon_buck4_lshift_ov	R/W	1111b	
9:5	trim_vmon_buck3_rdiv_uv	R/W	10001b	
4:0	trim_vmon_buck3_vref_dac	R/W	10000b	

3.2.6 EEPROM_0_05 Register (Offset = 0x5) [Reset = 0x1EF8C1EF]

EEPROM_0_05 is shown in Figure 3-200 and described in Table 3-204.

Return to the Table 3-197.

Figure 3-200. EEPROM_0_05 Register

31	30	29	28	27	26	25	24
RESERVED	swc_buck_hiz_5	trim_vmon_ldo1_lshift_uv					trim_vmon_ldo1_lshift_ov
R-	R/W-0b	R/W-1111b					R/W-1111b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
trim_vmon_ldo1_lshift_ov				trim_vmon_buck5_rdiv_uv			
R/W-1111b				R/W-10001b			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
trim_vmon_buck5_rdiv_uv	trim_vmon_buck5_vref_dac					trim_vmon_buck5_lshift_uv	
R/W-10001b	R/W-10000b					R/W-1111b	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
trim_vmon_buck5_lshift_uv			trim_vmon_buck5_lshift_ov				
R/W-1111b			R/W-1111b				

Table 3-204. EEPROM_0_05 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0b	
30	swc_buck_hiz_5	R/W	0b	
29:25	trim_vmon_ldo1_lshift_uv	R/W	1111b	
24:20	trim_vmon_ldo1_lshift_ov	R/W	1111b	
19:15	trim_vmon_buck5_rdiv_uv	R/W	10001b	
14:10	trim_vmon_buck5_vref_dac	R/W	10000b	
9:5	trim_vmon_buck5_lshift_uv	R/W	1111b	
4:0	trim_vmon_buck5_lshift_ov	R/W	1111b	

3.2.7 EEPROM_0_06 Register (Offset = 0x6) [Reset = 0x2307BE30]

EEPROM_0_06 is shown in Figure 3-201 and described in Table 3-205.

Return to the Table 3-197.

Figure 3-201. EEPROM_0_06 Register

31	30	29	28	27	26	25	24
RESERVED		trim_vmon_ldo2_rdiv_uv					trim_vmon_ldo2_vref_dac
R-		R/W-10001b					R/W-10000b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
trim_vmon_ldo2_vref_dac				trim_vmon_ldo2_lshift_uv			
R/W-10000b				R/W-1111b			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
trim_vmon_ldo2_lshift_uv	trim_vmon_ldo2_lshift_ov					trim_vmon_ldo1_rdiv_uv	
R/W-1111b	R/W-1111b					R/W-10001b	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
trim_vmon_ldo1_rdiv_uv			trim_vmon_ldo1_vref_dac				
R/W-10001b			R/W-10000b				

Table 3-205. EEPROM_0_06 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:30	RESERVED	R	0b	
29:25	trim_vmon_ldo2_rdiv_uv	R/W	10001b	
24:20	trim_vmon_ldo2_vref_dac	R/W	10000b	
19:15	trim_vmon_ldo2_lshift_uv	R/W	1111b	
14:10	trim_vmon_ldo2_lshift_ov	R/W	1111b	
9:5	trim_vmon_ldo1_rdiv_uv	R/W	10001b	
4:0	trim_vmon_ldo1_vref_dac	R/W	10000b	

3.2.8 EEPROM_0_07 Register (Offset = 0x7) [Reset = 0x1EF8C1EF]

EEPROM_0_07 is shown in Figure 3-202 and described in Table 3-206.

Return to the Table 3-197.

Figure 3-202. EEPROM_0_07 Register

31	30	29	28	27	26	25	24
RESERVED		trim_vmon_ldo4_lshift_uv					trim_vmon_ldo4_lshift_ov
R-		R/W-1111b					R/W-1111b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
trim_vmon_ldo4_lshift_ov				trim_vmon_ldo3_rdiv_uv			
R/W-1111b				R/W-10001b			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
trim_vmon_ldo3_rdiv_uv	trim_vmon_ldo3_vref_dac					trim_vmon_ldo3_lshift_uv	
R/W-10001b	R/W-10000b					R/W-1111b	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
trim_vmon_ldo3_lshift_uv			trim_vmon_ldo3_lshift_ov				
R/W-1111b			R/W-1111b				

Table 3-206. EEPROM_0_07 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:30	RESERVED	R	0b	
29:25	trim_vmon_ldo4_lshift_uv	R/W	1111b	
24:20	trim_vmon_ldo4_lshift_ov	R/W	1111b	
19:15	trim_vmon_ldo3_rdiv_uv	R/W	10001b	
14:10	trim_vmon_ldo3_vref_dac	R/W	10000b	
9:5	trim_vmon_ldo3_lshift_uv	R/W	1111b	
4:0	trim_vmon_ldo3_lshift_ov	R/W	1111b	

3.2.9 EEPROM_0_08 Register (Offset = 0x8) [Reset = 0x2AF84230]

EEPROM_0_08 is shown in Figure 3-203 and described in Table 3-207.

Return to the Table 3-197.

Figure 3-203. EEPROM_0_08 Register

31	30	29	28	27	26	25	24
RESERVED		trim_vsys_mon_rdiv_1v2					trim_vsys_mon_bg
R-		R/W-10101b					R/W-1111b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
trim_vsys_mon_bg				trim_tsd_gain			
R/W-1111b				R/W-10000b			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
trim_tsd_gain	trim_tsd_offset					trim_vmon_ldo4_rdiv_uv	
R/W-10000b	R/W-10000b					R/W-10001b	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
trim_vmon_ldo4_rdiv_uv			trim_vmon_ldo4_vref_dac				
R/W-10001b			R/W-10000b				

Table 3-207. EEPROM_0_08 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:30	RESERVED	R	0b	
29:25	trim_vsys_mon_rdiv_1v2	R/W	10101b	
24:20	trim_vsys_mon_bg	R/W	1111b	
19:15	trim_tsd_gain	R/W	10000b	
14:10	trim_tsd_offset	R/W	10000b	
9:5	trim_vmon_ldo4_rdiv_uv	R/W	10001b	
4:0	trim_vmon_ldo4_vref_dac	R/W	10000b	

3.2.10 EEPROM_0_09 Register (Offset = 0x9) [Reset = 0x082042AF]

EEPROM_0_09 is shown in Figure 3-204 and described in Table 3-208.

Return to the Table 3-197.

Figure 3-204. EEPROM_0_09 Register

31	30	29	28	27	26	25	24
RESERVED				trim_ldo1_ref			
R-				R/W-100000b			
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
trim_ldo1_ref		trim_dco					
R/W-100000b		R/W-1000000b					
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
trim_dco	trim_rtc_ldo					trim_rtc_rdiv_1v2	
R/W-1000000b	R/W-10000b					R/W-10101b	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
trim_rtc_rdiv_1v2			trim_rtc_bg				
R/W-10101b			R/W-1111b				

Table 3-208. EEPROM_0_09 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:28	RESERVED	R	0b	
27:22	trim_ldo1_ref	R/W	100000b	
21:15	trim_dco	R/W	1000000b	
14:10	trim_rtc_ldo	R/W	10000b	
9:5	trim_rtc_rdiv_1v2	R/W	10101b	
4:0	trim_rtc_bg	R/W	1111b	

3.2.11 EEPROM_0_10 Register (Offset = 0xA) [Reset = 0x22082220]

EEPROM_0_10 is shown in Figure 3-205 and described in Table 3-209.

Return to the Table 3-197.

Figure 3-205. EEPROM_0_10 Register

31	30	29	28	27	26	25	24
RESERVED		trim_ldo2_ilim1				trim_ldo2	
R-		R/W-1000b				R/W-100000b	
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
trim_ldo2				trim_ldo2_ref			
R/W-100000b				R/W-100000b			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
trim_ldo2_ref		trim_ldo1_ilim2				trim_ldo1_ilim1	
R/W-100000b		R/W-1000b				R/W-1000b	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
trim_ldo1_ilim1		trim_ldo1					
R/W-1000b		R/W-100000b					

Table 3-209. EEPROM_0_10 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:30	RESERVED	R	0b	
29:26	trim_ldo2_ilim1	R/W	1000b	
25:20	trim_ldo2	R/W	100000b	
19:14	trim_ldo2_ref	R/W	100000b	
13:10	trim_ldo1_ilim2	R/W	1000b	
9:6	trim_ldo1_ilim1	R/W	1000b	
5:0	trim_ldo1	R/W	100000b	

3.2.12 EEPROM_0_11 Register (Offset = 0xB) [Reset = 0x8F888208]

EEPROM_0_11 is shown in Figure 3-206 and described in Table 3-210.

Return to the Table 3-197.

Figure 3-206. EEPROM_0_11 Register

31	30	29	28	27	26	25	24
trim_ldo4_sr_fast			trim_ldo4_low				
100b			R/W-1111b				
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
trim_ldo3_ilim2				trim_ldo3_ilim1			
R/W-1000b				R/W-1000b			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
trim_ldo3						trim_ldo3_ref	
R/W-100000b						R/W-100000b	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
trim_ldo3_ref				trim_ldo2_ilim2			
R/W-100000b				R/W-1000b			

Table 3-210. EEPROM_0_11 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:29	trim_ldo4_sr_fast		100b	
28:24	trim_ldo4_low	R/W	1111b	
23:20	trim_ldo3_ilim2	R/W	1000b	
19:16	trim_ldo3_ilim1	R/W	1000b	
15:10	trim_ldo3	R/W	100000b	
9:4	trim_ldo3_ref	R/W	100000b	
3:0	trim_ldo2_ilim2	R/W	1000b	

3.2.13 EEPROM_0_12 Register (Offset = 0xC) [Reset = 0x8DEE8D2E]

EEPROM_0_12 is shown in Figure 3-207 and described in Table 3-211.

Return to the Table 3-197.

Figure 3-207. EEPROM_0_12 Register

31	30	29	28	27	26	25	24
trim_ldo4_sr_slow			trim_vout_adc_gain_1			trim_loop_comp_1	
100b			R/W-11b			R/W-1111b	

23	22	21	20	19	18	17	16
trim_loop_comp_1			trim_level_shift_gain_1				
R/W-1111b			R/W-1110b				

15	14	13	12	11	10	9	8
trim_vref_dac_1					trim_ldo4_ilim		
R/W-10001b					R/W-1010b		

7	6	5	4	3	2	1	0
trim_ldo4_ilim	trim_ldo4_high						
R/W-1010b	R/W-101110b						

Table 3-211. EEPROM_0_12 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:29	trim_ldo4_sr_slow		100b	
28:26	trim_vout_adc_gain_1	R/W	11b	
25:21	trim_loop_comp_1	R/W	1111b	
20:16	trim_level_shift_gain_1	R/W	1110b	
15:11	trim_vref_dac_1	R/W	10001b	
10:7	trim_ldo4_ilim	R/W	1010b	
6:0	trim_ldo4_high	R/W	101110b	

3.2.14 EEPROM_0_13 Register (Offset = 0xD) [Reset = 0x10F7C447]

EEPROM_0_13 is shown in Figure 3-208 and described in Table 3-212.

Return to the Table 3-197.

Figure 3-208. EEPROM_0_13 Register

31	30	29	28	27	26	25	24
RESERVED			trim_fb_gain_stage_offset_1				trim_ls_replica_1
R-			R/W-1000b				R/W-1111b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
trim_ls_replica_1				trim_hs_replica_1			
R/W-1111b				R/W-1111b			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
trim_hs_replica_1	trim_artif_ramp_1				trim_emu_ramp_1		
R/W-1111b	R/W-1000b				R/W-1000b		
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
trim_emu_ramp_1	trim_vout_adc_ptat_1			trim_vout_adc_offset_1			
R/W-1000b	R/W-100b			R/W-111b			

Table 3-212. EEPROM_0_13 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:29	RESERVED	R	0b	
28:25	trim_fb_gain_stage_offset_1	R/W	1000b	
24:20	trim_ls_replica_1	R/W	1111b	
19:15	trim_hs_replica_1	R/W	1111b	
14:11	trim_artif_ramp_1	R/W	1000b	
10:7	trim_emu_ramp_1	R/W	1000b	
6:4	trim_vout_adc_ptat_1	R/W	100b	
3:0	trim_vout_adc_offset_1	R/W	111b	

3.2.15 EEPROM_0_14 Register (Offset = 0xE) [Reset = 0x1BDD1088]

EEPROM_0_14 is shown in Figure 3-209 and described in Table 3-213.

Return to the Table 3-197.

Figure 3-209. EEPROM_0_14 Register

31	30	29	28	27	26	25	24
RESERVED		trim_vout_adc_gain_2			trim_loop_comp_2		
R-		R/W-11b			R/W-1111b		
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
trim_loop_comp_2		trim_level_shift_gain_2					trim_vref_dac_2
R/W-1111b		R/W-1110b					R/W-10001b
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
trim_vref_dac_2				trim_spare_1			
R/W-10001b				R/W-0b			
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
trim_slope_1				trim_integ_dac_gain_1			
R/W-1000b				R/W-1000b			

Table 3-213. EEPROM_0_14 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:30	RESERVED	R	0b	
29:27	trim_vout_adc_gain_2	R/W	11b	
26:22	trim_loop_comp_2	R/W	1111b	
21:17	trim_level_shift_gain_2	R/W	1110b	
16:12	trim_vref_dac_2	R/W	10001b	
11:8	trim_spare_1	R/W	0b	
7:4	trim_slope_1	R/W	1000b	
3:0	trim_integ_dac_gain_1	R/W	1000b	

3.2.16 EEPROM_0_15 Register (Offset = 0xF) [Reset = 0x10F7C447]

EEPROM_0_15 is shown in Figure 3-210 and described in Table 3-214.

Return to the Table 3-197.

Figure 3-210. EEPROM_0_15 Register

31	30	29	28	27	26	25	24
RESERVED			trim_fb_gain_stage_offset_2				trim_ls_replica_2
R-			R/W-1000b				R/W-1111b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
trim_ls_replica_2				trim_hs_replica_2			
R/W-1111b				R/W-1111b			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
trim_hs_replica_2	trim_artif_ramp_2				trim_emu_ramp_2		
R/W-1111b	R/W-1000b				R/W-1000b		
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
trim_emu_ramp_2	trim_vout_adc_ptat_2			trim_vout_adc_offset_2			
R/W-1000b	R/W-100b			R/W-111b			

Table 3-214. EEPROM_0_15 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:29	RESERVED	R	0b	
28:25	trim_fb_gain_stage_offset_2	R/W	1000b	
24:20	trim_ls_replica_2	R/W	1111b	
19:15	trim_hs_replica_2	R/W	1111b	
14:11	trim_artif_ramp_2	R/W	1000b	
10:7	trim_emu_ramp_2	R/W	1000b	
6:4	trim_vout_adc_ptat_2	R/W	100b	
3:0	trim_vout_adc_offset_2	R/W	111b	

3.2.17 EEPROM_0_16 Register (Offset = 0x10) [Reset = 0x1BDD1088]

EEPROM_0_16 is shown in Figure 3-211 and described in Table 3-215.

Return to the Table 3-197.

Figure 3-211. EEPROM_0_16 Register

31	30	29	28	27	26	25	24
RESERVED		trim_vout_adc_gain_3			trim_loop_comp_3		
R-		R/W-11b			R/W-1111b		
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
trim_loop_comp_3		trim_level_shift_gain_3					trim_vref_dac_3
R/W-1111b		R/W-1110b					R/W-10001b
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
trim_vref_dac_3				trim_spare_2			
R/W-10001b				R/W-0b			
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
trim_slope_2				trim_integ_dac_gain_2			
R/W-1000b				R/W-1000b			

Table 3-215. EEPROM_0_16 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:30	RESERVED	R	0b	
29:27	trim_vout_adc_gain_3	R/W	11b	
26:22	trim_loop_comp_3	R/W	1111b	
21:17	trim_level_shift_gain_3	R/W	1110b	
16:12	trim_vref_dac_3	R/W	10001b	
11:8	trim_spare_2	R/W	0b	
7:4	trim_slope_2	R/W	1000b	
3:0	trim_integ_dac_gain_2	R/W	1000b	

3.2.18 EEPROM_0_17 Register (Offset = 0x11) [Reset = 0x10F7C447]

EEPROM_0_17 is shown in Figure 3-212 and described in Table 3-216.

Return to the Table 3-197.

Figure 3-212. EEPROM_0_17 Register

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td></tr>
<tr><td colspan="3">RESERVED</td><td colspan="4">trim_fb_gain_stage_offset_3</td><td>trim_ls_replica_3</td></tr>
<tr><td colspan="3">R-</td><td colspan="4">R/W-1000b</td><td>R/W-1111b</td></tr>
<tr><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="4">trim_ls_replica_3</td><td colspan="4">trim_hs_replica_3</td></tr>
<tr><td colspan="4">R/W-1111b</td><td colspan="4">R/W-1111b</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr>
<tr><td>trim_hs_replica_3</td><td colspan="4">trim_artif_ramp_3</td><td colspan="3">trim_emu_ramp_3</td></tr>
<tr><td>R/W-1111b</td><td colspan="4">R/W-1000b</td><td colspan="3">R/W-1000b</td></tr>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>trim_emu_ramp_3</td><td colspan="3">trim_vout_adc_ptat_3</td><td colspan="4">trim_vout_adc_offset_3</td></tr>
<tr><td>R/W-1000b</td><td colspan="3">R/W-100b</td><td colspan="4">R/W-111b</td></tr>
</table>

Table 3-216. EEPROM_0_17 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:29	RESERVED	R	0b	
28:25	trim_fb_gain_stage_offset_3	R/W	1000b	
24:20	trim_ls_replica_3	R/W	1111b	
19:15	trim_hs_replica_3	R/W	1111b	
14:11	trim_artif_ramp_3	R/W	1000b	
10:7	trim_emu_ramp_3	R/W	1000b	
6:4	trim_vout_adc_ptat_3	R/W	100b	
3:0	trim_vout_adc_offset_3	R/W	111b	

3.2.19 EEPROM_0_18 Register (Offset = 0x12) [Reset = 0x1BDD1088]

EEPROM_0_18 is shown in Figure 3-213 and described in Table 3-217.

Return to the Table 3-197.

Figure 3-213. EEPROM_0_18 Register

31	30	29	28	27	26	25	24
RESERVED		trim_vout_adc_gain_4			trim_loop_comp_4		
R-		R/W-11b			R/W-1111b		
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
trim_loop_comp_4		trim_level_shift_gain_4					trim_vref_dac_4
R/W-1111b		R/W-1110b					R/W-10001b
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
trim_vref_dac_4				trim_spare_3			
R/W-10001b				R/W-0b			
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
trim_slope_3				trim_integ_dac_gain_3			
R/W-1000b				R/W-1000b			

Table 3-217. EEPROM_0_18 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:30	RESERVED	R	0b	
29:27	trim_vout_adc_gain_4	R/W	11b	
26:22	trim_loop_comp_4	R/W	1111b	
21:17	trim_level_shift_gain_4	R/W	1110b	
16:12	trim_vref_dac_4	R/W	10001b	
11:8	trim_spare_3	R/W	0b	
7:4	trim_slope_3	R/W	1000b	
3:0	trim_integ_dac_gain_3	R/W	1000b	

3.2.20 EEPROM_0_19 Register (Offset = 0x13) [Reset = 0x10F7C447]

EEPROM_0_19 is shown in Figure 3-214 and described in Table 3-218.

Return to the Table 3-197.

Figure 3-214. EEPROM_0_19 Register

31	30	29	28	27	26	25	24
RESERVED			trim_fb_gain_stage_offset_4				trim_ls_replica_4
R-			R/W-1000b				R/W-1111b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
trim_ls_replica_4				trim_hs_replica_4			
R/W-1111b				R/W-1111b			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
trim_hs_replica_4	trim_artif_ramp_4				trim_emu_ramp_4		
R/W-1111b	R/W-1000b				R/W-1000b		
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
trim_emu_ramp_4	trim_vout_adc_ptat_4			trim_vout_adc_offset_4			
R/W-1000b	R/W-100b			R/W-111b			

Table 3-218. EEPROM_0_19 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:29	RESERVED	R	0b	
28:25	trim_fb_gain_stage_offset_4	R/W	1000b	
24:20	trim_ls_replica_4	R/W	1111b	
19:15	trim_hs_replica_4	R/W	1111b	
14:11	trim_artif_ramp_4	R/W	1000b	
10:7	trim_emu_ramp_4	R/W	1000b	
6:4	trim_vout_adc_ptat_4	R/W	100b	
3:0	trim_vout_adc_offset_4	R/W	111b	

3.2.21 EEPROM_0_20 Register (Offset = 0x14) [Reset = 0x1BDD1088]

EEPROM_0_20 is shown in Figure 3-215 and described in Table 3-219.

Return to the Table 3-197.

Figure 3-215. EEPROM_0_20 Register

31	30	29	28	27	26	25	24
RESERVED		trim_vout_adc_gain_5			trim_loop_comp_5		
R-		R/W-11b			R/W-1111b		
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
trim_loop_comp_5		trim_level_shift_gain_5					trim_vref_dac_5
R/W-1111b		R/W-1110b					R/W-10001b
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
trim_vref_dac_5				trim_spare_4			
R/W-10001b				R/W-0b			
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
trim_slope_4				trim_integ_dac_gain_4			
R/W-1000b				R/W-1000b			

Table 3-219. EEPROM_0_20 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:30	RESERVED	R	0b	
29:27	trim_vout_adc_gain_5	R/W	11b	
26:22	trim_loop_comp_5	R/W	1111b	
21:17	trim_level_shift_gain_5	R/W	1110b	
16:12	trim_vref_dac_5	R/W	10001b	
11:8	trim_spare_4	R/W	0b	
7:4	trim_slope_4	R/W	1000b	
3:0	trim_integ_dac_gain_4	R/W	1000b	

3.2.22 EEPROM_0_21 Register (Offset = 0x15) [Reset = 0x10F7C447]

EEPROM_0_21 is shown in Figure 3-216 and described in Table 3-220.

Return to the Table 3-197.

Figure 3-216. EEPROM_0_21 Register

31	30	29	28	27	26	25	24
RESERVED			trim_fb_gain_stage_offset_5				trim_ls_replica_5
R-			R/W-1000b				R/W-1111b

23	22	21	20	19	18	17	16
trim_ls_replica_5				trim_hs_replica_5			
R/W-1111b				R/W-1111b			

15	14	13	12	11	10	9	8
trim_hs_replica_5	trim_artif_ramp_5				trim_emu_ramp_5		
R/W-1111b	R/W-1000b				R/W-1000b		

7	6	5	4	3	2	1	0
trim_emu_ramp_5	trim_vout_adc_ptat_5			trim_vout_adc_offset_5			
R/W-1000b	R/W-100b			R/W-111b			

Table 3-220. EEPROM_0_21 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:29	RESERVED	R	0b	
28:25	trim_fb_gain_stage_offset_5	R/W	1000b	
24:20	trim_ls_replica_5	R/W	1111b	
19:15	trim_hs_replica_5	R/W	1111b	
14:11	trim_artif_ramp_5	R/W	1000b	
10:7	trim_emu_ramp_5	R/W	1000b	
6:4	trim_vout_adc_ptat_5	R/W	100b	
3:0	trim_vout_adc_offset_5	R/W	111b	

3.2.23 EEPROM_0_22 Register (Offset = 0x16) [Reset = 0x8002B088]

EEPROM_0_22 is shown in Figure 3-217 and described in Table 3-221.

Return to the Table 3-197.

Figure 3-217. EEPROM_0_22 Register

31	30	29	28	27	26	25	24
trim_tsd_offset2					XCOORD		
R/W-10000b					R/W-0b		
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
XCOORD					trim_monitor_osc_20mhz		
R/W-0b					R/W-101011b		
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
trim_monitor_osc_20mhz				trim_spare_5			
R/W-101011b				R/W-0b			
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
trim_slope_5				trim_integ_dac_gain_5			
R/W-1000b				R/W-1000b			

Table 3-221. EEPROM_0_22 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:27	trim_tsd_offset2	R/W	10000b	
26:19	XCOORD	R/W	0b	
18:12	trim_monitor_osc_20mhz	R/W	101011b	
11:8	trim_spare_5	R/W	0b	
7:4	trim_slope_5	R/W	1000b	
3:0	trim_integ_dac_gain_5	R/W	1000b	

3.2.24 EEPROM_0_23 Register (Offset = 0x17) [Reset = 0x0]

EEPROM_0_23 is shown in Figure 3-218 and described in Table 3-222.

Return to the Table 3-197.

Figure 3-218. EEPROM_0_23 Register

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RESERVED		WAFERLOTNUM_15_8								WAFERLOTNUM_23_16					
R-		R/W-0b								R/W-0b					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WAFERLOTNUM_23_16		WAFERNUM						YCOORD							
R/W-0b		R/W-0b						R/W-0b							

Table 3-222. EEPROM_0_23 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:30	RESERVED	R	0b	
29:22	WAFERLOTNUM_15_8	R/W	0b	
21:14	WAFERLOTNUM_23_16	R/W	0b	
13:8	WAFERNUM	R/W	0b	
7:0	YCOORD	R/W	0b	

3.2.25 EEPROM_0_24 Register (Offset = 0x18) [Reset = 0x80000000]

EEPROM_0_24 is shown in Figure 3-219 and described in Table 3-223.

Return to the Table 3-197.

Figure 3-219. EEPROM_0_24 Register

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
trim_tsd_gain2					REGMAP_TRIM_CRC16_LOW								REGMAP_TRIM_CRC16_HIGH		
R/W-10000b					R/W-0b								R/W-0b		
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
REGMAP_TRIM_CRC16_HIGH					WAFERFAB			WAFERLOTNUM_7_0							
R/W-0b					R/W-0b			R/W-0b							

Table 3-223. EEPROM_0_24 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:27	trim_tsd_gain2	R/W	10000b	
26:19	REGMAP_TRIM_CRC16_LOW	R/W	0b	
18:11	REGMAP_TRIM_CRC16_HIGH	R/W	0b	
10:8	WAFERFAB	R/W	0b	
7:0	WAFERLOTNUM_7_0	R/W	0b	

3.2.26 EEPROM_0_25 Register (Offset = 0x19) [Reset = 0x0]

EEPROM_0_25 is shown in Figure 3-220 and described in Table 3-224.

Return to the Table 3-197.

Figure 3-220. EEPROM_0_25 Register

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RESERVED								TI_NVM_REV							
R-								R/W-0b							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
TI_NVM_ID								TI_DEVICE_ID							
R/W-0b								R/W-0b							

Table 3-224. EEPROM_0_25 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:24	RESERVED	R	0b	
23:16	TI_NVM_REV	R/W	0b	
15:8	TI_NVM_ID	R/W	0b	
7:0	TI_DEVICE_ID	R/W	0b	

3.2.27 EEPROM_0_26 Register (Offset = 0x1A) [Reset = 0xA02AA02A]

EEPROM_0_26 is shown in Figure 3-221 and described in Table 3-225.

Return to the Table 3-197.

Figure 3-221. EEPROM_0_26 Register

31	30	29	28	27	26	25	24
BUCK3_RV_SEL	RESERVED	BUCK3_PLDN	BUCK3_VMON_EN	BUCK3_VSEL	BUCK3_FPWM_MP	BUCK3_FPWM	BUCK3_EN
R/W-1b	R-	R/W-1b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
RESERVED		BUCK2_ILIM			BUCK2_SLEW_RATE		
R-		R/W-101b			R/W-10b		
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
BUCK2_RV_SEL	RESERVED	BUCK2_PLDN	BUCK2_VMON_EN	BUCK2_VSEL	RESERVED	BUCK2_FPWM	BUCK2_EN
R/W-1b	R-	R/W-1b	R/W-0b	R/W-0b	R-	R/W-0b	R/W-0b
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RESERVED		BUCK1_ILIM			BUCK1_SLEW_RATE		
R-		R/W-101b			R/W-10b		

Table 3-225. EEPROM_0_26 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	BUCK3_RV_SEL	R/W	1b	
30	RESERVED	R	0b	
29	BUCK3_PLDN	R/W	1b	
28	BUCK3_VMON_EN	R/W	0b	
27	BUCK3_VSEL	R/W	0b	
26	BUCK3_FPWM_MP	R/W	0b	
25	BUCK3_FPWM	R/W	0b	
24	BUCK3_EN	R/W	0b	
23:22	RESERVED	R	0b	
21:19	BUCK2_ILIM	R/W	101b	
18:16	BUCK2_SLEW_RATE	R/W	10b	
15	BUCK2_RV_SEL	R/W	1b	
14	RESERVED	R	0b	
13	BUCK2_PLDN	R/W	1b	
12	BUCK2_VMON_EN	R/W	0b	
11	BUCK2_VSEL	R/W	0b	
10	RESERVED	R	0b	
9	BUCK2_FPWM	R/W	0b	
8	BUCK2_EN	R/W	0b	
7:6	RESERVED	R	0b	
5:3	BUCK1_ILIM	R/W	101b	
2:0	BUCK1_SLEW_RATE	R/W	10b	

3.2.28 EEPROM_0_27 Register (Offset = 0x1B) [Reset = 0xA02AA02A]

EEPROM_0_27 is shown in Figure 3-222 and described in Table 3-226.

Return to the Table 3-197.

Figure 3-222. EEPROM_0_27 Register

31	30	29	28	27	26	25	24
BUCK5_RV_SEL	RESERVED	BUCK5_PLDN	BUCK5_VMON_EN	BUCK5_VSEL	RESERVED	BUCK5_FPWM	BUCK5_EN
R/W-1b	R-	R/W-1b	R/W-0b	R/W-0b	R-	R/W-0b	R/W-0b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
RESERVED		BUCK4_ILIM			BUCK4_SLEW_RATE		
R-		R/W-101b			R/W-10b		
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
BUCK4_RV_SEL	RESERVED	BUCK4_PLDN	BUCK4_VMON_EN	BUCK4_VSEL	RESERVED	BUCK4_FPWM	BUCK4_EN
R/W-1b	R-	R/W-1b	R/W-0b	R/W-0b	R-	R/W-0b	R/W-0b
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RESERVED		BUCK3_ILIM			BUCK3_SLEW_RATE		
R-		R/W-101b			R/W-10b		

Table 3-226. EEPROM_0_27 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	BUCK5_RV_SEL	R/W	1b	
30	RESERVED	R	0b	
29	BUCK5_PLDN	R/W	1b	
28	BUCK5_VMON_EN	R/W	0b	
27	BUCK5_VSEL	R/W	0b	
26	RESERVED	R	0b	
25	BUCK5_FPWM	R/W	0b	
24	BUCK5_EN	R/W	0b	
23:22	RESERVED	R	0b	
21:19	BUCK4_ILIM	R/W	101b	
18:16	BUCK4_SLEW_RATE	R/W	10b	
15	BUCK4_RV_SEL	R/W	1b	
14	RESERVED	R	0b	
13	BUCK4_PLDN	R/W	1b	
12	BUCK4_VMON_EN	R/W	0b	
11	BUCK4_VSEL	R/W	0b	
10	RESERVED	R	0b	
9	BUCK4_FPWM	R/W	0b	
8	BUCK4_EN	R/W	0b	
7:6	RESERVED	R	0b	
5:3	BUCK3_ILIM	R/W	101b	
2:0	BUCK3_SLEW_RATE	R/W	10b	

3.2.29 EEPROM_0_28 Register (Offset = 0x1C) [Reset = 0x4141411A]

EEPROM_0_28 is shown in Figure 3-223 and described in Table 3-227.

Return to the Table 3-197.

Figure 3-223. EEPROM_0_28 Register

31	30	29	28	27	26	25	24
BUCK2_VSET1							
R/W-1000001b							

23	22	21	20	19	18	17	16
BUCK1_VSET2							
R/W-1000001b							

15	14	13	12	11	10	9	8
BUCK1_VSET1							
R/W-1000001b							

7	6	5	4	3	2	1	0
RESERVED		BUCK5_ILIM			BUCK5_SLEW_RATE		
R-		R/W-11b			R/W-10b		

Table 3-227. EEPROM_0_28 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:24	BUCK2_VSET1	R/W	1000001b	
23:16	BUCK1_VSET2	R/W	1000001b	
15:8	BUCK1_VSET1	R/W	1000001b	
7:6	RESERVED	R	0b	
5:3	BUCK5_ILIM	R/W	11b	
2:0	BUCK5_SLEW_RATE	R/W	10b	

3.2.30 EEPROM_0_29 Register (Offset = 0x1D) [Reset = 0x73373741]

EEPROM_0_29 is shown in Figure 3-224 and described in Table 3-228.

Return to the Table 3-197.

Figure 3-224. EEPROM_0_29 Register

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BUCK4_VSET1								BUCK3_VSET2							
R/W-1110011b								R/W-110111b							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BUCK3_VSET1								BUCK2_VSET2							
R/W-110111b								R/W-1000001b							

Table 3-228. EEPROM_0_29 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:24	BUCK4_VSET1	R/W	1110011b	
23:16	BUCK3_VSET2	R/W	110111b	
15:8	BUCK3_VSET1	R/W	110111b	
7:0	BUCK2_VSET2	R/W	1000001b	

3.2.31 EEPROM_0_30 Register (Offset = 0x1E) [Reset = 0x1B414173]

EEPROM_0_30 is shown in Figure 3-225 and described in Table 3-229.

Return to the Table 3-197.

Figure 3-225. EEPROM_0_30 Register

31	30	29	28	27	26	25	24
RESERVED		BUCK1_UV_THR			BUCK1_OV_THR		
R-		R/W-11b			R/W-11b		
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
BUCK5_VSET2							
R/W-1000001b							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
BUCK5_VSET1							
R/W-1000001b							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
BUCK4_VSET2							
R/W-1110011b							

Table 3-229. EEPROM_0_30 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:30	RESERVED	R	0b	
29:27	BUCK1_UV_THR	R/W	11b	
26:24	BUCK1_OV_THR	R/W	11b	
23:16	BUCK5_VSET2	R/W	1000001b	
15:8	BUCK5_VSET1	R/W	1000001b	
7:0	BUCK4_VSET2	R/W	1110011b	

3.2.32 EEPROM_0_31 Register (Offset = 0x1F) [Reset = 0xE06DB6DB]

EEPROM_0_31 is shown in Figure 3-226 and described in Table 3-230.

Return to the Table 3-197.

Figure 3-226. EEPROM_0_31 Register

31	30	29	28	27	26	25	24
LDO1_RV_SEL	LDO1_PLDN		LDO1_VMON_EN	RESERVED		LDO1_SLOW_RAMP	LDO1_EN
R/W-1b	R/W-11b		R/W-0b	R-		R/W-0b	R/W-0b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
BUCK5_UV_THR			BUCK5_OV_THR			BUCK4_UV_THR	
R/W-11b			R/W-11b			R/W-11b	
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
BUCK4_UV_THR	BUCK4_OV_THR			BUCK3_UV_THR			BUCK3_OV_THR
R/W-11b	R/W-11b			R/W-11b			R/W-11b
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
BUCK3_OV_THR		BUCK2_UV_THR			BUCK2_OV_THR		
R/W-11b		R/W-11b			R/W-11b		

Table 3-230. EEPROM_0_31 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	LDO1_RV_SEL	R/W	1b	
30:29	LDO1_PLDN	R/W	11b	
28	LDO1_VMON_EN	R/W	0b	
27:26	RESERVED	R	0b	
25	LDO1_SLOW_RAMP	R/W	0b	
24	LDO1_EN	R/W	0b	
23:21	BUCK5_UV_THR	R/W	11b	
20:18	BUCK5_OV_THR	R/W	11b	
17:15	BUCK4_UV_THR	R/W	11b	
14:12	BUCK4_OV_THR	R/W	11b	
11:9	BUCK3_UV_THR	R/W	11b	
8:6	BUCK3_OV_THR	R/W	11b	
5:3	BUCK2_UV_THR	R/W	11b	
2:0	BUCK2_OV_THR	R/W	11b	

3.2.33 EEPROM_0_32 Register (Offset = 0x20) [Reset = 0x00E0E0E0]

EEPROM_0_32 is shown in Figure 3-227 and described in Table 3-231.

Return to the Table 3-197.

Figure 3-227. EEPROM_0_32 Register

31	30	29	28	27	26	25	24
RESERVED							
R-							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
LDO4_RV_SEL	LDO4_PLDN		LDO4_VMON_EN	RESERVED		LDO4_SLOW_RAMP	LDO4_EN
R/W-1b	R/W-11b		R/W-0b	R-		R/W-0b	R/W-0b
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
LDO3_RV_SEL	LDO3_PLDN		LDO3_VMON_EN	RESERVED		LDO3_SLOW_RAMP	LDO3_EN
R/W-1b	R/W-11b		R/W-0b	R-		R/W-0b	R/W-0b
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
LDO2_RV_SEL	LDO2_PLDN		LDO2_VMON_EN	RESERVED		LDO2_SLOW_RAMP	LDO2_EN
R/W-1b	R/W-11b		R/W-0b	R-		R/W-0b	R/W-0b

Table 3-231. EEPROM_0_32 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:24	RESERVED	R	0b	
23	LDO4_RV_SEL	R/W	1b	
22:21	LDO4_PLDN	R/W	11b	
20	LDO4_VMON_EN	R/W	0b	
19:18	RESERVED	R	0b	
17	LDO4_SLOW_RAMP	R/W	0b	
16	LDO4_EN	R/W	0b	
15	LDO3_RV_SEL	R/W	1b	
14:13	LDO3_PLDN	R/W	11b	
12	LDO3_VMON_EN	R/W	0b	
11:10	RESERVED	R	0b	
9	LDO3_SLOW_RAMP	R/W	0b	
8	LDO3_EN	R/W	0b	
7	LDO2_RV_SEL	R/W	1b	
6:5	LDO2_PLDN	R/W	11b	
4	LDO2_VMON_EN	R/W	0b	
3:2	RESERVED	R	0b	
1	LDO2_SLOW_RAMP	R/W	0b	
0	LDO2_EN	R/W	0b	

3.2.34 EEPROM_0_33 Register (Offset = 0x21) [Reset = 0x1038F400]

EEPROM_0_33 is shown in Figure 3-228 and described in Table 3-232.

Return to the Table 3-197.

Figure 3-228. EEPROM_0_33 Register

31	30	29	28	27	26	25	24
LDO3_BYPASS	LDO3_VSET						RESERVED
R/W-0b	R/W-1000b						R-
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
LDO2_BYPASS	LDO2_VSET						RESERVED
R/W-0b	R/W-11100b						R-
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
LDO1_BYPASS	LDO1_VSET						RESERVED
R/W-1b	R/W-111010b						R-
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RESERVED							
R-							

Table 3-232. EEPROM_0_33 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	LDO3_BYPASS	R/W	0b	
30:25	LDO3_VSET	R/W	1000b	
24	RESERVED	R	0b	
23	LDO2_BYPASS	R/W	0b	
22:17	LDO2_VSET	R/W	11100b	
16	RESERVED	R	0b	
15	LDO1_BYPASS	R/W	1b	
14:9	LDO1_VSET	R/W	111010b	
8:0	RESERVED	R	0b	

3.2.35 EEPROM_0_34 Register (Offset = 0x22) [Reset = 0x36DB6DB9]

EEPROM_0_34 is shown in Figure 3-229 and described in Table 3-233.

Return to the Table 3-197.

Figure 3-229. EEPROM_0_34 Register

31	30	29	28	27	26	25	24
RESERVED	LDO4_UV_THR			LDO4_OV_THR			LDO3_UV_THR
R-	R/W-11b			R/W-11b			R/W-11b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
LDO3_UV_THR		LDO3_OV_THR			LDO2_UV_THR		
R/W-11b		R/W-11b			R/W-11b		
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
LDO2_OV_THR			LDO1_UV_THR			LDO1_OV_THR	
R/W-11b			R/W-11b			R/W-11b	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
LDO1_OV_THR	LDO4_VSET						
R/W-11b	R/W-111001b						

Table 3-233. EEPROM_0_34 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0b	
30:28	LDO4_UV_THR	R/W	11b	
27:25	LDO4_OV_THR	R/W	11b	
24:22	LDO3_UV_THR	R/W	11b	
21:19	LDO3_OV_THR	R/W	11b	
18:16	LDO2_UV_THR	R/W	11b	
15:13	LDO2_OV_THR	R/W	11b	
12:10	LDO1_UV_THR	R/W	11b	
9:7	LDO1_OV_THR	R/W	11b	
6:0	LDO4_VSET	R/W	111001b	

3.2.36 EEPROM_0_35 Register (Offset = 0x23) [Reset = 0x4242223F]

EEPROM_0_35 is shown in Figure 3-230 and described in Table 3-234.

Return to the Table 3-197.

Figure 3-230. EEPROM_0_35 Register

31	30	29	28	27	26	25	24
GPIO3_SEL			GPIO3_DEGLITCH_EN	GPIO3_PU_PD_EN	GPIO3_PU_SEL	GPIO3_OD	GPIO3_DIR
R/W-10b			R/W-0b	R/W-0b	R/W-0b	R/W-1b	R/W-0b
23	22	21	20	19	18	17	16
GPIO2_SEL			GPIO2_DEGLITCH_EN	GPIO2_PU_PD_EN	GPIO2_PU_SEL	GPIO2_OD	GPIO2_DIR
R/W-10b			R/W-0b	R/W-0b	R/W-0b	R/W-1b	R/W-0b
15	14	13	12	11	10	9	8
GPIO1_SEL			GPIO1_DEGLITCH_EN	GPIO1_PU_PD_EN	GPIO1_PU_SEL	GPIO1_OD	GPIO1_DIR
R/W-1b			R/W-0b	R/W-0b	R/W-0b	R/W-1b	R/W-0b
7	6	5	4	3	2	1	0
RESERVED	VCCA_PG_SET	VCCA_UV_THR			VCCA_OV_THR		
R-	R/W-0b	R/W-111b			R/W-111b		

Table 3-234. EEPROM_0_35 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:29	GPIO3_SEL	R/W	10b	
28	GPIO3_DEGLITCH_EN	R/W	0b	
27	GPIO3_PU_PD_EN	R/W	0b	
26	GPIO3_PU_SEL	R/W	0b	
25	GPIO3_OD	R/W	1b	
24	GPIO3_DIR	R/W	0b	
23:21	GPIO2_SEL	R/W	10b	
20	GPIO2_DEGLITCH_EN	R/W	0b	
19	GPIO2_PU_PD_EN	R/W	0b	
18	GPIO2_PU_SEL	R/W	0b	
17	GPIO2_OD	R/W	1b	
16	GPIO2_DIR	R/W	0b	
15:13	GPIO1_SEL	R/W	1b	
12	GPIO1_DEGLITCH_EN	R/W	0b	
11	GPIO1_PU_PD_EN	R/W	0b	
10	GPIO1_PU_SEL	R/W	0b	
9	GPIO1_OD	R/W	1b	
8	GPIO1_DIR	R/W	0b	
7	RESERVED	R	0b	
6	VCCA_PG_SET	R/W	0b	
5:3	VCCA_UV_THR	R/W	111b	
2:0	VCCA_OV_THR	R/W	111b	

3.2.37 EEPROM_0_36 Register (Offset = 0x24) [Reset = 0x2A2121DA]

EEPROM_0_36 is shown in Figure 3-231 and described in Table 3-235.

Return to the Table 3-197.

Figure 3-231. EEPROM_0_36 Register

31	30	29	28	27	26	25	24
GPIO7_SEL			GPIO7_DEGLITCH_EN	GPIO7_PU_PD_EN	GPIO7_PU_SEL	GPIO7_OD	GPIO7_DIR
R/W-1b			R/W-0b	R/W-1b	R/W-0b	R/W-1b	R/W-0b
23	22	21	20	19	18	17	16
GPIO6_SEL			GPIO6_DEGLITCH_EN	GPIO6_PU_PD_EN	GPIO6_PU_SEL	GPIO6_OD	GPIO6_DIR
R/W-1b			R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-1b
15	14	13	12	11	10	9	8
GPIO5_SEL			GPIO5_DEGLITCH_EN	GPIO5_PU_PD_EN	GPIO5_PU_SEL	GPIO5_OD	GPIO5_DIR
R/W-1b			R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-1b
7	6	5	4	3	2	1	0
GPIO4_SEL			GPIO4_DEGLITCH_EN	GPIO4_PU_PD_EN	GPIO4_PU_SEL	GPIO4_OD	GPIO4_DIR
R/W-110b			R/W-1b	R/W-1b	R/W-0b	R/W-1b	R/W-0b

Table 3-235. EEPROM_0_36 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:29	GPIO7_SEL	R/W	1b	
28	GPIO7_DEGLITCH_EN	R/W	0b	
27	GPIO7_PU_PD_EN	R/W	1b	
26	GPIO7_PU_SEL	R/W	0b	
25	GPIO7_OD	R/W	1b	
24	GPIO7_DIR	R/W	0b	
23:21	GPIO6_SEL	R/W	1b	
20	GPIO6_DEGLITCH_EN	R/W	0b	
19	GPIO6_PU_PD_EN	R/W	0b	
18	GPIO6_PU_SEL	R/W	0b	
17	GPIO6_OD	R/W	0b	
16	GPIO6_DIR	R/W	1b	
15:13	GPIO5_SEL	R/W	1b	
12	GPIO5_DEGLITCH_EN	R/W	0b	
11	GPIO5_PU_PD_EN	R/W	0b	
10	GPIO5_PU_SEL	R/W	0b	
9	GPIO5_OD	R/W	0b	
8	GPIO5_DIR	R/W	1b	
7:5	GPIO4_SEL	R/W	110b	
4	GPIO4_DEGLITCH_EN	R/W	1b	
3	GPIO4_PU_PD_EN	R/W	1b	
2	GPIO4_PU_SEL	R/W	0b	
1	GPIO4_OD	R/W	1b	
0	GPIO4_DIR	R/W	0b	

3.2.38 EEPROM_0_37 Register (Offset = 0x25) [Reset = 0x412A2372]

EEPROM_0_37 is shown in Figure 3-232 and described in Table 3-236.

Return to the Table 3-197.

Figure 3-232. EEPROM_0_37 Register

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td></tr>
<tr><td colspan="3">GPIO11_SEL</td><td>GPIO11_DEGLITCH_EN</td><td>GPIO11_PU_PD_EN</td><td>GPIO11_PU_SEL</td><td>GPIO11_OD</td><td>GPIO11_DIR</td></tr>
<tr><td colspan="3">R/W-10b</td><td>R/W-0b</td><td>R/W-0b</td><td>R/W-0b</td><td>R/W-0b</td><td>R/W-1b</td></tr>
<tr><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="3">GPIO10_SEL</td><td>GPIO10_DEGLITCH_EN</td><td>GPIO10_PU_PD_EN</td><td>GPIO10_PU_SEL</td><td>GPIO10_OD</td><td>GPIO10_DIR</td></tr>
<tr><td colspan="3">R/W-1b</td><td>R/W-0b</td><td>R/W-1b</td><td>R/W-0b</td><td>R/W-1b</td><td>R/W-0b</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr>
<tr><td colspan="3">GPIO9_SEL</td><td>GPIO9_DEGLITCH_EN</td><td>GPIO9_PU_PD_EN</td><td>GPIO9_PU_SEL</td><td>GPIO9_OD</td><td>GPIO9_DIR</td></tr>
<tr><td colspan="3">R/W-1b</td><td>R/W-0b</td><td>R/W-0b</td><td>R/W-0b</td><td>R/W-1b</td><td>R/W-1b</td></tr>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="3">GPIO8_SEL</td><td>GPIO8_DEGLITCH_EN</td><td>GPIO8_PU_PD_EN</td><td>GPIO8_PU_SEL</td><td>GPIO8_OD</td><td>GPIO8_DIR</td></tr>
<tr><td colspan="3">R/W-11b</td><td>R/W-1b</td><td>R/W-0b</td><td>R/W-0b</td><td>R/W-1b</td><td>R/W-0b</td></tr>
</table>

Table 3-236. EEPROM_0_37 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:29	GPIO11_SEL	R/W	10b	
28	GPIO11_DEGLITCH_EN	R/W	0b	
27	GPIO11_PU_PD_EN	R/W	0b	
26	GPIO11_PU_SEL	R/W	0b	
25	GPIO11_OD	R/W	0b	
24	GPIO11_DIR	R/W	1b	
23:21	GPIO10_SEL	R/W	1b	
20	GPIO10_DEGLITCH_EN	R/W	0b	
19	GPIO10_PU_PD_EN	R/W	1b	
18	GPIO10_PU_SEL	R/W	0b	
17	GPIO10_OD	R/W	1b	
16	GPIO10_DIR	R/W	0b	
15:13	GPIO9_SEL	R/W	1b	
12	GPIO9_DEGLITCH_EN	R/W	0b	
11	GPIO9_PU_PD_EN	R/W	0b	
10	GPIO9_PU_SEL	R/W	0b	
9	GPIO9_OD	R/W	1b	
8	GPIO9_DIR	R/W	1b	
7:5	GPIO8_SEL	R/W	11b	
4	GPIO8_DEGLITCH_EN	R/W	1b	
3	GPIO8_PU_PD_EN	R/W	0b	
2	GPIO8_PU_SEL	R/W	0b	
1	GPIO8_OD	R/W	1b	
0	GPIO8_DIR	R/W	0b	

3.2.39 EEPROM_0_38 Register (Offset = 0x26) [Reset = 0x05500018]

EEPROM_0_38 is shown in Figure 3-233 and described in Table 3-237.

Return to the Table 3-197.

Figure 3-233. EEPROM_0_38 Register

31	30	29	28	27	26	25	24
RESERVED					BUCK4_GRP_SEL		BUCK3_GRP_SEL
R-					R/W-10b		R/W-10b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
BUCK3_GRP_SEL	BUCK2_GRP_SEL		BUCK1_GRP_SEL		GPIO11_OUT	GPIO10_OUT	GPIO9_OUT
R/W-10b	R/W-10b		R/W-10b		R/W-0b	R/W-0b	R/W-0b
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
GPIO8_OUT	GPIO7_OUT	GPIO6_OUT	GPIO5_OUT	GPIO4_OUT	GPIO3_OUT	GPIO2_OUT	GPIO1_OUT
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
NPWRON_SEL		ENABLE_POL	ENABLE_DEGLITCH_EN	ENABLE_PU_PD_EN	ENABLE_PU_SEL	RESERVED	NRSTOUT_OD
R/W-0b		R/W-0b	R/W-1b	R/W-1b	R/W-0b	R-	R/W-0b

Table 3-237. EEPROM_0_38 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:27	RESERVED	R	0b	
26:25	BUCK4_GRP_SEL	R/W	10b	
24:23	BUCK3_GRP_SEL	R/W	10b	
22:21	BUCK2_GRP_SEL	R/W	10b	
20:19	BUCK1_GRP_SEL	R/W	10b	
18	GPIO11_OUT	R/W	0b	
17	GPIO10_OUT	R/W	0b	
16	GPIO9_OUT	R/W	0b	
15	GPIO8_OUT	R/W	0b	
14	GPIO7_OUT	R/W	0b	
13	GPIO6_OUT	R/W	0b	
12	GPIO5_OUT	R/W	0b	
11	GPIO4_OUT	R/W	0b	
10	GPIO3_OUT	R/W	0b	
9	GPIO2_OUT	R/W	0b	
8	GPIO1_OUT	R/W	0b	
7:6	NPWRON_SEL	R/W	0b	
5	ENABLE_POL	R/W	0b	
4	ENABLE_DEGLITCH_EN	R/W	1b	
3	ENABLE_PU_PD_EN	R/W	1b	
2	ENABLE_PU_SEL	R/W	0b	
1	RESERVED	R	0b	
0	NRSTOUT_OD	R/W	0b	

3.2.40 EEPROM_0_39 Register (Offset = 0x27) [Reset = 0x15515659]

EEPROM_0_39 is shown in Figure 3-234 and described in Table 3-238.

Return to the Table 3-197.

Figure 3-234. EEPROM_0_39 Register

31	30	29	28	27	26	25	24
RESERVED		GPIO4_FSM_MASK_POL	GPIO4_FSM_MASK	GPIO3_FSM_MASK_POL	GPIO3_FSM_MASK	GPIO2_FSM_MASK_POL	GPIO2_FSM_MASK
R-		R/W-0b	R/W-1b	R/W-0b	R/W-1b	R/W-0b	R/W-1b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
GPIO1_FSM_MASK_POL	GPIO1_FSM_MASK	MODERATE_ERR_TRIG		SEVERE_ERR_TRIG		OTHER_RAIL_TRIG	
R/W-0b	R/W-1b	R/W-1b		R/W-0b		R/W-1b	
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
SOC_RAIL_TRIG		MCU_RAIL_TRIG		VCCA_GRP_SEL		LDO4_GRP_SEL	
R/W-1b		R/W-1b		R/W-1b		R/W-10b	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
LDO3_GRP_SEL		LDO2_GRP_SEL		LDO1_GRP_SEL		BUCK5_GRP_SEL	
R/W-1b		R/W-1b		R/W-10b		R/W-1b	

Table 3-238. EEPROM_0_39 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:30	RESERVED	R	0b	
29	GPIO4_FSM_MASK_POL	R/W	0b	
28	GPIO4_FSM_MASK	R/W	1b	
27	GPIO3_FSM_MASK_POL	R/W	0b	
26	GPIO3_FSM_MASK	R/W	1b	
25	GPIO2_FSM_MASK_POL	R/W	0b	
24	GPIO2_FSM_MASK	R/W	1b	
23	GPIO1_FSM_MASK_POL	R/W	0b	
22	GPIO1_FSM_MASK	R/W	1b	
21:20	MODERATE_ERR_TRIG	R/W	1b	
19:18	SEVERE_ERR_TRIG	R/W	0b	
17:16	OTHER_RAIL_TRIG	R/W	1b	
15:14	SOC_RAIL_TRIG	R/W	1b	
13:12	MCU_RAIL_TRIG	R/W	1b	
11:10	VCCA_GRP_SEL	R/W	1b	
9:8	LDO4_GRP_SEL	R/W	10b	
7:6	LDO3_GRP_SEL	R/W	1b	
5:4	LDO2_GRP_SEL	R/W	1b	
3:2	LDO1_GRP_SEL	R/W	10b	
1:0	BUCK5_GRP_SEL	R/W	1b	

3.2.41 EEPROM_0_40 Register (Offset = 0x28) [Reset = 0x88881555]

EEPROM_0_40 is shown in Figure 3-235 and described in Table 3-239.

Return to the Table 3-197.

Figure 3-235. EEPROM_0_40 Register

31	30	29	28	27	26	25	24
BUCK4_ILIM_MASK	RESERVED	BUCK4_UV_MASK	BUCK4_OV_MASK	BUCK3_ILIM_MASK	RESERVED	BUCK3_UV_MASK	BUCK3_OV_MASK
R/W-1b	R-	R/W-0b	R/W-0b	R/W-1b	R-	R/W-0b	R/W-0b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
BUCK1_ILIM_MASK	RESERVED	BUCK2_UV_MASK	BUCK2_OV_MASK	BUCK2_ILIM_MASK	RESERVED	BUCK1_UV_MASK	BUCK1_OV_MASK
R/W-1b	R-	R/W-0b	R/W-0b	R/W-1b	R-	R/W-0b	R/W-0b
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
RESERVED		GPIO11_FSM_MASK_POL	GPIO11_FSM_MASK	GPIO10_FSM_MASK_POL	GPIO10_FSM_MASK	GPIO9_FSM_MASK_POL	GPIO9_FSM_MASK
R-		R/W-0b	R/W-1b	R/W-0b	R/W-1b	R/W-0b	R/W-1b
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
GPIO8_FSM_MASK_POL	GPIO8_FSM_MASK	GPIO7_FSM_MASK_POL	GPIO7_FSM_MASK	GPIO6_FSM_MASK_POL	GPIO6_FSM_MASK	GPIO5_FSM_MASK_POL	GPIO5_FSM_MASK
R/W-0b	R/W-1b	R/W-0b	R/W-1b	R/W-0b	R/W-1b	R/W-0b	R/W-1b

Table 3-239. EEPROM_0_40 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	BUCK4_ILIM_MASK	R/W	1b	
30	RESERVED	R	0b	
29	BUCK4_UV_MASK	R/W	0b	
28	BUCK4_OV_MASK	R/W	0b	
27	BUCK3_ILIM_MASK	R/W	1b	
26	RESERVED	R	0b	
25	BUCK3_UV_MASK	R/W	0b	
24	BUCK3_OV_MASK	R/W	0b	
23	BUCK1_ILIM_MASK	R/W	1b	
22	RESERVED	R	0b	
21	BUCK2_UV_MASK	R/W	0b	
20	BUCK2_OV_MASK	R/W	0b	
19	BUCK2_ILIM_MASK	R/W	1b	
18	RESERVED	R	0b	
17	BUCK1_UV_MASK	R/W	0b	
16	BUCK1_OV_MASK	R/W	0b	
15:14	RESERVED	R	0b	
13	GPIO11_FSM_MASK_POL	R/W	0b	
12	GPIO11_FSM_MASK	R/W	1b	
11	GPIO10_FSM_MASK_POL	R/W	0b	
10	GPIO10_FSM_MASK	R/W	1b	
9	GPIO9_FSM_MASK_POL	R/W	0b	
8	GPIO9_FSM_MASK	R/W	1b	
7	GPIO8_FSM_MASK_POL	R/W	0b	

Table 3-239. EEPROM_0_40 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
6	GPIO8_FSM_MASK	R/W	1b	
5	GPIO7_FSM_MASK_POL	R/W	0b	
4	GPIO7_FSM_MASK	R/W	1b	
3	GPIO6_FSM_MASK_POL	R/W	0b	
2	GPIO6_FSM_MASK	R/W	1b	
1	GPIO5_FSM_MASK_POL	R/W	0b	
0	GPIO5_FSM_MASK	R/W	1b	

3.2.42 EEPROM_0_41 Register (Offset = 0x29) [Reset = 0x3FC88888]

EEPROM_0_41 is shown in Figure 3-236 and described in Table 3-240.

Return to the Table 3-197.

Figure 3-236. EEPROM_0_41 Register

31	30	29	28	27	26	25	24
RESERVED		GPIO8_FALL_MASK	GPIO7_FALL_MASK	GPIO6_FALL_MASK	GPIO5_FALL_MASK	GPIO4_FALL_MASK	GPIO3_FALL_MASK
R-		R/W-1b	R/W-1b	R/W-1b	R/W-1b	R/W-1b	R/W-1b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
GPIO2_FALL_MASK	GPIO1_FALL_MASK	VCCA_UV_MASK	VCCA_OV_MASK	LDO4_ILIM_MASK	RESERVED	LDO4_UV_MASK	LDO4_OV_MASK
R/W-1b	R/W-1b	R/W-0b	R/W-0b	R/W-1b	R-	R/W-0b	R/W-0b
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
LDO3_ILIM_MASK	RESERVED	LDO3_UV_MASK	LDO3_OV_MASK	LDO2_ILIM_MASK	RESERVED	LDO2_UV_MASK	LDO2_OV_MASK
R/W-1b	R-	R/W-0b	R/W-0b	R/W-1b	R-	R/W-0b	R/W-0b
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
LDO1_ILIM_MASK	RESERVED	LDO1_UV_MASK	LDO1_OV_MASK	BUCK5_ILIM_MASK	RESERVED	BUCK5_UV_MASK	BUCK5_OV_MASK
R/W-1b	R-	R/W-0b	R/W-0b	R/W-1b	R-	R/W-0b	R/W-0b

Table 3-240. EEPROM_0_41 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:30	RESERVED	R	0b	
29	GPIO8_FALL_MASK	R/W	1b	
28	GPIO7_FALL_MASK	R/W	1b	
27	GPIO6_FALL_MASK	R/W	1b	
26	GPIO5_FALL_MASK	R/W	1b	
25	GPIO4_FALL_MASK	R/W	1b	
24	GPIO3_FALL_MASK	R/W	1b	
23	GPIO2_FALL_MASK	R/W	1b	
22	GPIO1_FALL_MASK	R/W	1b	
21	VCCA_UV_MASK	R/W	0b	
20	VCCA_OV_MASK	R/W	0b	
19	LDO4_ILIM_MASK	R/W	1b	
18	RESERVED	R	0b	
17	LDO4_UV_MASK	R/W	0b	
16	LDO4_OV_MASK	R/W	0b	
15	LDO3_ILIM_MASK	R/W	1b	
14	RESERVED	R	0b	
13	LDO3_UV_MASK	R/W	0b	
12	LDO3_OV_MASK	R/W	0b	
11	LDO2_ILIM_MASK	R/W	1b	
10	RESERVED	R	0b	
9	LDO2_UV_MASK	R/W	0b	
8	LDO2_OV_MASK	R/W	0b	
7	LDO1_ILIM_MASK	R/W	1b	
6	RESERVED	R	0b	
5	LDO1_UV_MASK	R/W	0b	

Table 3-240. EEPROM_0_41 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
4	LDO1_OV_MASK	R/W	0b	
3	BUCK5_ILIM_MASK	R/W	1b	
2	RESERVED	R	0b	
1	BUCK5_UV_MASK	R/W	0b	
0	BUCK5_OV_MASK	R/W	0b	

3.2.43 EEPROM_0_42 Register (Offset = 0x2A) [Reset = 0x20207FF7]

EEPROM_0_42 is shown in Figure 3-237 and described in Table 3-241.

Return to the Table 3-197.

Figure 3-237. EEPROM_0_42 Register

31	30	29	28	27	26	25	24
NRSTOUT_READBACK_MASK	NINT_READBACK_MASK	NPWRON_LONG_MASK	SPMI_ERR_MASK	RESERVED	REG_CRC_ERR_MASK	BIST_FAIL_MASK	RESERVED
R/W-0b	R/W-0b	R/W-1b	R/W-0b	R-	R/W-0b	R/W-0b	R-
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
TWARN_MASK	RESERVED	EXT_CLK_MASK	BIST_PASS_MASK	SOFT_REBOOT_MASK	FSD_MASK	RESERVED	
R/W-0b	R-	R/W-1b	R/W-0b	R/W-0b	R/W-0b	R-	
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
ENABLE_MASK	NPWRON_START_MASK	GPIO11_RISE_MASK	GPIO10_RISE_MASK	GPIO9_RISE_MASK	GPIO11_FALL_MASK	GPIO10_FALL_MASK	GPIO9_FALL_MASK
R/W-0b	R/W-1b	R/W-1b	R/W-1b	R/W-1b	R/W-1b	R/W-1b	R/W-1b
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
GPIO8_RISE_MASK	GPIO7_RISE_MASK	GPIO6_RISE_MASK	GPIO5_RISE_MASK	GPIO4_RISE_MASK	GPIO3_RISE_MASK	GPIO2_RISE_MASK	GPIO1_RISE_MASK
R/W-1b	R/W-1b	R/W-1b	R/W-1b	R/W-0b	R/W-1b	R/W-1b	R/W-1b

Table 3-241. EEPROM_0_42 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	NRSTOUT_READBACK_MASK	R/W	0b	
30	NINT_READBACK_MASK	R/W	0b	
29	NPWRON_LONG_MASK	R/W	1b	
28	SPMI_ERR_MASK	R/W	0b	
27	RESERVED	R	0b	
26	REG_CRC_ERR_MASK	R/W	0b	
25	BIST_FAIL_MASK	R/W	0b	
24	RESERVED	R	0b	
23	TWARN_MASK	R/W	0b	
22	RESERVED	R	0b	
21	EXT_CLK_MASK	R/W	1b	
20	BIST_PASS_MASK	R/W	0b	
19	SOFT_REBOOT_MASK	R/W	0b	
18	FSD_MASK	R/W	0b	
17:16	RESERVED	R	0b	
15	ENABLE_MASK	R/W	0b	
14	NPWRON_START_MASK	R/W	1b	
13	GPIO11_RISE_MASK	R/W	1b	
12	GPIO10_RISE_MASK	R/W	1b	
11	GPIO9_RISE_MASK	R/W	1b	
10	GPIO11_FALL_MASK	R/W	1b	
9	GPIO10_FALL_MASK	R/W	1b	
8	GPIO9_FALL_MASK	R/W	1b	
7	GPIO8_RISE_MASK	R/W	1b	

Table 3-241. EEPROM_0_42 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
6	GPIO7_RISE_MASK	R/W	1b	
5	GPIO6_RISE_MASK	R/W	1b	
4	GPIO5_RISE_MASK	R/W	1b	
3	GPIO4_RISE_MASK	R/W	0b	
2	GPIO3_RISE_MASK	R/W	1b	
1	GPIO2_RISE_MASK	R/W	1b	
0	GPIO1_RISE_MASK	R/W	1b	

3.2.44 EEPROM_0_43 Register (Offset = 0x2B) [Reset = 0x0]

EEPROM_0_43 is shown in Figure 3-238 and described in Table 3-242.

Return to the Table 3-197.

Figure 3-238. EEPROM_0_43 Register

31	30	29	28	27	26	25	24
PGOOD_SEL_BUCK5		PGOOD_SEL_BUCK4		PGOOD_SEL_BUCK3		PGOOD_SEL_BUCK2	
R/W-0b		R/W-0b		R/W-0b		R/W-0b	
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
PGOOD_SEL_BUCK1		ESM_MCU_RST_MASK	ESM_MCU_FAIL_MASK	ESM_MCU_PIN_MASK	ESM_SOC_RST_MASK	ESM_SOC_FAIL_MASK	ESM_SOC_PIN_MASK
R/W-0b		R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
NRSTOUT_SOC_READBACK_MASK	RESERVED		EN_DRV_READBACK_MASK	I2C2_ADR_ERR_MASK	RESERVED	I2C2_CRC_ERR_MASK	RESERVED
R/W-0b	R-		R/W-0b	R/W-0b	R-	R/W-0b	R-
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
COMM_ADR_ERR_MASK	RESERVED	COMM_CRC_ERR_MASK	COMM_FRM_ERR_MASK	SOC_PWR_ERR_MASK	MCU_PWR_ERR_MASK	ORD_SHUTDOWN_MASK	IMM_SHUTDOWN_MASK
R/W-0b	R-	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-242. EEPROM_0_43 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:30	PGOOD_SEL_BUCK5	R/W	0b	
29:28	PGOOD_SEL_BUCK4	R/W	0b	
27:26	PGOOD_SEL_BUCK3	R/W	0b	
25:24	PGOOD_SEL_BUCK2	R/W	0b	
23:22	PGOOD_SEL_BUCK1	R/W	0b	
21	ESM_MCU_RST_MASK	R/W	0b	
20	ESM_MCU_FAIL_MASK	R/W	0b	
19	ESM_MCU_PIN_MASK	R/W	0b	
18	ESM_SOC_RST_MASK	R/W	0b	
17	ESM_SOC_FAIL_MASK	R/W	0b	
16	ESM_SOC_PIN_MASK	R/W	0b	
15	NRSTOUT_SOC_READBACK_MASK	R/W	0b	
14:13	RESERVED	R	0b	
12	EN_DRV_READBACK_MASK	R/W	0b	
11	I2C2_ADR_ERR_MASK	R/W	0b	
10	RESERVED	R	0b	
9	I2C2_CRC_ERR_MASK	R/W	0b	
8	RESERVED	R	0b	
7	COMM_ADR_ERR_MASK	R/W	0b	
6	RESERVED	R	0b	
5	COMM_CRC_ERR_MASK	R/W	0b	
4	COMM_FRM_ERR_MASK	R/W	0b	
3	SOC_PWR_ERR_MASK	R/W	0b	

Table 3-242. EEPROM_0_43 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
2	MCU_PWR_ERR_MASK	R/W	0b	
1	ORD_SHUTDOWN_MASK	R/W	0b	
0	IMM_SHUTDOWN_MASK	R/W	0b	

3.2.45 EEPROM_0_44 Register (Offset = 0x2C) [Reset = 0x03008000]

EEPROM_0_44 is shown in Figure 3-239 and described in Table 3-243.

Return to the Table 3-197.

Figure 3-239. EEPROM_0_44 Register

31	30	29	28	27	26	25	24
RESERVED		BB_VEOC		BB_ICHR	BB_CHARGER_EN	NSLEEP2_MASK	NSLEEP1_MASK
R-		R/W-0b		R/W-0b	R/W-0b	R/W-1b	R/W-1b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
EN_ILIM_FSM_CTRL	I2C2_HS	I2C1_HS	RESERVED	TSD_ORD_LEVEL	TWARN_LEVEL	EXT_CLK_FREQ	
R/W-0b	R/W-0b	R/W-0b	R-	R/W-0b	R/W-0b	R/W-0b	
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
PGOOD_WINDOW	PGOOD_POL	PGOOD_SEL_NRSTOUT_SOC	PGOOD_SEL_NRSTOUT	PGOOD_SEL_TDIE_WARN	RESERVED		PGOOD_SEL_VCCA
R/W-1b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R-		R/W-0b
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
PGOOD_SEL_LDO4		PGOOD_SEL_LDO3		PGOOD_SEL_LDO2		PGOOD_SEL_LDO1	
R/W-0b		R/W-0b		R/W-0b		R/W-0b	

Table 3-243. EEPROM_0_44 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:30	RESERVED	R	0b	
29:28	BB_VEOC	R/W	0b	
27	BB_ICHR	R/W	0b	
26	BB_CHARGER_EN	R/W	0b	
25	NSLEEP2_MASK	R/W	1b	
24	NSLEEP1_MASK	R/W	1b	
23	EN_ILIM_FSM_CTRL	R/W	0b	
22	I2C2_HS	R/W	0b	
21	I2C1_HS	R/W	0b	
20	RESERVED	R	0b	
19	TSD_ORD_LEVEL	R/W	0b	
18	TWARN_LEVEL	R/W	0b	
17:16	EXT_CLK_FREQ	R/W	0b	
15	PGOOD_WINDOW	R/W	1b	
14	PGOOD_POL	R/W	0b	
13	PGOOD_SEL_NRSTOUT_SOC	R/W	0b	
12	PGOOD_SEL_NRSTOUT	R/W	0b	
11	PGOOD_SEL_TDIE_WARN	R/W	0b	
10:9	RESERVED	R	0b	
8	PGOOD_SEL_VCCA	R/W	0b	
7:6	PGOOD_SEL_LDO4	R/W	0b	
5:4	PGOOD_SEL_LDO3	R/W	0b	
3:2	PGOOD_SEL_LDO2	R/W	0b	

Table 3-243. EEPROM_0_44 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
1:0	PGOOD_SEL_LDO1	R/W	0b	

3.2.46 EEPROM_0_45 Register (Offset = 0x2D) [Reset = 0x60000005]

EEPROM_0_45 is shown in Figure 3-240 and described in Table 3-244.

Return to the Table 3-197.

Figure 3-240. EEPROM_0_45 Register

31	30	29	28	27	26	25	24
PFSM_DELAY_STEP					BUCK5_FREQ_SEL	BUCK4_FREQ_SEL	BUCK3_FREQ_SEL
R/W-1100b					R/W-0b	R/W-0b	R/W-0b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
BUCK2_FREQ_SEL	BUCK1_FREQ_SEL	SS_PARAM2				SS_PARAM1	
R/W-0b	R/W-0b	R/W-0b				R/W-0b	
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
SS_PARAM1		SS_EN	SS_MODE		SS_DEPTH		BUCK5_RESET
R/W-0b		R/W-0b	R/W-0b		R/W-0b		R/W-0b
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
BUCK4_RESET	BUCK3_RESET	BUCK2_RESET	BUCK1_RESET	RECOV_CNT_THR			
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-101b			

Table 3-244. EEPROM_0_45 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:27	PFSM_DELAY_STEP	R/W	1100b	
26	BUCK5_FREQ_SEL	R/W	0b	
25	BUCK4_FREQ_SEL	R/W	0b	
24	BUCK3_FREQ_SEL	R/W	0b	
23	BUCK2_FREQ_SEL	R/W	0b	
22	BUCK1_FREQ_SEL	R/W	0b	
21:18	SS_PARAM2	R/W	0b	
17:14	SS_PARAM1	R/W	0b	
13	SS_EN	R/W	0b	
12:11	SS_MODE	R/W	0b	
10:9	SS_DEPTH	R/W	0b	
8	BUCK5_RESET	R/W	0b	
7	BUCK4_RESET	R/W	0b	
6	BUCK3_RESET	R/W	0b	
5	BUCK2_RESET	R/W	0b	
4	BUCK1_RESET	R/W	0b	
3:0	RECOV_CNT_THR	R/W	101b	

3.2.47 EEPROM_0_46 Register (Offset = 0x2E) [Reset = 0x18C0FFFF]

EEPROM_0_46 is shown in Figure 3-241 and described in Table 3-245.

Return to the Table 3-197.

Figure 3-241. EEPROM_0_46 Register

31	30	29	28	27	26	25	24
RESERVED			STARTUP_DEST		FAST_BIST	LP_STANDBY_SEL	XTAL_SEL
R-			R/W-11b		R/W-0b	R/W-0b	R/W-1b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
XTAL_SEL	XTAL_EN	ESM_SOC_EN	ESM_MCU_EN	USER_SPARE_4	USER_SPARE_3	USER_SPARE_2	USER_SPARE_1
R/W-1b	R/W-1b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
LDO4_RV_TIMEOUT				LDO3_RV_TIMEOUT			
R/W-1111b				R/W-1111b			
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
LDO2_RV_TIMEOUT				LDO1_RV_TIMEOUT			
R/W-1111b				R/W-1111b			

Table 3-245. EEPROM_0_46 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:29	RESERVED	R	0b	
28:27	STARTUP_DEST	R/W	11b	
26	FAST_BIST	R/W	0b	
25	LP_STANDBY_SEL	R/W	0b	
24:23	XTAL_SEL	R/W	1b	
22	XTAL_EN	R/W	1b	
21	ESM_SOC_EN	R/W	0b	
20	ESM_MCU_EN	R/W	0b	
19	USER_SPARE_4	R/W	0b	
18	USER_SPARE_3	R/W	0b	
17	USER_SPARE_2	R/W	0b	
16	USER_SPARE_1	R/W	0b	
15:12	LDO4_RV_TIMEOUT	R/W	1111b	
11:8	LDO3_RV_TIMEOUT	R/W	1111b	
7:4	LDO2_RV_TIMEOUT	R/W	1111b	
3:0	LDO1_RV_TIMEOUT	R/W	1111b	

3.2.48 EEPROM_0_47 Register (Offset = 0x2F) [Reset = 0x0]

EEPROM_0_47 is shown in Figure 3-242 and described in Table 3-246.

Return to the Table 3-197.

Figure 3-242. EEPROM_0_47 Register

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PFSM_DELAY4								PFSM_DELAY3							
R/W-0b								R/W-0b							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PFSM_DELAY2								PFSM_DELAY1							
R/W-0b								R/W-0b							

Table 3-246. EEPROM_0_47 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:24	PFSM_DELAY4	R/W	0b	
23:16	PFSM_DELAY3	R/W	0b	
15:8	PFSM_DELAY2	R/W	0b	
7:0	PFSM_DELAY1	R/W	0b	

3.2.49 EEPROM_0_48 Register (Offset = 0x30) [Reset = 0x044401A0]

EEPROM_0_48 is shown in Figure 3-243 and described in Table 3-247.

Return to the Table 3-197.

Figure 3-243. EEPROM_0_48 Register

31	30	29	28	27	26	25	24
RESERVED				BUCK3_SEL_VOUT_ADC_LEVEL		BUCK3_SEL_ISENSE_SLOPE_COMPENSATION	
R-				R/W-1b		R/W-0b	
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
BUCK2_SEL_VOUT_ADC_LEVEL		BUCK2_SEL_ISENSE_SLOPE_COMPENSATION		BUCK1_SEL_VOUT_ADC_LEVEL		BUCK1_SEL_ISENSE_SLOPE_COMPENSATION	
R/W-1b		R/W-0b		R/W-1b		R/W-0b	
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
SEL_RAMP_ARTIF		BUCK5_SEL_NEG_OCP_HYST	BUCK4_SEL_NEG_OCP_HYST	BUCK3_SEL_NEG_OCP_HYST	BUCK2_SEL_NEG_OCP_HYST	BUCK1_SEL_NEG_OCP_HYST	SEL_FB_FILTER
R/W-0b		R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-11b
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
SEL_FB_FILTER	SEL_TRAD_NON_OVERLAP	SEL_GATE_EARLY_SENSE	RESERVED			SEL_LOOP_NEG_HYST	
R/W-11b	R/W-0b	R/W-1b	R-			R/W-0b	

Table 3-247. EEPROM_0_48 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:28	RESERVED	R	0b	
27:26	BUCK3_SEL_VOUT_ADC_LEVEL	R/W	1b	
25:24	BUCK3_SEL_ISENSE_SLOPE_COMPENSATION	R/W	0b	
23:22	BUCK2_SEL_VOUT_ADC_LEVEL	R/W	1b	
21:20	BUCK2_SEL_ISENSE_SLOPE_COMPENSATION	R/W	0b	
19:18	BUCK1_SEL_VOUT_ADC_LEVEL	R/W	1b	
17:16	BUCK1_SEL_ISENSE_SLOPE_COMPENSATION	R/W	0b	
15:14	SEL_RAMP_ARTIF	R/W	0b	
13	BUCK5_SEL_NEG_OCP_HYST	R/W	0b	
12	BUCK4_SEL_NEG_OCP_HYST	R/W	0b	
11	BUCK3_SEL_NEG_OCP_HYST	R/W	0b	
10	BUCK2_SEL_NEG_OCP_HYST	R/W	0b	
9	BUCK1_SEL_NEG_OCP_HYST	R/W	0b	
8:7	SEL_FB_FILTER	R/W	11b	
6	SEL_TRAD_NON_OVERLAP	R/W	0b	
5	SEL_GATE_EARLY_SENSE	R/W	1b	

Table 3-247. EEPROM_0_48 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
4:2	RESERVED	R	0b	
1:0	SEL_LOOP_NEG_HYST	R/W	0b	

3.2.50 EEPROM_0_49 Register (Offset = 0x31) [Reset = 0x0679F150]

EEPROM_0_49 is shown in Figure 3-244 and described in Table 3-248.

Return to the Table 3-197.

Figure 3-244. EEPROM_0_49 Register

31	30	29	28	27	26	25	24
RESERVED					EN_DOT_MODE	EN_PFM_LOAD	RESERVED
R-					R/W-1b	R/W-1b	R-
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
RESERVED	EN_POS_OCP	EN_NEG_OCP	EN_LONG_PFM_EXIT_CNTR	SEL_HS_DETECTOR	RESERVED		BUCK5_SEL_POS_OCP_HYST
R-	R/W-1b	R/W-1b	R/W-1b	R/W-1b	R-		R/W-1b
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
BUCK4_SEL_POS_OCP_HYST	BUCK3_SEL_POS_OCP_HYST	BUCK2_SEL_POS_OCP_HYST	BUCK1_SEL_POS_OCP_HYST	EN_PLL_PROP_EXTEND	EN_SMART_OCP_BLANK	RESERVED	EN_LOAD_COMP_BLANK
R/W-1b	R/W-1b	R/W-1b	R/W-1b	R/W-0b	R/W-0b	R-	R/W-1b
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
BUCK3_SEL_PHASE_ADD		BUCK1_SEL_PHASE_ADD		BUCK3_SEL_PHASE_SHEDD	BUCK1_SEL_PHASE_SHEDD	EN_IAVE_LOOP_INJECTOR	EN_SW_RT_SHORT_DETECTORS
R/W-1b		R/W-1b		R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-248. EEPROM_0_49 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:27	RESERVED	R	0b	
26	EN_DOT_MODE	R/W	1b	
25	EN_PFM_LOAD	R/W	1b	
24:23	RESERVED	R	0b	
22	EN_POS_OCP	R/W	1b	
21	EN_NEG_OCP	R/W	1b	
20	EN_LONG_PFM_EXIT_CNTR	R/W	1b	
19	SEL_HS_DETECTOR	R/W	1b	
18:17	RESERVED	R	0b	
16	BUCK5_SEL_POS_OCP_HYST	R/W	1b	
15	BUCK4_SEL_POS_OCP_HYST	R/W	1b	
14	BUCK3_SEL_POS_OCP_HYST	R/W	1b	
13	BUCK2_SEL_POS_OCP_HYST	R/W	1b	
12	BUCK1_SEL_POS_OCP_HYST	R/W	1b	
11	EN_PLL_PROP_EXTEND	R/W	0b	
10	EN_SMART_OCP_BLANK	R/W	0b	
9	RESERVED	R	0b	
8	EN_LOAD_COMP_BLANK	R/W	1b	

Table 3-248. EEPROM_0_49 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
7:6	BUCK3_SEL_PHASE_ADD	R/W	1b	
5:4	BUCK1_SEL_PHASE_ADD	R/W	1b	
3	BUCK3_SEL_PHASE_SHEDD	R/W	0b	
2	BUCK1_SEL_PHASE_SHEDD	R/W	0b	
1	EN_IAVE_LOOP_INJECTOR	R/W	0b	
0	EN_SW_RT_SHORT_DETECTORS	R/W	0b	

3.2.51 EEPROM_0_50 Register (Offset = 0x32) [Reset = 0x04407D55]

EEPROM_0_50 is shown in Figure 3-245 and described in Table 3-249.

Return to the Table 3-197.

Figure 3-245. EEPROM_0_50 Register

31	30	29	28	27	26	25	24
RESERVED				BUCK5_SEL_VOUT_ADC_LEVEL		BUCK5_SEL_ISENSE_SLOPE_COMPENSATION	
R-				R/W-1b		R/W-0b	
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
BUCK4_SEL_VOUT_ADC_LEVEL		BUCK4_SEL_ISENSE_SLOPE_COMPENSATION		BUCK5_EN_RADAR_MODE	BUCK4_EN_RADAR_MODE	BUCK3_EN_RADAR_MODE	BUCK2_EN_RADAR_MODE
R/W-1b		R/W-0b		R/W-0b	R/W-0b	R/W-0b	R/W-0b
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
BUCK1_EN_RADAR_MODE	EN_AUTO_LOOP_COEFFS	LOOP_COEFF_FB_MSB	EN_LONG_ZERO_CROSS_FILTER	SEL_ZERO_CROSS_FILTER_AVE	LOOP_COEFF_I_BALANCE	BUCK5_SEL_OUTPUT_CAPS	
R/W-0b	R/W-1b	R/W-1b	R/W-1b	R/W-1b	R/W-1b	R/W-1b	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
BUCK4_SEL_OUTPUT_CAPS		BUCK3_SEL_OUTPUT_CAPS		BUCK2_SEL_OUTPUT_CAPS		BUCK1_SEL_OUTPUT_CAPS	
R/W-1b		R/W-1b		R/W-1b		R/W-1b	

Table 3-249. EEPROM_0_50 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:28	RESERVED	R	0b	
27:26	BUCK5_SEL_VOUT_ADC_LEVEL	R/W	1b	
25:24	BUCK5_SEL_ISENSE_SLOPE_COMPENSATION	R/W	0b	
23:22	BUCK4_SEL_VOUT_ADC_LEVEL	R/W	1b	
21:20	BUCK4_SEL_ISENSE_SLOPE_COMPENSATION	R/W	0b	
19	BUCK5_EN_RADAR_MODE	R/W	0b	
18	BUCK4_EN_RADAR_MODE	R/W	0b	
17	BUCK3_EN_RADAR_MODE	R/W	0b	
16	BUCK2_EN_RADAR_MODE	R/W	0b	
15	BUCK1_EN_RADAR_MODE	R/W	0b	
14	EN_AUTO_LOOP_COEFFS	R/W	1b	
13	LOOP_COEFF_FB_MSB	R/W	1b	
12	EN_LONG_ZERO_CROSS_FILTER	R/W	1b	
11	SEL_ZERO_CROSS_FILTER_AVE	R/W	1b	
10	LOOP_COEFF_I_BALANCE	R/W	1b	
9:8	BUCK5_SEL_OUTPUT_CAPS	R/W	1b	

Table 3-249. EEPROM_0_50 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
7:6	BUCK4_SEL_OUTPUT_CAPS	R/W	1b	
5:4	BUCK3_SEL_OUTPUT_CAPS	R/W	1b	
3:2	BUCK2_SEL_OUTPUT_CAPS	R/W	1b	
1:0	BUCK1_SEL_OUTPUT_CAPS	R/W	1b	

3.2.52 EEPROM_0_51 Register (Offset = 0x33) [Reset = 0x31C3EA56]

EEPROM_0_51 is shown in Figure 3-246 and described in Table 3-250.

Return to the Table 3-197.

Figure 3-246. EEPROM_0_51 Register

31	30	29	28	27	26	25	24
RESERVED	I_COEFF		EN_PWM_LS_DETECTION	RESERVED	EN_LS_AFTER_HIZ	EN_FAST_INTEGRATION_BYPASS_RAMP_RES	EN_M_10M_TRAN_DETECTOR
R-	R/W-1b		R/W-1b	R-	R/W-0b	R/W-0b	R/W-1b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
EN_SLOW_PLL_0P3	EN_CONSTANT_PLL_DVS_COEFF	RESERVED	DIS_DVS_WAIT_COMPARATORS	DIS_PFM_WAITS_HS_DETECTOR	DIS_PFM_WAITS_LS_DETECTOR	EN_I_BALANCE_INTEGRATOR	EN_FAST_VOUT_INTEGRATION
R/W-1b	R/W-1b	R-	R/W-0b	R/W-0b	R/W-0b	R/W-1b	R/W-1b
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
EN_FAST_PLL_0P7	EN_SW_SHORT_DETECTORS	BUCK5_SEL_RAMP		BUCK4_SEL_RAMP		BUCK3_SEL_RAMP	
R/W-1b	R/W-1b	R/W-10b		R/W-10b		R/W-10b	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RESERVED	BUCK2_SEL_RAMP		BUCK1_SEL_RAMP		EN_SLOW_PLL_COEFFS	EN_PFM_PULSE_WAIT_LS_OCP	FORCE_SS_ADAPT
R-	R/W-10b		R/W-10b		R/W-1b	R/W-1b	R/W-0b

Table 3-250. EEPROM_0_51 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	RESERVED	R	0b	
30:29	I_COEFF	R/W	1b	
28	EN_PWM_LS_DETECTION	R/W	1b	
27	RESERVED	R	0b	
26	EN_LS_AFTER_HIZ	R/W	0b	
25	EN_FAST_INTEGRATION_BYPASS_RAMP_RES	R/W	0b	
24	EN_M_10M_TRAN_DETECTOR	R/W	1b	
23	EN_SLOW_PLL_0P3	R/W	1b	
22	EN_CONSTANT_PLL_DVS_COEFF	R/W	1b	
21	RESERVED	R	0b	
20	DIS_DVS_WAIT_COMPARATORS	R/W	0b	
19	DIS_PFM_WAITS_HS_DETECTOR	R/W	0b	
18	DIS_PFM_WAITS_LS_DETECTOR	R/W	0b	
17	EN_I_BALANCE_INTEGRATOR	R/W	1b	
16	EN_FAST_VOUT_INTEGRATION	R/W	1b	
15	EN_FAST_PLL_0P7	R/W	1b	

Table 3-250. EEPROM_0_51 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
14	EN_SW_SHORT_DETECTORS	R/W	1b	
13:12	BUCK5_SEL_RAMP	R/W	10b	
11:10	BUCK4_SEL_RAMP	R/W	10b	
9:8	BUCK3_SEL_RAMP	R/W	10b	
7	RESERVED	R	0b	
6:5	BUCK2_SEL_RAMP	R/W	10b	
4:3	BUCK1_SEL_RAMP	R/W	10b	
2	EN_SLOW_PLL_COEFFS	R/W	1b	
1	EN_PFM_PULSE_WAIT_LS_OCP	R/W	1b	
0	FORCE_SS_ADAPT	R/W	0b	

3.2.53 EEPROM_0_52 Register (Offset = 0x34) [Reset = 0x67218109]

EEPROM_0_52 is shown in Figure 3-247 and described in Table 3-251.

Return to the Table 3-197.

Figure 3-247. EEPROM_0_52 Register

31	30	29	28	27	26	25	24
EN_INITIALIZE_DPLL_RESTART	REG_CRC_EN	FAST_VCCA_OVP	PFSM_ERR_MASK	FREQ_SEL_UNLOCK	MAX_ILIM		
R/W-0b	R/W-1b	R/W-1b	R/W-0b	R/W-0b	R/W-111b		
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
RESERVED	ABIST_ERROR_MASK	VMON_ABIST_EN	FAST_BOOT_BIST	DIS_TSD	DIS_UVLO_OVP_RESET	PFSM_ERR_RESET_DIS	VSYS_DEAD_LOCK_EN
R-	R/W-0b	R/W-1b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-1b
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
EN_OVP	BUCK5_FREQ_8MHZ	BUCK4_FREQ_8MHZ	BUCK3_FREQ_8MHZ	BUCK2_FREQ_8MHZ	BUCK1_FREQ_8MHZ	LONG_SINGLE_SHOT	FIXED_SS_LENGTH
R/W-1b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-10000b
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
FIXED_SS_LENGTH				EN_ADAPTIVE_SINGLE_SHOT	BUCK_NEG_ILIM		
R/W-10000b				R/W-1b	R/W-1b		

Table 3-251. EEPROM_0_52 Register Field Descriptions

Bit	Field	Type	Reset	Description
31	EN_INITIALIZE_DPLL_RESTART	R/W	0b	
30	REG_CRC_EN	R/W	1b	
29	FAST_VCCA_OVP	R/W	1b	
28	PFSM_ERR_MASK	R/W	0b	
27	FREQ_SEL_UNLOCK	R/W	0b	
26:24	MAX_ILIM	R/W	111b	
23	RESERVED	R	0b	
22	ABIST_ERROR_MASK	R/W	0b	
21	VMON_ABIST_EN	R/W	1b	
20	FAST_BOOT_BIST	R/W	0b	
19	DIS_TSD	R/W	0b	
18	DIS_UVLO_OVP_RESET	R/W	0b	
17	PFSM_ERR_RESET_DIS	R/W	0b	
16	VSYS_DEAD_LOCK_EN	R/W	1b	
15	EN_OVP	R/W	1b	
14	BUCK5_FREQ_8MHZ	R/W	0b	
13	BUCK4_FREQ_8MHZ	R/W	0b	
12	BUCK3_FREQ_8MHZ	R/W	0b	
11	BUCK2_FREQ_8MHZ	R/W	0b	
10	BUCK1_FREQ_8MHZ	R/W	0b	
9	LONG_SINGLE_SHOT	R/W	0b	
8:4	FIXED_SS_LENGTH	R/W	10000b	
3	EN_ADAPTIVE_SINGLE_SHOT	R/W	1b	

Table 3-251. EEPROM_0_52 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
2:0	BUCK_NEG_ILIM	R/W	1b	

3.2.54 EEPROM_0_53 Register (Offset = 0x35) [Reset = 0x88D65E80]

EEPROM_0_53 is shown in Figure 3-248 and described in Table 3-252.

Return to the Table 3-197.

Figure 3-248. EEPROM_0_53 Register

31	30	29	28	27	26	25	24
SPMI_WD_RUNTIME_INTERVAL				SPMI_WD_BOOT_INTERVAL			
R/W-1000b				R/W-1000b			
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
SPMI_WAKEUP_EN	SPMI_WD_EN	SPMI_EN	SPMI_WD_AUTO_BOOT	SPMI_SLAVE_ASR_HOLD	SPMI_RETRY_LIMIT		SPMI_IF_SEL
R/W-1b	R/W-1b	R/W-0b	R/W-1b	R/W-0b	R/W-11b		R/W-0b
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
SPMI_SLAVE_PASSIVE	SPMI_CLK_SEL		SPMI_MASTER_SEL	SPMI_CRC_EN	I2C2_CRC_EN	I2C1_SPI_CRC_EN	I2C_SPI_SEL
R/W-0b	R/W-10b		R/W-1b	R/W-1b	R/W-1b	R/W-1b	R/W-0b
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
WD_EN_EE	DISABLE_USE_TRIMS	DISABLE_CHANGE_BG	DISABLE_VM_NARROW_LIMITS	SEL_RC_OSC	EN_FIXED_DPLL_FREQ	SLOW_AUTOZERO_SEL	DIS_NRSTOUT_MCU_I2C_SPI_RESET
R/W-1b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-252. EEPROM_0_53 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:28	SPMI_WD_RUNTIME_INTERVAL	R/W	1000b	
27:24	SPMI_WD_BOOT_INTERVAL	R/W	1000b	
23	SPMI_WAKEUP_EN	R/W	1b	
22	SPMI_WD_EN	R/W	1b	
21	SPMI_EN	R/W	0b	
20	SPMI_WD_AUTO_BOOT	R/W	1b	
19	SPMI_SLAVE_ASR_HOLD	R/W	0b	
18:17	SPMI_RETRY_LIMIT	R/W	11b	
16	SPMI_IF_SEL	R/W	0b	
15	SPMI_SLAVE_PASSIVE	R/W	0b	
14:13	SPMI_CLK_SEL	R/W	10b	
12	SPMI_MASTER_SEL	R/W	1b	
11	SPMI_CRC_EN	R/W	1b	
10	I2C2_CRC_EN	R/W	1b	
9	I2C1_SPI_CRC_EN	R/W	1b	
8	I2C_SPI_SEL	R/W	0b	
7	WD_EN_EE	R/W	1b	
6	DISABLE_USE_TRIMS	R/W	0b	
5	DISABLE_CHANGE_BG	R/W	0b	
4	DISABLE_VM_NARROW_LIMITS	R/W	0b	
3	SEL_RC_OSC	R/W	0b	
2	EN_FIXED_DPLL_FREQ	R/W	0b	
1	SLOW_AUTOZERO_SEL	R/W	0b	

Table 3-252. EEPROM_0_53 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
0	DIS_NRSTOUT_MCU_I2C_SPI_RESET	R/W	0b	

3.2.55 EEPROM_0_54 Register (Offset = 0x36) [Reset = 0x05008888]

EEPROM_0_54 is shown in Figure 3-249 and described in Table 3-253.

Return to the Table 3-197.

Figure 3-249. EEPROM_0_54 Register

31	30	29	28	27	26	25	24
RESERVED		SPMI_MID		SPMI_SID			
R-		R/W-0b		R/W-101b			
23	22	21	20	19	18	17	16
BOOT_DELAY							
R/W-0b							
15	14	13	12	11	10	9	8
SPMI_WD_RUNTIME_BIST_TIMEOUT				SPMI_WD_BOOT_BIST_TIMEOUT			
R/W-1000b				R/W-1000b			
7	6	5	4	3	2	1	0
SPMI_PFSM_RESPONSE_TIMEOUT				SPMI_WD_RESPONSE_TIMEOUT			
R/W-1000b				R/W-1000b			

Table 3-253. EEPROM_0_54 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:30	RESERVED	R	0b	
29:28	SPMI_MID	R/W	0b	
27:24	SPMI_SID	R/W	101b	
23:16	BOOT_DELAY	R/W	0b	
15:12	SPMI_WD_RUNTIME_BIST_TIMEOUT	R/W	1000b	
11:8	SPMI_WD_BOOT_BIST_TIMEOUT	R/W	1000b	
7:4	SPMI_PFSM_RESPONSE_TIMEOUT	R/W	1000b	
3:0	SPMI_WD_RESPONSE_TIMEOUT	R/W	1000b	

3.2.56 EEPROM_0_55 Register (Offset = 0x37) [Reset = 0x00400948]

EEPROM_0_55 is shown in Figure 3-250 and described in Table 3-254.

Return to the Table 3-197.

Figure 3-250. EEPROM_0_55 Register

31	30	29	28	27	26	25	24
RESERVED				RTC_SPARE_3	RTC_SPARE_2	RTC_SPARE_1	RTC_SPARE_0
R-				0b	0b	0b	0b
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
MP_CONFIG			LDO4_PD_FORCE	LDO3_PD_FORCE	LDO2_PD_FORCE	LDO1_PD_FORCE	RESERVED
R/W-10b			R/W-0b	R/W-0b	R/W-0b	R/W-0b	R-
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
INT_LDO_PD_FORCE	FORCE_CLK_GATE	I2C2_ID					
R/W-0b	R/W-0b	R/W-10010b					
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
I2C2_ID	I2C1_ID						
R/W-10010b	R/W-1001000b						

Table 3-254. EEPROM_0_55 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:28	RESERVED	R	0b	
27	RTC_SPARE_3		0b	
26	RTC_SPARE_2		0b	
25	RTC_SPARE_1		0b	
24	RTC_SPARE_0		0b	
23:21	MP_CONFIG	R/W	10b	
20	LDO4_PD_FORCE	R/W	0b	
19	LDO3_PD_FORCE	R/W	0b	
18	LDO2_PD_FORCE	R/W	0b	
17	LDO1_PD_FORCE	R/W	0b	
16	RESERVED	R	0b	
15	INT_LDO_PD_FORCE	R/W	0b	
14	FORCE_CLK_GATE	R/W	0b	
13:7	I2C2_ID	R/W	10010b	
6:0	I2C1_ID	R/W	1001000b	

3.2.57 EEPROM_0_56 Register (Offset = 0x38) [Reset = 0x0]

EEPROM_0_56 is shown in Figure 3-251 and described in Table 3-255.

Return to the Table 3-197.

Figure 3-251. EEPROM_0_56 Register

31	30	29	28	27	26	25	24
RESERVED							
R-							
23	22	21	20	19	18	17	16
RESERVED							
R-							
15	14	13	12	11	10	9	8
RESERVED					SPMI_SLAVE_CNT		
R-					R/W-0b		
7	6	5	4	3	2	1	0
CUSTOMER_NVM_ID							
R/W-0b							

Table 3-255. EEPROM_0_56 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:11	RESERVED	R	0b	
10:8	SPMI_SLAVE_CNT	R/W	0b	
7:0	CUSTOMER_NVM_ID	R/W	0b	

3.2.58 EEPROM_0_57 Register (Offset = 0x39) [Reset = 0xA50000A0]

EEPROM_0_57 is shown in Figure 3-252 and described in Table 3-256.

Return to the Table 3-197.

Figure 3-252. EEPROM_0_57 Register

31	30	29	28	27	26	25	24
USER_EE_PROG_UNLOCK_CODE							
R/W-10100101b							

23	22	21	20	19	18	17	16
RESERVED	BUCK5_CHANGE_2MHZ_BELOW_0V5	BUCK4_CHANGE_2MHZ_BELOW_0V5	BUCK3_CHANGE_2MHZ_BELOW_0V5	BUCK2_CHANGE_2MHZ_BELOW_0V5	BUCK1_CHANGE_2MHZ_BELOW_0V5	BUCK5_DOUBLE_PFM_PULSE	BUCK4_DOUBLE_PFM_PULSE
R-	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

15	14	13	12	11	10	9	8
BUCK3_DOUBLE_PFM_PULSE	BUCK2_DOUBLE_PFM_PULSE	BUCK1_DOUBLE_PFM_PULSE	BUCK5_EN_P_10M_BODY_DIODE	BUCK4_EN_P_10M_BODY_DIODE	BUCK3_EN_P_10M_BODY_DIODE	BUCK2_EN_P_10M_BODY_DIODE	BUCK1_EN_P_10M_BODY_DIODE
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

7	6	5	4	3	2	1	0
BUCK1_RV_SEL	RESERVED	BUCK1_PLDN	BUCK1_VMON_EN	BUCK1_VSEL	BUCK1_FPWM_MP	BUCK1_FPWM	BUCK1_EN
R/W-1b	R-	R/W-1b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-256. EEPROM_0_57 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:24	USER_EE_PROG_UNLOCK_CODE	R/W	10100101b	
23	RESERVED	R	0b	
22	BUCK5_CHANGE_2MHZ_BELOW_0V5	R/W	0b	
21	BUCK4_CHANGE_2MHZ_BELOW_0V5	R/W	0b	
20	BUCK3_CHANGE_2MHZ_BELOW_0V5	R/W	0b	
19	BUCK2_CHANGE_2MHZ_BELOW_0V5	R/W	0b	
18	BUCK1_CHANGE_2MHZ_BELOW_0V5	R/W	0b	
17	BUCK5_DOUBLE_PFM_PULSE	R/W	0b	
16	BUCK4_DOUBLE_PFM_PULSE	R/W	0b	
15	BUCK3_DOUBLE_PFM_PULSE	R/W	0b	
14	BUCK2_DOUBLE_PFM_PULSE	R/W	0b	
13	BUCK1_DOUBLE_PFM_PULSE	R/W	0b	
12	BUCK5_EN_P_10M_BODY_DIODE	R/W	0b	
11	BUCK4_EN_P_10M_BODY_DIODE	R/W	0b	
10	BUCK3_EN_P_10M_BODY_DIODE	R/W	0b	

Table 3-256. EEPROM_0_57 Register Field Descriptions (continued)

Bit	Field	Type	Reset	Description
9	BUCK2_EN_P_10M_BODY_DIODE	R/W	0b	
8	BUCK1_EN_P_10M_BODY_DIODE	R/W	0b	
7	BUCK1_RV_SEL	R/W	1b	
6	RESERVED	R	0b	
5	BUCK1_PLDN	R/W	1b	
4	BUCK1_VMON_EN	R/W	0b	
3	BUCK1_VSEL	R/W	0b	
2	BUCK1_FPWM_MP	R/W	0b	
1	BUCK1_FPWM	R/W	0b	
0	BUCK1_EN	R/W	0b	

3.2.59 EEPROM_0_58 Register (Offset = 0x3A) [Reset = 0x1D80]

EEPROM_0_58 is shown in Figure 3-253 and described in Table 3-257.

Return to the Table 3-197.

Figure 3-253. EEPROM_0_58 Register

31	30	29	28	27	26	25	24
RESERVED				ldo2_dis_ov_pldn	ldo2_en_cp_low_sr	ldo2_dis_cp_leak_comp	ldo2_dis_ilim
R-				R/W-0b	R/W-0b	R/W-0b	R/W-0b

23	22	21	20	19	18	17	16
ldo2_dis_short_prot	ldo2_en_short_cp	ldo1_dis_ov_pldn	ldo1_en_cp_low_sr	ldo1_dis_cp_leak_comp	ldo1_dis_ilim	ldo1_dis_short_prot	ldo1_en_short_cp
R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

15	14	13	12	11	10	9	8
vbg_filt_config		xtal_bias_fine				xtal_comp_bias_lvl	xtal_amp_reg_mode
R/W-0b		R/W-111b				R/W-0b	R/W-1b

7	6	5	4	3	2	1	0
xtal_amp_reg_en	safety_sel_ibias		safety_bg_buf_hi_bw	safety_speedup	refsys_sel_ibias		refsys_bg_buf_hi_bw
R/W-1b	R/W-0b		R/W-0b	R/W-0b	R/W-0b		R/W-0b

Table 3-257. EEPROM_0_58 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:28	RESERVED	R	0b	
27	ldo2_dis_ov_pldn	R/W	0b	
26	ldo2_en_cp_low_sr	R/W	0b	
25	ldo2_dis_cp_leak_comp	R/W	0b	
24	ldo2_dis_ilim	R/W	0b	
23	ldo2_dis_short_prot	R/W	0b	
22	ldo2_en_short_cp	R/W	0b	
21	ldo1_dis_ov_pldn	R/W	0b	
20	ldo1_en_cp_low_sr	R/W	0b	
19	ldo1_dis_cp_leak_comp	R/W	0b	
18	ldo1_dis_ilim	R/W	0b	
17	ldo1_dis_short_prot	R/W	0b	
16	ldo1_en_short_cp	R/W	0b	
15:14	vbg_filt_config	R/W	0b	
13:10	xtal_bias_fine	R/W	111b	
9	xtal_comp_bias_lvl	R/W	0b	
8	xtal_amp_reg_mode	R/W	1b	
7	xtal_amp_reg_en	R/W	1b	
6:5	safety_sel_ibias	R/W	0b	
4	safety_bg_buf_hi_bw	R/W	0b	
3	safety_speedup	R/W	0b	
2:1	refsys_sel_ibias	R/W	0b	
0	refsys_bg_buf_hi_bw	R/W	0b	

3.2.60 EEPROM_0_59 Register (Offset = 0x3B) [Reset = 0x1180]

EEPROM_0_59 is shown in Figure 3-254 and described in Table 3-258.

Return to the Table 3-197.

Figure 3-254. EEPROM_0_59 Register

31	30	29	28	27	26	25	24
RESERVED							
R-							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
RESERVED							
R-							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
RESERVED		PFSM_PROXY_BRANCH_DIS	SRAM_RELOAD_LIMIT		PFSM_SEQ_MIN_TIME		ldo4_filter_current
R-		R/W-0b	R/W-10b		R/W-0b		R/W-11b
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
ldo4_filter_current	ldo4_sel_low_ilim	ldo3_dis_ov_pldn	ldo3_en_cp_low_sr	ldo3_dis_cp_leak_comp	ldo3_dis_ilim	ldo3_dis_short_prot	ldo3_en_short_cp
R/W-11b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b	R/W-0b

Table 3-258. EEPROM_0_59 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:14	RESERVED	R	0b	
13	PFSM_PROXY_BRANCH_DIS	R/W	0b	
12:11	SRAM_RELOAD_LIMIT	R/W	10b	
10:9	PFSM_SEQ_MIN_TIME	R/W	0b	
8:7	ldo4_filter_current	R/W	11b	
6	ldo4_sel_low_ilim	R/W	0b	
5	ldo3_dis_ov_pldn	R/W	0b	
4	ldo3_en_cp_low_sr	R/W	0b	
3	ldo3_dis_cp_leak_comp	R/W	0b	
2	ldo3_dis_ilim	R/W	0b	
1	ldo3_dis_short_prot	R/W	0b	
0	ldo3_en_short_cp	R/W	0b	

3.2.61 EEPROM_0_60 Register (Offset = 0x3C) [Reset = 0xFF03]

EEPROM_0_60 is shown in Figure 3-255 and described in Table 3-259.

Return to the Table 3-197.

Figure 3-255. EEPROM_0_60 Register

31	30	29	28	27	26	25	24
RESERVED							
R-							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
RESERVED							
R-							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
WD_LONGWIN							
R/W-11111111b							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
LPM_EN_DISABLES_VCCA_VMON	VMON_DEGLITCH_SEL	RESERVED				VCCA_VMON_EN	WD_EN
R/W-0b	R/W-0b	R-				R/W-1b	R/W-1b

Table 3-259. EEPROM_0_60 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:16	RESERVED	R	0b	
15:8	WD_LONGWIN	R/W	11111111b	
7	LPM_EN_DISABLES_VCCA_VMON	R/W	0b	
6	VMON_DEGLITCH_SEL	R/W	0b	
5:2	RESERVED	R	0b	
1	VCCA_VMON_EN	R/W	1b	
0	WD_EN	R/W	1b	

3.2.62 EEPROM_0_61 Register (Offset = 0x3D) [Reset = 0x0]

EEPROM_0_61 is shown in Figure 3-256 and described in Table 3-260.

Return to the Table 3-197.

Figure 3-256. EEPROM_0_61 Register

31	30	29	28	27	26	25	24
REGMAP_USER_EXCLUDE_PERSIST_CRC16_HIGH							
R/W-0b							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
REGMAP_USER_EXCLUDE_PERSIST_CRC16_LOW							
R/W-0b							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
REGMAP_USER_INCLUDE_PERSIST_CRC16_HIGH							
R/W-0b							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
REGMAP_USER_INCLUDE_PERSIST_CRC16_LOW							
R/W-0b							

Table 3-260. EEPROM_0_61 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:24	REGMAP_USER_EXCLUDE_PERSIST_CRC16_HIGH	R/W	0b	
23:16	REGMAP_USER_EXCLUDE_PERSIST_CRC16_LOW	R/W	0b	
15:8	REGMAP_USER_INCLUDE_PERSIST_CRC16_HIGH	R/W	0b	
7:0	REGMAP_USER_INCLUDE_PERSIST_CRC16_LOW	R/W	0b	

3.2.63 EEPROM_0_62 Register (Offset = 0x3E) [Reset = 0x0]

EEPROM_0_62 is shown in Figure 3-257 and described in Table 3-261.

Return to the Table 3-197.

Figure 3-257. EEPROM_0_62 Register

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SRAM_BANK0_CRC16_HIGH								SRAM_BANK0_CRC16_LOW							
R/W-0b								R/W-0b							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
REGMAP_CONFIG_CRC16_HIGH								REGMAP_CONFIG_CRC16_LOW							
R/W-0b								R/W-0b							

Table 3-261. EEPROM_0_62 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:24	SRAM_BANK0_CRC16_HIGH	R/W	0b	
23:16	SRAM_BANK0_CRC16_LOW	R/W	0b	
15:8	REGMAP_CONFIG_CRC16_HIGH	R/W	0b	
7:0	REGMAP_CONFIG_CRC16_LOW	R/W	0b	

3.2.64 EEPROM_0_63 Register (Offset = 0x3F) [Reset = 0x0]

EEPROM_0_63 is shown in Figure 3-258 and described in Table 3-262.

Return to the Table 3-197.

Figure 3-258. EEPROM_0_63 Register

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SRAM_BANK2_CRC16_HIGH								SRAM_BANK2_CRC16_LOW							
R/W-0b								R/W-0b							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SRAM_BANK1_CRC16_HIGH								SRAM_BANK1_CRC16_LOW							
R/W-0b								R/W-0b							

Table 3-262. EEPROM_0_63 Register Field Descriptions

Bit	Field	Type	Reset	Description
31:24	SRAM_BANK2_CRC16_HIGH	R/W	0b	
23:16	SRAM_BANK2_CRC16_LOW	R/W	0b	
15:8	SRAM_BANK1_CRC16_HIGH	R/W	0b	
7:0	SRAM_BANK1_CRC16_LOW	R/W	0b	

4 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from Revision A (May 2020) to Revision B (February 2021)	Page
• Updated the numbering format for tables, figures, and cross-references throughout the document	5
• Added notes to the *Signal Descriptions* table in the *Digital Description Signals* section	7
• Updated the numbering format for tables, figures, and cross-references throughout the document	15
• Updated the *Functional Diagram* figure in the *Functional Diagram* section	16
• Updated *Mission State Configuration* section	22
• Added the *Configuration Memory Organization and Sequence Execution* section	25
• Updated the *On Requests* section	27
• Updated the *WKUP1 and WKUP2 Functions* section	29
• Updated the *Power Rail Output Error* section	30
• Updated the *Catastrophic Error* section	31
• Updated the *Buck Regulators Overview* section	35
• Updated the *Adaptive Voltage Scaling (AVS) and Dynamic Voltage Scaling (DVS)* section	38
• Updated the *Low Dropout Regulators (LDOs)* section	43
• Updated the *Output Voltage Selection for LDO1, LDO2, and LDO3* table in the *LDO1, LDO2, and LDO3* section	44
• Updated the *Output Voltage Selection for LDO4* table in the *Low-Noise LDO (LDO4)* section	45
• Updated the *Backup Supply Power-Path* section	46
• Updated the *General-Purpose I/Os (GPIO Pins)* section	50
• Updated the *RTC Interrupts* section	63
• Updated the *Watchdog Disable Function* section	66
• Removed the *Correct and Incorrect WD Q&A Sequence Run Scenarios for WD Q&A* table in the *Question Generation* section	72
• Updated the *ESM Start-Up in PWM Mode* section	88
• Updated the *Register Maps* section and added the *EEPROM_map Registers* section	109

Changes from Revision * (September 2019) to Revision A (May 2020)	Page
• Updated TRM	5