

LP876242-Q1 Four 8.8-MHz Buck Converters for AWR and IWR MMICs

1 Features

- AEC-Q100 Qualified with the following results:
 - Input voltage: 3.0 V to 5.5 V
 - Device temperature grade 1: -40°C to $+125^{\circ}\text{C}$ ambient operating temperature range
 - Device HBM ESD classification Level 2
 - Device CDM ESD classification Level C4B
- Functional Safety-Compliant
 - Developed for functional safety applications
 - Documentation available to aid ISO 26262 system design up to ASIL-D
 - Documentation available to aid IEC 61508 system design up to SIL-3
 - Systematic capability up to ASIL-D
 - Hardware integrity up to ASIL-D
 - Windowed voltage and over-current monitors
 - Watchdog with selectable trigger / Q&A mode
 - Level or PWM error signal monitoring (ESM)
 - Thermal monitoring with high temperature warning and thermal shutdown
 - Bit-integrity (CRC) error detection on configuration registers and non-volatile memory
- 4 high-efficiency step-down DC/DC converters:
 - Output voltage: 0.9 V to 1.9 V
 - Maximum output current: 4/3/3/2 A
 - Programmable output voltage slew-rate: 0.5 mV/ μs to 33 mV/ μs
 - Switching frequency: 8.8 MHz
- 10 configurable general purpose I/O (GPIO)
- SPMI interface for multi-PMIC synchronization
- Input overvoltage monitor (OVP) and undervoltage lockout (UVLO)

2 Applications

- [Imaging radar](#) using cascaded configuration
- [Long range radar](#)
- [Radar module without processing](#)

3 Description

The LP876242-Q1 device is designed to meet the power management requirements of the latest processors and platforms in various safety-relevant automotive and industrial applications. The device has four step-down DC/DC converter cores, generating four 1-phase outputs. The device settings can be changed by I²C-compatible serial interface or by a SPI serial interface.

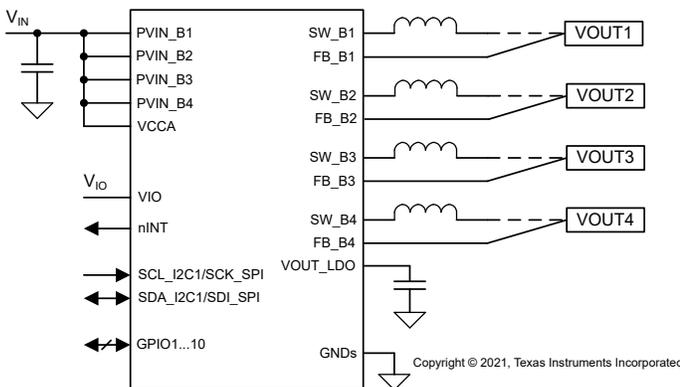
The switching clock is forced to PWM mode and the phases are interleaved. The switching can be synchronized to an external clock and spread-spectrum mode can be enabled to minimize the disturbances.

The LP876242-Q1 device includes protection and diagnostic mechanisms such as register and interface CRC, short-circuit protection, thermal monitoring, and shutdown. The LP876242-Q1 device can notify the processor of these events through the interrupt handler, allowing the processor to take action in response.

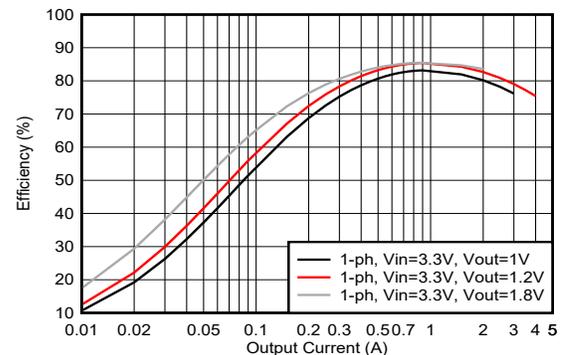
Device Information⁽¹⁾

| PART NUMBER | PACKAGE | BODY SIZE (NOM) |
|-------------|--------------|-------------------|
| LP876242-Q1 | VQFN-HR (32) | 5.50 mm × 5.00 mm |

- (1) For all available packages, see the orderable addendum at the end of the data sheet.



Simplified Schematic



Efficiency vs Output Current (1-phase)



Table of Contents

| | | | |
|---|----|--|-----|
| 1 Features | 1 | 6.17 Serial Peripheral Interface (SPI)..... | 27 |
| 2 Applications | 1 | 6.18 Typical Characteristics..... | 29 |
| 3 Description | 1 | 7 Detailed Description | 30 |
| 4 Revision History | 2 | 7.1 Overview..... | 30 |
| 5 Pin Configuration and Functions | 3 | 7.2 Functional Block Diagram..... | 31 |
| 5.1 Digital Signal Descriptions..... | 8 | 7.3 Feature Description..... | 31 |
| 6 Specifications | 14 | 7.4 Device Functional Modes..... | 95 |
| 6.1 Absolute Maximum Ratings..... | 14 | 7.5 Control Interfaces..... | 124 |
| 6.2 ESD Ratings..... | 14 | 7.6 NVM Configurable Registers..... | 132 |
| 6.3 Recommended Operating Conditions..... | 15 | 7.7 Register Map..... | 135 |
| 6.4 Thermal Information..... | 15 | 8 Application and Implementation | 274 |
| 6.5 Internal Low Drop-Out Regulators (LDOVINT)..... | 16 | 8.1 Application Information..... | 274 |
| 6.6 BUCK1, BUCK2, BUCK3, and BUCK4 Regulators..... | 16 | 8.2 Typical Applications..... | 274 |
| 6.7 Reference Generator (REFOUT)..... | 18 | 8.3 Layout..... | 284 |
| 6.8 Monitoring Functions..... | 18 | 8.4 Power Supply Recommendations..... | 287 |
| 6.9 Clocks, Oscillators, and DPLL..... | 21 | 9 Device and Documentation Support | 288 |
| 6.10 Thermal Monitoring and Shutdown..... | 22 | 9.1 Receiving Notification of Documentation Updates..... | 288 |
| 6.11 System Control Thresholds..... | 23 | 9.2 Support Resources..... | 288 |
| 6.12 Current Consumption..... | 23 | 9.3 Trademarks..... | 288 |
| 6.13 Digital Input Signal Parameters..... | 24 | 9.4 Electrostatic Discharge Caution..... | 288 |
| 6.14 Digital Output Signal Parameters..... | 24 | 9.5 Glossary..... | 288 |
| 6.15 I/O Pullup and Pulldown Resistance..... | 25 | 10 Mechanical, Packaging, and Orderable Information | 288 |
| 6.16 I ² C Interface..... | 25 | | |

4 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

| Changes from Revision * (June 2021) to Revision A (September 2022) | Page |
|--|------|
| • Changed the device status from Advance Information to Production Data..... | 1 |

5 Pin Configuration and Functions

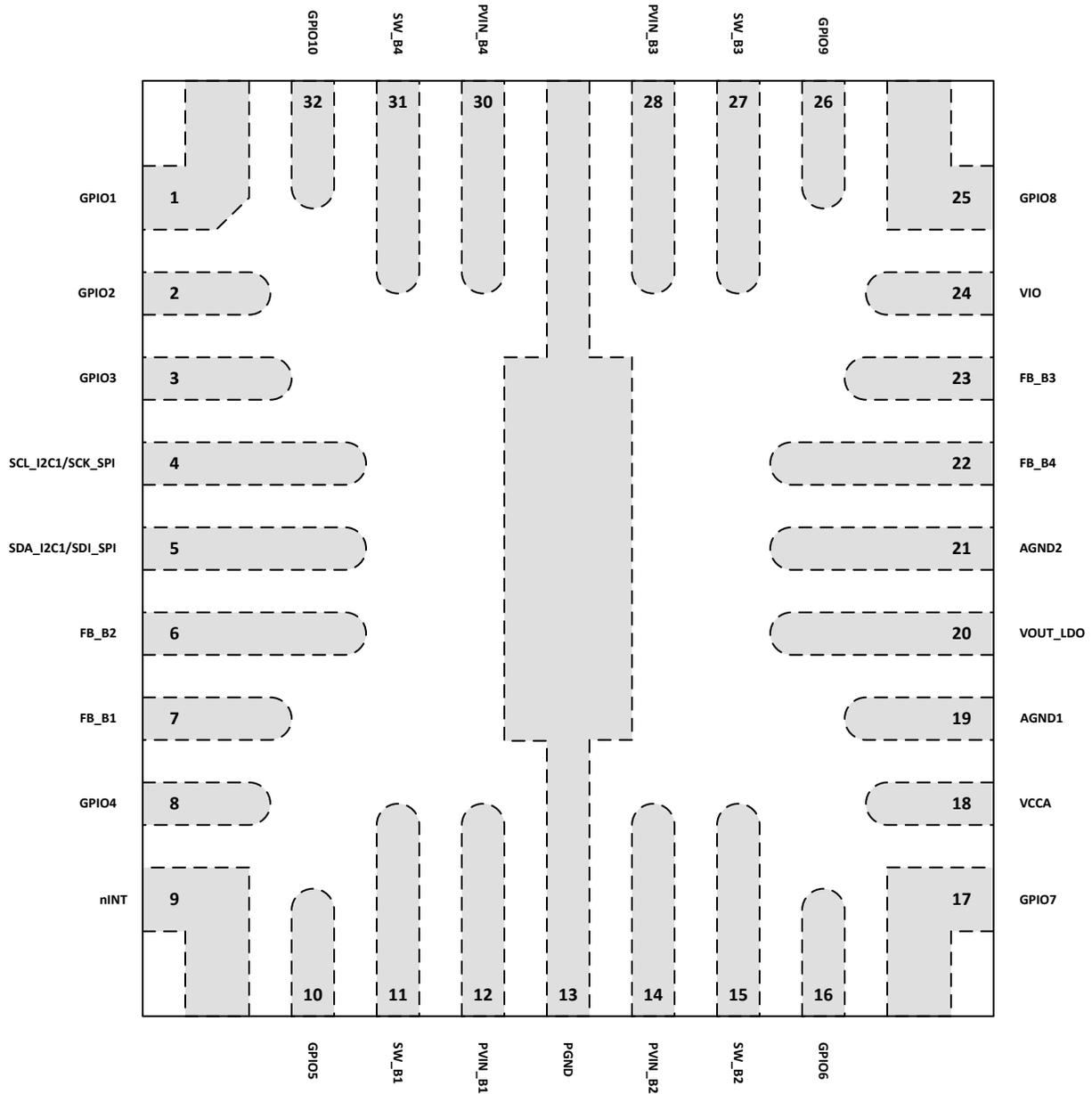


Figure 5-1. RQK Package 32-Pin VQFN-HR Top View

Table 5-1. Pin Functions

| PIN | | I/O | TYPE | DESCRIPTION | CONNECTION IF NOT USED |
|-----|----------------------|-----|---------|---|---------------------------------|
| NO. | NAME | | | | |
| 1 | GPIO1 | I/O | Digital | Primary function: General Purpose Input/Output signal. When configured as an output pin, it can be included as part of the power sequencer output signal to enable an external regulator. | Input: Ground, Output: Floating |
| | | O | Digital | Alternative programmable function: EN_DRV - Enable Drive output pin to indicate the device entering safe state (set low when ENABLE_DRV bit is '0'). | Floating |
| | | O | Digital | Alternative programmable function: nRSTOUT_SOC - System reset or power on reset output (low = reset). | Floating |
| | | O | Digital | Alternative programmable function: PGOOD - Programmable Power Good indication pin. | Floating |
| | | I | Digital | Alternative programmable function: nSLEEP1 or nSLEEP2, which are the sleep request signals for the device to go to lower power states (Active Low). | Ground |
| | | I | Digital | Alternative programmable function: WKUP1 or WKUP2, which are the wake-up request signals for the device to go to higher power states. | Ground |
| 2 | GPIO2 | I/O | Digital | Primary function: General Purpose Input/Output signal. When configured as an output pin, it can be included as part of the power sequencer output signal to enable an external regulator. | Input: Ground, Output: Floating |
| | | I | Digital | Alternative programmable function: SCL_I2C2 - Serial interface clock input for I2C access. | Ground |
| | | I | Digital | Alternative programmable function: CS_SPI - Serial interface Chip Select signal for SPI access. | Ground |
| | | I | Digital | Alternative programmable function: TRIG_WDOG - Trigger signal for trigger mode watchdog. | Ground |
| | | I | Digital | Alternative programmable function: nSLEEP1 or nSLEEP2, which are the sleep request signals for the device to go to lower power states (Active Low). | Ground |
| | | I | Digital | Alternative programmable function: WKUP1 or WKUP2, which are the wake-up request signals for the device to go to higher power states. | Ground |
| 3 | GPIO3 | I/O | Digital | Primary function: General Purpose Input/Output signal. When configured as an output pin, it can be included as part of the power sequencer output signal to enable an external regulator. | Input: Ground, Output: Floating |
| | | I/O | Digital | Alternative programmable function: SDA_I2C2 - Serial interface data input and output for I2C access. | Ground |
| | | O | Digital | Alternative programmable function: SDO_SPI - Serial interface data output signal for SPI access. | Floating |
| | | I | Digital | Alternative programmable function: nSLEEP1 or nSLEEP2, which are the sleep request signals for the device to go to lower power states (Active Low). | Ground |
| | | I | Digital | Alternative programmable function: WKUP1 or WKUP2, which are the wake-up request signals for the device to go to higher power states. | Ground |
| 4 | SCL_I2C1/ SCK_SPI | I | Digital | If SPI is not used: SCL_I2C1 - Serial interface clock input for I2C access. | Ground |
| | | I | Digital | If SPI is used: SCK_SPI - Serial interface clock input for SPI access. | Ground |
| 5 | SDA_I2C1/ SDI_SPI | I/O | Digital | If SPI is not used: SDA_I2C1 - Serial interface data input and output for I2C access. | Ground |
| | | I | Digital | If SPI is used: SDI_SPI - Serial interface data input signal for SPI access. | Ground |
| 6 | FB_B2 | — | Analog | Output voltage feedback (positive) for BUCK2. | Ground |
| 7 | FB_B1 | — | Analog | Output voltage feedback (positive) for BUCK1. | Ground |

Table 5-1. Pin Functions (continued)

| PIN | | I/O | TYPE | DESCRIPTION | CONNECTION IF NOT USED |
|-----|---------|-----|---------|---|---------------------------------|
| NO. | NAME | | | | |
| 8 | GPIO4 | I/O | Digital | Primary function: General Purpose Input/Output signal. When configured as an output pin, it can be included as part of the power sequencer output signal to enable an external regulator. | Input: Ground, Output: Floating |
| | | I | Digital | Alternative programmable function: ENABLE - External power-on control. | Ground |
| | | I | Digital | Alternative programmable function: TRIG_WDOG - Trigger signal for trigger mode watchdog. | Ground |
| | | — | Analog | Alternative programmable function: BUCK1_VMON - Voltage monitoring input for BUCK1 regulator. | Ground |
| | | I | Digital | Alternative programmable function: nSLEEP1 or nSLEEP2, which are the sleep request signals for the device to go to lower power states (Active Low). | Ground |
| | | I | Digital | Alternative programmable function: WKUP1 or WKUP2, which are the wake-up request signals for the device to go to higher power states. | Ground |
| 9 | nINT | O | Digital | Open-drain interrupt output, active LOW. | Floating |
| 10 | GPIO5 | I/O | Digital | Primary function: General Purpose Input/Output signal. When configured as an output pin, it can be included as part of the power sequencer output signal to enable an external regulator. | Input: Ground, Output: Floating |
| | | I | Digital | Alternative programmable function: SYNCCLKIN - External switching clock input for Buck regulators. | Ground |
| | | O | Digital | Alternative programmable function: SYNCCLKOUT - Switching clock output for external regulators. | Floating |
| | | O | Digital | Alternative programmable function: nRSTOUT_SOC - System reset or power on reset output (low = reset). | Floating |
| | | I | Digital | Alternative programmable function: nSLEEP1 or nSLEEP2, which are the sleep request signals for the device to go to lower power states (Active Low). | Ground |
| | | I | Digital | Alternative programmable function: WKUP1 or WKUP2, which are the wake-up request signals for the device to go to higher power states. | Ground |
| 11 | SW_B1 | — | Analog | BUCK1 switch node. | Floating |
| 12 | PVIN_B1 | — | Power | Power input for BUCK1. The separate power pins PVIN_Bx are not connected together internally – PVIN_Bx and VCCA pins must be connected together in the application and be locally bypassed. | System supply |
| 13 | PGND | — | Ground | Power ground for Buck regulators. | Ground |
| 14 | PVIN_B2 | — | Power | Power input for BUCK2. The separate power pins PVIN_Bx are not connected together internally – PVIN_Bx and VCCA pins must be connected together in the application and be locally bypassed. | System supply |
| 15 | SW_B2 | — | Analog | BUCK2 switch node. | Floating |
| 16 | GPIO6 | I/O | Digital | Primary function: General Purpose Input/Output signal. When configured as an output pin, it can be included as part of the power sequencer output signal to enable an external regulator. | Input: Ground, Output: Floating |
| | | I | Digital | Alternative programmable function: nERR_MCU - System error count down input signal from the MCU. | Floating |
| | | O | Digital | Alternative programmable function: SYNCCLKOUT - Switching clock output for external regulators. | Floating |
| | | O | Digital | Alternative programmable function: PGOOD - Programmable Power Good indication pin. | Floating |
| | | I | Digital | Alternative programmable function: nSLEEP1 or nSLEEP2, which are the sleep request signals for the device to go to lower power states (Active Low). | Ground |
| | | I | Digital | Alternative programmable function: WKUP1 or WKUP2, which are the wake-up request signals for the device to go to higher power states. | Ground |

Table 5-1. Pin Functions (continued)

| PIN | | I/O | TYPE | DESCRIPTION | CONNECTION IF NOT USED |
|-----|----------|-----|---------|---|---------------------------------|
| NO. | NAME | | | | |
| 17 | GPIO7 | I/O | Digital | Primary function: General Purpose Input/Output signal. When configured as an output pin, it can be included as part of the power sequencer output signal to enable an external regulator. | Input: Ground, Output: Floating |
| | | I | Digital | Alternative programmable function: nERR_MCU - System error count down input signal from the MCU. | Floating |
| | | O | Analog | Alternative programmable function: REFOUT - Buffered bandgap output. | Floating |
| | | I | Analog | Alternative programmable function: VMON1 - External voltage monitoring input. | Ground |
| | | I | Digital | Alternative programmable function: nSLEEP1 or nSLEEP2, which are the sleep request signals for the device to go to lower power states (Active Low). | Ground |
| | | I | Digital | Alternative programmable function: WKUP1 or WKUP2, which are the wake-up request signals for the device to go to higher power states. | Ground |
| 18 | VCCA | — | Power | Supply voltage for internal LDO. VCCA and PVIN_Bx pins must be connected together in the application and be locally bypassed. | System supply |
| 19 | AGND1 | — | Ground | Ground | Ground |
| 20 | VOUT_LDO | — | Power | LDO regulator filter node. LDO is used for internal purposes. | — |
| 21 | AGND2 | — | Ground | Ground | Ground |
| 22 | FB_B4 | — | Analog | Output voltage feedback (positive) for BUCK4. | Ground |
| 23 | FB_B3 | — | Analog | Output voltage feedback (positive) for BUCK3. | Ground |
| 24 | VIO | — | Power | Supply voltage for selected digital outputs. | Ground |
| 25 | GPIO8 | I/O | Digital | Primary function: General Purpose Input/Output signal. When configured as an output pin, it can be included as part of the power sequencer output signal to enable an external regulator. | Input: Ground, Output: Floating |
| | | I/O | Digital | Alternative programmable function: SCLK_SPMI - Multi-PMIC SPMI serial interface clock signal. This pin is an output pin for the master SPMI device, and an input pin for the slave SPMI device. | Ground |
| | | I | Analog | Alternative programmable function: VMON2 - External voltage monitoring input. | Ground |
| | | I | Digital | Alternative programmable function: nSLEEP1 or nSLEEP2, which are the sleep request signals for the device to go to lower power states (Active Low). | Ground |
| | | I | Digital | Alternative programmable function: WKUP1 or WKUP2, which are the wake-up request signals for the device to go to higher power states. | Ground |
| 26 | GPIO9 | I/O | Digital | Primary function: General Purpose Input/Output signal. When configured as an output pin, it can be included as part of the power sequencer output signal to enable an external regulator. | Input: Ground, Output: Floating |
| | | I/O | Digital | Alternative programmable function: SDATA_SPMI - Multi-PMIC SPMI serial interface bidirectional data signal | Floating |
| | | O | Digital | Alternative programmable function: PGOOD - Programmable Power Good indication pin. | Floating |
| | | I | Digital | Alternative programmable function: SYNCCLKIN - External switching clock input for Buck regulators. | Ground |
| | | I | Digital | Alternative programmable function: nSLEEP1 or nSLEEP2, which are the sleep request signals for the device to go to lower power states (Active Low). | Ground |
| | | I | Digital | Alternative programmable function: WKUP1 or WKUP2, which are the wake-up request signals for the device to go to higher power states. | Ground |
| 27 | SW_B3 | — | Analog | BUCK3 switch node. | Floating |
| 28 | PVIN_B3 | — | Power | Power input for BUCK3. The separate power pins PVIN_Bx are not connected together internally – PVIN_Bx and VCCA pins must be connected together in the application and be locally bypassed. | System supply |
| 30 | PVIN_B4 | — | Power | Power input for BUCK4. The separate power pins PVIN_Bx are not connected together internally – PVIN_Bx and VCCA pins must be connected together in the application and be locally bypassed. | System supply |

Table 5-1. Pin Functions (continued)

| PIN | | I/O | TYPE | DESCRIPTION | CONNECTION IF NOT USED |
|-----|--------|-----|---------|---|---------------------------------|
| NO. | NAME | | | | |
| 31 | SW_B4 | — | Analog | BUCK4 switch node. | Floating |
| 32 | GPIO10 | I/O | Digital | Primary function: General Purpose Input/Output signal. When configured as an output pin, it can be included as part of the power sequencer output signal to enable an external regulator. | Input: Ground, Output: Floating |
| | | O | Digital | Alternative programmable function: nRSTOUT - System reset or power on reset output (low = reset). | Floating |
| | | O | Digital | Alternative programmable function: nRSTOUT_SOC - System reset or power on reset output (low = reset). | Floating |
| | | I | Digital | Alternative programmable function: nSLEEP1 or nSLEEP2, which are the sleep request signals for the device to go to lower power states (Active Low). | Ground |
| | | I | Digital | Alternative programmable function: WKUP1 or WKUP2, which are the wake-up request signals for the device to go to higher power states. | Ground |

5.1 Digital Signal Descriptions

Table 5-2. Signal Descriptions

| SIGNAL NAME | I/O | Threshold Level | INPUT TYPE SELECTION | | OUTPUT TYPE SELECTION | | Internal PU/PD ⁽²⁾ | RECOMMENDED EXTERNAL PU/PD ⁽³⁾ | Control Registers |
|--|------------------|---|----------------------|--|-----------------------|-------------------------------------|--|---|--|
| | | | Power Domain | DEGLITCH TIME ⁽⁵⁾ | Power Domain | Push-pull/Open-drain ⁽⁴⁾ | | | |
| ENABLE (Selectable function of GPIO4 pin) ⁽¹⁾ | Input | V_{IL} , V_{IH} | VINT | 8 μ s | | | 400 k Ω SPU to VINT, or 400 k Ω SPD to GND | None | GPIO4_SEL GPIO4_DEGLITCH_EN GPIO4_PU_PD_EN GPIO4_PU_SEL ENABLE_POL |
| EN_DRV (Selectable function of GPIO1 pin) ⁽¹⁾ | Output | $V_{OL_20\text{ mA}}$ | | | VCCA/ PVIN_B4 | PP with 10k Ω PU to VCCA | 10 k Ω PU to VCCA | None | GPIO1_SEL FORCE_EN_DRV_LOW ENABLE_DRV |
| SCL_I2C1 (Selectable function of SCL_I2C1/ SCK_SPI pin) ⁽¹⁾ | Input | V_{IL} , V_{IH} | VINT | High-speed mode: 10 ns All other modes: 50 ns | | | None | PU to VIO | NVM-configuration ⁽⁶⁾ I2C1_HS |
| SDA_I2C1 (Selectable function of SDA_I2C1/ SDI_SPI pin) ⁽¹⁾ | Input/ output | V_{IL} , V_{IH} $V_{OL_20\text{ mA}}$ | VINT | High-speed mode: 10 ns All other modes: 50 ns | VIO | OD | None | PU to VIO | NVM-configuration ⁽⁶⁾ I2C1_HS |
| SCL_I2C2 (Selectable function of GPIO2 pin) ⁽¹⁾ | Input | V_{IL} , V_{IH} | VINT | High-speed mode: 10 ns All other modes: 50 ns | | | None | PU to VIO | NVM-configuration ⁽⁶⁾ GPIO2_SEL I2C2_HS |
| SDA_I2C2 (Selectable function of GPIO3 pin) ⁽¹⁾ | Input/ output | V_{IL} , V_{IH} $V_{OL_20\text{ mA}}$ | VINT | High-speed mode: 10 ns All other modes: 50 ns | VIO | OD | None | PU to VIO | NVM-configuration ⁽⁶⁾ GPIO3_SEL I2C2_HS |
| SCK_SPI (Selectable function of SCL_I2C1/ SCK_SPI pin) ⁽¹⁾ | Input | V_{IL} , V_{IH} | VINT | None | | | None | None | NVM-configuration ⁽⁶⁾ |
| SDI_SPI (Selectable function of SDA_I2C1/ SDI_SPI pin) ⁽¹⁾ | Input | V_{IL} , V_{IH} | VINT | None | | | None | None | NVM-configuration ⁽⁶⁾ |

Table 5-2. Signal Descriptions (continued)

| SIGNAL NAME | I/O | Threshold Level | INPUT TYPE SELECTION | | OUTPUT TYPE SELECTION | | Internal PU/PD ⁽²⁾ | RECOMMENDED EXTERNAL PU/PD ⁽³⁾ | Control Registers |
|--|--|--|----------------------|------------------------------|-----------------------|-------------------------------------|------------------------------------|---|---|
| | | | Power Domain | DEGLITCH TIME ⁽⁵⁾ | Power Domain | Push-pull/Open-drain ⁽⁴⁾ | | | |
| CS_SPI (Selectable function of GPIO2 pin) ⁽¹⁾ | Input | V_{IL}, V_{IH} | VINT | None | | | None | None | NVM-configuration ⁽⁶⁾ GPIO2_SEL |
| SDO_SPI (Selectable function of GPIO3 pin) ⁽¹⁾ | Output | $V_{OL_20\text{ mA}}, V_{OH(VIO)}$ | | | VIO | PP / HiZ | None | None | NVM-configuration ⁽⁶⁾ GPIO3_SEL |
| SCLK_SPMI (Configurable function of GPIO8 pin) ⁽¹⁾ | Input in Slave Mode Output in Master Mode | V_{IL}, V_{IH} $V_{OL_20\text{ mA}}$ $V_{OH(VINT)}$ | VINT | None | VINT | PP | 400 kΩ SPD to GND | None | GPIO8_SEL GPIO8_PU_PD_EN NVM-configuration ⁽⁶⁾ |
| SDATA_SPMI (Configurable function of GPIO9 pin) ⁽¹⁾ | Input/output | V_{IL}, V_{IH} $V_{OL_20\text{ mA}}$ $V_{OH(IO)}$ | VINT | None | VINT | PP / HiZ | 400 kΩ SPD to GND | None | GPIO9_SEL GPIO9_PU_PD_EN NVM-configuration ⁽⁶⁾ |
| nINT | Output | $V_{OL_20\text{ mA}}$ | | | VIO | OD | None | PU to VIO | |
| nRSTOUT(Configurable function of GPIO10 pin) ⁽¹⁾ | Output | $V_{OL_20\text{ mA}}$ | | | VIO | PP with 10 kΩ PU to VIO or OD | 10kΩ PU to VIO if configured as PP | PU to VIO or VCCA if Open-drain | GPIO10_SEL GPIO10_OD |
| nRSTOUT_SOC (Configurable function of GPIO1, GPIO5, GPIO10 pins) ⁽¹⁾ | Output | $V_{OL_20\text{ mA}}$ | | | VIO | PP with 10kΩ PU to VIO or OD | 10kΩ PU to VIO if configured as PP | PPU to VIO or VCCA if Open-drain | GPIO1_SEL GPIO1_OD GPIO5_SEL GPIO5_OD GPIO10_SEL GPIO10_OD |
| PGOOD (Configurable function of GPIO1, GPIO6, GPIO9 pins) ⁽¹⁾ | Output | GPIO1: $V_{OL_20\text{ mA}}, V_{OH(VIO)}$ GPIO6: $V_{OL_3\text{ mA}}, V_{OH(VIO)}$ GPIO9: $V_{OL_20\text{ mA}}, V_{OH(VINT)}$ | | | VIO / VINT | PP or OD | None | PU to VIO if Open-drain | GPIO1_SEL GPIO1_OD GPIO6_SEL GPIO6_OD GPIO9_SEL GPIO9_OD PGOOD_POL PGOOD_WINDOW PGOOD_SEL_x |

Table 5-2. Signal Descriptions (continued)

| SIGNAL NAME | I/O | Threshold Level | INPUT TYPE SELECTION | | OUTPUT TYPE SELECTION | | Internal PU/PD ⁽²⁾ | RECOMMEND ED EXTERNAL PU/PD ⁽³⁾ | Control Registers |
|--|------------------|---|----------------------|------------------------------|-----------------------|---|---|--|---|
| | | | Power Domain | DEGLITCH TIME ⁽⁵⁾ | Power Domain | Push-pull/ Open-drain ⁽⁴⁾ | | | |
| nERR_MCU (Configurable function of GPIO6, GPIO7 pins) ⁽¹⁾ | Input | V_{IL} , V_{IH} | VINT | 8 μ s | | | 400 k Ω PD to GND | None | GPIO6_SEL GPIO7_SEL |
| TRIG_WDOG (Configurable function of GPIO2, GPIO4 pins) ⁽¹⁾ | Input | V_{IL} , V_{IH} | VINT | 30 μ s | | | 400 k Ω SPD to GND | None | GPIO2_SEL GPIO2_PU_PD_EN GPIO4_SEL GPIO4_PU_PD_EN |
| nSLEEP1 (Configurable function of all GPIO pins) ⁽¹⁾ | Input | V_{IL} , V_{IH} | VINT | 8 μ s | | | GPIO4, 7, 8 or 9: 400 k Ω SPU to VINT GPIO1, 2, 3, 5, 6, or 10: 400 k Ω SPU to VIO | None | GPIOx_SEL GPIOx_PU_PD_EN |
| nSLEEP2 (Configurable function of all GPIO pins) ⁽¹⁾ | Input | V_{IL} , V_{IH} | VINT | 8 μ s | | | GPIO4, 7, 8 or 9: 400 k Ω SPU to VINT GPIO1, 2, 3, 5, 6, or 10: 400 k Ω SPU to VIO | None | GPIOx_SEL GPIOx_PU_PD_EN |
| WKUP1 (Configurable function of all GPIO pins) ⁽¹⁾ | Input | V_{IL} , V_{IH} | VINT | 8 μ s | | | GPIO4, 7, 8 or 9: 400 k Ω SPU to VINT 400 k Ω SPD to GND GPIO1, 2, 3, 5, 6, or 10: 400 k Ω SPU to VIO 400 k Ω SPD to GND | None | GPIOx_SEL GPIOx_DEGLITCH_EN GPIOx_PU_PD_EN GPIOx_PU_SEL |
| WKUP2 (Configurable function of all GPIO pins) ⁽¹⁾ | Input | V_{IL} , V_{IH} | VINT | 8 μ s | | | GPIO4, 7, 8 or 9: 400 k Ω SPU to VINT 400 k Ω SPD to GND GPIO1, 2, 3, 5, 6, or 10: 400 k Ω SPU to VIO 400 k Ω SPD to GND | None | GPIOx_SEL GPIOx_DEGLITCH_EN GPIOx_PU_PD_EN GPIOx_PU_SEL |
| GPIO1 (Configurable function of GPIO1 pin) ⁽¹⁾ | Input/ output | V_{IL} , V_{IH} $V_{OL,20\text{ mA}}$ $V_{OH(VIO)}$ | VINT | 8 μ s | VIO | PP or OD | 400 k Ω SPU to VIO 400 k Ω SPD to GND | PU to VIO or VCCA if Open-drain | GPIO1_SEL GPIO1_DEGLITCH_EN GPIO1_PU_PD_EN GPIO1_PU_SEL GPIO1_OD GPIO1_DIR |

Table 5-2. Signal Descriptions (continued)

| SIGNAL NAME | I/O | Threshold Level | INPUT TYPE SELECTION | | OUTPUT TYPE SELECTION | | Internal PU/PD ⁽²⁾ | RECOMMENDED EXTERNAL PU/PD ⁽³⁾ | Control Registers |
|--|--------------|---|----------------------|------------------------------|-----------------------|-------------------------------------|---|---|---|
| | | | Power Domain | DEGLITCH TIME ⁽⁵⁾ | Power Domain | Push-pull/Open-drain ⁽⁴⁾ | | | |
| GPIO2 (Configurable function of GPIO2 pin) ⁽¹⁾ | Input/output | V_{IL} , V_{IH} $V_{OL_3\text{ mA}}$ $V_{OH(VIO)}$ | VINT | 8 μs | VIO | PP or OD | 400 k Ω SPU to VIO 400 k Ω SPD to GND | PU to VIO or VCCA if Open-drain | GPIO2_SEL GPIO2_DEGLITCH_EN GPIO2_PU_PD_EN GPIO2_PU_SEL GPIO2_OD GPIO2_DIR |
| GPIO3 (Configurable function of GPIO3 pin) ⁽¹⁾ | Input/output | V_{IL} , V_{IH} $V_{OL_20\text{ mA}}$ $V_{OH(VIO)}$ | VINT | 8 μs | VIO | PP or OD | 400 k Ω SPU to VIO 400 k Ω SPD to GND | PU to VIO or VCCA if Open-drain | GPIO3_SEL GPIO3_DEGLITCH_EN GPIO3_PU_PD_EN GPIO3_PU_SEL GPIO3_OD GPIO3_DIR |
| GPIO4 (Configurable function of GPIO4 pin) ⁽¹⁾ | Input/output | V_{IL} , V_{IH} $V_{OL_3\text{ mA}}$ $V_{OH(VINT)}$ | VINT | 8 μs | VINT | PP or OD | 400 k Ω SPU to VINT 400 k Ω SPD to GND | PU to VIO if Open-drain | GPIO4_SEL GPIO4_DEGLITCH_EN GPIO4_PU_PD_EN GPIO4_PU_SEL GPIO4_OD GPIO4_DIR |
| GPIO5 (Configurable function of GPIO5 pin) ⁽¹⁾ | Input/output | V_{IL} , V_{IH} $V_{OL_20\text{ mA}}$ $V_{OH(VIO)}$ | VINT | 8 μs | VIO | PP or OD | 400 k Ω SPU to VIO 400 k Ω SPD to GND | PU to VIO or VCCA if Open-drain | GPIO5_SEL GPIO5_DEGLITCH_EN GPIO5_PU_PD_EN GPIO5_PU_SEL GPIO5_OD GPIO5_DIR |
| GPIO6 (Configurable function of GPIO6 pin) ⁽¹⁾ | Input/output | V_{IL} , V_{IH} $V_{OL_3\text{ mA}}$ $V_{OH(VIO)}$ | VINT | 8 μs | VIO | PP or OD | 400 k Ω SPU to VIO 400 k Ω SPD to GND | PU to VIO or VCCA if Open-drain | GPIO6_SEL GPIO6_DEGLITCH_EN GPIO6_PU_PD_EN GPIO6_PU_SEL GPIO6_OD GPIO6_DIR |
| GPIO7 (Configurable function of GPIO7 pin) ⁽¹⁾ | Input/output | V_{IL} , V_{IH} $V_{OL_3\text{ mA}}$ $V_{OH(VINT)}$ | VINT | 8 μs | VINT | PP or OD | 400 k Ω SPU to VINT 400 k Ω SPD to GND | PU to VIO if Open-drain | GPIO7_SEL GPIO7_DEGLITCH_EN GPIO7_PU_PD_EN GPIO7_PU_SEL GPIO7_OD GPIO7_DIR |
| GPIO8 (Configurable function of GPIO8 pin) ⁽¹⁾ | Input/output | V_{IL} , V_{IH} $V_{OL_20\text{ mA}}$ $V_{OH(VINT)}$ | VINT | 8 μs | VINT | PP or OD | 400 k Ω SPU to VINT 400 k Ω SPD to GND | PU to VIO if Open-drain | GPIO8_SEL GPIO8_DEGLITCH_EN GPIO8_PU_PD_EN GPIO8_PU_SEL GPIO8_OD GPIO8_DIR |

Table 5-2. Signal Descriptions (continued)

| SIGNAL NAME | I/O | Threshold Level | INPUT TYPE SELECTION | | OUTPUT TYPE SELECTION | | Internal PU/PD ⁽²⁾ | RECOMMENDED EXTERNAL PU/PD ⁽³⁾ | Control Registers |
|---|--------------|---|----------------------|------------------------------|-----------------------|-------------------------------------|---|---|---|
| | | | Power Domain | DEGLITCH TIME ⁽⁵⁾ | Power Domain | Push-pull/Open-drain ⁽⁴⁾ | | | |
| GPIO9 (Configurable function of GPIO9 pin) ⁽¹⁾ | Input/output | V_{IL} , V_{IH} $V_{OL_20\text{ mA}}$ $V_{OH(VINT)}$ | VINT | 8 μ s | VINT | PP or OD | 400 k Ω SPU to VINT 400 k Ω SPD to GND | PU to VIO if Open-drain | GPIO9_SEL GPIO9_DEGLITCH_EN GPIO9_PU_PD_EN GPIO9_PU_SEL GPIO9_OD GPIO9_DIR |
| GPIO10 (Configurable function of GPIO10 pin) ⁽¹⁾ | Input/output | V_{IL} , V_{IH} $V_{OL_20\text{ mA}}$ $V_{OH(VIO)}$ | VINT | 8 μ s | VIO | PP or OD | 400 k Ω SPU to VIO 400 k Ω SPD to GND | PU to VIO or VCCA if Open-drain | GPIO10_SEL GPIO10_DEGLITCH_EN GPIO10_PU_PD_EN GPIO10_PU_SEL GPIO10_OD GPIO10_DIR |
| SYNCCLKIN (Configurable function of GPIO5, GPIO9 pins) ⁽¹⁾ | Input | V_{IL} , V_{IH} | VINT | None | | | 400 k Ω SPD to GND | None | GPIO5_SEL GPIO5_PU_PD_EN GPIO9_SEL GPIO9_PU_PD_EN |
| SYNCCLKOUT (Configurable function of GPIO5, GPIO6 pins) ⁽¹⁾ | Output | GPIO5: $V_{OL_20\text{ mA}}$, $V_{OH(VIO)}$ GPIO6: $V_{OL_3\text{ mA}}$, $V_{OH(VIO)}$ | | | VIO | PP | None | None | GPIO5_SEL GPIO6_SEL |
| VMON1 (Configurable function of GPIO7 pin) ⁽¹⁾ | Input | Analog | | | | | None | None | GPIO7_SEL VMON1_EN VMON1_RANGE_SEL |
| VMON2 (Configurable function of GPIO8 pin) ⁽¹⁾ | Input | Analog | | | | | None | None | GPIO8_SEL VMON2_EN VMON2_RANGE_SEL |
| BUCK1_VMON (Configurable function of GPIO4 pin) ⁽¹⁾ | Input | Analog | | | | | None | None | GPIO4_SEL BUCK1_VMON_EN |
| REFOUT (Configurable function of GPIO7 pin) ⁽¹⁾ | Output | Analog | | | | | 400 k Ω PD to GND when REFOUT_EN = 0 | | GPIO7_SEL REFOUT_EN |

(1) Configurable function through NVM register setting.

(2) PU = Pullup, PD = Pulldown, SPU = Software-configurable pullup, SPD = Software-configurable pulldown.

(3) The internal pull-up and pull-down resistors are automatically disabled when the device is configured as a push-pull output pin unless otherwise noted.

- (4) PP = Push-pull, OD = Open-drain.
- (5) Deglitch time is only applicable when option is enabled.
- (6) NVM-configuration for I2C/SPI and SPMI cannot be overwritten during operation.

6 Specifications

6.1 Absolute Maximum Ratings

Over operating free-air temperature range (unless otherwise noted). Voltage level refers to the AGNDx ground of the device.
(1)

| POS | | | MIN | MAX | UNIT |
|--------|--|--|------|--|-------|
| M1.1 | Voltage on supply input pin | VCCA | -0.3 | 6 | V |
| M1.2 | Voltage on all buck supply voltage input pins | PVIN_Bx | -0.3 | 6 | V |
| M1.3 | Voltage difference between supply input pins | Between VCCA and each PVIN_Bx | -0.5 | 0.5 | V |
| M1.4a | Voltage on all buck switch nodes | SW_Bx pins | -0.3 | $V_{PVIN_Bx} + 0.3\text{ V}$, up to 6 V | V |
| M1.4b | Voltage on all buck switch nodes | SW_Bx pins, 10-ns transient | -2 | 10 | V |
| M1.5 | Voltage on all buck voltage sense nodes | FB_Bx | -0.3 | 4 | V |
| M1.6 | Voltage on all buck power ground pins | PGND | -0.3 | 0.3 | V |
| M1.7 | Voltage on internal LDO output pin | VOUT_LDO | -0.3 | 2 | V |
| M1.8 | Voltage on I/O supply pin | VIO | -0.3 | 6 | V |
| M1.9 | Voltage on logic pins (input or output) | I ² C and SPI pins, nINT pin, and all GPIO pins | -0.3 | 6 | V |
| M1.13a | Voltage rise slew-rate on input supply pins | VCCA, PVIN_Bx (voltage below 2.7 V) | | 60 | mV/μs |
| M1.13b | | VIO (only when VCCA < 2 V) | | 60 | |
| M1.10a | Peak output current | All pins other than power resources | | 20 | mA |
| M1.10b | | Buck regulators: PVIN_Bx, SW_Bx, and PGNDx per phase | | 6 | |
| M1.10c | | GPIOx pins, source current | | 3 | |
| M1.10d | Average output current, 100 k hour, T _J = 125°C | GPIO1/3/5/8/9/10, SDA_I2C1/SDI_SPI and nINT pins, sink current | | 8 | mA |
| M1.10e | | GPIO2/4/6/7 pins, sink current | | 3 | |
| M1.10f | | Buck regulators | | 3.5 | |
| M1.11 | Junction temperature, T _J | | -45 | 160 | °C |
| M1.12 | Storage temperature, T _{stg} | | -65 | 150 | °C |

- (1) Operation outside the *Absolute Maximum Ratings* may cause permanent device damage. *Absolute Maximum Ratings* do not imply functional operation of the device at these or any other conditions beyond those listed under *Recommended Operating Conditions*. If used outside the *Recommended Operating Conditions* but within the *Absolute Maximum Ratings*, the device may not be fully functional, and this may affect device reliability, functionality, performance, and shorten the device lifetime.

6.2 ESD Ratings

| POS | | | | VALUE | UNIT |
|-------|--------------------|-------------------------|---|-------|------|
| M1.13 | V _(ESD) | Electrostatic discharge | Human-body model (HBM), per AEC Q100-002 ⁽¹⁾ | ±2000 | V |
| M1.14 | V _(ESD) | Electrostatic discharge | Charged-device model (CDM), per AEC Q100-011 | ±500 | V |

- (1) AEC Q100-002 indicates that HBM stressing must be in accordance with the ANSI/ESDA/JEDEC JS-001 specification.

6.3 Recommended Operating Conditions

Over operating free-air temperature range (unless otherwise noted). Voltage level refers to the AGNDx ground of the device.

| POS | | | MIN | NOM | MAX | UNIT |
|--------|--|-------------------------------|-------|---------------------|-------------------------------------|------|
| R1.1 | Voltage on supply input pin | VCCA | 2.8 | 3.3 | 5.5 | V |
| R1.2 | Voltage on all buck supply input pins | PVIN_Bx | 2.8 | 3.3 | 5.5 | V |
| R1.3 | Voltage difference between supply input pins | Between VCCA and each PVIN_Bx | -0.2 | | 0.2 | V |
| R1.4 | Voltage on all buck switch nodes | SW_Bx pins | 0 | | 5.5 | V |
| R1.5 | Voltage on all buck voltage sense nodes | FB_Bx | 0 | $V_{OUT(BUCKx)max}$ | | V |
| R1.6 | Voltage on all buck power ground pins | Between PGND and AGNDx | | 0 | | V |
| R1.7 | Voltage on internal LDO output pin | VOUT_LDO | 1.65 | | 1.95 | V |
| R1.8a | Voltage on I/O supply pin | $V_{VIO} = 1.8\text{ V}$ | 1.7 | 1.8 | 1.9 | V |
| R1.8b | | $V_{VIO} = 3.3\text{ V}$ | 3.135 | 3.3 | V_{VCCA} , up to 3.465V | |
| R1.9 | Voltage on logic pins (input) ⁽²⁾ | | 0 | | 5.5 | |
| R1.10a | Voltage on logic pins (output, push-pull) in VIO domain ⁽²⁾ | | 0 | | V_{VIO} | V |
| R1.10b | Voltage on logic pins (output, push-pull) in LDOVINT domain ⁽²⁾ | | 0 | | V_{VOUT_LDO} | V |
| R1.10c | Voltage on logic pins (output, open-drain) ⁽²⁾ | | 0 | | 5.5 | |
| R1.11 | Voltage on logic pins (output) in VCCA domain | EN_DRV | 0 | | V_{VCCA} / V_{VIO} ⁽³⁾ | |
| R1.12 | Voltage on AGND ground pins | AGND1 and AGND2 | | 0 | | V |
| R1.13 | Operating free-air temperature ⁽¹⁾ | | -40 | 25 | 125 | °C |
| R1.14 | Junction temperature, T_J | Operational | -40 | 25 | 150 | °C |

(1) Additional cooling strategies may be necessary to keep junction temperature at recommended limits.

(2) Internal pull-up resistor is disabled if pin voltage is above V_{VOUT_LDO} (LDOVINT domain pins) or V_{VIO} (VIO domain pins)

(3) V_{VCCA} or V_{VIO} , whichever has lower voltage.

6.4 Thermal Information

| THERMAL METRIC ⁽¹⁾ | | LP876x-Q1 | UNIT |
|-------------------------------|--|---------------|------|
| | | RQK (VQFN-HR) | |
| | | 32 PINS | |
| $R_{\theta JA}$ | Junction-to-ambient thermal resistance | 30.5 | °C/W |
| $R_{\theta JC(top)}$ | Junction-to-case (top) thermal resistance | 9.0 | °C/W |
| $R_{\theta JB}$ | Junction-to-board thermal resistance | 6.0 | °C/W |
| ψ_{JT} | Junction-to-top characterization parameter | 0.2 | °C/W |
| ψ_{JB} | Junction-to-board characterization parameter | 5.8 | °C/W |
| $R_{\theta JC(bot)}$ | Junction-to-case (bottom) thermal resistance | 7.5 | °C/W |

(1) For more information about traditional and new thermal metrics, see the [Semiconductor and ICPackage Thermal Metrics application report](#).

6.5 Internal Low Drop-Out Regulators (LDOVINT)

Over operating free-air temperature range, $V_{VCCA} = 3.3\text{ V}$ (unless otherwise noted). Voltage level refers to the AGNDx ground of the device.

| POS | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT | | |
|-----------------------------------|--------------------|---|---|-----|-----|------|-----|---------------|
| Electrical Characteristics | | | | | | | | |
| 2.1 | $C_{OUT(LDOVINT)}$ | Output filtering capacitance ⁽¹⁾ | Connected from VOUT_LDO to AGNDx | | 1 | 2.2 | 4 | μF |
| 2.3 | $V_{OUT(LDOVINT)}$ | LDOVINT output voltage | | | | 1.8 | | V |
| 2.7 | $I_{Qon(LDOVINT)}$ | Quiescent current, on mode | LDOVINT under valid operating condition, $I_{LOAD} = 0\text{ mA}$ | | | 3 | 10 | μA |
| 2.8 | $R_{DIS(LDOVINT)}$ | Pulldown discharge resistance at LDOVINT output | Off mode, pulldown enabled and LDO disabled | | 60 | 125 | 190 | Ω |

- (1) When DC voltage is applied to a ceramic capacitor, the effective capacitance is reduced due to DC bias effect. The table above therefore lists the minimum value as CAPACITANCE. In order to meet the minimum capacitance requirement, the nominal value of the capacitor may have to be scaled accordingly to take the drop of capacitance into account for a given dc voltage at the outputs of regulators.

6.6 BUCK1, BUCK2, BUCK3, and BUCK4 Regulators

Over operating free-air temperature range (unless otherwise noted). Voltage level refers to the AGNDx ground of the device.

| POS | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT | | |
|--|--------------------------------|---|--|-----|-------|--------|-------|-------------------|
| Electrical Characteristics - Output Voltage | | | | | | | | |
| 3.1b | $V_{VOUT_Bx_Step}$ | Output voltage programmable step size | $0.9\text{ V} \leq V_{VOUT_Bx} < 1.1\text{ V}$ | | 5 | | mV | |
| 3.1c | | | $1.1\text{ V} \leq V_{VOUT_Bx} < 1.66\text{ V}$ | | 10 | | | |
| 3.1d | | | $1.66\text{ V} \leq V_{VOUT_Bx} \leq 1.9\text{ V}$ | | 20 | | | |
| 3.3 | | Input and output voltage difference | Minimum voltage between PVIN_Bx and VOUT_Bx to fulfill the electrical characteristics | | 0.7 | | V | |
| 3.4a | $V_{VOUT_Bx_Slew_Rate}$ | Output voltage slew-rate programmable range ^{(5) (6) (8)} | BUCKn_SLEW_RATE[2:0] = 000b | | 26.6 | 33.3 | 36.6 | mV/ μs |
| 3.4b | | | BUCKn_SLEW_RATE[2:0] = 001b | | 17 | 20 | 22 | |
| 3.4c | | | BUCKn_SLEW_RATE[2:0] = 010b | | 9 | 10 | 11 | |
| 3.4d | | | BUCKn_SLEW_RATE[2:0] = 011b | | 4.5 | 5 | 5.5 | |
| 3.4e | | | BUCKn_SLEW_RATE[2:0] = 100b | | 2.25 | 2.5 | 2.75 | |
| 3.4f | | | BUCKn_SLEW_RATE[2:0] = 101b | | 1.12 | 1.25 | 1.38 | |
| 3.4g | | | BUCKn_SLEW_RATE[2:0] = 110b | | 0.56 | 0.625 | 0.69 | |
| 3.4h | | | BUCKn_SLEW_RATE[2:0] = 111b | | 0.281 | 0.3125 | 0.344 | |
| Electrical Characteristics - Output Current, Limits and Thresholds | | | | | | | | |
| 3.6e | I_{OUT_Bx} | Output current ^{(3) (4)} | Buck1 | | 2 | | A | |
| 3.6f | | | Buck2 | | 4 | | | |
| 3.6g | | | Buck3 / Buck4 | | 3 | | | |
| 3.8 | $I_{LIM\ FWD\ PEAK\ Range}$ | Forward current limit (peak during each switching cycle) Programmable range | | | 2.5 | 6.5 | A | |
| 3.9 | $I_{LIM\ FWD\ PEAK\ Step}$ | Forward current limit step Size | | | | 1 | A | |
| 3.10a | $I_{LIM\ FWD\ PEAK\ Accuracy}$ | Forward current limit accuracy | $I_{LIM} = 2.5\text{ A}, 3.5\text{ A}, 4.5\text{ A}, 3.0\text{ V} \leq V_{PVIN_Bx} \leq 5.5\text{ V}$ | | -0.55 | 0.55 | A | |
| 3.10b | | | $I_{LIM} = 5.5\text{ A}, 6.5\text{ A}, 4.5\text{ V} \leq V_{PVIN_Bx} \leq 5.5\text{ V}$ | | -10% | 10% | | |
| 3.10c | | | $I_{LIM} = 5.5\text{ A}, 6.5\text{ A}, 3.0\text{ V} \leq V_{PVIN_Bx} \leq 4.5\text{ V}$ | | -15% | 10% | | |
| 3.11 | $I_{LIM\ NEG}$ | Negative current limit (peak during each switching cycle) | | | 1.5 | 2 | 2.6 | A |
| Electrical Characteristics - Current Consumption, On Resistance, and Output Pulldown Resistance | | | | | | | | |

6.6 BUCK1, BUCK2, BUCK3, and BUCK4 Regulators (continued)

Over operating free-air temperature range (unless otherwise noted). Voltage level refers to the AGNDx ground of the device.

| POS | PARAMETER | | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|---|---------------------------|--|---|-----|---------|-----|------------------|
| 3.17 | I_{off} | Shutdown current, BUCKx disabled | | | 1 | | μ A |
| 3.19 | $R_{DS(ON) HS FET}$ | On-resistance, high-side FET | $I_{OUT_Bx} = 1$ A | | 26 | 65 | m Ω |
| 3.20 | $R_{DS(ON) LS FET}$ | On-resistance, low-side FET | $I_{OUT_Bx} = 1$ A | | 16 | 35 | m Ω |
| 3.21 | R_{DIS_Bx} | Output pulldown discharge resistance | Regulator disabled, per phase, BUCKx_PLDN = 1, between SW_Bx and PGND pins | 50 | 100 | 150 | Ω |
| 3.21b | $V_{TH_SC_RV_Bx}$ | Threshold voltage for Short Circuit and Residual Voltage Detection | | 140 | 150 | 160 | mV |
| 3.22 | R_{SW_SC} | Resistance threshold for Short circuit detection at the SW pin | | 3 | 5 | 25 | Ω |
| Electrical Characteristics - 8.8MHz Single-Phase Configuration | | | | | | | |
| 3.96 | V_{PVIN_Bx} | Input voltage range | | 3.0 | 3.3 | 5.5 | V |
| 3.97a | V_{VOUT_Bx} | Output voltage programmable range | $I_{OUT_Bx} \leq 3$ A | 0.9 | | 1.9 | V |
| 3.97b | | | $I_{OUT_Bx} \leq 4$ A | 0.9 | | 1.4 | |
| 3.98 | C_{IN_Bx} | Input filtering capacitance ^{(1) (2)} | | 3 | 22 | | μ F |
| 3.99a | $C_{OUT_Local_Bx}$ | Output capacitance, local with filter ⁽²⁾ | | 24 | 44 | 106 | μ F |
| 3.99b | $C_{OUT_Filter_Bx}$ | Output capacitance, after filter ⁽²⁾ | Capacitance after every filter | 20 | 33 | 80 | μ F |
| 3.99c | $C_{OUT_No_Filter_Bx}$ | Output capacitance, no filter ⁽²⁾ | | 24 | 44 | 106 | μ F |
| 3.100a | L_{Bx} | Power inductor | Inductance | 329 | 470 | 611 | nH |
| 3.100b | | | DCR | | 10 | | |
| 3.100c | L_{Ext} | Filter1 | Inductor, 100nH | 70 | 100 | 130 | nH |
| 3.100d | L_{Ext} | Filter2 | Ferrite, MPZ2012S300A | | | | |
| 3.101 | I_{Q_PWM} | PWM mode Quiescent current | $I_{OUT_Bx} = 0$ mA | | 30 | | mA |
| 3.167a | $V_{OUT_DC_Bx}$ | DC output voltage accuracy, includes voltage reference, DC load and line regulations and temperature | $V_{VOUT_Bx} < 1$ V, PWM mode | -10 | | 10 | mV |
| 3.167b | | | $V_{VOUT_Bx} \geq 1$ V, PWM mode | -1% | | 1% | |
| 3.103a | T_{LDSR} | Transient load step response, 1V8 ⁽⁷⁾ | C_{Local} as in Pos3.99a, no additional filter: $V_{VOUT_Bx} = 1.8$ V, $I_{OUT_Bx} = 0.5$ A to 1.8 A, $t_r = t_f = 4$ μ s | | 18 | | mV |
| 3.103c | | Transient load step response, 1V2 ⁽⁷⁾ | C_{Local} as in Pos3.99a, no additional filter: $V_{VOUT_Bx} = 1.2$ V, $I_{OUT_Bx} = 1.95$ A to 3 A, $t_r = t_f = 1$ μ s | | 20 | | |
| 3.103e | | Transient load step response, 1V0 ⁽⁷⁾ | C_{Local} as in Pos3.99a, no additional filter: $V_{VOUT_Bx} = 1.0$ V, $I_{OUT_Bx} = 0$ A to 2.5 A, $t_r = t_f = 4$ μ s | | 34 | | |
| 3.104 | T_{LNSR} | Transient line response, no filter | V_{PVIN_Bx} stepping between 3 V and 3.5 V or between 4.8 V and 5.3 V, $t_r = t_f = 10$ μ s, $I_{OUT_Bx} = I_{OUT(max)}$ | -25 | ± 5 | 25 | mV |
| 3.105 | V_{OUT_Ripple} | Ripple voltage ⁽⁷⁾ | PWM mode | | 4 | | mV _{PP} |
| Timing Requirements | | | | | | | |
| 3.108 | | Settling time after voltage scaling | From end of voltage ramp to V_{OUT} within 15 mV from $V_{OUT_DC_Bx}$ ⁽⁹⁾ | | | 105 | μ s |
| 3.109 | | Start-up delay | From enable to start of output voltage rise | 100 | 150 | 200 | μ s |

6.6 BUCK1, BUCK2, BUCK3, and BUCK4 Regulators (continued)

Over operating free-air temperature range (unless otherwise noted). Voltage level refers to the AGNDx ground of the device.

| POS | PARAMETER | | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|----------------------------------|---------------------------|---|---|-----|-----|------|---------------|
| 3.110 | $t_{\text{delay_OC}}$ | Over-current detection delay | Peak current limit triggering during every switching cycle | | | 7 | μs |
| 3.111 | $t_{\text{deglitch_OC}}$ | Over-current detection signal deglitch time | Digital deglitch time for detected signal. Time duration to filter out short positive and negative pulses | 19 | | 23 | μs |
| 3.112 | $t_{\text{latency_OC}}$ | Over-current signal latency time from detection | Total delay from over-current detection to interrupt or PFSM trigger | | | 30 | μs |
| Switching Characteristics | | | | | | | |
| 3.106c | f_{sw} | Switching frequency, PWM mode NVM programmable | 8.8 MHz setting, internal clock | 8 | 8.8 | 9.6 | MHz |
| 3.106f | | | 8.8 MHz setting, internal clock, spread spectrum enabled | 7.0 | 8.8 | 10.6 | |
| 3.106i | | | 8.8 MHz setting, synchronized to external clock | 7.0 | 8.8 | 10.6 | |

- (1) Input capacitors must be placed as close as possible to the device pins.
- (2) When DC voltage is applied to a ceramic capacitor, the effective capacitance is reduced due to DC bias effect. The table above therefore lists the minimum value as CAPACITANCE. In order to meet the minimum capacitance requirement, the nominal value of the capacitor may have to be scaled accordingly to take the drop of capacitance into account for a given dc voltage at the outputs of regulators.
- (3) The maximum output current can be limited by the forward current limit $I_{\text{LIM FWD}}$. The maximum output current is also limited by the junction temperature and maximum average current over lifetime. The power dissipation inside the die increases the junction temperature and limits the maximum current depending of the length of the current pulse, efficiency, board and ambient temperature.
- (4) Advance thermal design is required to avoid thermal shutdown.
- (5) SLEW_RATEx[2:0] register default comes from NVM memory, and can be re-programed by software. Output capacitance, forward and negative current limits and load current may limit the maximum and minimum slew rates.
- (6) A high slew-rate setting can generate over and undershoot during voltage change. See Application Section for more information.
- (7) Please refer to the applications section of the data sheet regarding the power delivery network (PDN) used for the transient load step and output ripple test conditions. All ripple specs are defined across POL capacitor in the described PDN.
- (8) Slew-rate is measured from 10% to 90% of the voltage ramp with voltage step ≥ 500 mV.
- (9) Voltage ramp is calculated using slew-rate from minimum column.

6.7 Reference Generator (REFOUT)

Over operating free-air temperature range (unless otherwise noted). Voltage level refers to the AGNDx ground of the device.

| POS | PARAMETER | | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|-----------------------------------|----------------------|-----------------------------------|---|------|-----|------|---------------|
| Electrical Characteristics | | | | | | | |
| 4.1 | | Max capacitance for REFOUT signal | Capacitance between REFOUT signal and ground | | | 100 | pF |
| 4.2 | | Output voltage | Measured at the REFOUT signal | 1.17 | 1.2 | 1.23 | V |
| Timing Requirements | | | | | | | |
| 4.4 | $t_{\text{SU_REF}}$ | Start-up time | From REFOUT_EN=1 to the time REFOUT voltage settles | | 30 | | μs |

6.8 Monitoring Functions

Over operating free-air temperature range (unless otherwise noted). Voltage level refers to the AGNDx ground of the device.

| POS | PARAMETER | | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|--|-----------|--|-----------------|-----|-----|-----|------|
| Electrical Characteristics: BUCK REGULATORS OUTPUT, VMONx INPUT | | | | | | | |

6.8 Monitoring Functions (continued)

Over operating free-air temperature range (unless otherwise noted). Voltage level refers to the AGNDx ground of the device.

| POS | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT | | |
|------|--|--|--|-----|-----|-------|-------|-------|
| 5.1a | V _{BUCK_OV_TH} , V _{VMON_OV_TH} | Overvoltage monitoring for buck output and VMONx pin input, programable threshold accuracy, V _{OUT_Bx} / V _{VMONx} > 1 V ⁽¹⁾ | BUCKn_OV_THR / VMONn_OV_THR = 0x0, VMONn_RANGE_SEL = 0 | | | 2% | 3% | 4% |
| 5.1b | | | BUCKn_OV_THR / VMONn_OV_THR = 0x1, VMONn_RANGE_SEL = 0 | | | 2.5% | 3.5% | 4.5% |
| 5.1c | | | BUCKn_OV_THR / VMONn_OV_THR = 0x2, VMONn_RANGE_SEL = 0 | | | 3% | 4% | 5% |
| 5.1d | | | BUCKn_OV_THR / VMONn_OV_THR = 0x3, VMONn_RANGE_SEL = 0 | | | 4% | 5% | 6% |
| 5.1e | | | BUCKn_OV_THR / VMONn_OV_THR = 0x4, VMONn_RANGE_SEL = 0 | | | 5% | 6% | 7% |
| 5.1f | | | BUCKn_OV_THR / VMONn_OV_THR = 0x5, VMONn_RANGE_SEL = 0 | | | 6% | 7% | 8% |
| 5.1g | | | BUCKn_OV_THR / VMONn_OV_THR = 0x6, VMONn_RANGE_SEL = 0 | | | 7% | 8% | 9% |
| 5.1h | | | BUCKn_OV_THR / VMONn_OV_THR = 0x7, VMONn_RANGE_SEL = 0 | | | 9% | 10% | 11% |
| 5.2a | V _{BUCK_OV_TH_mv} , V _{VMON_OV_TH_mv} | Overvoltage monitoring for buck output and VMONx pin input, programable threshold accuracy, V _{OUT_Bx} / V _{VMONx} ≤ 1 V ⁽¹⁾ | BUCKn_OV_THR / VMONn_OV_THR = 0x0, VMONn_RANGE_SEL = 0 | | | 20 | 30 | 40 |
| 5.2b | | | BUCKn_OV_THR / VMONn_OV_THR = 0x1, VMONn_RANGE_SEL = 0 | | | 25 | 35 | 45 |
| 5.2c | | | BUCKn_OV_THR / VMONn_OV_THR = 0x2, VMONn_RANGE_SEL = 0 | | | 30 | 40 | 50 |
| 5.2d | | | BUCKn_OV_THR / VMONn_OV_THR = 0x3, VMONn_RANGE_SEL = 0 | | | 40 | 50 | 60 |
| 5.2e | | | BUCKn_OV_THR / VMONn_OV_THR = 0x4, VMONn_RANGE_SEL = 0 | | | 50 | 60 | 70 |
| 5.2f | | | BUCKn_OV_THR / VMONn_OV_THR = 0x5, VMONn_RANGE_SEL = 0 | | | 60 | 70 | 80 |
| 5.2g | | | BUCKn_OV_THR / VMONn_OV_THR = 0x6, VMONn_RANGE_SEL = 0 | | | 70 | 80 | 90 |
| 5.2h | | | BUCKn_OV_THR / VMONn_OV_THR = 0x7, VMONn_RANGE_SEL = 0 | | | 90 | 100 | 110 |
| 5.3a | V _{BUCK_UV_TH} , V _{VMON_UV_TH} | Undervoltage monitoring for buck output and VMONx pin input, programable threshold accuracy, V _{OUT_Bx} / V _{VMONx} > 1 V ⁽¹⁾ | BUCKn_UV_THR / VMONn_UV_THR = 0x0, VMONn_RANGE_SEL = 0 | | | -4% | -3% | -2% |
| 5.3b | | | BUCKn_UV_THR / VMONn_UV_THR = 0x1, VMONn_RANGE_SEL = 0 | | | -4.5% | -3.5% | -2.5% |
| 5.3c | | | BUCKn_UV_THR / VMONn_UV_THR = 0x2, VMONn_RANGE_SEL = 0 | | | -5% | -4% | -3% |
| 5.3d | | | BUCKn_UV_THR / VMONn_UV_THR = 0x3, VMONn_RANGE_SEL = 0 | | | -6% | -5% | -4% |
| 5.3e | | | BUCKn_UV_THR / VMONn_UV_THR = 0x4, VMONn_RANGE_SEL = 0 | | | -7% | -6% | -5% |
| 5.3f | | | BUCKn_UV_THR / VMONn_UV_THR = 0x5, VMONn_RANGE_SEL = 0 | | | -8% | -7% | -6% |
| 5.3g | | | BUCKn_UV_THR / VMONn_UV_THR = 0x6, VMONn_RANGE_SEL = 0 | | | -9% | -8% | -7% |
| 5.3h | | | BUCKn_UV_THR / VMONn_UV_THR = 0x7, VMONn_RANGE_SEL = 0 | | | -11% | -10% | -9% |

6.8 Monitoring Functions (continued)

Over operating free-air temperature range (unless otherwise noted). Voltage level refers to the AGNDx ground of the device.

| POS | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|---|--|--|------|------|------|------|
| 5.4a | $V_{BUCK_UV_TH_mv}$, $V_{VMON_UV_TH_mv}$ Undervoltage monitoring for buck output and VMONx pin input, programmable threshold accuracy, $V_{OUT_BX} / V_{VMONx} \leq 1 V^{(1)}$ | BUCKn_UV_THR / VMONn_UV_THR = 0x0, VMONn_RANGE_SEL = 0 | -40 | -30 | -20 | mV |
| 5.4b | | BUCKn_UV_THR / VMONn_UV_THR = 0x1, VMONn_RANGE_SEL = 0 | -45 | -35 | -25 | |
| 5.4c | | BUCKn_UV_THR / VMONn_UV_THR = 0x2, VMONn_RANGE_SEL = 0 | -50 | -40 | -30 | |
| 5.4d | | BUCKn_UV_THR / VMONn_UV_THR = 0x3, VMONn_RANGE_SEL = 0 | -60 | -50 | -40 | |
| 5.4e | | BUCKn_UV_THR / VMONn_UV_THR = 0x4, VMONn_RANGE_SEL = 0 | -70 | -60 | -50 | |
| 5.4f | | BUCKn_UV_THR / VMONn_UV_THR = 0x5, VMONn_RANGE_SEL = 0 | -80 | -70 | -60 | |
| 5.4g | | BUCKn_UV_THR / VMONn_UV_THR = 0x6, VMONn_RANGE_SEL = 0 | -90 | -80 | -70 | |
| 5.4h | | BUCKn_UV_THR / VMONn_UV_THR = 0x7, VMONn_RANGE_SEL = 0 | -110 | -100 | -90 | |
| 5.5a | $V_{VMON_OV_TH2}$ Overvoltage monitoring for VMONx pin input with extended range ⁽¹⁾ | VMONn_OV_THR = 0x0, VMONn_RANGE_SEL = 1 | 100 | 150 | 200 | mV |
| 5.5b | | VMONn_OV_THR = 0x1, VMONn_RANGE_SEL = 1 | 125 | 175 | 225 | |
| 5.5c | | VMONn_OV_THR = 0x2, VMONn_RANGE_SEL = 1 | 150 | 200 | 250 | |
| 5.5d | | VMONn_OV_THR = 0x3, VMONn_RANGE_SEL = 1 | 200 | 250 | 300 | |
| 5.5e | | VMONn_OV_THR = 0x4, VMONn_RANGE_SEL = 1 | 250 | 300 | 350 | |
| 5.5f | | VMONn_OV_THR = 0x5, VMONn_RANGE_SEL = 1 | 300 | 350 | 400 | |
| 5.5g | | VMONn_OV_THR = 0x6, VMONn_RANGE_SEL = 1 | 350 | 400 | 450 | |
| 5.5h | | VMONn_OV_THR = 0x7, VMONn_RANGE_SEL = 1 | 450 | 500 | 550 | |
| 5.6a | $V_{VMON_UV_TH2}$ Undervoltage monitoring for VMONx pin input with extended range ⁽¹⁾ | VMONn_UV_THR = 0x0, VMONn_RANGE_SEL = 1 | -200 | -150 | -100 | mV |
| 5.6b | | VMONn_UV_THR = 0x1, VMONn_RANGE_SEL = 1 | -225 | -175 | -125 | |
| 5.6c | | VMONn_UV_THR = 0x2, VMONn_RANGE_SEL = 1 | -250 | -200 | -150 | |
| 5.6d | | VMONn_UV_THR = 0x3, VMONn_RANGE_SEL = 1 | -300 | -250 | -200 | |
| 5.6e | | VMONn_UV_THR = 0x4, VMONn_RANGE_SEL = 1 | -350 | -300 | -250 | |
| 5.6f | | VMONn_UV_THR = 0x5, VMONn_RANGE_SEL = 1 | -400 | -350 | -300 | |
| 5.6g | | VMONn_UV_THR = 0x6, VMONn_RANGE_SEL = 1 | -450 | -400 | -350 | |
| 5.6h | | VMONn_UV_THR = 0x7, VMONn_RANGE_SEL = 1 | -550 | -500 | -450 | |
| 5.6i | $V_{TH_RV(VMON)}$ Threshold voltage for Residual Voltage Detection at VMONx pins | | 140 | 150 | 160 | mV |
| Electrical Characteristics: VCCA INPUT | | | | | | |
| 5.7a | $V_{CCA_OV_TH}$ Overvoltage monitoring for VCCA input, programmable threshold accuracy ⁽²⁾ | VCCA_OV_THR = 0x0 | 2% | 3% | 4% | |
| 5.7b | | VCCA_OV_THR = 0x1 | 2.5% | 3.5% | 4.5% | |
| 5.7c | | VCCA_OV_THR = 0x2 | 3% | 4% | 5% | |
| 5.7d | | VCCA_OV_THR = 0x3 | 4% | 5% | 6% | |
| 5.7e | | VCCA_OV_THR = 0x4 | 5% | 6% | 7% | |
| 5.7f | | VCCA_OV_THR = 0x5 | 6% | 7% | 8% | |
| 5.7g | | VCCA_OV_THR = 0x6 | 7% | 8% | 9% | |
| 5.7h | | VCCA_OV_THR = 0x7 | 9% | 10% | 11% | |

6.8 Monitoring Functions (continued)

Over operating free-air temperature range (unless otherwise noted). Voltage level refers to the AGNDx ground of the device.

| POS | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT | | |
|----------------------------|----------------------------------|--|--|-----|-----|-------|-------|-------|
| 5.8a | VCCA _{UV_TH} | Undervoltage monitoring for VCCA input, programmable threshold accuracy ⁽²⁾ | VCCA_UV_THR = 0x0 | | | -4% | -3% | -2% |
| 5.8b | | | VCCA_UV_THR = 0x1 | | | -4.5% | -3.5% | -2.5% |
| 5.8c | | | VCCA_UV_THR = 0x2 | | | -5% | -4% | -3% |
| 5.8d | | | VCCA_UV_THR = 0x3 | | | -6% | -5% | -4% |
| 5.8e | | | VCCA_UV_THR = 0x4 | | | -7% | -6% | -5% |
| 5.8f | | | VCCA_UV_THR = 0x5 | | | -8% | -7% | -6% |
| 5.8g | | | VCCA_UV_THR = 0x6 | | | -9% | -8% | -7% |
| 5.8h | | | VCCA_UV_THR = 0x7 | | | -11% | -10% | -9% |
| Timing Requirements | | | | | | | | |
| 5.9a | t _{delay_OV_UV} | BUCK and VMON OV/UV detection delay | Detection delay with 5mV (V _{in} ≤ 1 V) or 0.5% (V _{in} > 1 V) over/underdrive | | | 8 | μs | |
| 5.9b | t _{delay_VCCA_OV_UV} | VCCA OV/UV detection delay | Detection delay with 30mV over/underdrive | | | 8 | μs | |
| 5.10a | t _{deglitch0_OV_UV} | VCCA, BUCK and VMON OV/UV signal deglitch time | VMON_DEGLITCH_SEL is 0.5 μs: Digital deglitch time for detected signal | | | 0.5 | 1 | |
| 5.10b | t _{deglitch1_OV_UV} | | VMON_DEGLITCH_SEL is 4 μs: Digital deglitch time for detected signal | | | 3.4 | 3.8 | 4.2 |
| 5.10c | t _{deglitch2_OV_UV} | | VMON_DEGLITCH_SEL is 20 μs: Digital deglitch time for detected signal | | | 18 | 20 | 22 |
| 5.11a | t _{latency0_OV_UV} | BUCK and VMON OV/UV signal latency time | VMON_DEGLITCH_SEL is 0.5 μs: Total delay from 5mV (V _{in} ≤ 1 V) or 0.5% (V _{in} > 1 V) over/underdrive to interrupt or PFSM trigger | | | 9 | μs | |
| 5.11b | t _{latency1_OV_UV} | | VMON_DEGLITCH_SEL is 4 μs: Total delay from 5mV (V _{in} ≤ 1 V) or 0.5% (V _{in} > 1 V) over/underdrive to interrupt or PFSM trigger | | | 13 | | |
| 5.11c | t _{latency2_OV_UV} | | VMON_DEGLITCH_SEL is 20 μs: Total delay from 5mV (V _{in} ≤ 1 V) or 0.5% (V _{in} > 1 V) over/underdrive to interrupt or PFSM trigger | | | 30 | | |
| 5.11d | t _{latency0_VCCA_OV_UV} | VCCA OV/UV signal latency time | VMON_DEGLITCH_SEL is 0.5 μs: Total delay from 30mV over/underdrive to interrupt or PFSM trigger | | | 9 | μs | |
| 5.11e | t _{latency1_VCCA_OV_UV} | | VMON_DEGLITCH_SEL is 4 μs: Total delay from 30mV over/underdrive to interrupt or PFSM trigger | | | 13 | | |
| 5.11f | t _{latency2_VCCA_OV_UV} | | VMON_DEGLITCH_SEL is 20 μs: Total delay from 30mV over/underdrive to interrupt or PFSM trigger | | | 30 | | |
| 5.12a | t _{deglitch_PGOOD_rise} | PGOOD signal additional deglitch time | Input signal transition from invalid to valid | | | 9.5 | 10.5 | |
| 5.12b | t _{deglitch_PGOOD_fall} | | Input signal transition from valid to invalid | | | 0 | μs | |

- (1) The default values of BUCKn_OV_THR, BUCKn_UV_THR, VMONn_OV_THR and VMONn_UV_THR registers come from the NVM memory, and can be re-programmed by software.
- (2) The default values of VCCA_OV_THR and VCCA_UV_THR registers come from the NVM memory, and can be re-programmed by software.

6.9 Clocks, Oscillators, and DPLL

Over operating free-air temperature range (unless otherwise noted).

| POS | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|--|-----------|--|-----|-----|-----|------|
| Switching Characteristics: 20-MHz and 128-kHz RC OSCILLATOR CLOCK | | | | | | |
| 6.2 | | 20 MHz RC Oscillator output frequency | 19 | 20 | 21 | MHz |
| 6.4 | | 128 kHz RC Oscillator output frequency | 121 | 128 | 135 | kHz |

6.9 Clocks, Oscillators, and DPLL (continued)

Over operating free-air temperature range (unless otherwise noted).

| POS | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|---|--|---|------|-------|-----|------|
| Switching Characteristics: DPLL, SYNCCLKIN, and SYNCCLKOUT | | | | | | |
| 6.6a | External input clock nominal frequency | EXT_CLK_FREQ = 0x0 | | 1.1 | | MHz |
| 6.6b | | EXT_CLK_FREQ = 0x1 | | 2.2 | | |
| 6.6c | | EXT_CLK_FREQ = 0x2 | | 4.4 | | |
| 6.6d | | EXT_CLK_FREQ = 0x3 | | 8.8 | | |
| 6.7a | External input clock required accuracy from nominal frequency ⁽¹⁾ | SS_DEPTH = 0x0 (Spread-spectrum disabled) | -18% | | 18% | |
| 6.7b | | SS_DEPTH = 0x1 | -10% | | 10% | |
| 6.7c | | SS_DEPTH = 0x2 | -8% | | 8% | |
| 6.8a | Logic low time for SYNCCLKIN clock | | 40 | | | ns |
| 6.8b | Logic high time for SYNCCLKIN clock | | 40 | | | ns |
| 6.9a | External clock detection delay for missing clock detection | | | | 1.8 | μs |
| 6.10 | Clock change delay (internal to external) | Delay from valid clock detection to use of external clock | | 600 | | μs |
| 6.11a | SYNCCLKOUT clock nominal frequency | SYNCCLKOUT_FREQ_SEL = 0x1 | | 1.1 | | MHz |
| 6.11b | | SYNCCLKOUT_FREQ_SEL = 0x2 | | 2.2 | | |
| 6.11c | | SYNCCLKOUT_FREQ_SEL = 0x3 | | 4.4 | | |
| 6.12 | SYNCCLKOUT duty-cycle | Cycle-to-cycle | 40% | 50% | 60% | |
| 6.13 | SYNCCLKOUT output buffer external load | T _J = 25°C | 5 | 35 | 50 | pF |
| 6.15a | Spread spectrum variation from nominal frequency | SS_DEPTH = 0x1 | | ±6.3% | | |
| 6.15b | | SS_DEPTH = 0x2 | | ±8.4% | | |
| 6.17a | t _{latency_CLKfail} Clock Monitor Failure signal latency from detection | Failure on 20 MHz system clock | | | 10 | μs |
| 6.17b | | Failure on 128 kHz monitoring clock | | | 40 | μs |
| 6.18 | t _{latency_CLKdrift} Clock Monitor Drift signal latency from detection | | | | 115 | μs |
| 6.19 | f _{sysclk} Internal system clock | | 19 | 20 | 21 | MHz |
| 6.20 | CLKdrift_TH | Threshold for internal system clock frequency drift detection | | -20% | 20% | |
| 6.21 | CLKfail_TH | Threshold for internal system clock stuck at high or stuck at low detection | | | 10 | MHz |

 (1) With 8.8 MHz frequency setting the maximum switching clock frequency may be limited by minimum high-side FET T_{on} time

6.10 Thermal Monitoring and Shutdown

Over operating free-air temperature range (unless otherwise noted).

| POS | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT | |
|-----------------------------------|---------------------------|---|-------------------|-----|-----|------|----|
| Electrical Characteristics | | | | | | | |
| 7.1a | T _{WARN_0} | Thermal warning threshold (no hysteresis) | TWARN_LEVEL = 0 | 120 | 130 | 140 | °C |
| 7.1b | T _{WARN_1} | | TWARN_LEVEL = 1 | 130 | 140 | 150 | |
| 7.2a | T _{SD_orderly_0} | Thermal orderly shutdown rising threshold | TSD_ORD_LEVEL = 0 | 130 | 140 | 150 | °C |
| 7.2b | T _{SD_orderly_1} | | TSD_ORD_LEVEL = 1 | 135 | 145 | 155 | |
| 7.2c | | Thermal orderly shutdown hysteresis | TSD_ORD_LEVEL = 0 | | 10 | | °C |
| 7.2d | | | TSD_ORD_LEVEL = 1 | | 5 | | |
| 7.3a | T _{SD_imm} | Thermal immediate shutdown rising threshold | | 140 | 150 | 160 | °C |
| 7.3b | | Thermal immediate shutdown hysteresis | | | 5 | | °C |
| Timing Requirements | | | | | | | |

Over operating free-air temperature range (unless otherwise noted).

| POS | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|-----|--------------------------|-----------------------------------|-----|-----|-----|------|
| 7.4 | t _{latency_TSD} | TSD signal latency from detection | | | 425 | μs |

6.11 System Control Thresholds

Over operating free-air temperature range (unless otherwise noted). Voltage level refers to the AGNDx ground of the device.

| POS | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT | | |
|-----------------------------------|-------------------------------|---|-----------------------------------|-----|-----|------|-----|---|
| Electrical Characteristics | | | | | | | | |
| 8.1a | V _{POR_Falling} | VCCA UVLO/POR falling threshold | Measured on VCCA pin, trimmed | | 2.7 | 2.75 | 2.8 | V |
| 8.1b | V _{POR_Rising} | VCCA UVLO/POR rising threshold | Measured on VCCA pin, untrimmed | | 2.7 | | 3 | V |
| 8.1c | V _{POR_Hyst} | VCCA UVLO/POR hysteresis | | 100 | | | mV | |
| 8.2a | V _{OVP_Rising} | VCCA OVP rising threshold | Measured on VCCA pin, trimmed | | 5.6 | 5.7 | 5.8 | V |
| 8.2b | V _{OVP_Hyst} | VCCA OVP hysteresis | | 50 | | | mV | |
| Timing Requirements | | | | | | | | |
| 8.3 | t _{latency_VCCAOP} | VCCA_OVP signal latency from detection | | | | 15 | μs | |
| 8.6 | t _{latency_VCCAUVLO} | VCCA_UVLO signal latency from detection | | | | 10 | μs | |
| 8.8 | t _{latency_VINT} | LDOVINT OVP and UVLO signal latency from detection | With 25-mV over/underdrive | | | 12 | μs | |
| 8.12a | t _{INIT_REF_CLK_LDO} | Device initialization time to start up references, LDOVINT and EEPROM LDO | From NO SUPPLY state | | | 2.2 | ms | |
| 8.12b | t _{INIT_LDO} | Device initialization time to start up references and EEPROM LDO | From LP_STANDBY state | | | 0.95 | ms | |
| 8.13 | t _{INIT_NVM_ANALOG} | Device initialization time to load default values for NVM programmable registers and use trimmed values | | | | 0.6 | ms | |
| 8.11 | t _{LBISTrun} | Run time for LBIST | FAST_BIST=0 | | | 1.4 | ms | |
| 8.10 | t _{ABISTrun} | Run time for ABIST | Stand-alone device ⁽¹⁾ | | | 0.25 | ms | |

(1) SPMI BIST is part of ABIST and can increase the time for devices in multi-PMIC platforms.

6.12 Current Consumption

Over operating free-air temperature range (unless otherwise noted).

| POS | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT | |
|-----------------------------------|-----------------------------|--|---|-----|-----|------|----|
| Electrical Characteristics | | | | | | | |
| 9.1a | I _{LP_STANDBY_3V3} | Low Power Standby current consumption ⁽¹⁾ | From VCCA and PVIN_Bx pins. VCCA = PVIN_Bx = 3.3 V. VIO = 0V. T _J = 25°C | | | 3000 | μA |
| 9.1b | I _{LP_STANDBY_5V0} | | From VCCA and PVIN_Bx pins. VCCA = PVIN_Bx = 5.0 V. VIO = 0V. T _J = 25°C | | | 3000 | |
| 9.2a | I _{STANDBY_3V3} | Standby current consumption | From VCCA and PVIN_Bx pins. VCCA OV/UV monitoring disabled. VCCA = PVIN_Bx = 3.3 V. VIO = 0V. T _J = 25°C | | | 34 | μA |
| 9.2b | I _{STANDBY_5V0} | | From VCCA and PVIN_Bx pins. VCCA OV/UV monitoring disabled. VCCA = PVIN_Bx = 5.0 V. VIO = 0V. T _J = 25°C | | | 36 | |

Over operating free-air temperature range (unless otherwise noted).

| POS | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|------|-----------------------------|---|-----|-----|-----|------|
| 9.4e | I _{ACTIVE_3V3_8M8} | From VCCA and PVIN_Bx pins. VCCA = PVIN_Bx = VIO = 3.3 V. Fsw = 8.8 MHz. All buck regulators are enabled in forced PWM mode with no load. All buck and VCCA OV/UV monitoring enabled. T _J = 25°C | | 119 | | mA |
| 9.4f | I _{ACTIVE_5V0_8M8} | From VCCA and PVIN_Bx pins. VCCA = PVIN_Bx = 5.0 V. VIO = 3.3V. Fsw = 8.8 MHz. All buck regulators are enabled in forced PWM mode with no load. All buck and VCCA OV/UV monitoring enabled. T _J = 25°C | | 175 | | |

(1) LP_STANDBY mode is not preferred to use because of high current consumption.

6.13 Digital Input Signal Parameters

Over operating free-air temperature range, VIO refers to the VIO pin, VCCA refers to the VCCA pin (unless otherwise noted).

| POS | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT | |
|---|---------------------------|--------------------------------------|-----------------------|-----|------|------|----|
| Electrical Characteristics: All Digital Input Signals in GPIOx, SCL_I2C1, SDA_I2C1, SCK_SPI, SDI_SPI | | | | | | | |
| 10.1 | V _{IL} | Low-level input voltage | | | 0.54 | V | |
| 10.2 | V _{IH} | High-level input voltage | 1.26 | | | V | |
| 10.3 | | Hysteresis | 150 | | | mV | |
| Timing Requirements: ENABLE | | | | | | | |
| 10.4 | t _{degl_ENABLE} | ENABLE signal deglitch time | GPIOx_DEGLITCH_EN = 1 | 6 | 8 | 10 | μs |
| 10.5 | t _{degl_nSLEEPx} | nSLEEPx signal deglitch time | | 6 | 8 | 10 | μs |
| Timing Requirements: GPIOx | | | | | | | |
| 10.6 | t _{degl_ESMx} | nERR signal deglitch time | | 13 | 15 | 17 | μs |
| 10.7 | t _{WD_pulse} | TRIG_WDOG input signal deglitch time | | 24 | 30 | 36 | μs |
| 10.8 | t _{degl_GPIOx} | GPIOx, WKUPx signal deglitch time | GPIOx_DEGLITCH_EN = 1 | 6 | 8 | 10 | μs |
| 10.9 | t _{no_degl_PW} | Minimum pulse width for GPIOx, WKUPx | GPIOx_DEGLITCH_EN = 0 | 200 | | | ns |

6.14 Digital Output Signal Parameters

Over operating free-air temperature range, VIO refers to the VIO pin, VLDO refers to the LDO_VOUT pin, VCCA refers to the VCCA pin (unless otherwise noted).

| POS | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|--|-----------------------|--|-------------------------|-----------|------|------|
| Electrical Characteristics: nINT, SDA_I2Cx | | | | | | |
| 11.1 | V _{OL_20mA} | Low-level output voltage | I _{OL} = 20 mA | 0 | 0.4 | V |
| Electrical Characteristics: PGOOD(GPIO1), nRSTOUT_SOC(GPIO1), SDO_SPI, nRSTOUT_SOC(GPIO5), SYNCCLKOUT(GPIO5), nRSTOUT(GPIO10) and GPIO Output Signals through GPIO1, GPIO3, GPIO5 and GPIO10 pins | | | | | | |
| 11.2 | V _{OL_20mA} | Low-level output voltage, push-pull and open-drain | I _{OL} = 20 mA | 0 | 0.4 | V |
| 11.3 | V _{OH(VIO)} | High-level output voltage, push-pull | I _{OH} = 3 mA | VIO – 0.4 | VIO | V |
| Electrical Characteristics: PGOOD(GPIO6), SYNCCLKOUT(GPIO6) and GPIO Output Signals through GPIO2, GPIO6 pins | | | | | | |
| 11.4 | V _{OL_3mA} | Low-level output voltage, push-pull and open-drain | I _{OL} = 3 mA | 0 | 0.4 | V |
| 11.5 | V _{OH(VIO)} | High-level output voltage, push-pull | I _{OH} = 3 mA | VIO – 0.4 | VIO | V |
| Electrical Characteristics: SCLK_SPMI(GPIO8), SDATA_SPMI(GPIO9), PGOOD(GPIO9) and GPIO Output Signals through GPIO8 and GPIO9 pins | | | | | | |
| 11.6 | V _{OL_20mA} | Low-level output voltage, push-pull and open-drain | I _{OL} = 20 mA | 0 | 0.4 | V |
| 11.7 | V _{OH(VINT)} | High-level output voltage, push-pull | I _{OH} = 3 mA | 1.4 | VLDO | V |
| Electrical Characteristics: GPIO Output Signals through GPIO4 and GPIO7 pins | | | | | | |
| 11.8 | V _{OL_3mA} | Low-level output voltage, push-pull | I _{OL} = 3 mA | 0 | 0.4 | V |

6.14 Digital Output Signal Parameters (continued)

Over operating free-air temperature range, VIO refers to the VIO pin, VLDO refers to the LDO_VOUT pin, VCCA refers to the VCCA pin (unless otherwise noted).

| POS | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|----------------------------|----------------------------|--------------------------------------|--|-----|------|---------|
| 11.9 | V _{OH(VINT)} | High-level output voltage, push-pull | I _{OH} = 3 mA | | VLDO | V |
| 11.9b | V _{OH(VCCA)} | High-level output voltage, push-pull | EN_DRV pin, 10 kΩ internal pull-up, V _{VIO} = V _{VCCA} , no load | | VCCA | V |
| Timing Requirements | | | | | | |
| 11.10 | t _{gate_readback} | Gating time for readback monitor | Signal level change or GPIO selection (GPIO _n _SEL) | | 8.3 | 10.3 μs |

6.15 I/O Pullup and Pulldown Resistance

Over operating free-air temperature range, VIO refers to the VIO_IN pin, VCCA refers to the VCCA pin (unless otherwise noted).

| POS | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|-----------------------------------|-----------------------------|--------------------------------|--|-----|-----|------------|
| Electrical Characteristics | | | | | | |
| 12.2 | R _{PU_GPIO} | IO signals pullup resistance | GPIO4, GPIO7, GPIO8 and GPIO9 pins with internal pullup to VOUT_LDO | | 280 | 400 520 kΩ |
| 12.3 | R _{PU_GPIO} | IO signals pullup resistance | GPIO1, GPIO2, GPIO3, GPIO5, GPIO6 and GPIO10 pins with internal pullup to VIO | | 280 | 400 520 kΩ |
| 12.4 | R _{PD_GPIO} | IO signals pulldown resistance | GPIO1 - 10 pins with internal pulldown to ground | | 280 | 400 520 kΩ |
| 12.5 | R _{PU_NRSTOUT} | nRSTOUT pullup resistance | Internal pullup to VIO supply when output driven high in push-pull configuration | | 8 | 10 12 kΩ |
| 12.6 | R _{PU_NRSTOUT_SOC} | nRSTOUT_SOC pullup resistance | Internal pullup to VIO supply when output driven high in push-pull configuration | | 8 | 10 12 kΩ |
| 12.7 | R _{PU_EN_DRV} | EN_DRV pullup resistance | Internal pullup to VCCA supply when output driven high | | 8 | 10 12 kΩ |

6.16 I²C Interface

Over operating free-air temperature range (unless otherwise noted). Device supports standard mode (100 kHz), fast mode (400 kHz), and fast mode+ (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.

| POS | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|-----------------------------------|-------------------|---------------------------------|--|-----|------|------|
| Electrical Characteristics | | | | | | |
| 13.1 | C _B | Capacitive load for SDA and SCL | | | 400 | pF |
| Timing Requirements | | | | | | |
| 13.2a | f _{SCL} | Serial clock frequency | Standard mode | | 100 | kHz |
| 13.2b | | | Fast mode | | 400 | |
| 13.2c | | | Fast mode+ | | 1 | MHz |
| 13.2d | | | High-speed mode, C _b = 100 pF | | 3.4 | |
| 13.2e | | | High-speed mode, C _b = 400 pF | | 1.7 | |
| 13.3a | t _{LOW} | SCL low time | Standard mode | | 4.7 | μs |
| 13.3b | | | Fast mode | | 1.3 | |
| 13.3c | | | Fast mode+ | | 0.5 | |
| 13.3d | | | High-speed mode, C _b = 100 pF | | 160 | ns |
| 13.3e | | | High-speed mode, C _b = 400 pF | | 320 | |
| 13.4a | t _{HIGH} | SCL high time | Standard mode | | 4 | μs |
| 13.4b | | | Fast mode | | 0.6 | |
| 13.4c | | | Fast mode+ | | 0.26 | |
| 13.4d | | | High-speed mode, C _b = 100 pF | | 60 | ns |
| 13.4e | | | High-speed mode, C _b = 400 pF | | 120 | |

6.16 I²C Interface (continued)

Over operating free-air temperature range (unless otherwise noted). Device supports standard mode (100 kHz), fast mode (400 kHz), and fast mode+ (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.

| POS | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT | |
|--------|----------------------|---|--|-----|------|------|------|
| 13.5a | t _{SU, DAT} | Data setup time | Standard mode | | 250 | ns | |
| 13.5b | | | Fast mode | | 100 | | |
| 13.5c | | | Fast mode+ | | 50 | | |
| 13.5d | | | High-speed mode | | 10 | | |
| 13.6a | t _{HD, DAT} | Data hold time | Standard mode | | 10 | ns | |
| 13.6b | | | Fast mode | | 10 | | 3450 |
| 13.6c | | | Fast mode+ | | 10 | | 900 |
| 13.6d | | | High-speed mode, C _b = 100 pF | | 10 | 70 | ns |
| 13.6e | | | High-speed mode, C _b = 400 pF | | 10 | 150 | |
| 13.7a | t _{SU, STA} | Setup time for a start or a REPEATED START condition | Standard mode | | 4.7 | μs | |
| 13.7b | | | Fast mode | | 0.6 | | |
| 13.7c | | | Fast mode+ | | 0.26 | | |
| 13.7d | | | High-speed mode | | 160 | ns | |
| 13.8a | t _{HD, STA} | Hold time for a start or a REPEATED START condition | Standard mode | | 4 | μs | |
| 13.8b | | | Fast mode | | 0.6 | | |
| 13.8c | | | Fast mode+ | | 0.26 | | |
| 13.8d | | | High-speed mode | | 160 | ns | |
| 13.9a | t _{BUF} | Bus free time between a STOP and START condition | Standard mode | | 4.7 | μs | |
| 13.9b | | | Fast mode | | 1.3 | | |
| 13.9c | | | Fast mode+ | | 0.5 | | |
| 13.10a | t _{SU, STO} | Setup time for a STOP condition | Standard mode | | 4 | μs | |
| 13.10b | | | Fast mode | | 0.6 | | |
| 13.10c | | | Fast mode+ | | 0.26 | | |
| 13.10d | | | High-speed mode | | 160 | ns | |
| 13.11a | t _{rDA} | Rise time of SDA signal | Standard mode | | 1000 | ns | |
| 13.11b | | | Fast mode | | 20 | | 300 |
| 13.11c | | | Fast mode+ | | | | 120 |
| 13.11d | | | High-speed mode, C _b = 100 pF | | 10 | | 80 |
| 13.11e | | | High-speed mode, C _b = 400 pF | | 20 | | 160 |
| 13.12a | t _{fDA} | Fall time of SDA signal | Standard mode | | 300 | ns | |
| 13.12b | | | Fast mode | | 6.5 | | 300 |
| 13.12c | | | Fast mode+ | | 6.5 | | 120 |
| 13.12d | | | High-speed mode, C _b = 100 pF | | 10 | | 80 |
| 13.12e | | | High-speed mode, C _b = 400 pF | | 13 | | 160 |
| 13.13a | t _{rCL} | Rise time of SCL signal | Standard mode | | 1000 | ns | |
| 13.13b | | | Fast mode | | 20 | | 300 |
| 13.13c | | | Fast mode+ | | | | 120 |
| 13.13d | | | High-speed mode, C _b = 100 pF | | 10 | | 40 |
| 13.13e | | | High-speed mode, C _b = 400 pF | | 20 | | 80 |
| 13.14a | t _{rCL1} | Rise time of SCL signal after a repeated start condition and after an acknowledge bit | High-speed mode, C _b = 100 pF | | 10 | ns | |
| 13.14b | | | High-speed mode, C _b = 400 pF | | 20 | | 160 |
| 13.15a | t _{rCL} | Fall time of SCL signal | Standard mode | | 300 | ns | |
| 13.15b | | | Fast mode | | 6.5 | | 300 |
| 13.15c | | | Fast mode+ | | 6.5 | | 120 |
| 13.15d | | | High-speed mode, C _b = 100 pF | | 10 | | 40 |
| 13.15e | | | High-speed mode, C _b = 400 pF | | 20 | | 80 |

6.16 I²C Interface (continued)

Over operating free-air temperature range (unless otherwise noted). Device supports standard mode (100 kHz), fast mode (400 kHz), and fast mode+ (1 MHz) when V_{I/O} is 3.3 V or 1.8 V, and high-speed mode (3.4 MHz) only when V_{I/O} is 1.8 V.

| POS | PARAMETER | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|--------|---|--|-----|-----|-----|------|
| 13.16a | t _{SP} Pulse width of spike suppressed (SCL and SDA spikes that are less than the indicated width are suppressed) | Standard mode, fast mode, and fast mode+ | | | 50 | ns |
| 13.16b | | High-speed mode, C _b = 400 pf | | | 10 | |

6.17 Serial Peripheral Interface (SPI)

These specifications are ensured by design, V_{I/O} = 1.8 V or 3.3 V (unless otherwise noted).

| POS | PARAMETERS | TEST CONDITIONS | MIN | NOM | MAX | UNIT |
|-----------------------------------|----------------------------|--|--------------------------|-----|-----|------|
| Electrical Characteristics | | | | | | |
| 15.1 | Capacitive load on pin SDO | | | | 30 | pF |
| Timing Requirements | | | | | | |
| 15.2 | 1 | Cycle time | | 200 | | ns |
| 15.3 | 2 | Enable lead time | | 150 | | ns |
| 15.4 | 3 | Enable lag time | | 150 | | ns |
| 15.5 | 4 | Clock low time | | 60 | | ns |
| 15.6 | 5 | Clock high time | | 60 | | ns |
| 15.7 | 6 | Data setup time | | 15 | | ns |
| 15.8 | 7 | Data hold time | | 15 | | ns |
| 15.9 | 8 | Output data valid after SCLK falling | | 4 | | ns |
| 15.10a | 9 | New output data valid after SCLK falling | V _{I/O} = 1.8 V | | 72 | ns |
| 15.10b | | | V _{I/O} = 3.3 V | | 110 | |
| 15.11 | 10 | Disable time | | | 30 | ns |
| 15.12 | 11 | CS inactive time | | 100 | | ns |

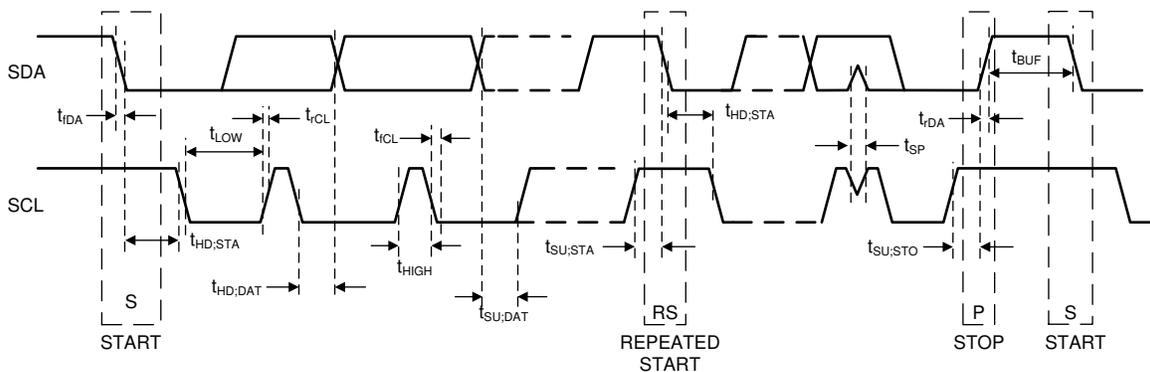


Figure 6-1. I²C Timing

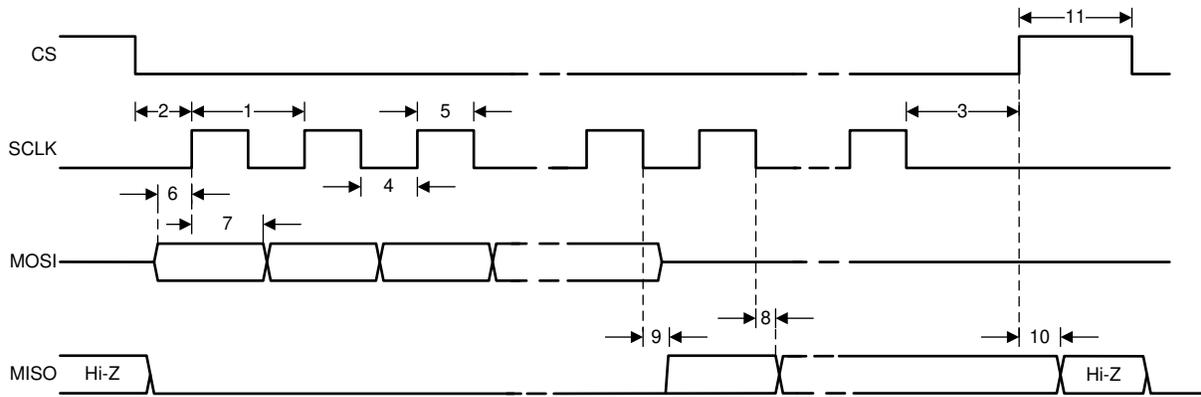


Figure 6-2. SPI Timing

6.18 Typical Characteristics

Unless otherwise specified: $T_A = 25^\circ\text{C}$, $V_{IN} = 3.3\text{ V}$, $V_{OUT} = 1\text{ V}$.

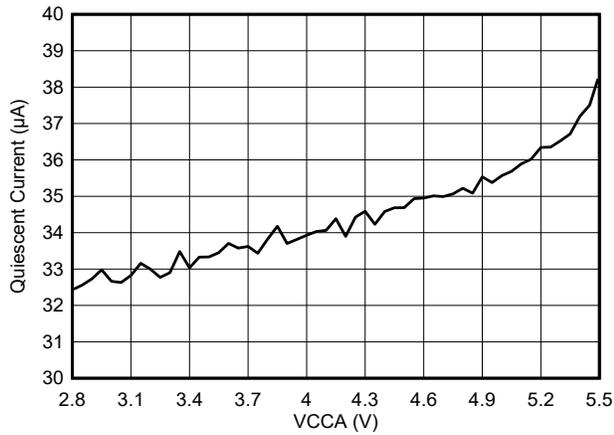


Figure 6-3. Quiescent Current Consumption vs Input Voltage

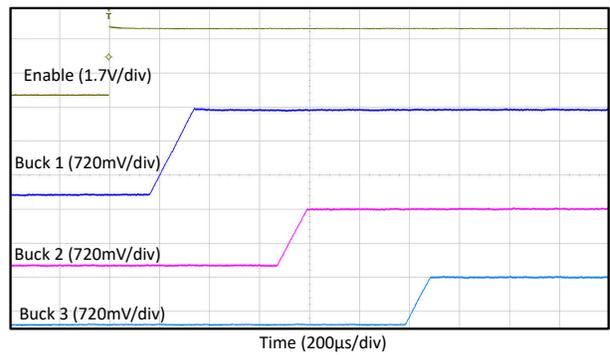


Figure 6-4. Startup, LP876242B0

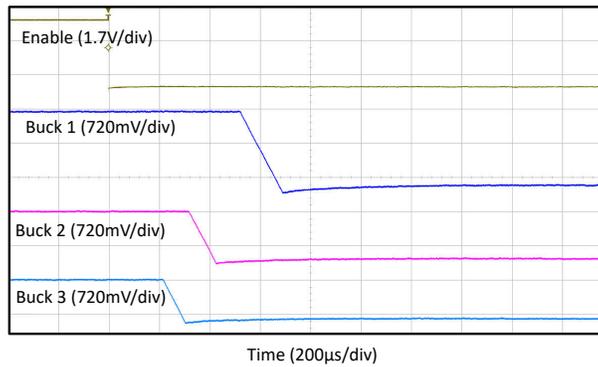


Figure 6-5. Shutdown, LP876242B0

7 Detailed Description

7.1 Overview

The LP876242-Q1 device is a power-management integrated circuit (PMIC), available in a 32-pin, 0.5-mm pitch, 5.5-mm × 5-mm QFN HotRod package. The device is designed for powering embedded systems or system on chip (SoC) in Automotive or Industrial applications. All buck converters have the capability 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, I²C or state transition. A DPLL enables the BUCK converters to synchronizing to an external clock input, with phase delays between the output rails.

Two I²C interface channels or one SPI channel can be used to configure the power rails and the power state of the LP876242-Q1 device. I²C channel 1 (I2C1) is the main channel with access to the registers that control the configurable power sequencer, the states and the outputs of power rails, and the device operating states. I²C channel 2 (I2C2), which is available through GPIO2 and GPIO3 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 LP876242-Q1 device includes an internal RC oscillator to sequence all resources during power up and power down. An internal LDO (LDOVINT) generates the supply for the entire digital circuitry of the device as soon as the external input supply is available through the VCCA input.

LP876242-Q1 device has ten GPIOs each with multiple functions and configurable features. All of the GPIOs, when configured as a general purpose output pin, can be included in the power-up and power-down sequence and used as enable signals for external resources. In addition, each GPIO can be configured as a wake-up input or a sleep mode trigger. The default configuration of the GPIO port comes from the NVM memory, and can be re-configured by software if the external connection permits.

The LP876242-Q1 device includes a Q&A watchdog to monitor software lockup, and a system error monitoring input (nERR_MCU) with fault injection option to monitor the lock-step signal of the attached MCU. The device 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 signal, allowing the processor to take action in response.

An SPMI interface is included in the LP876242-Q1 device to distribute power state information to at most five satellite PMICs, thus enabling synchronous power state transition across multiple PMICs in the application system. This feature allows the consolidation of IO control signals from up to six PMICs powering the system into one primary LP876242-Q1 PMIC.

7.2 Functional Block Diagram

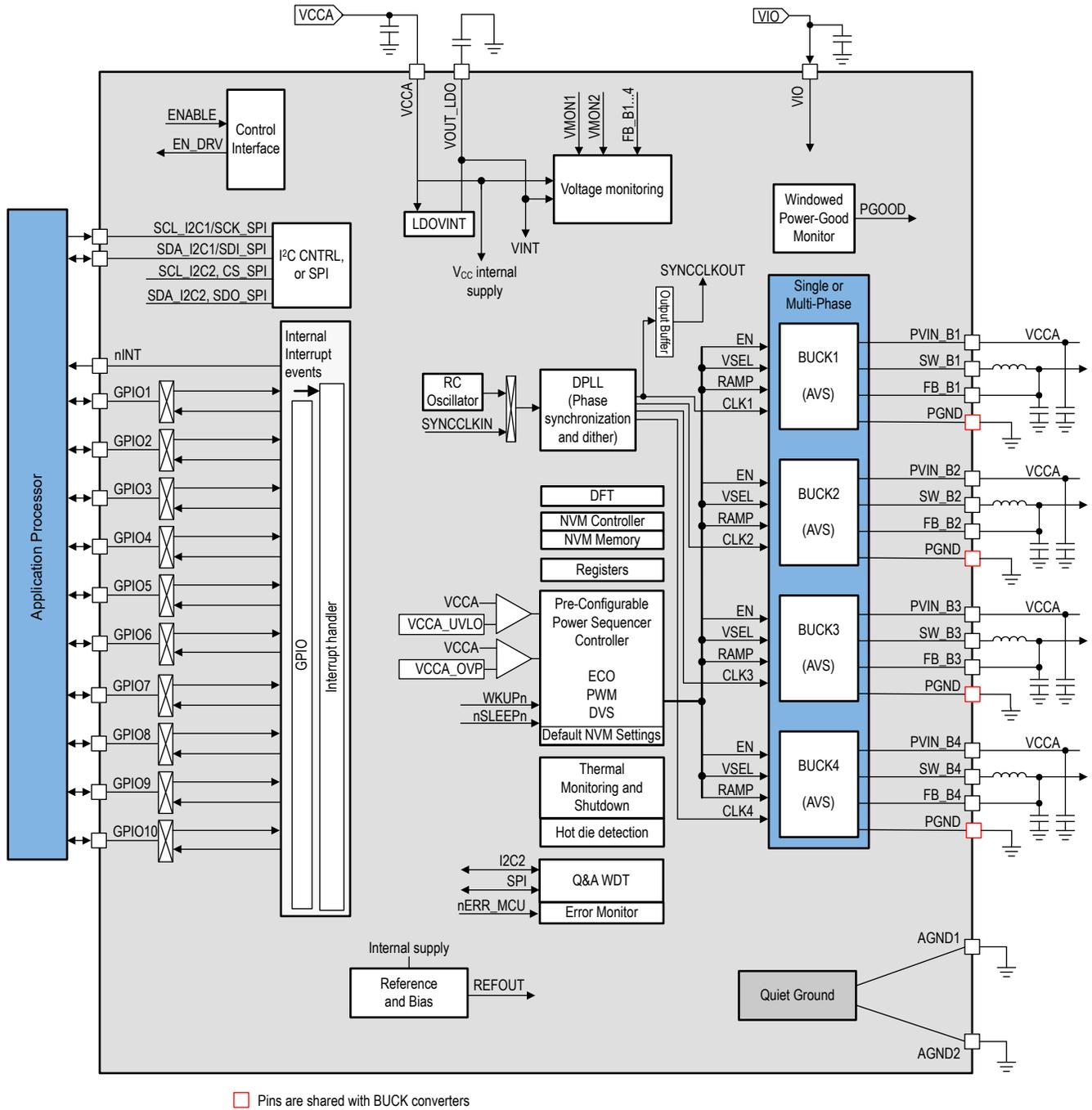


Figure 7-1. Functional Block Diagram

7.3 Feature Description

7.3.1 Input Voltage Monitor

The comparator module that monitors the voltage on the VCCA pins controls the power state machine of the LP876242-Q1 device. The [Figure 7-2](#) shows a block diagram of the VCCA input voltage monitoring. 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 LP876242-Q1 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 NO SUPPLY state and is completely shut down.

VCCA_OVP While the LP876242-Q1 device is in operation, if the voltage on VCCA pin rises above the VCCA_OVP threshold, the device clears the ENABLE_DRV bit and start the immediate shutdown sequence using pull-down resistors at the buck regulator outputs 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 range. Please refer to for additional detail on the operation of the VCCA OV/UV monitor function.

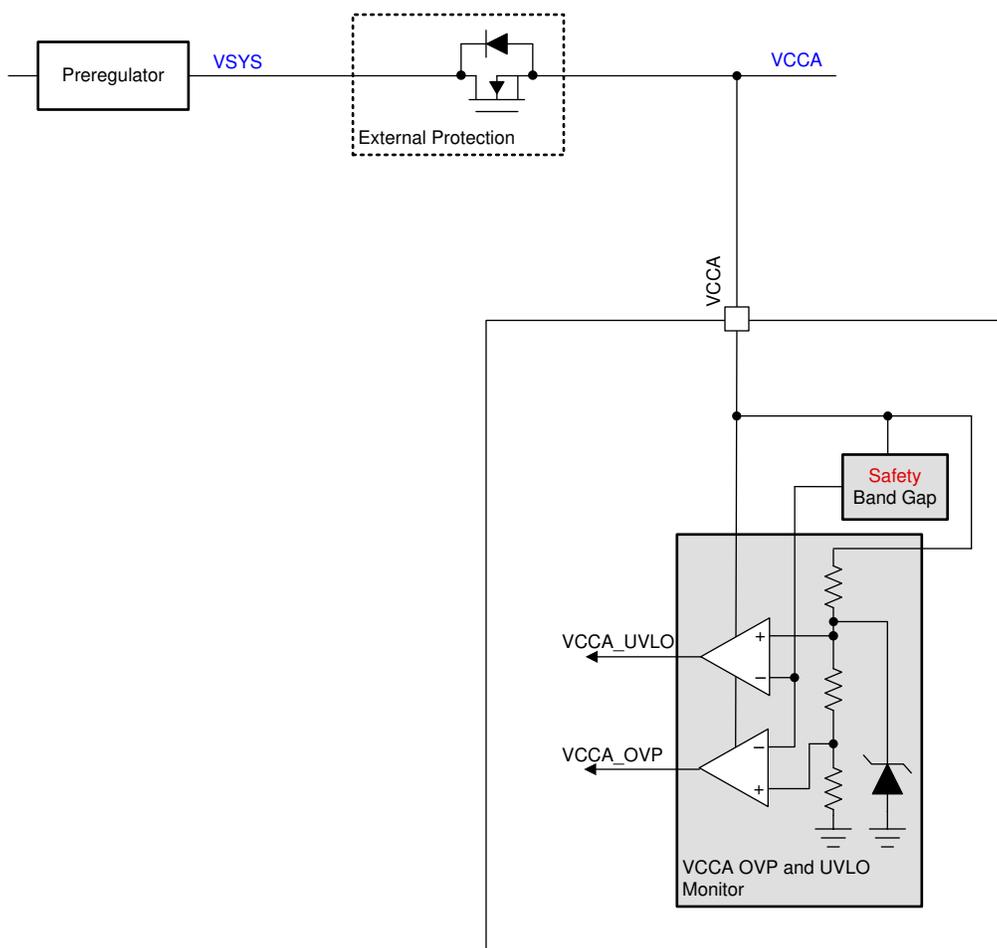


Figure 7-2. VCCA Monitor

7.3.2 Power Resources

The power resources provided by the LP876242-Q1 device include 4 inductor-based buck regulators. These supply resources provide the required power to the external processor cores, external components, and to

modules embedded in the LP876242-Q1 device. The supply of the bucks, the PVIN_Bx pins, must connect to the VCCA pin externally.

7.3.2.1 Buck Regulators

7.3.2.1.1 BUCK Regulator Overview

The LP876242-Q1 includes four synchronous buck converters. All of the buck converters support the following features:

- 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 monitors with configurable threshold
- Windowed voltage monitor for external supply when the buck converter is disabled
- Output Current Limit
- Short-to-Ground Detection on SW_Bx pins at start-up of the buck regulator

Figure 7-3 shows a block diagram of a single core.

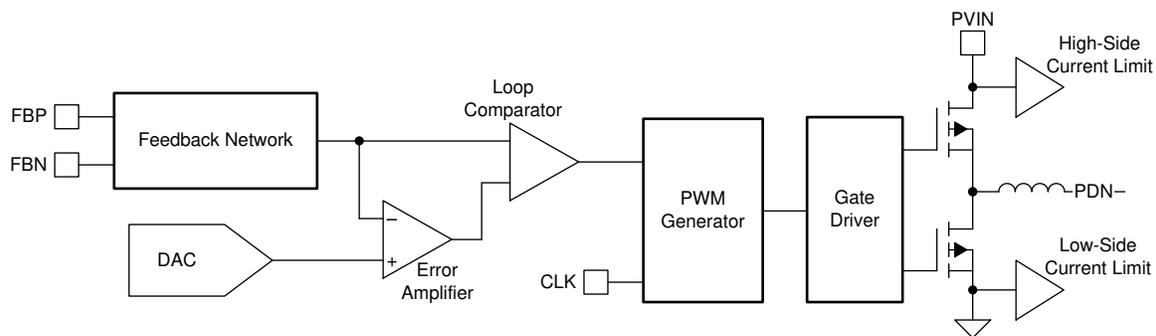


Figure 7-3. BUCK Core Block Diagram

7.3.2.1.2 Spread-Spectrum Mode

The LP876242-Q1 device supports spread-spectrum modulation of the switching clock signal used by 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.

The spread-spectrum modulation mode is pre-configured in NVM. Changing this modulation mode 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 $\pm 18\%$ 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.

7.3.2.1.3 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, that 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 BUCK regulators in the LP876242-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 regulator remains at the AVS/DVS voltage level instead of the default voltage from NVM until one of the following events occur:

- 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

Figure 7-4 shows the arbitration scheme for loading the output level of the BUCK regulator from the AVS register using the BUCKn_VSET control registers.

Figure 7-4. AVS/DVS Configuration Register Arbitration Diagram

The digital control block automatically updates the OV and UV threshold of the BUCK output voltage monitor 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 1.

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 1.

$$t_{PG_OV_UV_DELAY} = (dV / BUCKn_SLEW_RATE) + t_{settle_Bx} \quad (1)$$

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 regulator is enabled and the voltage is rising to the BUCKn_VSETx level. The duration of the mask starts from the time the BUCK regulator is enabled. The BUCK OV monitor output is masked for a fixed delay time of $t_{PG_OV_GATE}$, that is approximately 115 μ s – 128 μ s. The UV monitor output is masked for the time duration calculated by Equation 2. The 370- μ s additional delay time in the formula includes the start-up delay of the BUCK regulator, the fixed delay after the ramp, and the time for the BIST operation of the OV and UV monitors.

$$t_{PG_UV_GATE} = (BUCKn_VSET / BUCKn_SLEW_RATE) + 370 \mu s \quad (2)$$

Note

Because output capacitance, forward and negative current limits and load current of the BUCK regulator may affect the slew rate of the BUCK regulator output voltage, the delay time of $t_{PG_UV_GATE}$ may not be sufficient long for the slower slew rate setting when the target BUCK regulator output voltage is higher. Please refer to the PMIC User's Guide for detail information about the supported voltage level and slew rate setting combinations of a particular orderable part number.

Figure 7-5 and Figure 7-6 are timing diagrams illustrating the voltage change for AVS and DVS enabled BUCK regulators and the corresponding OV and UV monitor threshold changes.

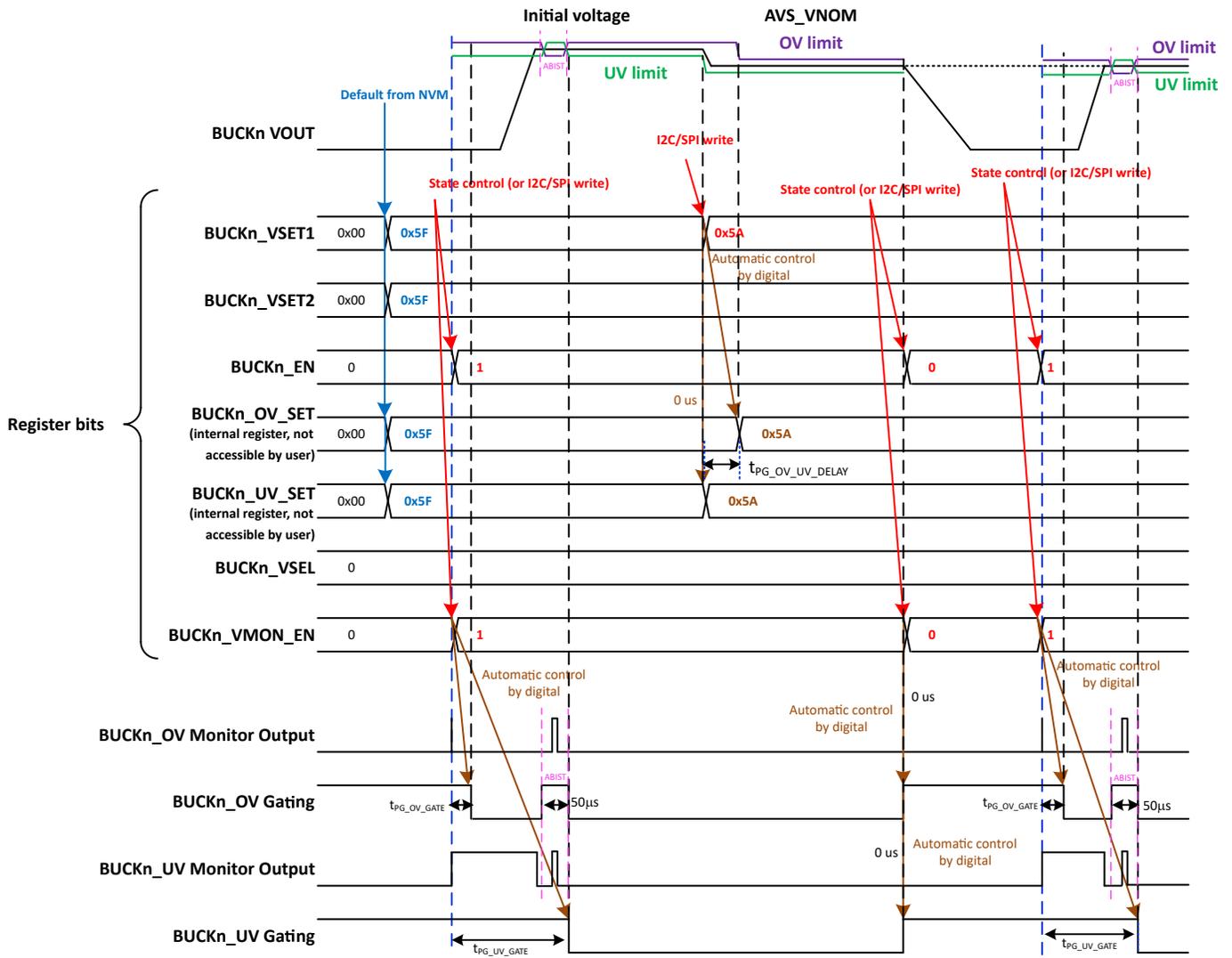


Figure 7-5. AVS Voltage and OV UV Threshold Level Change Timing Diagram

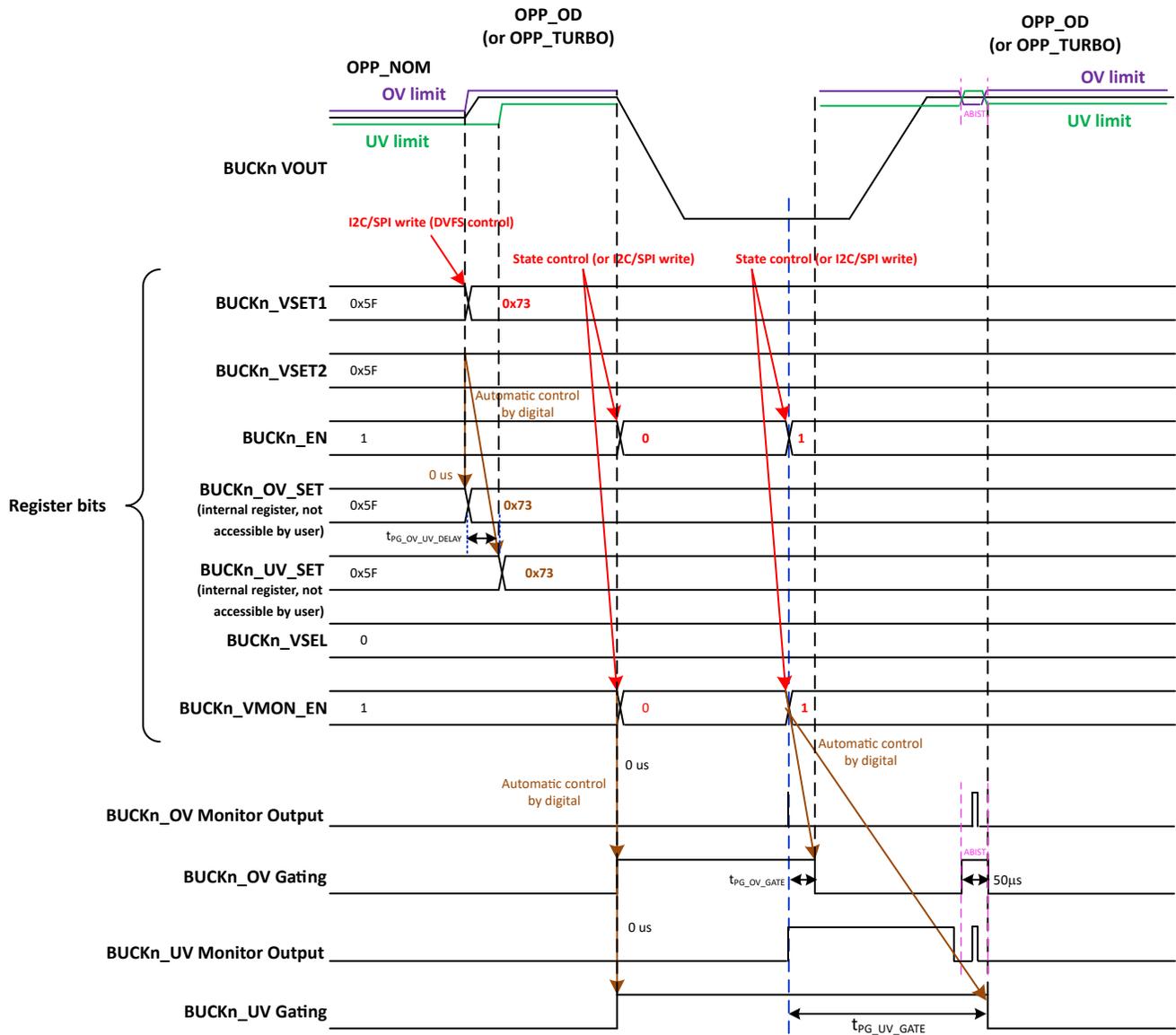


Figure 7-6. DVS Voltage and OV UV Threshold Level Change Timing Diagram

7.3.2.1.4 BUCK Output Voltage Setting

Table 7-1 shows the coding used to select the BUCK regulator output voltage.

Table 7-1. Output Voltage Selection for BUCK Regulators

| BUCKn_VSE Tn | Output Voltage [V] 20 mV steps | BUCKn_VSE Tn | Output Voltage [V] 5 mV steps | BUCKn_VSE Tn | Output Voltage [V] 5 mV steps | BUCKn_VSE Tn | Output Voltage [V] 10 mV steps | BUCKn_VSE Tn | Output Voltage [V] 20 mV steps | BUCKn_VSE Tn | 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 |
| 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 |

Table 7-1. Output Voltage Selection for BUCK Regulators (continued)

| BUCKn_VSE Tn | Output Voltage [V] 20 mV steps | BUCKn_VSE Tn | Output Voltage [V] 5 mV steps | BUCKn_VSE Tn | Output Voltage [V] 5 mV steps | BUCKn_VSE Tn | Output Voltage [V] 10 mV steps | BUCKn_VSE Tn | Output Voltage [V] 20 mV steps | BUCKn_VSE Tn | Output Voltage [V] 20 mV steps |
|-----------------|--------------------------------------|-----------------|-------------------------------------|-----------------|-------------------------------------|-----------------|--------------------------------------|-----------------|--------------------------------------|-----------------|--------------------------------------|
| | | 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 | | | | |

7.3.2.1.5 Sync Clock Functionality

The LP876242-Q1 device contains a SYNCCLKIN input to synchronize switching clock of the buck regulator with the external clock. The block diagram of the clocking and DPLL module is shown in Figure 7-7. 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, 4.4 MHz, or 8.8 MHz.

The EXT_CLK_INT interrupt is also generated in cases the external clock is expected, but it is not available.

The LP876242-Q1 device can also generate clock 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.

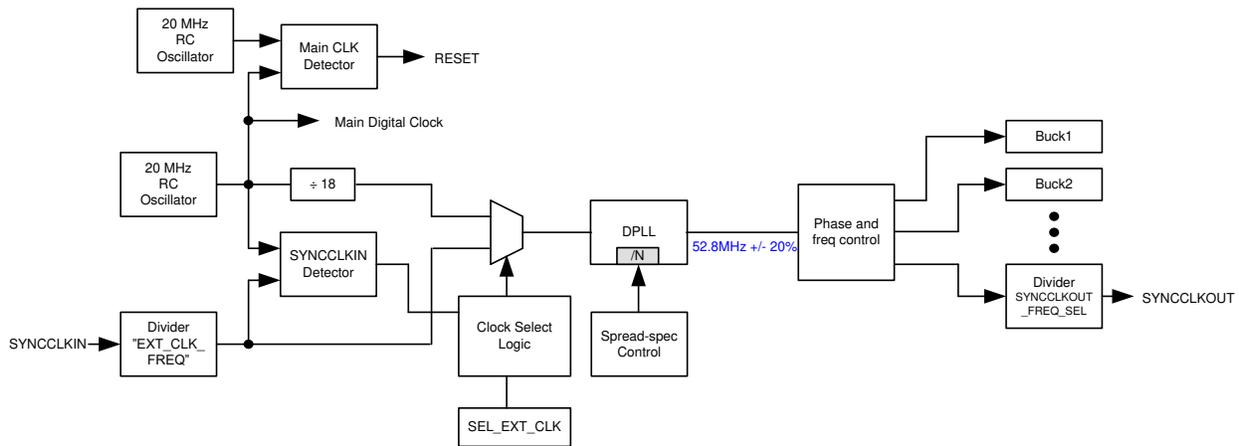


Figure 7-7. Sync Clock and DPLL Module

7.3.2.2 Internal Low Dropout Regulator (LDOVINT)

The LDOVINT voltage regulator is dedicated to supply the digital and analog functions of the LP876242-Q1 device. The LDOVINT regulator is controlled by VCCA supply, when VCCA is available the LDOVINT is enabled.

7.3.3 Residual Voltage Checking

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

This feature is enabled by the BUCKn_VMON_EN and BUCKn_RV_SEL bits for each buck regulators, and by the VMONn_EN and VMONn_RV_SEL bits for VMON inputs. When this feature is enabled, the voltage monitor of the corresponding regulator or monitoring input remains on after the monitoring is disabled, and remain on for the RV Timeout period. After the RV Timeout period elapses the monitored voltage is compared to the short circuit (SC) threshold of $V_{TH_SC_RV}$, and assert the corresponding BUCKn_SC_INT or VMONn_RV_INT interrupt bits if the residual voltage is still higher than the threshold voltage. The RV timeout period for the BUCK regulators and VMON inputs is automatically calculated by the digital controller inside the device by Equation 3 and Equation 4. If external voltage is monitored with buck feedback pin, buck is disabled all the time (BUCKn_EN = 0).

$$t_{BUCK_RV_TIMEOUT} = BUCKn_VSET / BUCKn_SLEW_RATE + 100 \mu s \quad (3)$$

$$t_{VMON_RV_TIMEOUT} = VMONn_PG_VSET / VMONn_SLEW_RATE + 100 \mu s \quad (4)$$

Figure 7-8 shows the timing diagram of the residual voltage checking operation that results in pass or fail results.

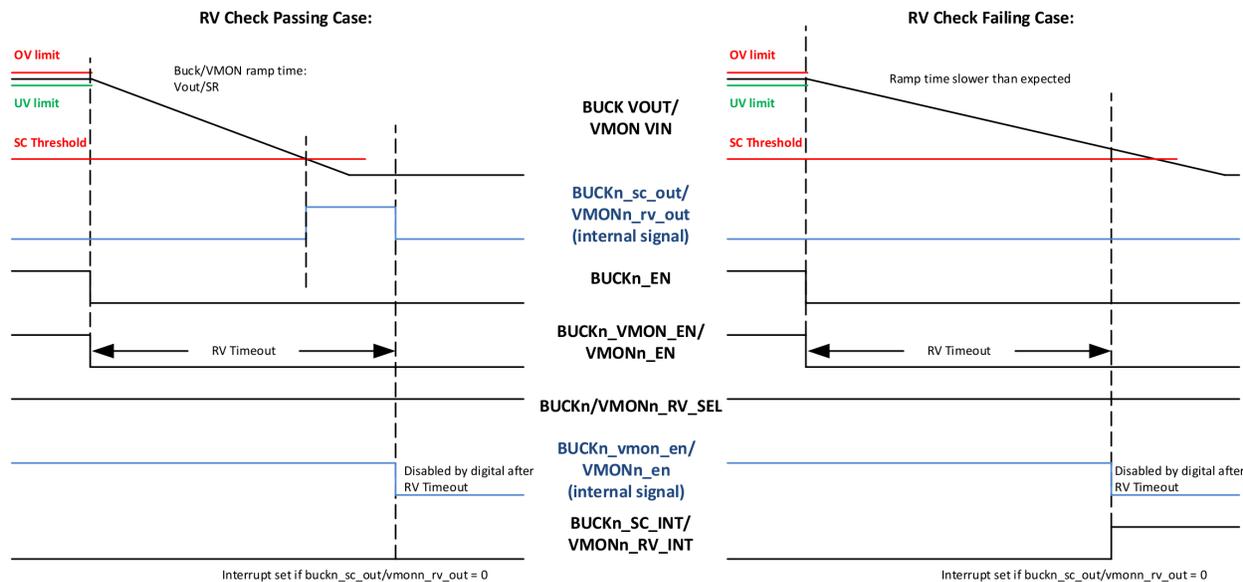


Figure 7-8. Residual Voltage Check Timing Diagram

Note

Unless mentioned otherwise in the User's Guide for the orderable part number, the residual voltage monitoring is enabled once, immediately after a power-up event on the VCCA input supply pin. Before the LP876242-Q1 executes the start-up sequence of the BUCK regulators, the LP876242-Q1 disables all residual voltage monitors. TI has made this implementation to prevent false-positive residual-voltage detection caused by charge accumulation in the output capacitors of the BUCK regulators or at the output voltage rails monitored through VMON_1 (at GPIO7 pin) or VMON_2 (at GPIO8 pin).

7.3.4 Output Voltage Monitor and PGOOD Generation

The LP876242-Q1 device monitors the undervoltage (UV) and overvoltage (OV) conditions on the output voltages of the BUCK regulators, the UV and OV conditions on the VMONn voltage monitoring input pins, and VCCA (when it is expected to be 5 V or 3.3 V), and has the option to indicate the result with a 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 primary phase for multi-phase regulator). When voltage and current are monitored, an active 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 PGOOD signal represents the momentary status of the included indications without latching. If the PGOOD signal goes low due to an indicated warning or error condition, the PGOOD signal goes high immediately when the previous indicated warning or error condition is no longer present.

The BUCKn_VMONn_EN bit enables the overvoltage (OV), undervoltage (UV) and short-circuit (SC) comparators. The current limit (ILIM) comparator of each BUCK regulator is activated as soon as the corresponding BUCK regulator is enabled. In order to add the current limit indication of a BUCK regulator to the PGOOD signal, the BUCKn_VMONn_EN bit of the corresponding BUCK regulator must be set. When a BUCK 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_VMONn_EN bit remains '1' while the BUCKn_EN bit is '0', it can be used as a voltage monitor for an external rail that is connected to the FB_Bn pin of the BUCK regulator.

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

The VCCA input voltage monitoring is enabled with VCCA_VMON_EN bit. The monitoring can be enabled by an NVM default setting, that starts 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 is 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 outputs of the voltage monitors 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, then undervoltage and overvoltage are monitored.

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

[Figure 7-9](#) shows the Power-Good generation block diagram, and [Figure 7-10](#) shows the Power-Good waveforms.

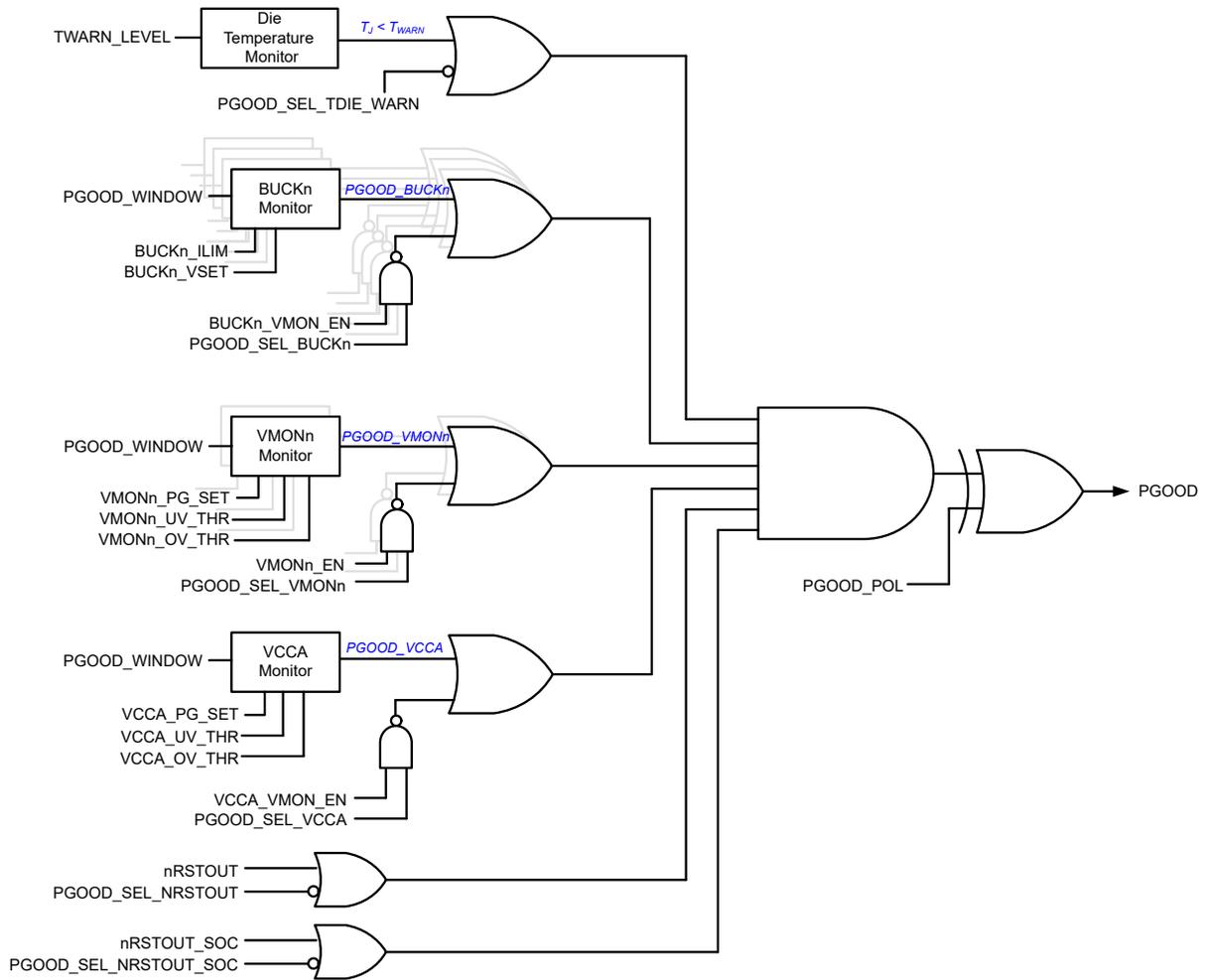


Figure 7-9. PGOOD Block Diagram

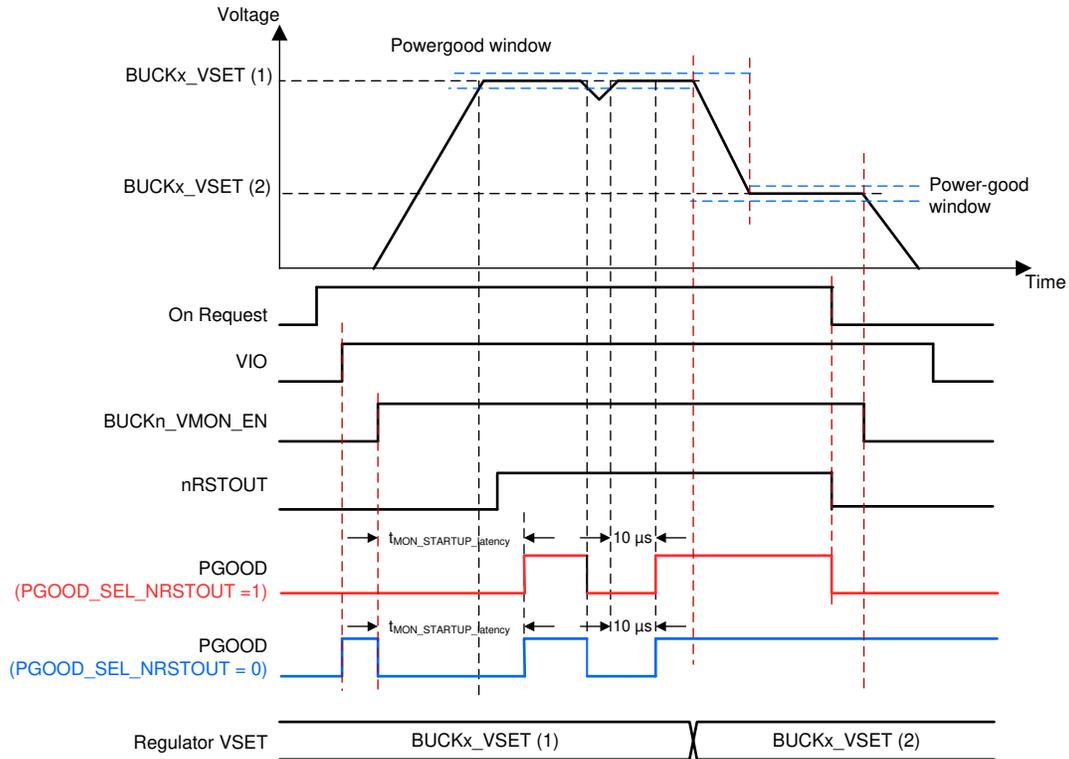


Figure 7-10. PGOOD Waveforms

The OV and UV threshold of the voltage monitors of the BUCK regulators and VMON input voltage monitor are updated automatically by the digital control block when the voltage setting changes. When the output voltage of the regulator is increased, the OV threshold is updated at the same time the `_VSET` of the regulator or the `PG_LEVEL` of the VMON 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 or `PG_SET` of the VMON 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 output voltage monitors and VMON input voltage monitors are calculated based on the target output voltage set by the corresponding `BUCKn_VSET1`, `BUCKn_VSET2`, or `VMONn_PG_SET` registers, and the deviation from the target output voltage set (the voltage window) by the corresponding `BUCKn_UV_THR`, `BUCKn_OV_THR`, `VMONn_UV_THR`, and the `VMONn_OV_THR` registers. For the OV and UV threshold of BUCK output monitors to update with the correct timing, the following operating procedures must be followed when updating the `_VSET` values of the regulators to avoid detection of OV/UV fault:

- BUCK regulators must be enabled at the same time as or earlier than as their VMON, such that the voltage reaches its target value before OV/UV self-test (BIST) is done
- New voltage level must not be set before the start-up 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: when a regulator is enabled, a voltage monitor self-test is performed to ensure proper operation. The monitoring function is disabled and gated during this time. [Figure 7-11](#) shows the timing diagram of the BUCK regulator UV/OV self-test. [Figure 7-12](#) shows the timing diagram of the VMON UV/OV self-test. The monitoring function is activated after the gating period.

The self-test for VCCA, BUCK and VMON 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 fails and `BIST_FAIL_INT` interrupt is set. In addition, a failed self-test for over-voltage comparator sets the over-voltage interrupt.

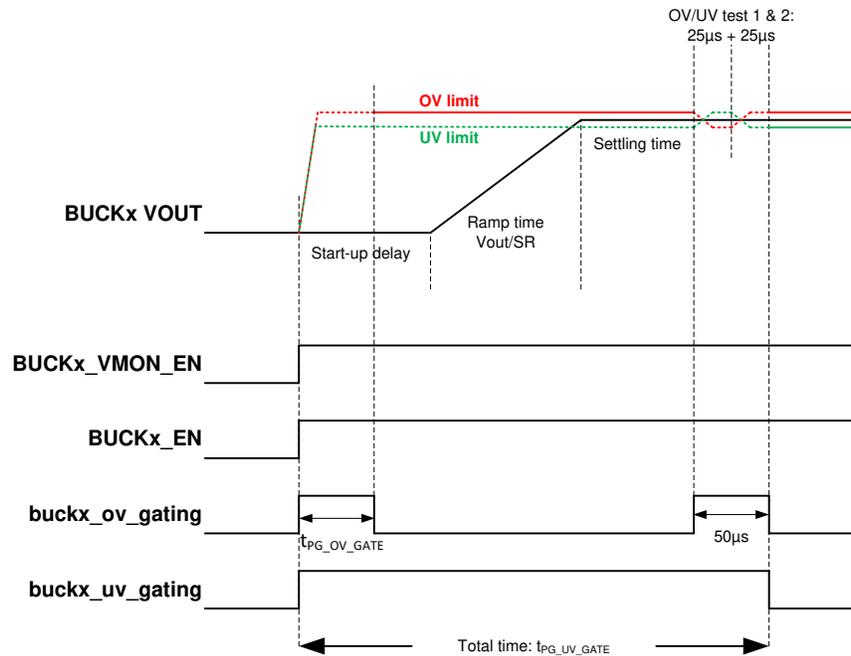


Figure 7-11. Timing of BUCK Regulator UV/OV Self-Test

The voltage monitors of unused BUCK regulators can be used for external supply rails monitoring. In three-phase configuration, the Voltage Monitor of BUCK3 (on FB_B3 pin) becomes a free available resource for monitoring an external supply voltage. In four-phase configuration, the Voltage Monitor of both BUCK3 (on FB_B3 pin) and BUCK4 (on FB_B4 pin) become free available resources for monitoring two external supply voltages. The target output voltage is set by the corresponding BUCKn_VSET1, BUCKn_VSET2 registers, and the deviation from the target output voltage set (the voltage window) by the corresponding BUCKn_UV_THR, BUCKn_OV_THR registers. Following aspects need to be taken into account if Voltage Monitors of unused BUCK regulators are used for monitoring external supply rails:

- For voltage monitors of unused BUCK : the maximum nominal supply voltage of the monitored supply rail is 3.3V
- For voltage monitors of unused BUCK regulators : the configured values for the BUCKn_VSET and the BUCKn_SLEW_RATE determine the delay-time for the voltage monitoring to become active after the corresponding BUCKn_VMON_EN bit is set. See equation (2) in [Section 7.3.2.1.3](#).

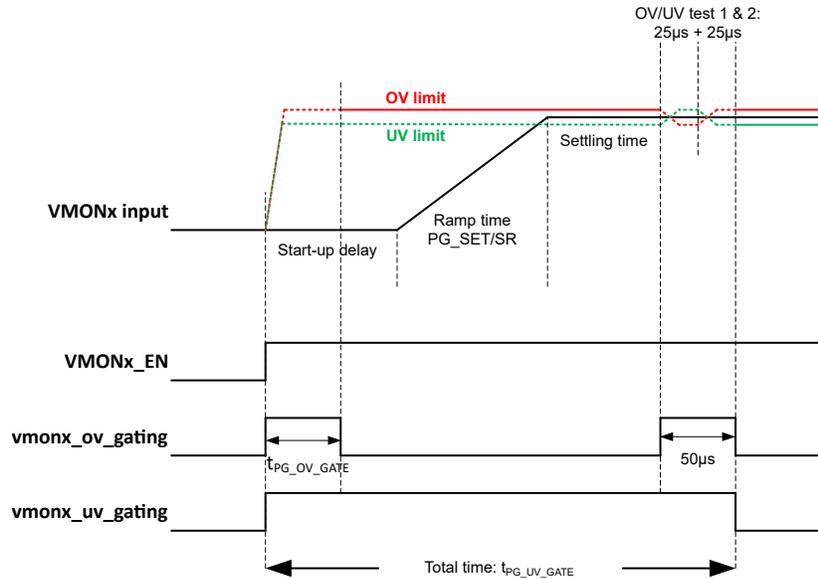


Figure 7-12. Timing of VMON Input UV/OV Self-Test

The voltage monitoring threshold for the VMONx input pins are shown in [Table 7-2](#).

Note

Unless mentioned otherwise in the User's Guide for the orderable part number, for automatically sequenced Voltage Monitors (either as part of automatic sequencing of BUCK regulators, or as stand-alone Voltage Monitor), the LP876242-Q1 unmask the UV/OV right before the release of the nRSTOUT resp. nRSTOUT_SoC pins. For Voltage Monitors which are enabled through system software (either as associated Voltage Monitor for BUCK regulators, or as stand-alone Voltage Monitor), the corresponding over voltage and under voltage interrupt masking must be enabled before enabling the voltage monitor. The Analog-BIST is performed for the voltage monitor as described in [Figure 7-11](#) and [Figure 7-12](#). The over voltage and under voltage interrupt masking must be disabled earliest at time calculated by programmed voltage level (Vout) divided by the programmed slew-rate and by adding 500 us time.

Note

For Voltage Monitors that are used to monitor external supply voltages, in order to not have a failing self-test or a false-positive UV/OV detection after completion of the self-test, the actual voltage level of the external supply (VOUT_actual) must satisfy following conditions:

- For $V_{OUT} > 1V$: $V_{OUT_actual} = V_{OUT_typical} \pm (75\% * \text{Typical UV/OV Threshold} - 1\%)$
- For $V_{OUT} \leq 1V$: $V_{OUT_actual} = V_{OUT_typical} \pm (\text{Typical UV/OV Threshold} - 15mV)$

VOUT_typical is the configured power-good voltage level for the used Voltage Monitor in registers BUCKn_VOUT1/2. This requirement also applies to the voltage on the VCCA pin in case the VCCA UV/OV Monitor is used.

Table 7-2. Monitoring Voltage Selection for VMONx External Voltage Pins

| VMO Nx_P G_SE T | Output Voltage [V] | | VMO Nx_P G_SE T | Output Voltage [V] | | VMO Nx_P G_SE T | Output Voltage [V] | | VMO Nx_P G_SE T | Output Voltage [V] | | VMO Nx_P G_SE T | Output Voltage [V] | | VMO Nx_P G_SE T | Output Voltage [V] | |
|--------------------------|-----------------------------|-------------------------|--------------------------|-----------------------------|-------------------------|--------------------------|-----------------------------|----------------------------|--------------------------|-----------------------------|----------------------------|--------------------------|-----------------------------|----------------------------|--------------------------|-----------------------------|----------------------------|
| | VMO Nx_R ANG E = 0 | VMONx _RANG E = 1 | | VMO Nx_R ANG E = 0 | VMON x_RAN GE = 1 | | VMO Nx_R ANG E = 0 | VMON x_RA NGE = 1 |
| 0x00 | 0.3 | Reserved | 0x0F | 0.6 | Reserved | 0x41 | 0.85 | 4.25 | 0x73 | 1.1 | Reserved | 0xAB | 1.66 | Reserved | 0xD6 | 2.52 | Reserved |
| 0x01 | 0.32 | Reserved | 0x10 | 0.605 | Reserved | 0x42 | 0.855 | 4.275 | 0x74 | 1.11 | Reserved | 0xAC | 1.68 | Reserved | 0xD7 | 2.54 | Reserved |
| 0x02 | 0.34 | Reserved | 0x11 | 0.61 | Reserved | 0x43 | 0.86 | 4.3 | 0x75 | 1.12 | Reserved | 0xAD | 1.7 | Reserved | 0xD8 | 2.56 | Reserved |
| 0x03 | 0.36 | Reserved | 0x12 | 0.615 | Reserved | 0x44 | 0.865 | 4.325 | 0x76 | 1.13 | Reserved | 0xAE | 1.72 | Reserved | 0xD9 | 2.58 | Reserved |
| 0x04 | 0.38 | Reserved | 0x13 | 0.62 | Reserved | 0x45 | 0.87 | 4.35 | 0x77 | 1.14 | Reserved | 0xAF | 1.74 | Reserved | 0xDA | 2.6 | Reserved |
| 0x05 | 0.4 | Reserved | 0x14 | 0.625 | Reserved | 0x46 | 0.875 | 4.375 | 0x78 | 1.15 | Reserved | 0xB0 | 1.76 | Reserved | 0xDB | 2.62 | Reserved |
| 0x06 | 0.42 | Reserved | 0x15 | 0.63 | Reserved | 0x47 | 0.88 | 4.4 | 0x79 | 1.16 | Reserved | 0xB1 | 1.78 | Reserved | 0xDC | 2.64 | Reserved |
| 0x07 | 0.44 | Reserved | 0x16 | 0.635 | Reserved | 0x48 | 0.885 | 4.425 | 0x7A | 1.17 | Reserved | 0xB2 | 1.8 | Reserved | 0xDD | 2.66 | Reserved |
| 0x08 | 0.46 | Reserved | 0x17 | 0.64 | Reserved | 0x49 | 0.89 | 4.45 | 0x7B | 1.18 | Reserved | 0xB3 | 1.82 | Reserved | 0xDE | 2.68 | Reserved |
| 0x09 | 0.48 | Reserved | 0x18 | 0.645 | Reserved | 0x4A | 0.895 | 4.475 | 0x7C | 1.19 | Reserved | 0xB4 | 1.84 | Reserved | 0xDF | 2.7 | Reserved |
| 0x0A | 0.5 | Reserved | 0x19 | 0.65 | Reserved | 0x4B | 0.9 | 4.5 | 0x7D | 1.2 | Reserved | 0xB5 | 1.86 | Reserved | 0xE0 | 2.72 | Reserved |
| 0x0B | 0.52 | Reserved | 0x1A | 0.655 | Reserved | 0x4C | 0.905 | 4.525 | 0x7E | 1.21 | Reserved | 0xB6 | 1.88 | Reserved | 0xE1 | 2.74 | Reserved |
| 0x0C | 0.54 | Reserved | 0x1B | 0.66 | Reserved | 0x4D | 0.91 | 4.55 | 0x7F | 1.22 | Reserved | 0xB7 | 1.9 | Reserved | 0xE2 | 2.76 | Reserved |
| 0x0D | 0.56 | Reserved | 0x1C | 0.665 | Reserved | 0x4E | 0.915 | 4.575 | 0x80 | 1.23 | Reserved | 0xB8 | 1.92 | Reserved | 0xE3 | 2.78 | Reserved |
| 0x0E | 0.58 | Reserved | 0x1D | 0.67 | 3.35 | 0x4F | 0.92 | 4.6 | 0x81 | 1.24 | Reserved | 0xB9 | 1.94 | Reserved | 0xE4 | 2.8 | Reserved |
| | | | 0x1E | 0.675 | 3.375 | 0x50 | 0.925 | 4.625 | 0x82 | 1.25 | Reserved | 0xBA | 1.96 | Reserved | 0xE5 | 2.82 | Reserved |
| | | | 0x1F | 0.68 | 3.4 | 0x51 | 0.93 | 4.65 | 0x83 | 1.26 | Reserved | 0xBB | 1.98 | Reserved | 0xE6 | 2.84 | Reserved |
| | | | 0x20 | 0.685 | 3.425 | 0x52 | 0.935 | 4.675 | 0x84 | 1.27 | Reserved | 0xBC | 2 | Reserved | 0xE7 | 2.86 | Reserved |
| | | | 0x21 | 0.69 | 3.45 | 0x53 | 0.94 | 4.7 | 0x85 | 1.28 | Reserved | 0xBD | 2.02 | Reserved | 0xE8 | 2.88 | Reserved |
| | | | 0x22 | 0.695 | 3.475 | 0x54 | 0.945 | 4.725 | 0x86 | 1.29 | Reserved | 0xBE | 2.04 | Reserved | 0xE9 | 2.9 | Reserved |

Table 7-2. Monitoring Voltage Selection for VMONx External Voltage Pins (continued)

| VMO Nx_P G_SE T | Output Voltage [V] | | VMO Nx_P G_SE T | Output Voltage [V] | | VMO Nx_P G_SE T | Output Voltage [V] | | VMO Nx_P G_SE T | Output Voltage [V] | | VMO Nx_P G_SE T | Output Voltage [V] | | VMO Nx_P G_SE T | Output Voltage [V] | |
|--------------------------|-----------------------------|-------------------------|--------------------------|-----------------------------|-------------------------|--------------------------|-----------------------------|----------------------------|--------------------------|-----------------------------|----------------------------|--------------------------|-----------------------------|----------------------------|--------------------------|-----------------------------|----------------------------|
| | VMO Nx_R ANG E = 0 | VMONx _RANG E = 1 | | VMO Nx_R ANG E = 0 | VMON x_RAN GE = 1 | | VMO Nx_R ANG E = 0 | VMON x_RA NGE = 1 |
| | | | 0x23 | 0.7 | 3.5 | 0x55 | 0.95 | 4.75 | 0x87 | 1.3 | Reserv ed | 0xBF | 2.06 | Reser ved | 0xEA | 2.92 | Reser ved |
| | | | 0x24 | 0.705 | 3.525 | 0x56 | 0.955 | 4.775 | 0x88 | 1.31 | Reserv ed | 0xC0 | 2.08 | Reser ved | 0xEB | 2.94 | Reser ved |
| | | | 0x25 | 0.71 | 3.55 | 0x57 | 0.96 | 4.8 | 0x89 | 1.32 | Reserv ed | 0xC1 | 2.1 | Reser ved | 0xEC | 2.96 | Reser ved |
| | | | 0x26 | 0.715 | 3.575 | 0x58 | 0.965 | 4.825 | 0x8A | 1.33 | Reserv ed | 0xC2 | 2.12 | Reser ved | 0xED | 2.98 | Reser ved |
| | | | 0x27 | 0.72 | 3.6 | 0x59 | 0.97 | 4.85 | 0x8B | 1.34 | Reserv ed | 0xC3 | 2.14 | Reser ved | 0xEE | 3.0 | Reser ved |
| | | | 0x28 | 0.725 | 3.625 | 0x5A | 0.975 | 4.875 | 0x8C | 1.35 | Reserv ed | 0xC4 | 2.16 | Reser ved | 0xEF | 3.02 | Reser ved |
| | | | 0x29 | 0.73 | 3.65 | 0x5B | 0.98 | 4.9 | 0x8D | 1.36 | Reserv ed | 0xC5 | 2.18 | Reser ved | 0xF0 | 3.04 | Reser ved |
| | | | 0x2A | 0.735 | 3.675 | 0x5C | 0.985 | 4.925 | 0x8E | 1.37 | Reserv ed | 0xC6 | 2.2 | Reser ved | 0xF1 | 3.06 | Reser ved |
| | | | 0x2B | 0.74 | 3.7 | 0x5D | 0.99 | 4.95 | 0x8F | 1.38 | Reserv ed | 0xC7 | 2.22 | Reser ved | 0xF2 | 3.08 | Reser ved |
| | | | 0x2C | 0.745 | 3.725 | 0x5E | 0.995 | 4.975 | 0x90 | 1.39 | Reserv ed | 0xC8 | 2.24 | Reser ved | 0xF3 | 3.1 | Reser ved |
| | | | 0x2D | 0.75 | 3.75 | 0x5F | 1.0 | 5.0 | 0x91 | 1.4 | Reserv ed | 0xC9 | 2.26 | Reser ved | 0xF4 | 3.12 | Reser ved |
| | | | 0x2E | 0.755 | 3.775 | 0x60 | 1.005 | Reserv ed | 0x92 | 1.41 | Reserv ed | 0xCA | 2.28 | Reser ved | 0xF5 | 3.14 | Reser ved |
| | | | 0x2F | 0.76 | 3.8 | 0x61 | 1.01 | Reserv ed | 0x93 | 1.42 | Reserv ed | 0xCB | 2.3 | Reser ved | 0xF6 | 3.16 | Reser ved |
| | | | 0x30 | 0.765 | 3.825 | 0x62 | 1.015 | Reserv ed | 0x94 | 1.43 | Reserv ed | 0xCC | 2.32 | Reser ved | 0xF7 | 3.18 | Reser ved |
| | | | 0x31 | 0.77 | 3.85 | 0x63 | 1.02 | Reserv ed | 0x95 | 1.44 | Reserv ed | 0xCD | 2.34 | Reser ved | 0xF8 | 3.2 | Reser ved |
| | | | 0x32 | 0.775 | 3.875 | 0x64 | 1.025 | Reserv ed | 0x96 | 1.45 | Reserv ed | 0xCE | 2.36 | Reser ved | 0xF9 | 3.22 | Reser ved |
| | | | 0x33 | 0.78 | 3.9 | 0x65 | 1.03 | Reserv ed | 0x97 | 1.46 | Reserv ed | 0xCF | 2.38 | Reser ved | 0xFA | 3.24 | Reser ved |
| | | | 0x34 | 0.785 | 3.925 | 0x66 | 1.035 | Reserv ed | 0x98 | 1.47 | Reserv ed | 0xD0 | 2.4 | Reser ved | 0xFB | 3.26 | Reser ved |
| | | | 0x35 | 0.79 | 3.95 | 0x67 | 1.04 | Reserv ed | 0x99 | 1.48 | Reserv ed | 0xD1 | 2.42 | Reser ved | 0xFC | 3.28 | Reser ved |
| | | | 0x36 | 0.795 | 3.975 | 0x68 | 1.045 | Reserv ed | 0x9A | 1.49 | Reserv ed | 0xD2 | 2.44 | Reser ved | 0xFD | 3.3 | Reser ved |

Table 7-2. Monitoring Voltage Selection for VMONx External Voltage Pins (continued)

| VMO Nx_P G_SE T | Output Voltage [V] | | VMO Nx_P G_SE T | Output Voltage [V] | | VMO Nx_P G_SE T | Output Voltage [V] | | VMO Nx_P G_SE T | Output Voltage [V] | | VMO Nx_P G_SE T | Output Voltage [V] | | VMO Nx_P G_SE T | Output Voltage [V] | |
|--------------------------|-----------------------------|-------------------------|--------------------------|-----------------------------|-------------------------|--------------------------|-----------------------------|----------------------------|--------------------------|-----------------------------|----------------------------|--------------------------|-----------------------------|----------------------------|--------------------------|-----------------------------|----------------------------|
| | VMO Nx_R ANG E = 0 | VMONx _RANG E = 1 | | VMO Nx_R ANG E = 0 | VMON x_RAN GE = 1 | | VMO Nx_R ANG E = 0 | VMON x_RA NGE = 1 |
| | | | 0x37 | 0.8 | 4.0 | 0x69 | 1.05 | Reserv ed | 0x9B | 1.5 | Reserv ed | 0xD3 | 2.46 | Reser ved | 0xFE | 3.32 | Reser ved |
| | | | 0x38 | 0.805 | 4.025 | 0x6A | 1.055 | Reserv ed | 0x9C | 1.51 | Reserv ed | 0xD4 | 2.48 | Reser ved | 0xFF | 3.34 | Reser ved |
| | | | 0x39 | 0.81 | 4.05 | 0x6B | 1.06 | Reserv ed | 0x9D | 1.52 | Reserv ed | 0xD5 | 2.5 | Reser ved | | | |
| | | | 0x3A | 0.815 | 4.075 | 0x6C | 1.065 | Reserv ed | 0x9E | 1.53 | Reserv ed | | | | | | |
| | | | 0x3B | 0.82 | 4.1 | 0x6D | 1.07 | Reserv ed | 0x9F | 1.54 | Reserv ed | | | | | | |
| | | | 0x3C | 0.825 | 4.125 | 0x6E | 1.075 | Reserv ed | 0xA0 | 1.55 | Reserv ed | | | | | | |
| | | | 0x3D | 0.83 | 4.15 | 0x6F | 1.08 | Reserv ed | 0xA1 | 1.56 | Reserv ed | | | | | | |
| | | | 0x3E | 0.835 | 4.175 | 0x70 | 1.085 | Reserv ed | 0xA2 | 1.57 | Reserv ed | | | | | | |
| | | | 0x3F | 0.84 | 4.2 | 0x71 | 1.09 | Reserv ed | 0xA3 | 1.58 | Reserv ed | | | | | | |
| | | | 0x40 | 0.845 | 4.225 | 0x72 | 1.095 | Reserv ed | 0xA4 | 1.59 | Reserv ed | | | | | | |
| | | | | | | | | | 0xA5 | 1.6 | Reserv ed | | | | | | |
| | | | | | | | | | 0xA6 | 1.61 | Reserv ed | | | | | | |
| | | | | | | | | | 0xA7 | 1.62 | Reserv ed | | | | | | |
| | | | | | | | | | 0xA8 | 1.63 | Reserv ed | | | | | | |
| | | | | | | | | | 0xA9 | 1.64 | Reserv ed | | | | | | |
| | | | | | | | | | 0xAA | 1.65 | Reserv ed | | | | | | |

7.3.5 General-Purpose I/Os (GPIO Pins)

The LP876242-Q1 device integrates ten configurable general-purpose I/Os that are multiplexed with alternative features as listed in Pin Configuration and Functions.

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 GPIO_n_CONF register.

- GPIOn_DEGLITCH_EN: Enables the 8 μ s 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:

- ENABLE
- EN_DRV
- nRSTOUT
- nRSTOUT_SOC
- PGOOD
- nERR_MCU
- TRIG_WDOG
- NSLEEP1, NSLEEP2
- WKUP1, 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 are 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, SCL_SPMI, SYNCCLKIN, or SYNCCLKOUT. Please refer to Digital Signal Descriptions 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. 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 7.4.1.2.4.3](#) and [Section 7.4.1.2.4.4](#).

Any of the GPIO pin can also be configured as part of the power-up sequence to enable external devices such as external BUCKs when it is configured as a general-purpose output port.

The nINT pin and the GPIO pins assigned as EN_DRV, nRSTOUT and nRSTOUT_SOC have readback monitoring to detect errors on the signals. The monitoring of the GPIO pin assigned as EN_DRV checks for mismatch in both low and high levels. For the nINT pin and the GPIO pins assigned as nRSTOUT and nRSTOUT_SOC, the readback monitoring only checks for mismatches in the low level, therefore it is allowed to combine these signals with other external pull-down sources. The readback mismatch is continuously monitored without deglitch circuitry during operation, and the monitoring is gated for $t_{\text{gate_readback}}$ period when the signal state is changed or when a new function is selected for the GPIO pin with the GPIOn_SEL bits. NINT_READBACK_INT, EN_DRV_READBACK_INT, NRSTOUT_READBACK_INT, and NRSTOUT_SOC_READBACK_INT are the interrupt bits that are set in an event of a readback mismatch for these pins, respectively.

Note

All GPIO pin are set to generic input pins with resistive pull-down before NVM memory is loaded during device power up. Therefore, if any GPIOs has external pull-up resistors connecting to a voltage domain that is energized before the NVM memory is loaded, the GPIO pin is pulled high before the configuration for the pin is loaded from the NVM.

Note

For GPIO pins with internal pull down enabled, additional leakage current flows into the GPIO pin if this pin is pulled-up to a voltage higher than the voltage level of its output power domain. If the internal pull down must be enabled, please use a resistor divider to divide down the input voltage, or use a series resistor to connect to the input source and ensure the voltage level at the GPIO pin is below the voltage level of its output power domain.

7.3.6 Thermal Monitoring

The LP876242-Q1 device includes several thermal monitoring functions for internal thermal protection of the PMIC.

The LP876242-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 BUCK modules. An over-temperature condition at either module first generates a warning to the system, and if the temperature continues to rise, then 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 7.3.6.1](#), that 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 7.3.6.2](#), that begins device shutdown orderly or immediately.

Thermal monitoring is automatically enabled when one of the BUCK outputs is enabled within the mission states. The thermal monitoring is disabled in the LP_STANDBY state, when only the internal regulator is 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 the 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 LP876242-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.

7.3.6.1 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 LP876242-Q1 device uses the TWARN_LEVEL register bit to set the thermal warning threshold temperature at 130°C or 140°C. There is no hysteresis 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).

7.3.6.2 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, the LP876242-Q1 device performs an orderly shutdown of all output power rails. If the die temperature raises rapidly and reaches the T_{SD_imm} level before the orderly shutdown process completes, the LP876242-Q1

device performs an immediate shutdown with activated pull-down on all output power rails, in order to turn off all of the output power rails as rapidly as possible. After the thermal shutdown takes place, the system cannot restart until the die temperature is below the thermal warning threshold.

7.3.7 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 (also indicated as SC interrupt) and over-current (ILIM) error conditions found on the BUCK regulators . |
| VMON ERROR | These interrupts indicate OV and UV or residual voltage error conditions found on the VMON1 and VMON2 inputs and on the VCCA supply. |
| SEVERE ERROR | These errors indicate severe device error conditions, such as thermal shutdown, PFSM sequencing and execution error and VCCA over-voltage, that 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. |
| MODERATE ERROR | These interrupts provide warnings to the system to indicate multiple restart attempts from SAFE RECOVERY state exceeding the allowed recovery count, multiple warm-reset executions exceeding the allowed recovery count, SPMI communication error, register CRC error, BIST failure, read-back error on nRSTOUT or nINT pins, or junction temperature 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. ¹ |
| 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. |
| START-UP SOURCE | These interrupts provide information to the system on the mechanism that caused the device to start up, which includes FSD, the activation of the ENABLE pin |
| GPIO DETECTION | These interrupts indicate the High/Rising-Edge or the Low/Falling-Edge detection at the GPIO1 through GPIO10 pins. |
| FSM ERROR INTERRUPT | These interrupts indicate the detection of an error that 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 bits 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 7.5.3](#) for more information on SPI status signals.

Hierarchical Structure of Interrupt Registers 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. Summary of Interrupt Signals summarizes the trigger and the clearing mechanism for all of the interrupt signals. This table also shows which interrupt sources 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 is not affected, and the event is not recorded. If an interrupt is masked after the event occurred, the interrupt register bit reflects the event until the bit is cleared. While the event is masked, the interrupt register bit is not over-written when a new event occurs.

¹ The SEVERE ERROR and the MODERATE ERROR are 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.

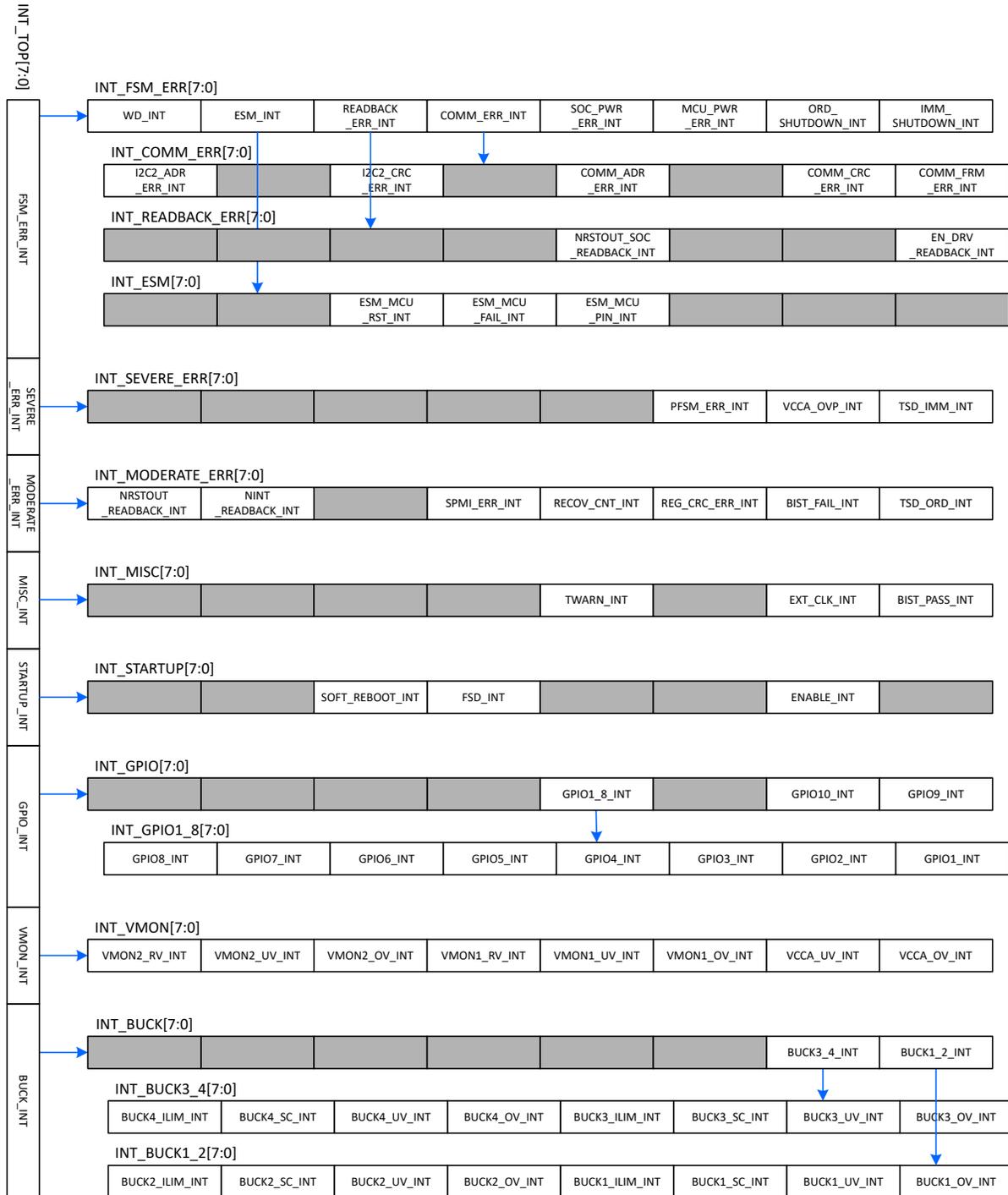


Figure 7-13. Hierarchical Structure of Interrupt Registers

Table 7-3. 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_CTR L=1: According to BUCKn_GRP_SEL and x_RAIL_TRIG bits EN_ILIM_FSM_CTR L=0: N/A | EN_ILIM_FSM_CTR L=1: Transition according to FSM trigger and interrupt EN_ILIM_FSM_CTR L=0: Interrupt only | Depends on PFSM configuration, see PFSM 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 |

Table 7-3. Summary of Interrupt Signals (continued)

| EVENT | TRIGGER FOR FSM | RESULT ⁽¹⁾ | RECOVERY | INTERRUPT BIT | MASK FOR INTERRUPT | LIVE STATUS BIT | INTERRUPT CLEAR |
|--|--|--|--|------------------|--------------------|-----------------|--|
| 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 PFSM configuration, see PFSM transition diagram | BUCKn_SC_INT = 1 | N/A | N/A | Write 1 to BUCKn_SC_INT bit Interrupt is not cleared if the BUCKn is enabled and the BUCKn output voltage is below the short-circuit threshold after elapse of expected ramp-up time interval |
| 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 PFSM configuration, see PFSM transition diagram | BUCKn_SC_INT = 1 | N/A | N/A | Write 1 to BUCKn_SC_INT bit If BUCKn_SC_INT was set due to a detected residual voltage, this bit can only be read by the MCU if the residual voltage condition is no longer present and the device has performed a successful power-up sequence |
| BUCK regulator overvoltage | According to BUCKn_GRP_SEL and x_RAIL_TRIG bits | Transition according to FSM trigger and interrupt | Depends on PFSM configuration, see PFSM transition diagram | BUCKn_OV_INT = 1 | BUCKn_OV_MASK | BUCKn_OV_STAT | Write 1 to BUCKn_OV_INT bit Interrupt is not cleared if the associated fault condition is still present |
| BUCK regulator undervoltage | According to BUCKn_GRP_SEL and x_RAIL_TRIG bits | Transition according to FSM trigger and interrupt | Depends on PFSM configuration, see PFSM transition diagram | BUCKn_UV_INT = 1 | BUCKn_UV_MASK | BUCKn_UV_STAT | Write 1 to BUCKn_UV_INT bit Interrupt is not cleared if the associated fault condition is still present |
| VCCA input overvoltage monitoring | According to VCCA_GRP_SEL and x_RAIL_TRIG bits | Transition according to FSM trigger and interrupt | Depends on PFSM configuration, see PFSM transition diagram | VCCA_OV_INT = 1 | VCCA_OV_MASK | VCCA_OV_STAT | Write 1 to VCCA_OV_INT bit Interrupt is not cleared if the associated fault condition is still present |
| VCCA input undervoltage monitoring | According to VCCA_GRP_SEL and x_RAIL_TRIG bits | Transition according to FSM trigger and interrupt | Depends on PFSM configuration, see PFSM transition diagram | VCCA_UV_INT = 1 | VCCA_UV_MASK | VCCA_UV_STAT | Write 1 to VCCA_UV_INT bit Interrupt is not cleared if the associated fault condition is still present |
| VMONx input overvoltage monitoring | According to VMONx_GRP_SEL and X_RAIL_TRIG bits | Transition according to FSM trigger and interrupt | Depends on PFSM configuration, see PFSM transition diagram | VMONx_OV_INT = 1 | VMONx_OV_MASK | VMONx_OV_STAT | Write 1 to VMONx_OV_INT bit Interrupt is not cleared if the associated fault condition is still present |
| VMONx input undervoltage monitoring | According to VMONx_GRP_SEL and X_RAIL_TRIG bits | Transition according to FSM trigger and interrupt | Depends on PFSM configuration, see PFSM transition diagram | VMONx_UV_INT = 1 | VMONx_UV_MASK | VMONx_UV_STAT | Write 1 to VMONx_UV_INT bit Interrupt is not cleared if the associated fault condition is still present |
| VMONx residual voltage violation | VMONx_RV_SEL = 1 According to VMONn_GRP_SEL and x_RAIL_TRIG bits VMONx_RV_SEL = 0 N/A | VMONx_RV_SEL = 1 Transition according to FSM trigger and interrupt VMONx_RV_SEL = 0 N/A | Depends on PFSM configuration, see PFSM transition diagram | VMONx_RV_INT = 1 | N/A | N/A | Write 1 to VMONx_RV_INT bit |

Table 7-3. Summary of Interrupt Signals (continued)

| EVENT | TRIGGER FOR FSM | RESULT ⁽¹⁾ | RECOVERY | INTERRUPT BIT | MASK FOR INTERRUPT | LIVE STATUS BIT | INTERRUPT CLEAR |
|--|-------------------------------------|---|---|-------------------------|----------------------|----------------------|--|
| 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 start-up 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 This interrupt bit can only be read by the MCU after the device has recovered from a previously occurred over-temperature event. |
| 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 start-up 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 This interrupt bit can only be read by the MCU after the device has recovered from a previously occurred over-temperature event. |
| BIST error | ORDERLY_SHUTDOWN (MODERATE_ERR_INT) | All regulators disabled and Output GPIOx set to low immediately and interrupt ⁽¹⁾ | Automatic start-up to STARTUP_DEST[1:0] state | BIST_FAIL_INT = 1 | BIST_FAIL_MASK | N/A | Write 1 to BIST_FAIL_INT bit |
| Register CRC error | ORDERLY_SHUTDOWN (MODERATE_ERR_INT) | All regulators disabled and Output GPIOx set to low immediately and interrupt ⁽¹⁾ | Automatic start-up 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 start-up 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 ⁽⁴⁾ | 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 ⁽⁴⁾ | 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 start-up 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 only | 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 the associated fault condition is still present |

Table 7-3. Summary of Interrupt Signals (continued)

| EVENT | TRIGGER FOR FSM | RESULT ⁽¹⁾ | RECOVERY | INTERRUPT BIT | MASK FOR INTERRUPT | LIVE STATUS BIT | INTERRUPT CLEAR |
|---|-------------------------------------|--|---|--------------------------------|---------------------------|---------------------------|--|
| NINT pin readback error (monitoring low state) | ORDERLY_SHUTDOWN (MODERATE_ERR_INT) | All regulators disabled and Output GPIOx set to low immediately and interrupt ⁽¹⁾ | Automatic start-up 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 the associated fault condition is still present |
| NRSTOUT pin readback error (monitoring low state) | ORDERLY_SHUTDOWN (MODERATE_ERR_INT) | All regulators disabled and Output GPIOx set to low immediately and interrupt ⁽¹⁾ | Automatic start-up 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 the associated fault condition is still present |
| 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 the associated fault condition is still present |
| 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 Interrupt is not cleared if the associated fault condition is still present |
| 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 Interrupt is not cleared if the associated fault condition is still present |
| 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 This bit can only be read by the MCU after the LP876242-Q1 has executed a warm-reset and the recovery counter does not exceed the recovery threshold |
| 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 Interrupt is not cleared if the associated fault condition is still present |
| 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 | Interrupt 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 This bit can only be read by the MCU after the LP876242-Q1 has executed a warm-reset and the recovery counter does not exceed the recovery threshold |

Table 7-3. Summary of Interrupt Signals (continued)

| EVENT | TRIGGER FOR FSM | RESULT ⁽¹⁾ | RECOVERY | INTERRUPT BIT | MASK FOR INTERRUPT | LIVE STATUS BIT | INTERRUPT CLEAR |
|--|---|---|---|----------------------------|------------------------------------|-----------------|---|
| 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 This bit can only be read by the MCU after the LP876242-Q1 has executed a warm-reset and the recovery counter does not exceed the recovery threshold |
| 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 start-up 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 (depending on NVM configuration with or without pull-down resistors) and Output GPIOx set to low immediately and interrupt ⁽¹⁾ | Automatic start-up 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 PFSM configuration, see PFSM 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 PFSM configuration, see PFSM 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 start-up 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 VCCA_OVP_INT bit This bit can only be read by the MCU if VCCA < VCCA _{OVP} level. As long as VCCA > VCCA _{OVP} level, device stays in SAFE_RECOVER state, and hence this interrupt cannot be not cleared. |
| 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 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 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 |
| 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 | Reset condition for all logic circuits | 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 |

Table 7-3. Summary of Interrupt Signals (continued)

| EVENT | TRIGGER FOR FSM | RESULT ⁽¹⁾ | RECOVERY | INTERRUPT BIT | MASK FOR INTERRUPT | LIVE STATUS BIT | INTERRUPT CLEAR |
|---|--|---|---------------------|---------------|--------------------|-----------------|------------------------|
| Main clock outside valid frequency | Reset condition for all logic circuits | 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} | Reset condition for all logic circuits | All regulators disabled with pull-down resistors and Output GPIOx set to low immediately ⁽¹⁾ | VCCA voltage rising | N/A | N/A | N/A | N/A |
| First supply detection, VCCA supply rising above VCCA _{UVLO} | TRIGGER_SU_x | Start-up 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 functional 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 does not occur if RECOV_CNT_THR = 0, even though RECOV_CNT continues to accumulate and increase, and eventually saturates when it reaches the maximum count of 15.
- (4) I2C1, I2C2, or SPI address error only occur in safety applications if the interface CRC feature is enabled, when both I2C1_SPI_CRC_EN and I2C2_CRC_EN are set to '1'.

7.3.8 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 pin when the watchdog detects correct operation of the MCU. When the watchdog detects an incorrect operation of the MCU, the LP876242-Q1 device pulls the pin low. This 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 that 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`. Watchdog Trigger Mode provides more details.

Q&A (question and answer) mode In Q&A mode, the MCU sends watchdog answers through the I2C bus. To select this mode, the MCU must set bit `WD_MODE_SELECT`. Watchdog Question-Answer Mode provides more details.

7.3.8.1 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 `=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, 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 (see PFSM Trigger Selections) and sets the error-flag, 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 and are latched until the MCU writes a '1' to these bits.

Overview of Watchdog Fail Counter Value Ranges and Corresponding Device Status gives an overview of the Watchdog Fail Counter value ranges and the corresponding device status.

Table 7-4. Overview of Watchdog Fail Counter Value Ranges and Corresponding Device Status

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

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

Refer to Watchdog Trigger Mode and Watchdog Q&A Related Definitions respectively for definitions of good events and bad events.

7.3.8.2 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 ($t_{\text{LONG_WINDOW}}$) 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 or 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 7.3.8.4](#))
- WD_RETURN_LONGWIN to configure whether to return to Long-Window or continue to the next sequence after the completion of the current watchdog sequence (more detail in [Section 7.3.8.4](#))
- 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 or 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

The device keeps 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 ($t_{\text{LONG_WINDOW}}$) 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 steps

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

$$t_{\text{LONG_WINDOW_MIN}} = \text{WD_LONGWIN}[7:0] \times 0.95 \quad (5)$$

$$t_{\text{LONG_WINDOW_MAX}} = \text{WD_LONGWIN}[7:0] \times 1.05 \quad (6)$$

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. The MCU must clear bit WD_PWRHOLD before setting bit WD_EN back to '1' to start the watchdog in 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]

7.3.8.3 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 before elapse of the Long Window time interval:

- Clear bits WD_PWRHOLD (more detail in [Section 7.3.8.4](#))

- 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 . This bit latched until the MCU writes a '1' to clear it.

7.3.8.4 Watchdog Disable Function

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 7.3.8.3](#)), the MCU must clear bit WD_RETURN_LONGWIN before the end of the first watchdog sequence in order to continue the watchdog sequence operation.

7.3.8.5 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 after one 20-MHz system clock cycle 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 pin if 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.

$$t_{WINDOW1_MIN} = (WD_WIN1[6:0] + 1) \times 0.55 \times 0.95 \text{ ms} \quad (7)$$

$$t_{WINDOW1_MAX} = (WD_WIN1[6:0] + 1) \times 0.55 \times 1.05 \text{ ms} \quad (8)$$

Use [Equation 9](#) and [Equation 10](#) to calculate the minimum and maximum values for the $t_{WINDOW2}$ time interval.

$$t_{WINDOW2_MIN} = (WD_WIN2[6:0] + 1) \times 0.55 \times 0.95 \text{ ms} \quad (9)$$

$$t_{WINDOW2_MAX} = (WD_WIN2[6:0] + 1) \times 0.55 \times 1.05 \text{ ms} \quad (10)$$

7.3.8.6 Watchdog Trigger Mode

When the LP876242-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 , 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 the internal 20-MHz 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.

WatchDog Flow Chart and Timing Diagrams in Trigger Mode shows the flow-chart of the watchdog in Trigger mode.

7.3.8.7 WatchDog Flow Chart and Timing Diagrams in Trigger Mode

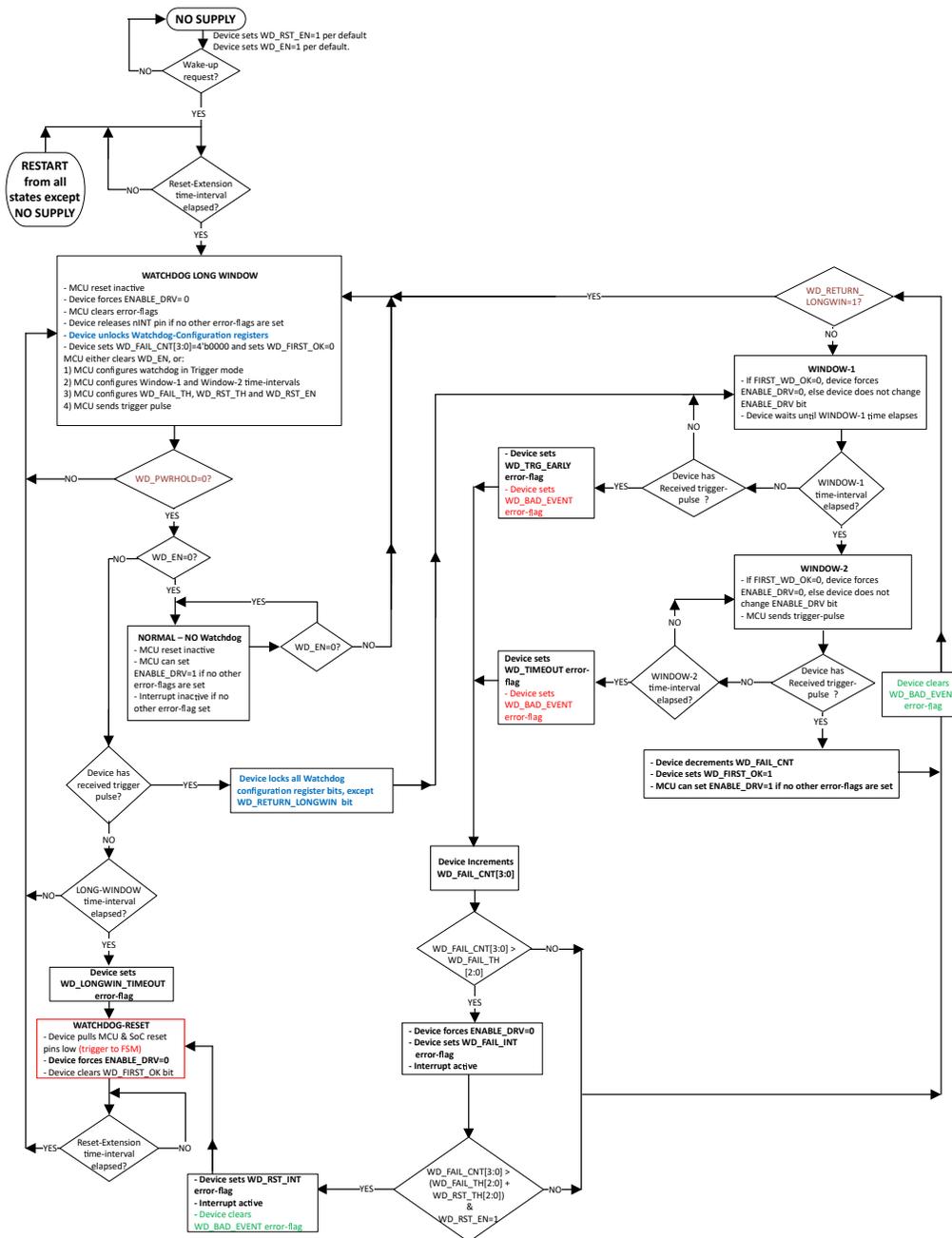


Figure 7-14. Flow Chart for WatchDog Monitor in Trigger Mode

Figure 7-15, Figure 7-16, Figure 7-17, Figure 7-18, and Figure 7-19 give examples of watchdog is trigger mode with good and bad events after device start-up. In these figures, the red bended arrows indicate a delay of one 20-MHz system clock cycle.

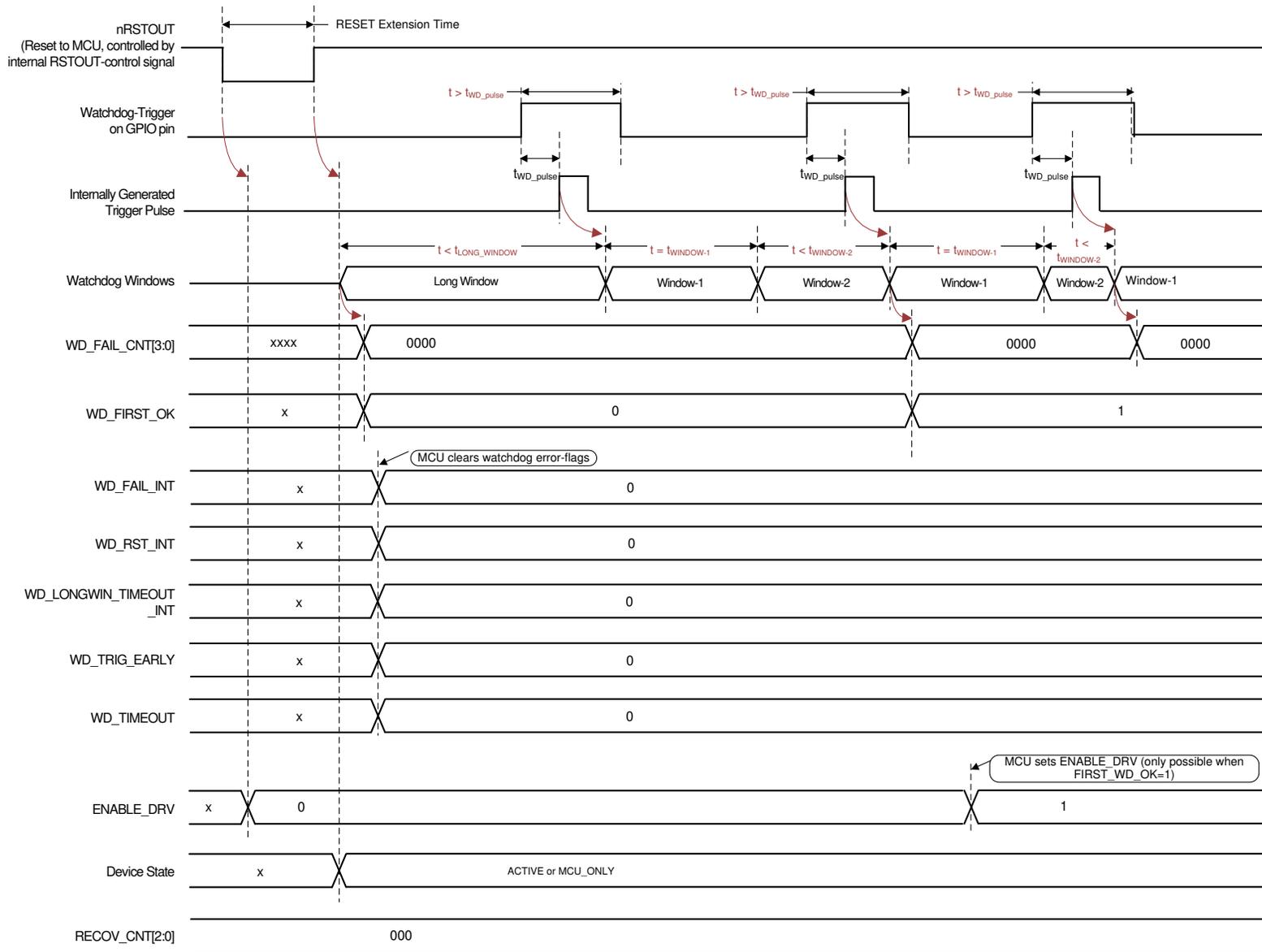


Figure 7-15. Watchdog in Trigger Mode – Normal MCU Start-up with Correct Watchdog-Triggers

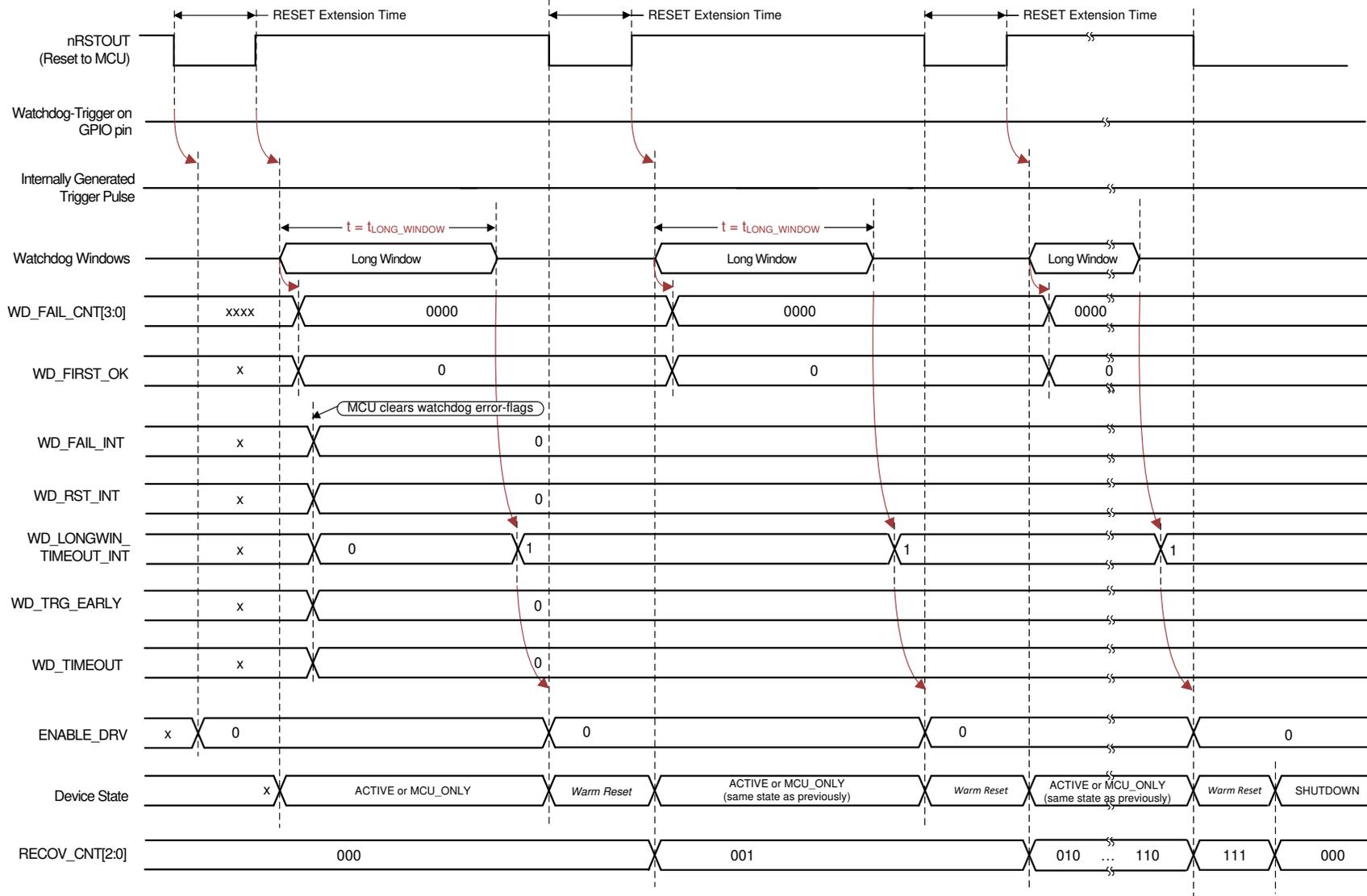


Figure 7-16. Watchdog in Trigger Mode – MCU Does Not Send Watchdog-Triggers After Start-up

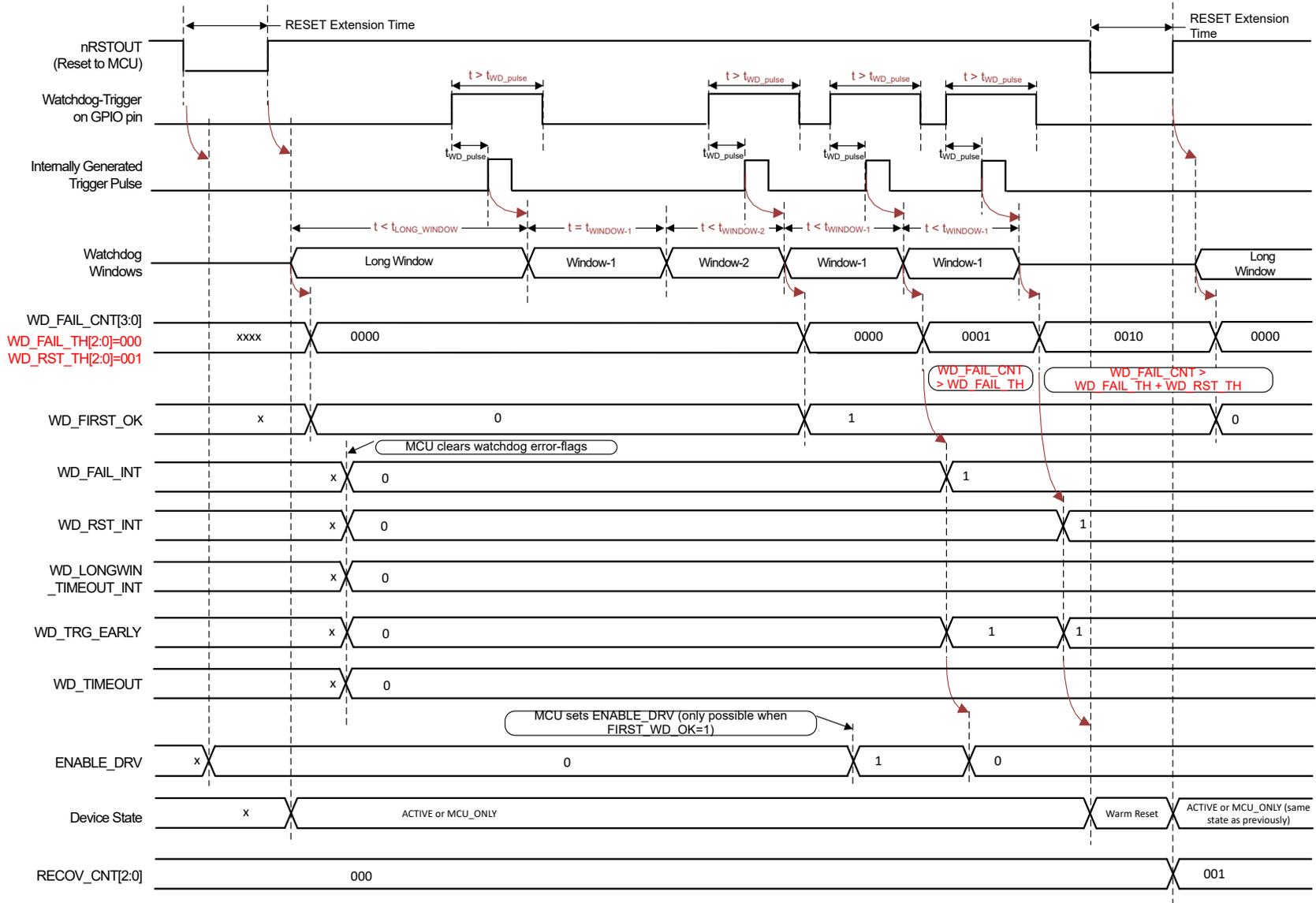


Figure 7-17. Watchdog in Trigger Mode – Bad Event (Watchdog-Triggers in Window-1) After Start-up

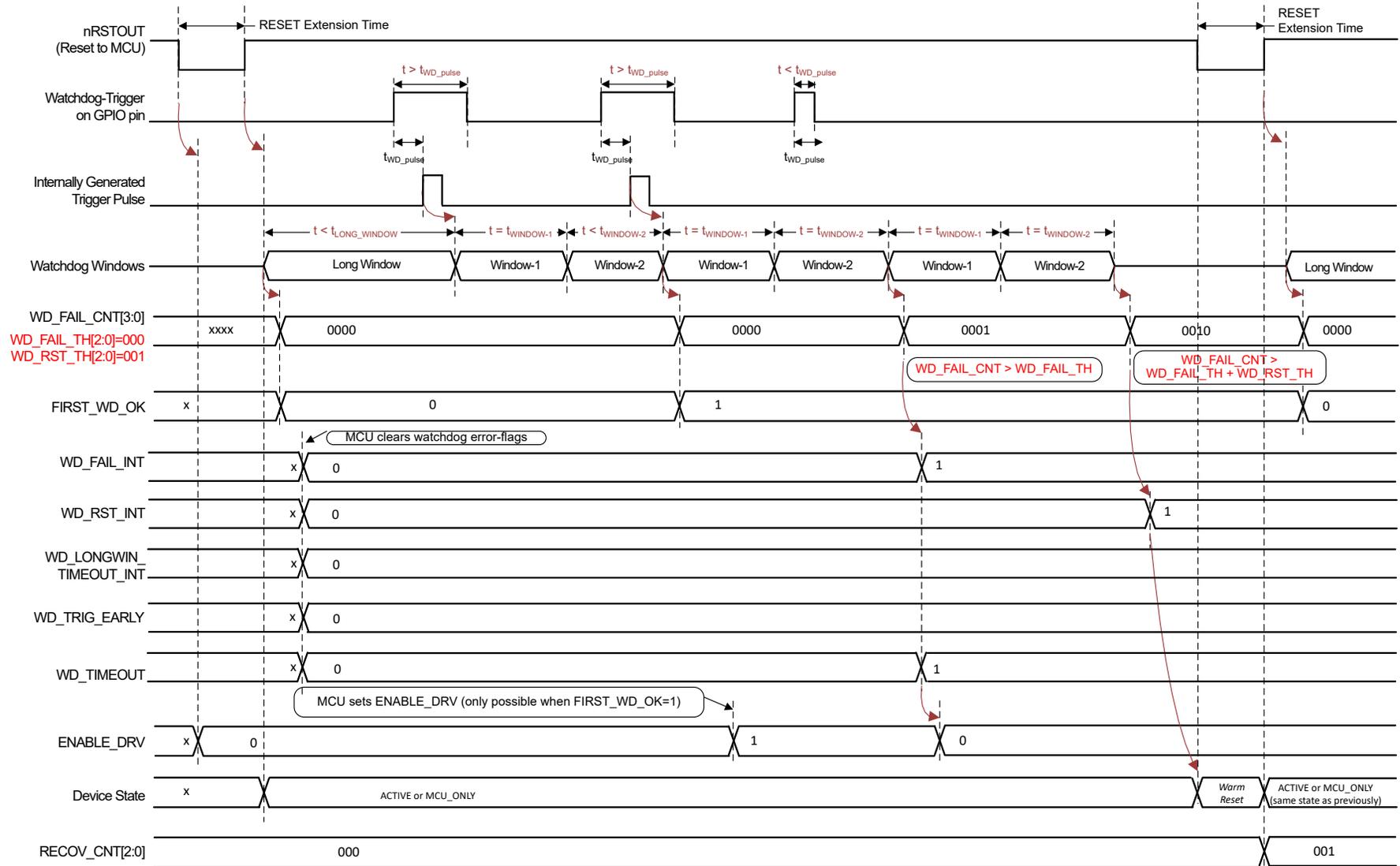


Figure 7-18. Watchdog in Trigger Mode – Bad Events (Too Short or no Trigger in Window-2) After Start-up

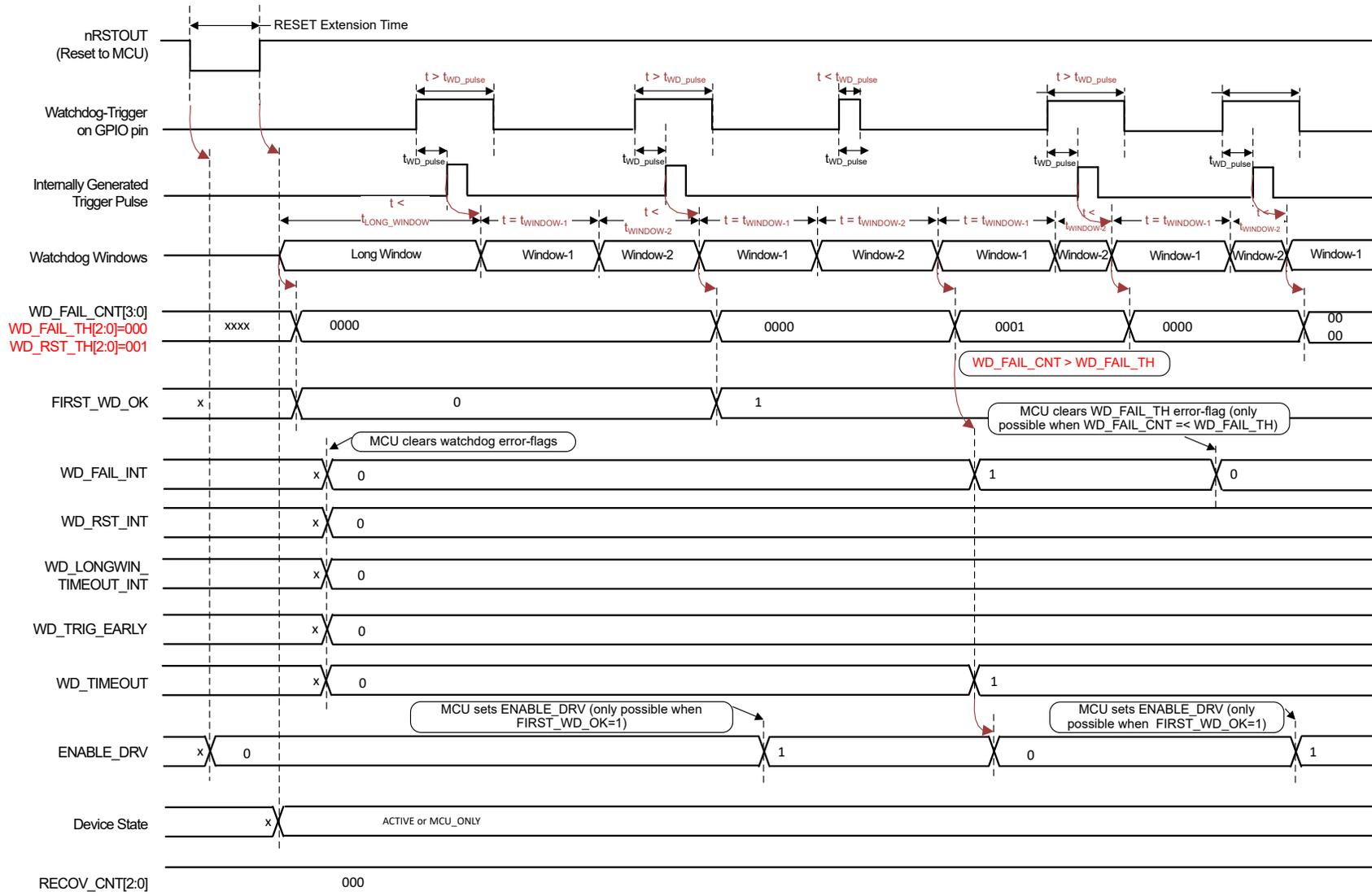


Figure 7-19. Watchdog in Trigger Mode – Good Events (Correct Watchdog-Triggers) After Start-up, Followed by a Bad-Event (No Watchdog-Triggers in Window-2) and After That Followed by a Good Event.

7.3.8.8 Watchdog Question-Answer Mode

When the LP876242-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.

The device provides a question for the MCU in WD_QUESTION[3:0] during operation. 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 I²C2 interface, mapped to GPIO2 and GPIO3 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.

7.3.8.8.1 Watchdog Q&A Related Definitions

A question and answer are defined as follows:

Question A question is a 4-bit word (see [Section 7.3.8.8.2](#)).

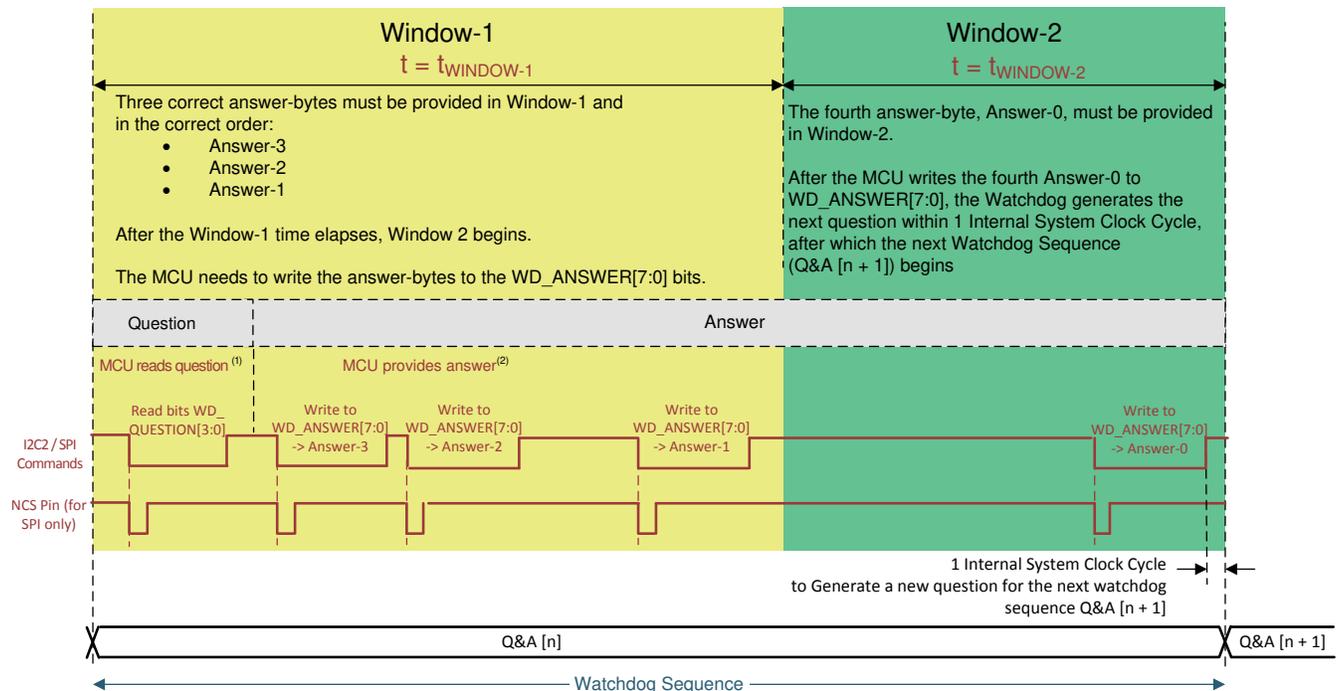
The watchdog provides the question to the MCU when the MCU reads the WD_QUESTION[3:0] bits.

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 that generates the question according to the circuit shown in [Figure 7-22](#). 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.



The register WD_QUESTION_ANSW_CNT has following bits:

- Bit 6: Reserved
- Bit 5-4: WD_ANSW_CNT. These bits give the number for the next expected answer-byte
- Bit 3-0: WD_QUESTION. These bits give the value of the current generated question

(1) For the question generation, 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 I²C or SPI commands in-between the write-commands to WD_ANSWER[7:0] (even re-requesting the question). The insertion of other commands in-between the write-commands to WD_ANSWER[7:0] 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 7-20. Watchdog Sequence in Q&A Mode

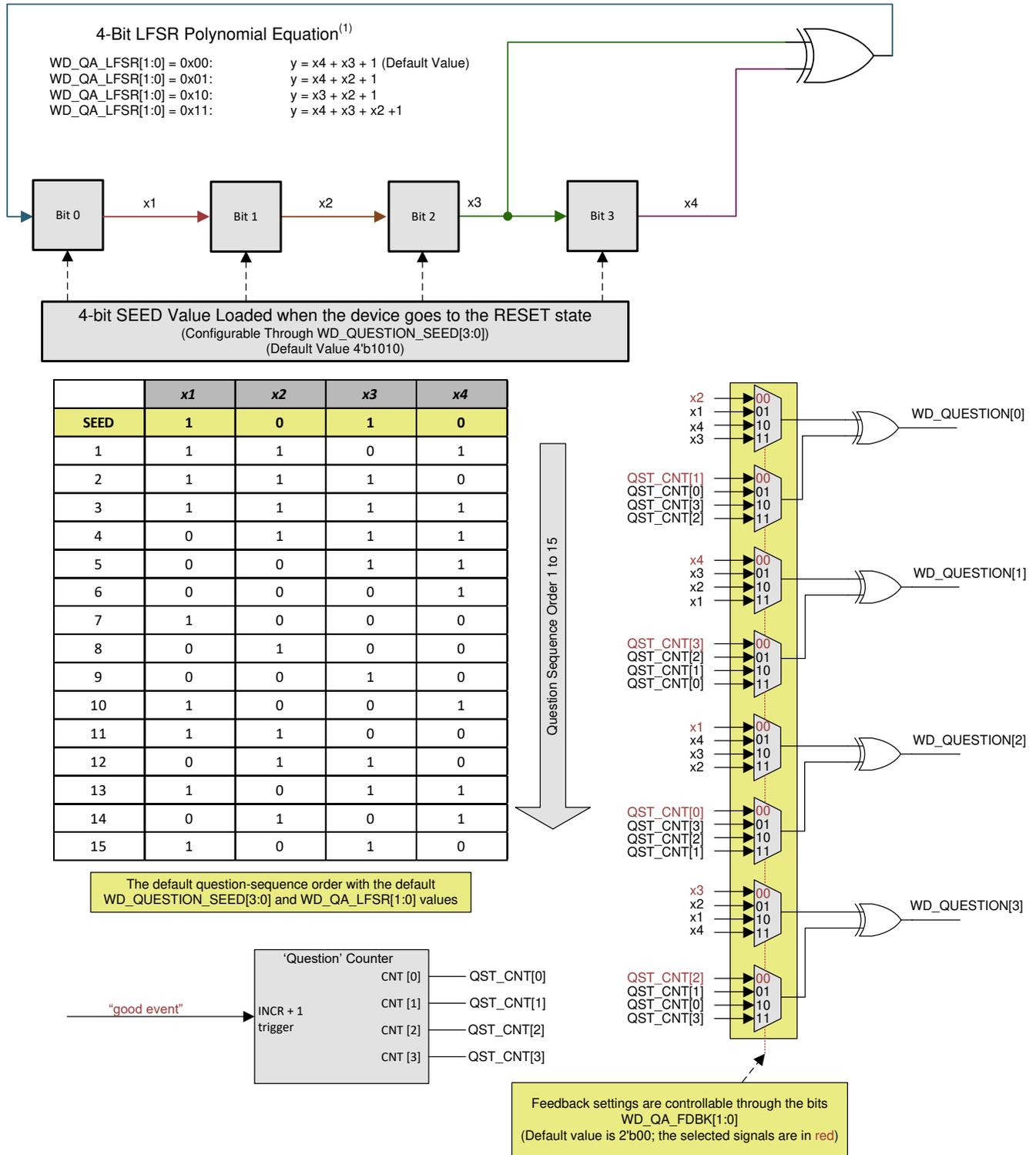
7.3.8.8.2 Question Generation

The watchdog uses a 4-bit *question counter* (QST_CNT[3:0] bits in [Figure 7-21](#)), 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.

[Figure 7-21](#) shows the logic combination for the WD_QUESTION[3:0] generation.

[Figure 7-22](#) shows how the logic combination of the question-counter with the WD_ANSW_CNT[1:0] status bits generates the reference answer-bytes.



(1) If the current value for bits (x1, x2, x3, x4) is 4'b0000, the next value for these bits (x1, x2, x3, x4) is 4'b0001, and all further question generation begins from this value.

Figure 7-21. Watchdog Question Generation

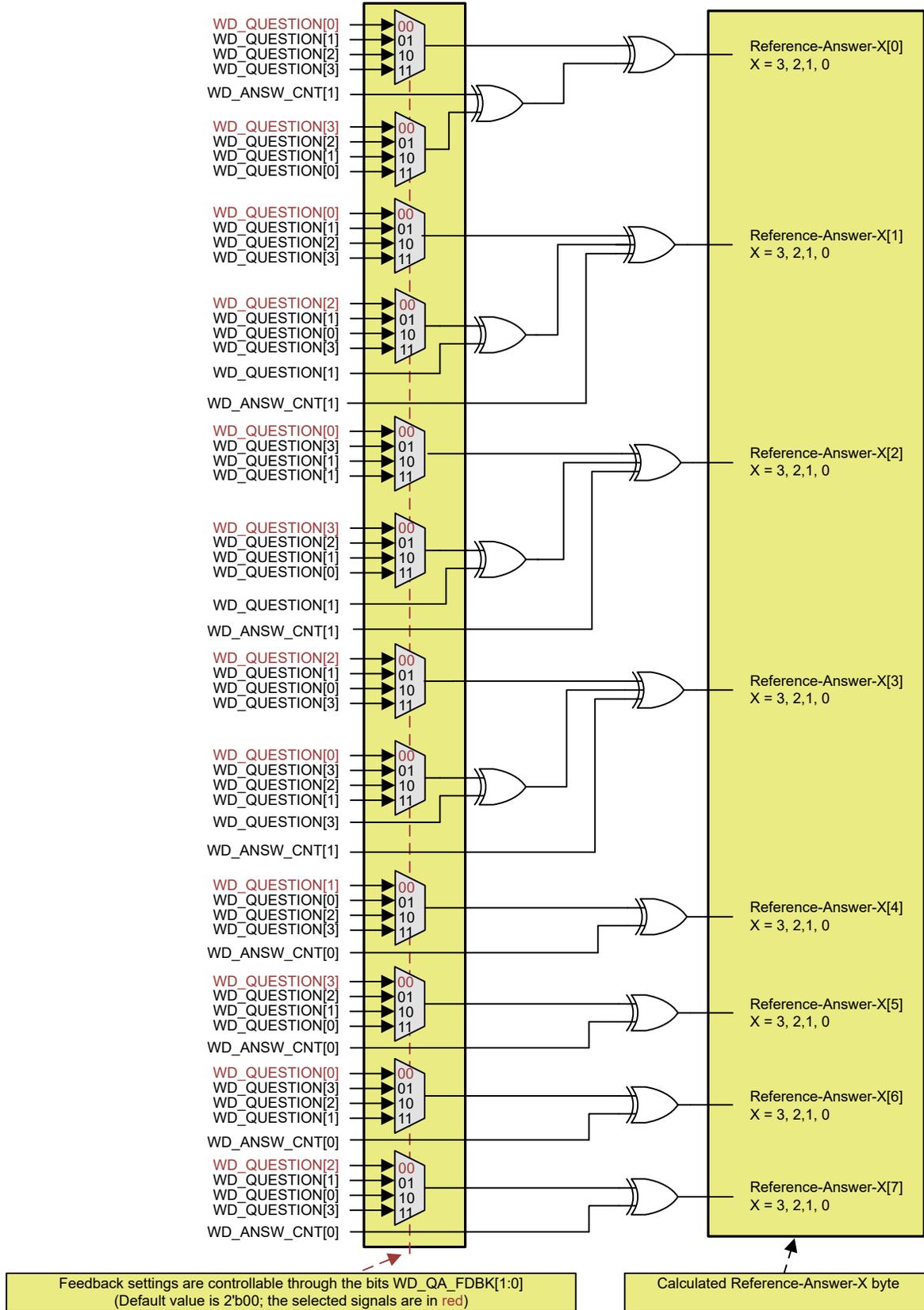


Figure 7-22. Watchdog Reference Answer Calculation

7.3.8.8.3 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 7-22](#). 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.

7.3.8.8.3.1 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 $2b'11$.

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

Table 7-5. 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 |

7.3.8.8.3.2 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 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 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 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 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 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.

[Figure 7-23](#) shows the flow-chart of the watchdog in Q&A mode.

7.3.8.8.3.3 Watchdog Q&A Sequence Scenarios
Table 7-6. Correct and Incorrect WD Q&A Sequence Run Scenarios

| NUMBER OF WD ANSWERS | | ACTION | WD STATUS BITS IN WDT_STATUS REGISTER | | | | COMMENTS |
|----------------------|---------------------|--|---------------------------------------|------------|---------|----------|--|
| RESPONSE WINDOW 1 | RESPONSE WINDOW 2 | | ANSW_ERR | ANSW_EARLY | SEQ_ERR | TIME_OUT | |
| 0 answers | 0 answers | -New WD cycle starts after the end of RESPONSE WINDOW 2 -Increment WD failure counter -New WD cycle starts with the same WD question | 0b | 0b | 1b | 1b | No answers |
| 0 answers | 4 INCORRECT answers | -New WD cycle starts after the 4th WD answer -Increment WD failure counter -New WD cycle starts with the same WD question | 1b | 0b | 1b | 0b | WD_ANSW_CNT[1:0] = 3 |
| 0 answers | 4 CORRECT answers | -New WD cycle starts after the 4th WD answer -Increment WD failure counter -New WD cycle starts with the same WD question | 0b | 0b | 1b | 0b | WD_ANSW_CNT[1:0] = 3 |
| 0 answers | 1 CORRECT answer | -New WD cycle starts after the end of RESPONSE WINDOW 2 -Increment WD failure counter -New WD cycle starts with the same WD question | 0b | 0b | 1b | 1b | Less than 3 CORRECT ANSWER in RESPONSE WINDOW 1 and 1 CORRECT ANSWER in RESPONSE WINDOW 2 (WD_ANSW_CNT[1:0] < 3) |
| 1 CORRECT answer | 1 CORRECT answer | | | | | | |
| 2 CORRECT answer | 1 CORRECT answer | | | | | | |
| 0 answers | 1 INCORRECT answer | -New WD cycle starts after the end of RESPONSE WINDOW 2 -Increment WD failure counter -New WD cycle starts with the same WD question | 1b | 0b | 1b | 1b | Less than 3 CORRECT ANSWER in RESPONSE WINDOW 1 and 1 INCORRECT ANSWER in RESPONSE WINDOW 2 (WD_ANSW_CNT[1:0] < 3) |
| 1 CORRECT answer | 1 INCORRECT answer | | | | | | |
| 2 CORRECT answers | 1 INCORRECT answer | | | | | | |
| 0 answers | 4 CORRECT answers | -New WD cycle starts after the 4th WD answer -Increment WD failure counter -New WD cycle starts with the same WD question | 0b | 0b | 1b | 0b | Less than 3 CORRECT ANSWER in WIN1 and more than 1 CORRECT ANSWER in RESPONSE WINDOW 2 (WD_ANSW_CNT[1:0] = 3) |
| 1 CORRECT answer | 3 CORRECT answers | | | | | | |
| 2 CORRECT answers | 2 CORRECT answers | | | | | | |
| 0 answers | 4 INCORRECT answers | -New WD cycle starts after the 4th WD answer -Increment WD failure counter -New WD cycle starts with the same WD question | 1b | 0b | 1b | 0b | Less than 3 CORRECT ANSWER in RESPONSE WINDOW 1 and more than 1 INCORRECT ANSWER in RESPONSE WINDOW 2 (WD_ANSW_CNT[1:0] = 3) |
| 1 CORRECT answer | 3 INCORRECT answers | | | | | | |
| 2 CORRECT answers | 2 INCORRECT answers | | | | | | |
| 0 answers | 3 CORRECT answers | -New WD cycle starts after the end of RESPONSE WINDOW 2 -Increment WD failure counter -New WD cycle starts with the same WD question | 0b | 0b | 1b | 1b | Less than 3 INCORRECT ANSWER in RESPONSE WINDOW 1 and more than 1 CORRECT ANSWER in RESPONSE WINDOW 2 (WD_ANSW_CNT[1:0] < 3) |
| 1 INCORRECT answer | 2 CORRECT answers | | | | | | |
| 2 INCORRECT answers | 1 CORRECT answer | -New WD cycle starts after the end of RESPONSE WINDOW 2 -Increment WD failure counter -New WD cycle starts with the same WD question | 1b | 0b | 1b | 1b | |
| 0 answers | 3 INCORRECT answers | -New WD cycle starts after the end of RESPONSE WINDOW 2 -Increment WD failure counter -New WD cycle starts with the same WD question | 1b | 0b | 1b | 1b | Less than 3 INCORRECT ANSWER in RESPONSE WINDOW 1 and more than 1 INCORRECT ANSWER in RESPONSE WINDOW 2 (WD_ANSW_CNT[1:0] < 3) |
| 1 INCORRECT answer | 2 INCORRECT answer | | | | | | |
| 2 INCORRECT answer | 1 INCORRECT answer | | | | | | |
| 0 answers | 4 CORRECT answers | -New WD cycle starts after the 4th WD answer -Increment WD failure counter -New WD cycle starts with the same WD question | 0b | 0b | 1b | 0b | Less than 3 INCORRECT ANSWER in RESPONSE WINDOW 1 and more than 1 CORRECT ANSWER in RESPONSE WINDOW 2 (WD_ANSW_CNT[1:0] = 3) |
| 1 INCORRECT answer | 3 CORRECT answers | | | | | | |
| 2 INCORRECT answers | 2 CORRECT answers | | 1b | 0b | 1b | 0b | |
| 0 answers | 4 INCORRECT answers | -New WD cycle starts after the 4th WD answer -Increment WD failure counter -New WD cycle starts with the same WD question | 1b | 0b | 1b | 0b | Less than 3 INCORRECT ANSWER in RESPONSE WINDOW 1 and more than 1 INCORRECT ANSWER in RESPONSE WINDOW 2 (WD_ANSW_CNT[1:0] = 3) |
| 1 INCORRECT answer | 3 INCORRECT answers | | | | | | |
| 2 INCORRECT answers | 2 INCORRECT answers | | | | | | |

Table 7-6. Correct and Incorrect WD Q&A Sequence Run Scenarios (continued)

| NUMBER OF WD ANSWERS | | ACTION | WD STATUS BITS IN WDT_STATUS REGISTER | | | | COMMENTS |
|---|---------------------|--|---------------------------------------|------------|---------|----------|--|
| RESPONSE WINDOW 1 | RESPONSE WINDOW 2 | | ANSW_ERR | ANSW_EARLY | SEQ_ERR | TIME_OUT | |
| 3 CORRECT answers | 0 answers | -New WD cycle starts after the end of RESPONSE WINDOW 2 -Increment WD failure counter -New WD cycle starts with the same WD Question | 0b | 0b | 0b | 1b | Less than 4 CORRECT ANSW in RESPONSE WINDOW 1 and more than 0 ANSWER in RESPONSE WINDOW 2 (WD_ANSW_CNT[1:0] < 3) |
| 2 CORRECT answers | 0 answers | | 0b | 0b | 1b | 1b | |
| 1 CORRECT answers | 0 answers | | 0b | 0b | 0b | 0b | |
| 3 CORRECT answers | 1 CORRECT answer | -New WD cycle starts after the 4th WD answer -Decrement WD failure counter -New WD cycle starts with a new WD question | 0b | 0b | 0b | 0b | CORRECT SEQUENCE |
| 3 CORRECT answers | 1 INCORRECT answers | -New WD cycle starts after the 4th WD answer -Increment WD failure counter -New WD cycle starts with the same WD question | 1b | 0b | 0b | 0b | WD_ANSW_CNT[1:0] = 3 |
| 3 INCORRECT answers | 0 answers | -New WD cycle starts after the end of RESPONSE WINDOW 2 -Increment WD failure counter -New WD cycle starts with the same WD question | 1b | 0b | 0b | 1b | WD_ANSW_CNT[1:0] < 3 |
| 3 INCORRECT answers | 1 CORRECT answer | -New WD cycle starts after the 4th WD answer -Increment WD failure counter -New WD cycle starts with the same WD question | 1b | 0b | 0b | 0b | WD_ANSW_CNT[1:0] = 3 |
| 3 INCORRECT answers | 1 INCORRECT answer | -New WD cycle starts after the 4th WD answer -Increment WD failure counter -New WD cycle starts with the same WD question | 1b | 0b | 0b | 0b | WD_ANSW_CNT[1:0] = 3 |
| 4 CORRECT answers | Not applicable | -New WD cycle starts after the 4th WD answer -Increment WD failure counter -New WD cycle starts with the same WD question | 0b | 1b | 0b | 0b | 4 CORRECT or INCORRECT ANSWER in RESPONSE WINDOW 1 |
| 3 CORRECT answers + 1 INCORRECT answer | Not applicable | -New WD cycle starts after the 4th WD answer -Increment WD failure counter -New WD cycle starts with the same WD question | 1b | 1b | 0b | 0b | |
| 2 CORRECT answers + 2 INCORRECT answers | Not applicable | | | | | | |
| 1 CORRECT answer + 3 INCORRECT answers | Not applicable | | | | | | |

7.3.9 Error Signal Monitor (ESM)

The LP876242-Q1 device has an error signal monitor (ESM), referred to as ESM_MCU throughout this document. This ESM_MCU monitors the MCU error output signal at the nERR_MCU input pin .

At device start-up, the ESM_MCU can be enabled or disabled through configuration bit ESM_MCU_EN. The value for this configuration bit is stored in the NVM memory of the device. To start the enabled ESM_MCU, the MCU sets the start bit ESM_MCU_START through software after the system is powered up and the initial software configuration is completed. If the MCU clears this start bit, the ESM_MCU stops monitoring its input pin. The MCU can set the ENABLE_DRV bit only when the MCU has either started or disabled the ESM_MCU. When the ESM_MCU 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

The ESM_MCU uses a deglitch-filter with deglitch-time $t_{\text{degl_ESMx}}$ to monitor its related input pin.

The MCU can configure the ESM_MCU in two different modes that are defined as follows:

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

To select this mode for the ESM_MCU, the MCU must clear bit ESM_MCU_MODE. See [Section 7.3.9.2](#) for further detail

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

The ESM_MCU has an error-counter (ESM_MCU_ERR_CNT[4:0]), which increments with +2 after each bad-event, and decrements with -1 after each good-event. The ESM_MCU 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. See [Section 7.3.9.3](#) for further details.

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

7.3.9.1 ESM Error-Handling Procedure

The ESM_MCU has two of its own configurable delay-timers that are reset when the device clears the ESM_MCU_START bit. Below steps describe the procedure through which the ESM_MCU goes in case it detects an ESM-error:

1. If the respective mask bit ESM_MCU_PIN_MASK=0, the device sets interrupt bit ESM_MCU_PIN_INT , and pulls the nINT pin low.
2. The ESM starts the delay-1 timer (configurable through related ESM_MCU_DELAY1[7:0] bits).
3. If the ESM-error is no longer present and MCU has cleared the interrupt bit ESM_MCU_PIN_INT before the delay-1 timer elapses, the device releases the nINTpin, the ESM resets the delay-1 and delay-2 timers and continues to monitor its input pin.

4. If the ESM-error is still present, or if MCU has not cleared the interrupt bit ESM_MCU_PIN_INT , and the delay-1 timer elapses, then the ESM clears the ENABLE_DRV bit if bit ESM_MCU_ENDRV=1.
5. If the delay-2 timer (configurable through related ESM_MCU_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. ESM starts the delay-2 timer,
 - b. If ESM_MCU_FAIL_MASK = 0, the device sets interrupt bit ESM_MCU_FAIL_INT and 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 releases the nINT pin, the ESM resets 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)
8. If the ESM-error is still present, or if MCU has not cleared the interrupt bits ESM_MCU_PIN_INT and ESM_MCU_FAIL_INT , and the delay-2 timer elapses, then the device:
 - a. clears the ESM_MCU_START BIT
 - b. sets interrupt bit ESM_MCU_FAIL_INT and ESM_MCU_RST_INT, which the device handles as an ESM_MCU_RST trigger for FSM, described in Summary of Interrupt Signals
 - c. After this trigger handling completes, the device re-initializes the ESM_MCU

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

$$\text{Min. } t_{\text{DELAY-1}} = (\text{ESM_MCU_DELAY1}[7:0] \times 2.048 \text{ ms}) \times 0.95 \quad (11)$$

$$\text{Max. } t_{\text{DELAY-1}} = (\text{ESM_MCU_DELAY1}[7:0] \times 2.048 \text{ ms}) \times 1.05 \quad (12)$$

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

$$\text{Min. } t_{\text{DELAY-2}} = (\text{ESM_MCU_DELAY2}[7:0] \times 2.048 \text{ ms}) \times 0.95 \quad (13)$$

$$\text{Max. } t_{\text{DELAY-2}} = (\text{ESM_MCU_DELAY2}[7:0] \times 2.048 \text{ ms}) \times 1.05 \quad (14)$$

7.3.9.2 Level Mode

In Level Mode, after MCU has set the start bit (bit ESM_MCU_START), the ESM_MCU monitors its nERR_MCU input pin. The ESM_MCU detects an ESM-error when the voltage level on its input pin remains low for a time equal or longer than the deglitch-time $t_{\text{degl_ESMx}}$. When the ESM_MCU detects an ESM-error, it starts the ESM Error-Handling procedure as described in [Section 7.3.9.1](#). The Error-Handling Procedure is stopped if, before elapse of the delay-1 or delay-2 interval, the voltage level on the input pin remains high for a time equal or longer than the deglitch-time $t_{\text{degl_ESMx}}$ and the MCU clears all corresponding interrupt bits. If the ESM-error persists such that the configured delay-1 and delay-2 times elapse, the ESM_MCU sends a ESM_MCU_RST trigger to the PFSM and the device clear the ESM_MCU_START bit. After the PFSM completes the handling of the ESM_MCU_RST trigger, the device re-initializes the ESM_MCU.

For a complete overview on how the ESM_MCU works in Level Mode, please refer to the flow-chart in [Figure 7-24](#). In this flow-chart, the $_x$ stands for $_MCU$. [Figure 7-25](#), [Figure 7-26](#), [Figure 7-27](#), and [Figure 7-28](#) show example wave forms for several error-cases for the ESM_MCU in Level Mode.

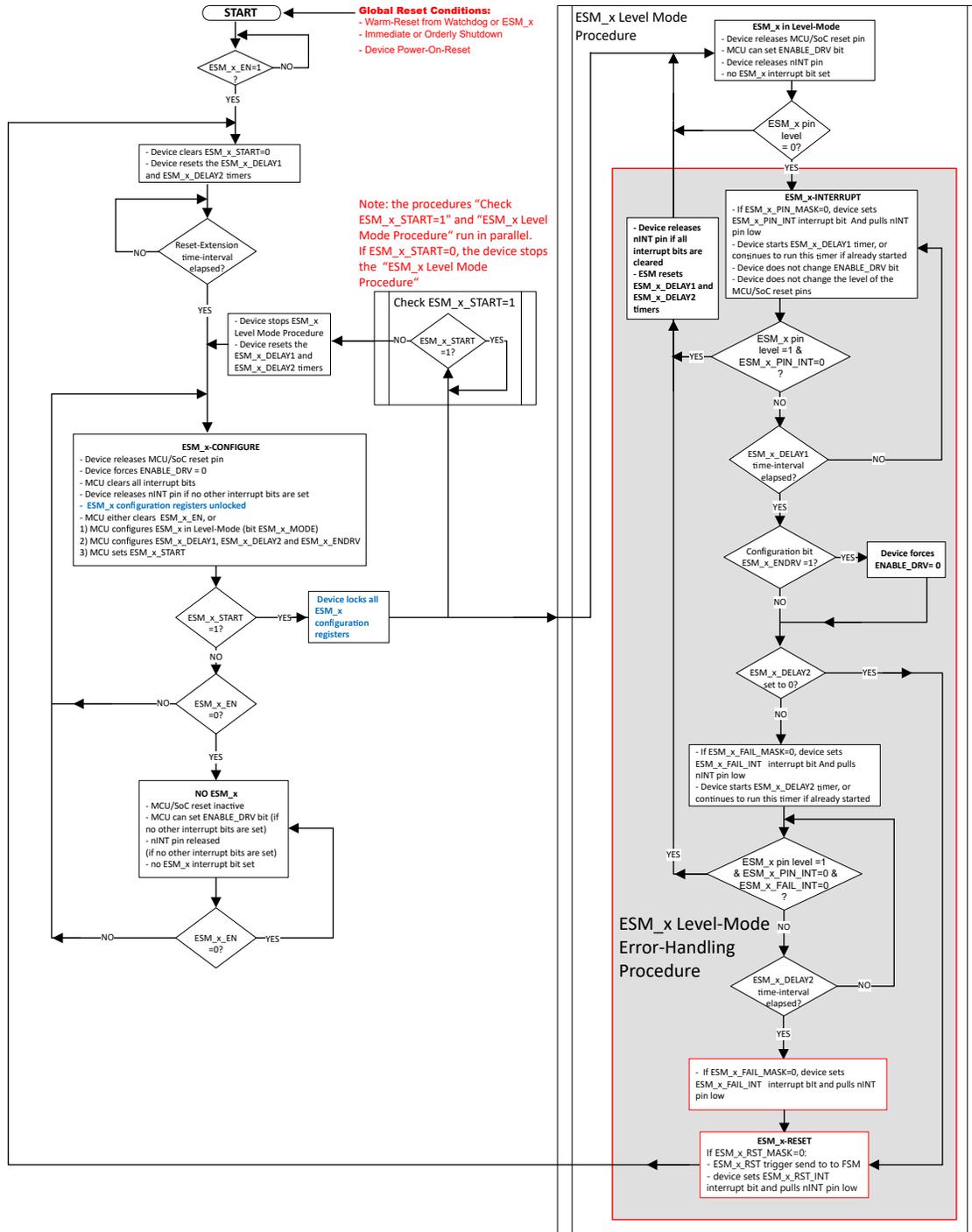
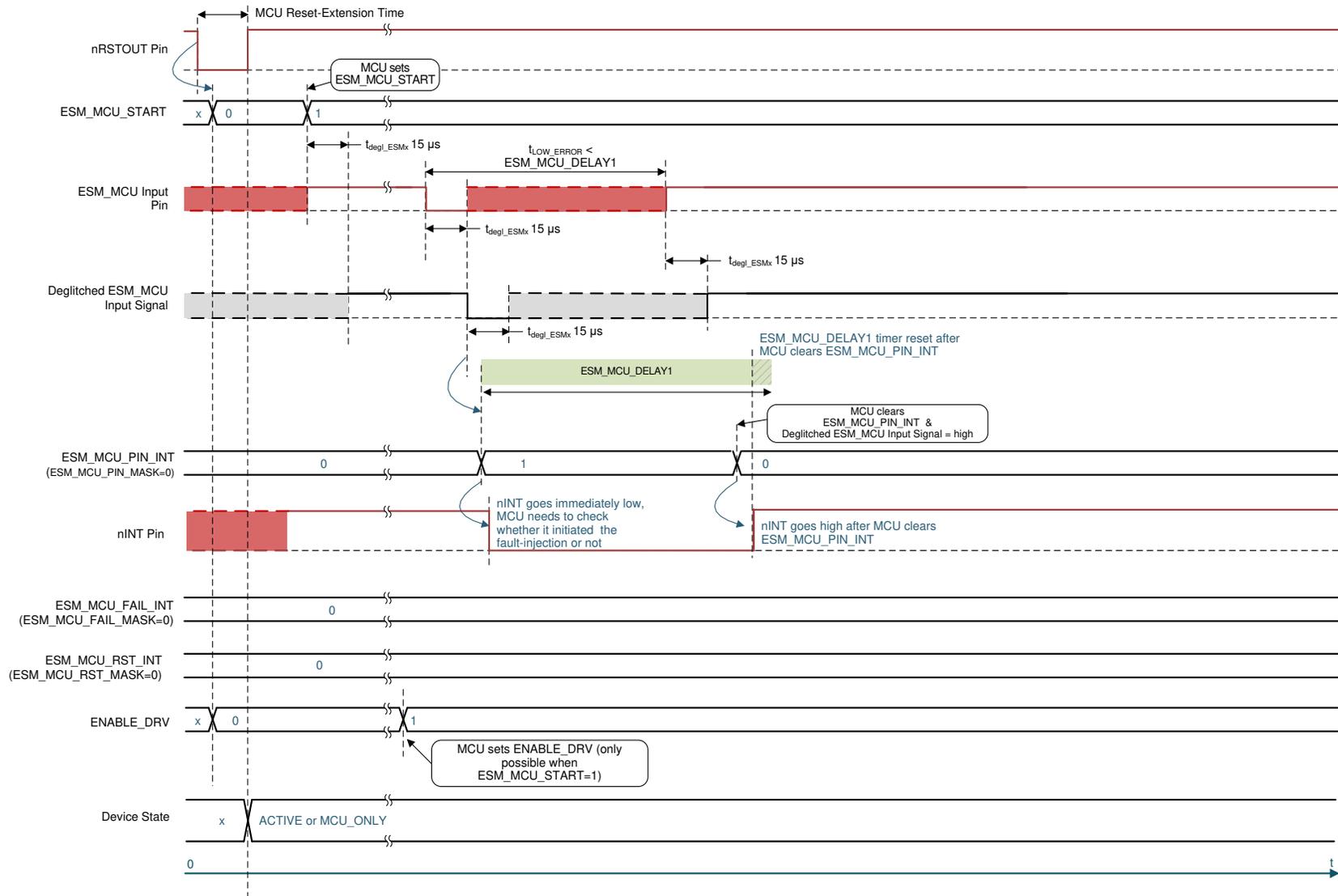
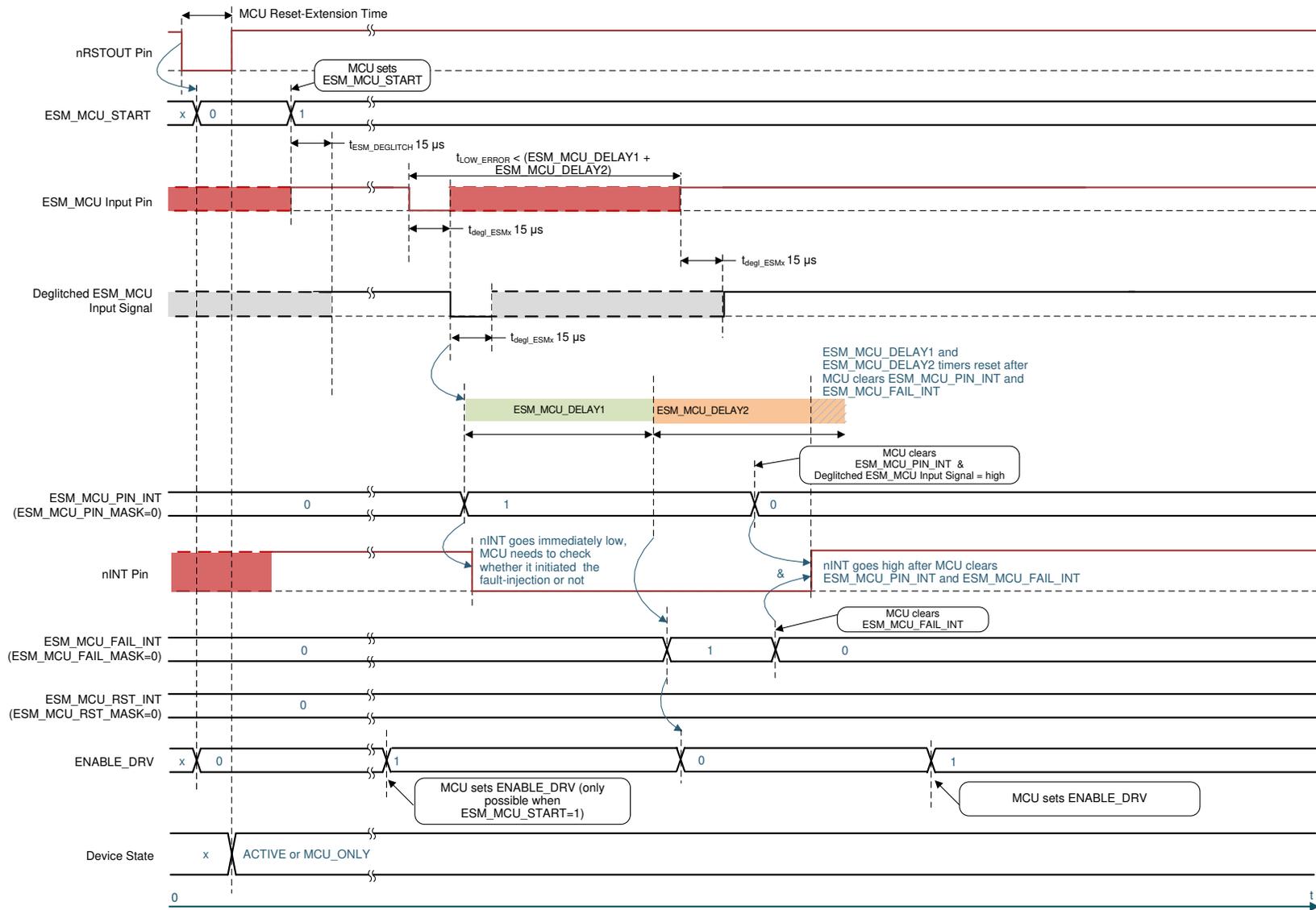


Figure 7-24. Flow Chart for Error Detection in Level Mode



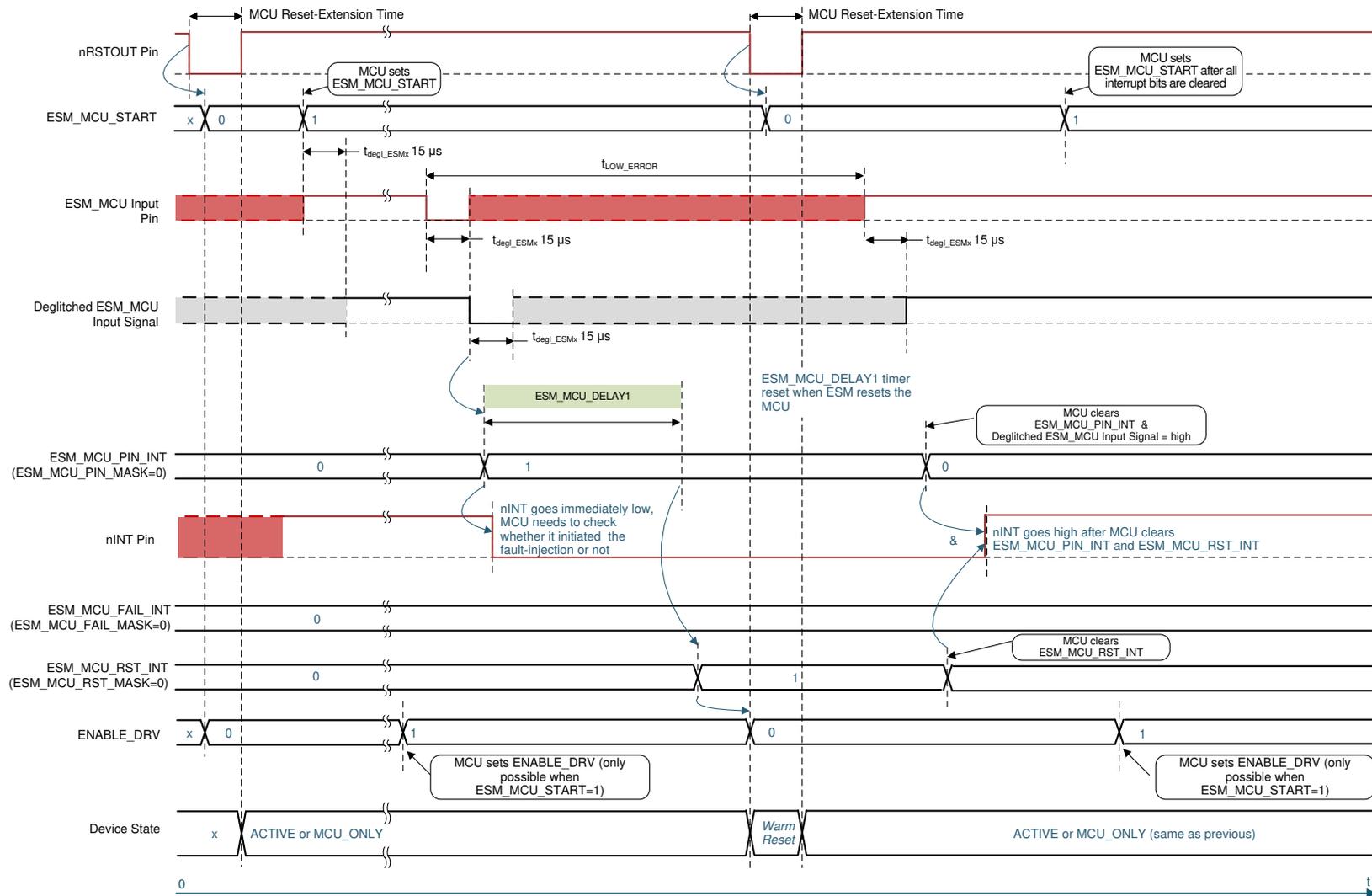
Case Number 1:
MCU initiated a fault-injection, and MCU clears the ESM_MCU_PIN_INT interrupt bit before elapse of ESM_MCU_DELAY1 time-interval

Figure 7-25. Example Waveform for ESM_MCU in Level Mode - Case Number 1: ESM_MCU Signal Recovers Before Elapse of Delay-1 time-interval



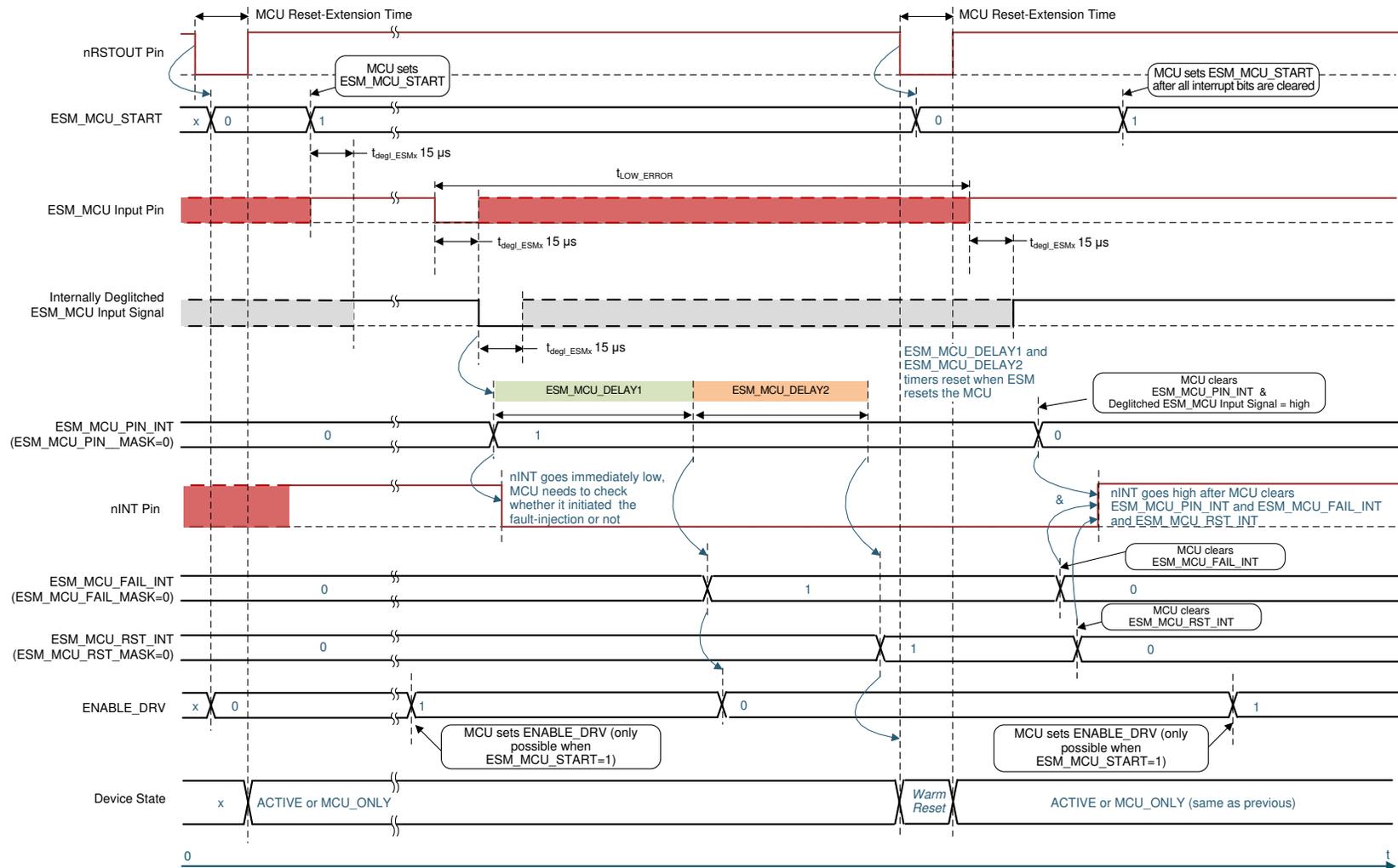
Case Number 2: $ESM_MCU_DELAY2 > 0$
An error event occurred in the MCU, but the MCU recovers and clears the interrupt bits before elapse of the ESM_MCU_DELAY2 time-interval

Figure 7-26. Example Waveform for ESM_MCU in Level Mode – Case Number 2: Delay-2 Not Set To 0 and ESM_MCU_ENDRV=1, ESM_MCU Signal Recovers Before Elapse of Delay-2 Time-Interval



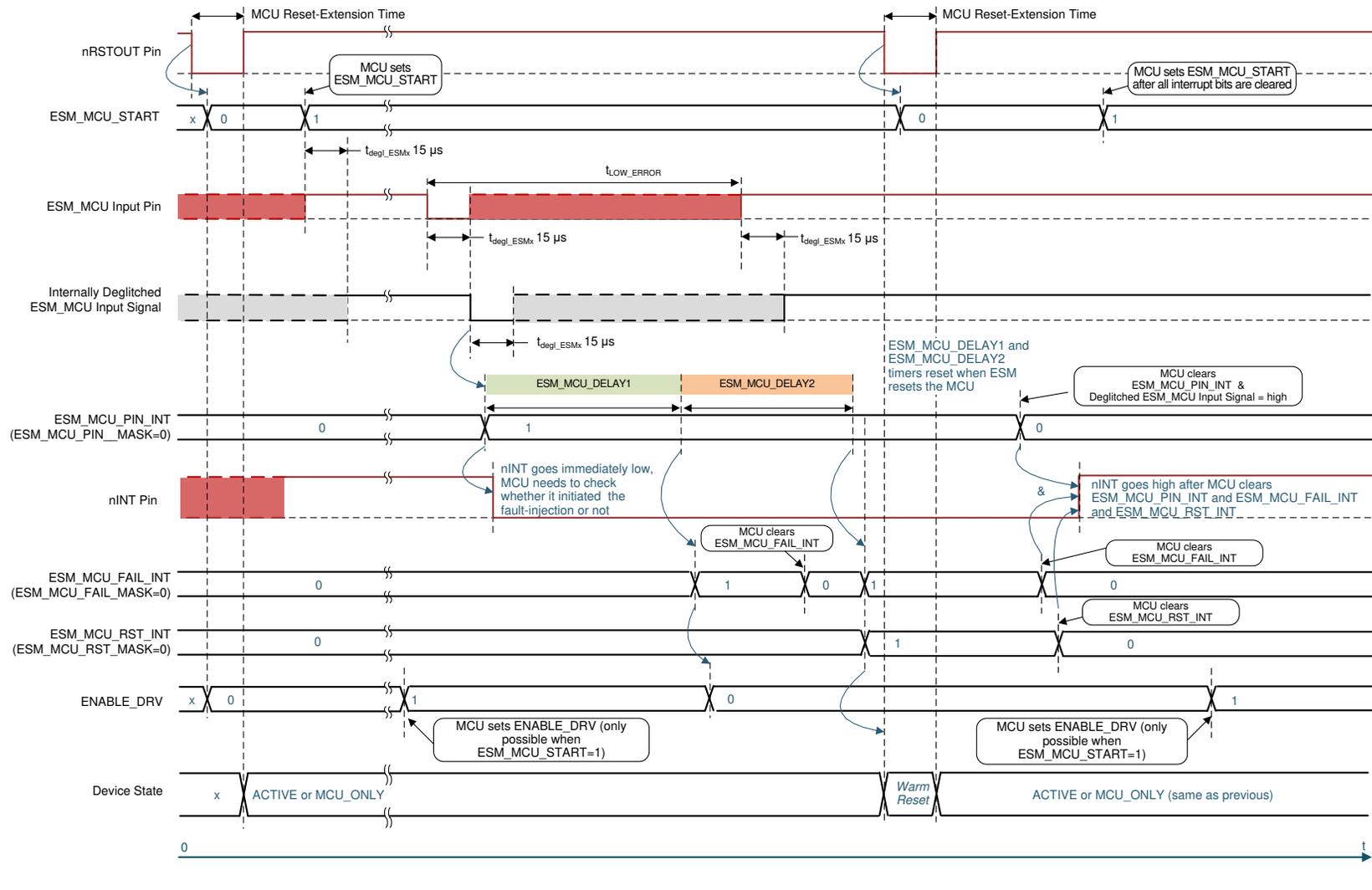
Case Number 3a: ESM_MCU_DELAY2 = 0
An error event occurred in the MCU, and the MCU is unable to correct the error before elapse of the ESM_MCU_DELAY1 time-interval. Hence the PMIC resets the MCU

Figure 7-27. Example Waveform for ESM_MCU 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



Case Number 3b: `ESM_MCU_DELAY2 > 0`
An error event occurred in the MCU, and the MCU is unable to correct the error before elapse of the `ESM_MCU_DELAY1` and `ESM_MCU_DELAY2` time-intervals. Hence the PMIC resets the MCU

Figure 7-28. Example Waveform for ESM_MCU 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



Case Number 3c: **ESM_MCU_DELAY2 > 0**
An error event occurred in the MCU, the MCU recovers and clears ESM_MCU_FAIL_INT, but fails to clear ESM_MCU_PIN_INT before elapse of the ESM_MCU_DELAY2 time-interval. Hence the PMIC resets the MCU and sets ESM_FAIL_INT (if ESM_MCU_FAIL_MASK=0).

Figure 7-29. Example Waveform for ESM_MCU in Level Mode – Case Number 3c: Delay-2 Not Set to 0 and ESM_MCU_ENDRV=1, MCU Fails to Clear ESM_MCU_PIN_INT Before Elapse of the ESM_MCU_DELAY2

7.3.9.3 PWM Mode

7.3.9.3.1 Good-Events and Bad-Events

In PWM mode, the ESM_MCU monitors the high-pulse and low-pulse duration times its PWM input signal as follows:

- After a falling edge, the ESM_MCU 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_MCU detects a bad event and the low-pulse duration counter reinitializes. Each time the signal further exceeds the maximum threshold, the ESM_MCU 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_MCU 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_MCU detects a bad event and the high-pulse duration counter reinitializes. Each time the signal further exceeds the maximum threshold, the ESM_MCU detects a bad event. On the next falling edge on the input signal, the ESM_MCU starts the low-pulse time-duration monitoring.

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

- A high-pulse time-duration that is longer than the maximum high-pulse time-threshold ($t_{HIGH_MAX_TH}$) that is configured in register bits ESM_MCU_HMAX[7:0].
- A high-pulse time-duration that is shorter than the minimum high-pulse time-threshold ($t_{HIGH_MIN_TH}$) that is configured in register bits ESM_MCU_HMIN[7:0].
- A low-pulse time-duration that is longer than the maximum low-pulse time-threshold ($t_{LOW_MAX_TH}$) that is configured in register bits ESM_MCU_LMAX[7:0].
- A low-pulse time-duration that is less than the minimum low-pulse time-threshold ($t_{LOW_MIN_TH}$) that is configured in register bits ESM_MCU_LMIN[7:0].

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

- 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

Register bits ESM_MCU_HMAX[7:0] set the maximum high-pulse time-threshold ($t_{HIGH_MAX_TH}$) for the ESM_MCU. Use [Equation 15](#) and [Equation 16](#) to calculate the worst-case values for the $t_{HIGH_MAX_TH}$:

$$\text{Min. } t_{HIGH_MAX_TH} = (15 \mu\text{s} + \text{ESM_MCU_HMAX}[7:0] \times 15 \mu\text{s}) \times 0.95 \quad (15)$$

$$\text{Max. } t_{HIGH_MAX_TH} = (15 \mu\text{s} + \text{ESM_MCU_HMAX}[7:0] \times 15 \mu\text{s}) \times 1.05 \quad (16)$$

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

$$\text{Min. } t_{HIGH_MIN_TH} = (15 \mu\text{s} + \text{ESM_MCU_HMIN}[7:0] \times 15 \mu\text{s}) \times 0.95 \quad (17)$$

$$\text{Max. } t_{HIGH_MIN_TH} = (15 \mu\text{s} + \text{ESM_MCU_HMIN}[7:0] \times 15 \mu\text{s}) \times 1.05 \quad (18)$$

ESM_MCU_LMAX[7:0] set the maximum low-pulse time-threshold ($t_{LOW_MAX_TH}$) for the ESM_MCU. Use [Equation 19](#) and [Equation 20](#) to calculate the worst-case values for the $t_{LOW_MAX_TH}$:

$$\text{Min. } t_{LOW_MAX_TH} = (15 \mu\text{s} + \text{ESM_MCU_LMAX}[7:0] \times 15 \mu\text{s}) \times 0.95 \quad (19)$$

$$\text{Max. } t_{LOW_MAX_TH} = (15 \mu\text{s} + \text{ESM_MCU_LMAX}[7:0] \times 15 \mu\text{s}) \times 1.05 \quad (20)$$

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

$$\text{Min. } t_{\text{LOW_MIN_TH}} = (15 \mu\text{s} + \text{ESM_MCU_LMIN}[7:0] \times 15 \mu\text{s}) \times 0.95 \quad (21)$$

$$\text{Max. } t_{\text{LOW_MIN_TH}} = (15 \mu\text{s} + \text{ESM_MCU_LMIN}[7:0] \times 15 \mu\text{s}) \times 1.05 \quad (22)$$

Please note that when setting up the minimum and the maximum low-pulse or high-pulse time-thresholds need to be configured such that clock tolerances from the LP876242-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:

$$\text{ESM_MCU_HMIN}[7:0] < 0.5 \times (\text{ESM_MCU_HMAX}[7:0] + \text{ESM_MCU_HMIN}[7:0]) \times 0.95 \times (1 - \text{MCUclock tolerance}) \quad (23)$$

$$\text{ESM_MCU_HMAX}[7:0] > 0.5 \times (\text{ESM_MCU_HMAX}[7:0] + \text{ESM_MCU_HMIN}[7:0]) \times 1.05 \times (1 + \text{MCU clock tolerance}) \quad (24)$$

$$\text{ESM_MCU_LMIN}[7:0] < 0.5 \times (\text{ESM_MCU_LMAX}[7:0] + \text{ESM_MCU_LMIN}[7:0]) \times 0.95 \times (1 - \text{MCUclock tolerance}) \quad (25)$$

$$\text{ESM_MCU_LMAX}[7:0] > 0.5 \times (\text{ESM_MCU_LMAX}[7:0] + \text{ESM_MCU_LMIN}[7:0]) \times 1.05 \times (1 + \text{MCU clock tolerance}) \quad (26)$$

7.3.9.3.2 ESM Error-Counter

If the ESM_MCU detects a bad-event, it increments its error-counter (bits ESM_MCU_ERR_CNT[4:0]) by 2. If the ESM_MCU detects a good-event, it decrements its error-counter (bits ESM_MCU_ERR_CNT[4:0]) by 1.

The device clears the ESM_MCU error counter when ESM_MCU_START=0.

The ESM_MCU error-counter has a threshold (bits ESM_MCU_ERR_CNT_TH[3:0]) that the MCU can configure if the ESM_MCU_START bit is 0. If the ESM_MCU error-counter value is above its configured threshold, the ESM_MCU has detected a so-called ESM-error and starts the Error-Handling Procedure as described in [Section 7.3.9.1](#). If the ESM_MCU error-counter reached a value equal or less its configured threshold before the elapse of the configured delay-1 or delay-2 intervals and the MCU software clears all ESM_MCU related interrupt bits, the ESM-error is no longer present and the ESM_MCU stops the Error-Handling Procedure as described in [Section 7.3.9.1](#). If the ESM-error persists such that the configured delay-1 and delay-2 times elapse, the ESM_MCU sends a ESM_MCU_RST trigger to the PFSM and the device clears the ESM_MCU_START bit. After the PFSM completes the handling of the ESM_MCU_RST trigger, the device re-initializes the ESM_MCU.

7.3.9.3.2.1 ESM Start-Up in PWM Mode

After the MCU has set the start bit of ESM (bit ESM_MCU_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 ESM starts the following procedure:
 - a. Start a timer with a time-length according the value configured in ESM_MCU_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 7.3.9.3.1](#). [Figure 7-31](#) shows an example this scenario as Case Number 1.
 - d. If the configured time-length (configured in ESM_MCU_LMAX[7:0]) elapses, the ESM detects a bad-event and increments the error-counter with +2. Hereafter, the ESM detects good-events or bad-events as described in [Section 7.3.9.3.1](#). [Figure 7-33](#) shows an example this scenario as Case Number 3.
 - e. If the ESM error-counter value is above its configured threshold, the ESM has detected a so-called ESM-error and starts the Error-Handling Procedure as described in [Section 7.3.9.3.1](#).
 - f. During this Error-Handling Procedure, the ESM continues to monitor its 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 ESM sends an ESM_RST trigger to the PFSM, which, depending on the PFSM configuration, resets the MCU . [Figure 7-34](#) shows a scenario in which the device resets the MCU as Case Number 4.
 - g. If the ESM error-counter reaches a value equal or less its configured threshold before the elapse of the configured delay-1 or delay-2 time-intervals and the MCU software clears all ESM related interrupt bits, the ESM-error is no longer present and the ESM stops the Error-Handling Procedure as described in [Section 7.3.9.3.1](#).

2. The deglitched signal monitored input pin has a high level at the moment the MCU sets the start bit. In this scenario, the ESM starts the following procedure:
 - a. Start a timer with a time-length according the value configured in ESM_MCU_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 7.3.9.3.1](#). [Figure 7-32](#) shows an example this scenario as Case Number 2.
 - d. If the configured time-length (configured in ESM_MCU_HMAX[7:0]) elapses, the ESM detects a bad-event and increments the error-counter with +2. Hereafter, the ESM detects good-events or bad-events as described in [Section 7.3.9.3.1](#).
 - e. If the ESM error-counter value is above its configured threshold, the ESM has detected a so-called ESM-error and starts the Error-Handling Procedure as described in [Section 7.3.9.3.1](#).
 - f. During this Error-Handling Procedure, the ESM continues to monitor its 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 ESM sends an ESM_RST trigger to the PFSM, which, depending on the PFSM configuration, resets the MCU , as Case Number 4.
 - g. If the ESM error-counter reaches a value equal or less its configured threshold before the elapse of the configured delay-1 or delay-2 time-intervals and the MCU software clears all ESM related interrupt bits, the ESM-error is no longer present and the ESM stops the Error-Handling Procedure as described in [Section 7.3.9.3.1](#).

7.3.9.3.3 ESM Flow Chart and Timing Diagrams in PWM Mode

For a complete overview on how the ESM_MCU works in PWM Mode, please refer to the flow-chart in [Figure 7-30](#). In this flow-chart, the `_x` stands for `_MCU` [Figure 7-31](#), [Figure 7-32](#), [Figure 7-33](#), and [Figure 7-34](#) show example waveforms for several error-cases for the ESM_MCU in PWM Mode. In this flow-chart, the `_x` stands for `_MCU`

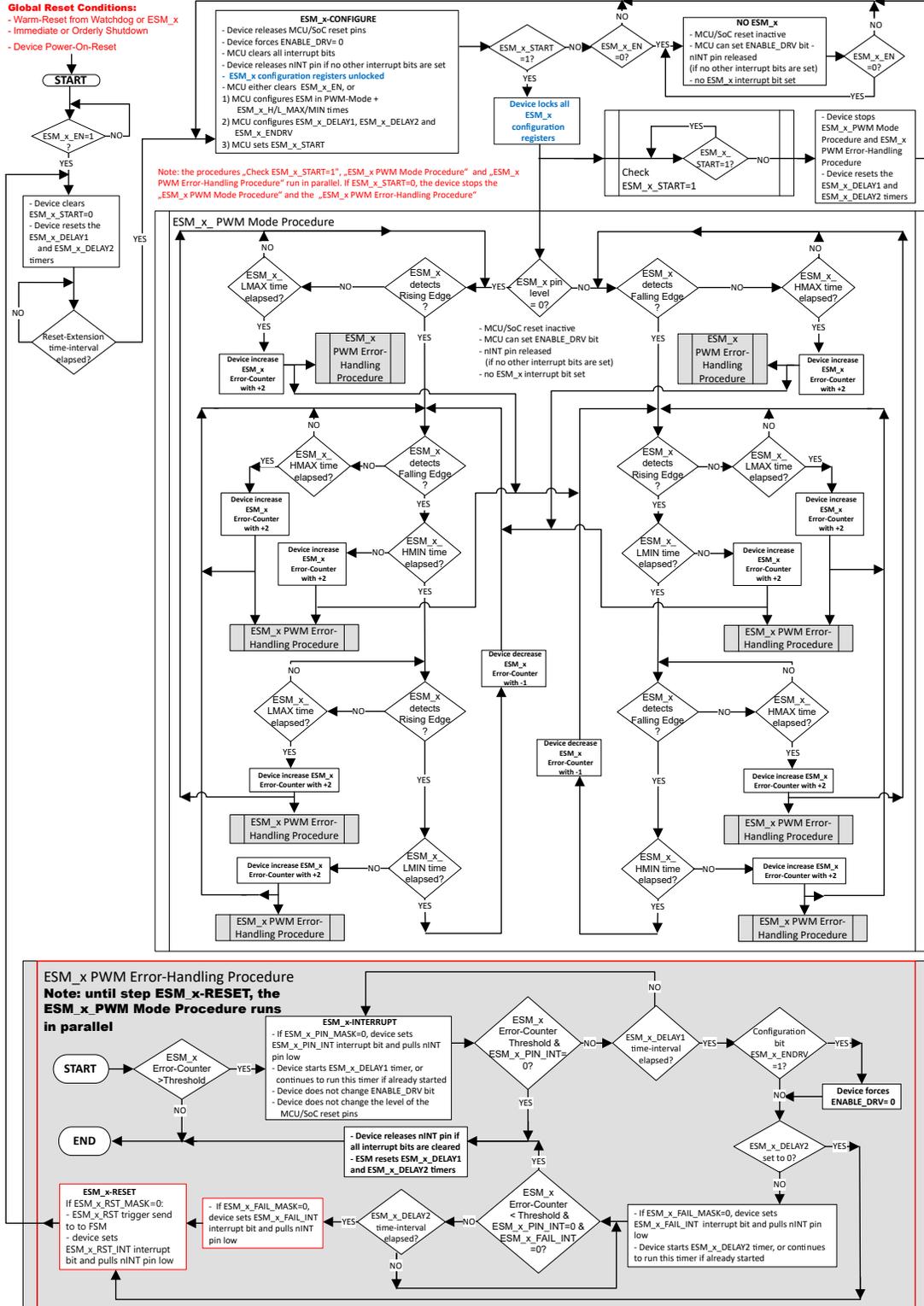


Figure 7-30. Flow-Chart for ESM_MCU in PWM Mode

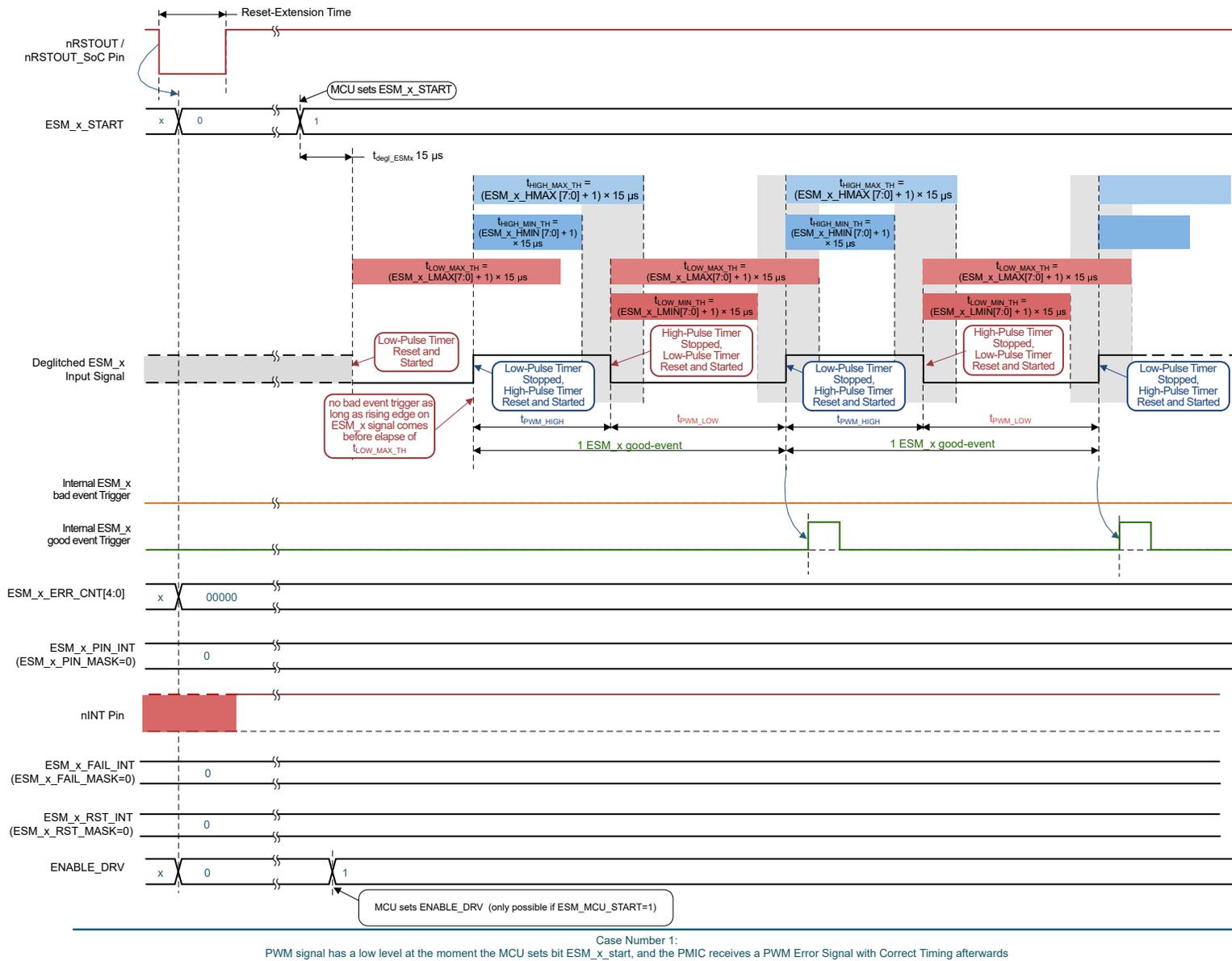


Figure 7-31. Example Waveform for ESM_MCU in PWM Mode – Case Number 1: ESM_MCU Starts with Low-Level at Deglitched Input signal, and Receives Correct PWM Signal Afterwards. (The _x stand for _MCU)

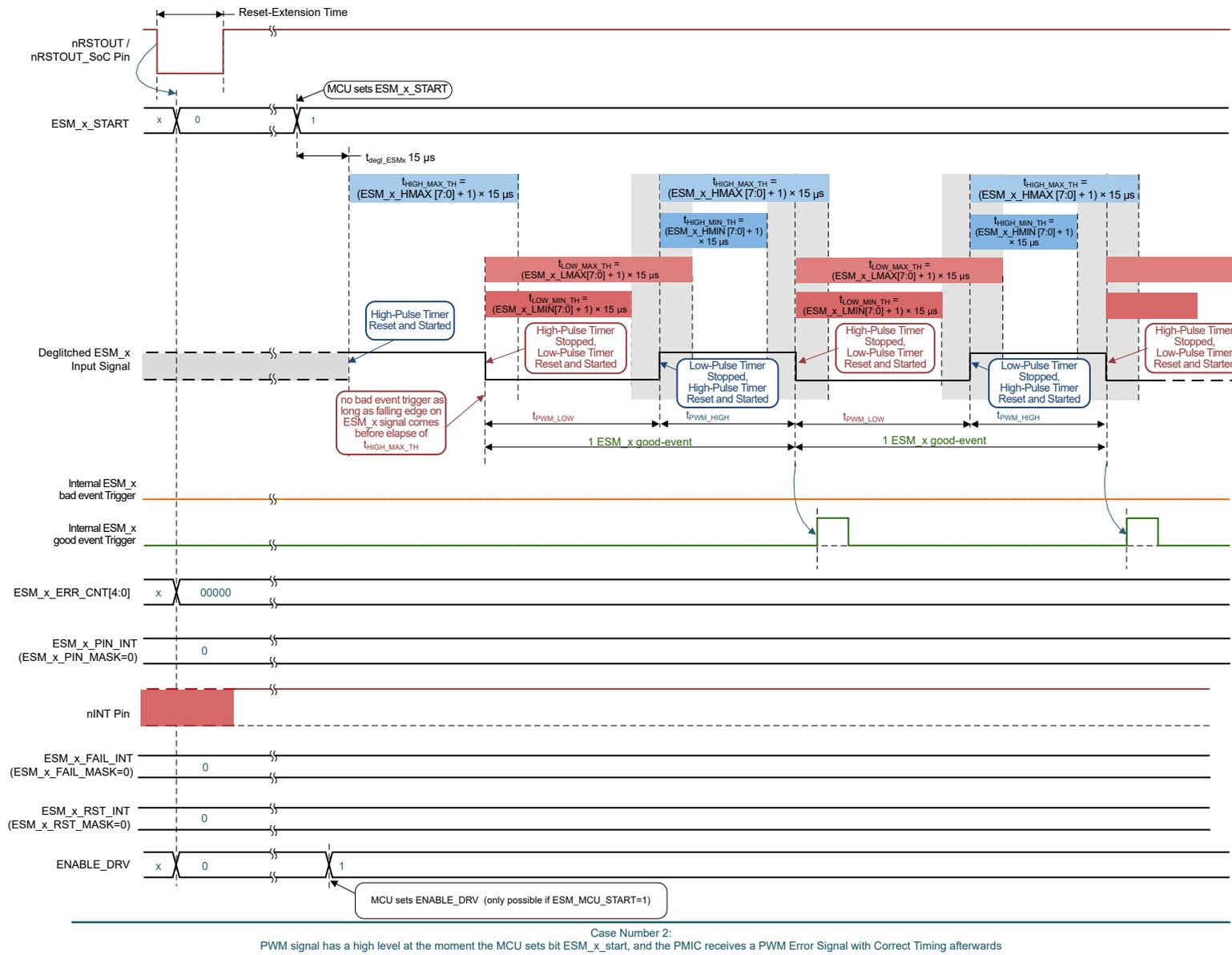
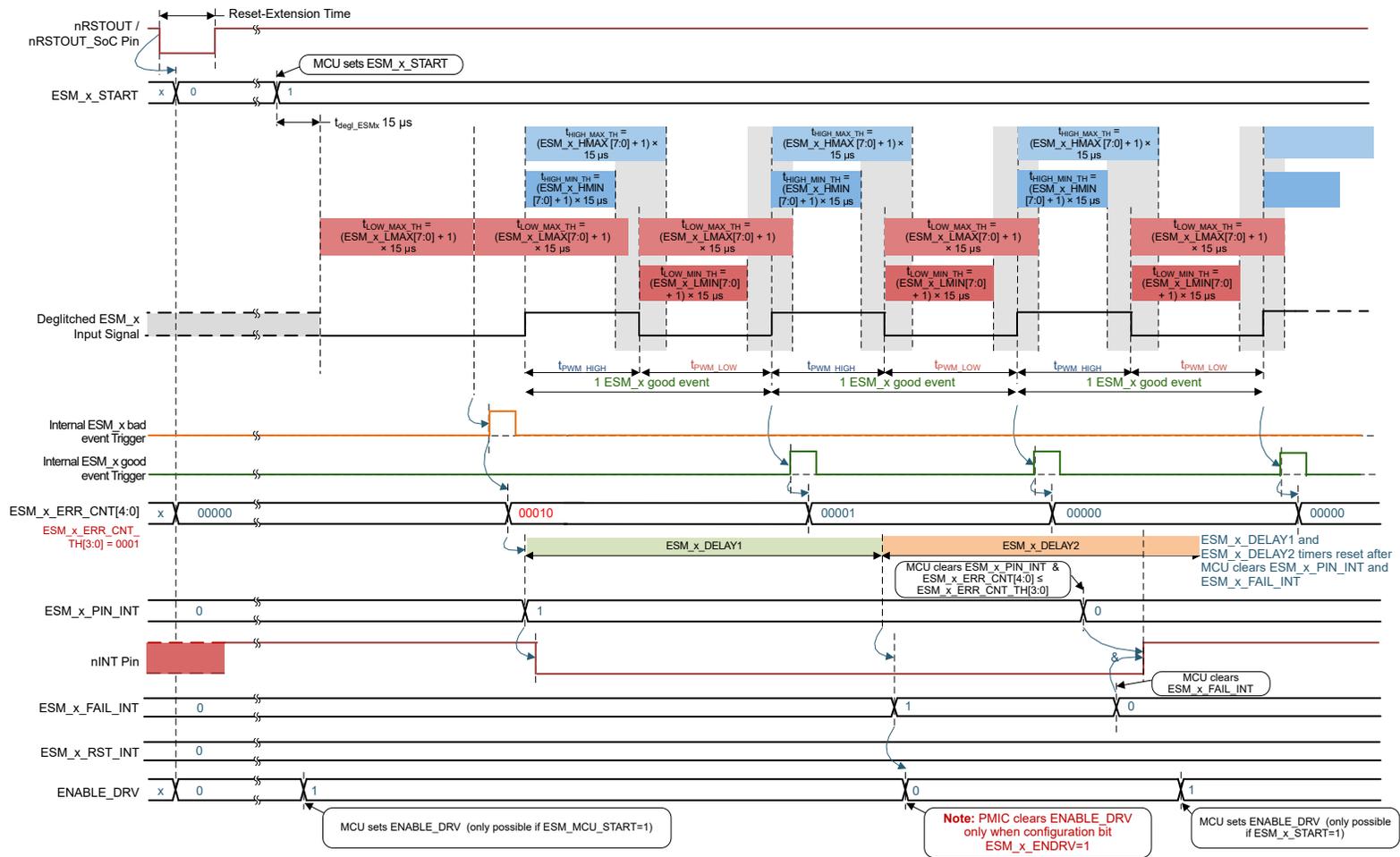
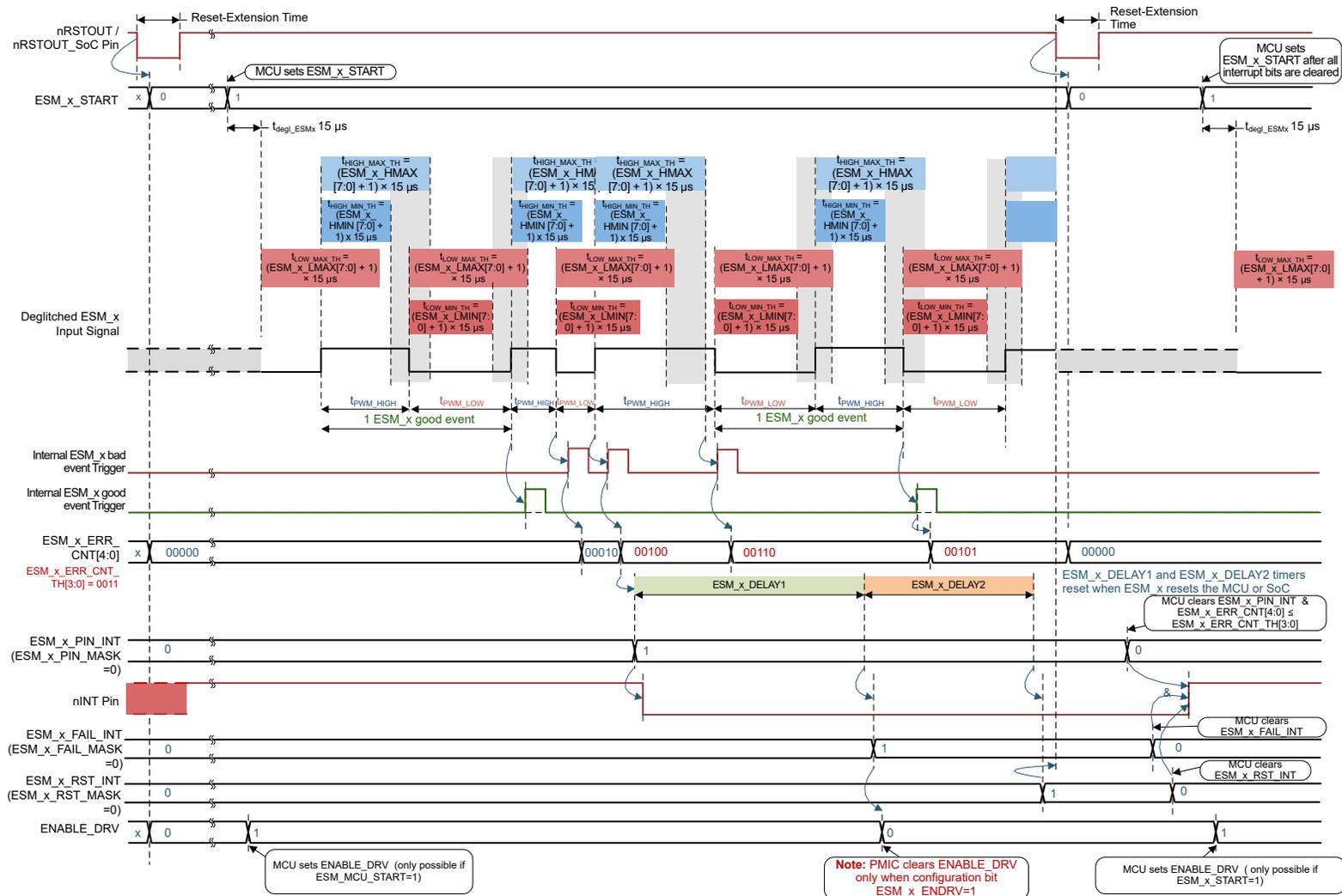


Figure 7-32. Example Waveform for ESM_MCU in PWM Mode – Case Number 2: ESM_MCU Starts with High-Level at Deglitched Input Signal, and Receives Correct PWM Signal Afterwards (The _x stand for _MCU)



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 7-33. Example Waveform for ESM in PWM Mode – Case Number 3: ESM_MCU Starts with Low-Level at Deglitched Input Signal, but Receives Too Late a Correct PWM Signal Afterwards (The _x stand for _MCU)



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

Figure 7-34. Example Waveform for ESM_MCU in PWM Mode – Case Number 4: ESM_MCU Starts with Low-Level at Deglitched Input Signal and Receives a Correct PWM Signal. Afterwards the ESM_MCU Detects Bad Events, and the PWM Signal Recovers Too Late which Leads to an ESM_MCU Reset Trigger to the PFSM (The _x stand for _MCU)

7.4 Device Functional Modes

7.4.1 Device State Machine

The LP876242-Q1 device integrates a finite state machine (FSM) engine that manages the state of the device during operating state transitions. The device supports NVM-configurable mission states with configurable input triggers for transitions between states. Any resources, including the 4 BUCK regulators, the two VMON voltage monitors and all of the digital IO pins including the 10 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 (I²C or SPI), or the FSM can automatically control the resource during state sequences.

The FSM is powered by an internal LDO that 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 that control the operational modes of the LP876242-Q1 device:

- Fixed Device Power Finite State Machine (FFSM)
- Pre-configurable Finite State Machine (PFSM) for Mission States (ACTIVE, MCU_ONLY, S2R, DEEP_SLEEP)
- Error Handling Operations

The PFSM provides configurable rail and voltage monitoring 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.

7.4.1.1 Fixed Device Power FSM

The Fixed Device Power portion of the FSM engine manages the power up of the device before the power rails are fully enabled and ready to power external loadings, and the power down of the device when in the event of insufficient power supply or device or 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. |
| LP_STANDBY | The device can enter this state from a mission state after receiving a valid OFF request by ENABLE pin or I2C trigger and the LP_STANDBY_SEL = 1. The internal LDO (LDOVINT) is enabled and VCCA monitoring is disabled to minimize power dissipation. As the accurate VCCA monitoring is disabled, the VCCA voltage must be above 1.7 V during LP_STANDBY state, or stay below 1.7 V for minimum 20 ms to ensure that the digital is reset correctly. If this requirement for VCCA cannot be met, STANDBY state must be used instead of LP_STANDBY. The wake-up from LP_STANDBY state can be initiated by active edge on ENABLE signal or by WKUPx pins. |
| INIT | The device is powered by a valid supply on the system power rail ($VCCA \geq VCCA_{UV}$). If the device was previously in LP_STANDBY state, it has received an external wake-up signal at the WKUP1/2 pins, or an On Request from the 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 accordingly. |
| BOOT BIST | The device is running the built-in self-test routine that includes both the ABIST, the LBIST/ and the CRC. 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 wakes up from the LP STANDBY state. When the device arrives at this state from the SAFE_RECOVERY state, LBIST is automatically skipped if it has not previously failed. If LBIST failed, but passed after multiple re-tries before exceeding the recovery counter limit, the device powers up normally. The following NVM bits are pre-configured options that can be set to disable parts of the ABIST or CRC tests if further sequence time reduction is required (please refer to the User's Guide of the orderable part number):

- 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

Note

Note: the BIST tests are executed as parallel processes, and the longest process determines the total BIST duration

RUNTIME BIST A request was received from the MCU to exercise a run-time built-in self-test (RUNTIME_BIST) on the device. No rails are modified and all external signals, including all I²C or SPI interface communications, are ignored during BIST. If the device passed BIST, it resumes the previous operation. If the device failed BIST, it shuts down all of the regulator outputs and proceed to the SAFE RECOVERY state. In order to avoid a register CRC error, all register writes must be avoided after the request for the BIST operation until the device pulls the nINT pin low to indicate the completion of BIST. The results of the BIST are indicated by the BIST_PASS_INT or the BIST_FAIL_INT bits.

Note

For executing the RUNTIME_BIST, the system software must perform following steps:

Before RUNTIME_BIST request:

- 1) Set VCCA_UV_MASK, VCCA_OV_MASK, all BUCKx_UV_MASK, all BUCKx_OV_MASK, VMONx_UV_MASK and VMONx_OV_MASK bits to 1
- 2) If PGOOD pin is used, clear all PGOOD_SEL_BUCKx and PGOOD_SEL_VMONx bits to 0

After completion of RUNTIME_BIST:

- 1) Clear VCCA_UV_MASK, VCCA_OV_MASK, all VMONx_UV_MASK, all VMONx_OV_MASK, all BUCKx_UV_MASK and all BUCKx_OV_MASK bits to 0
-

**SAFE
RECOVERY**

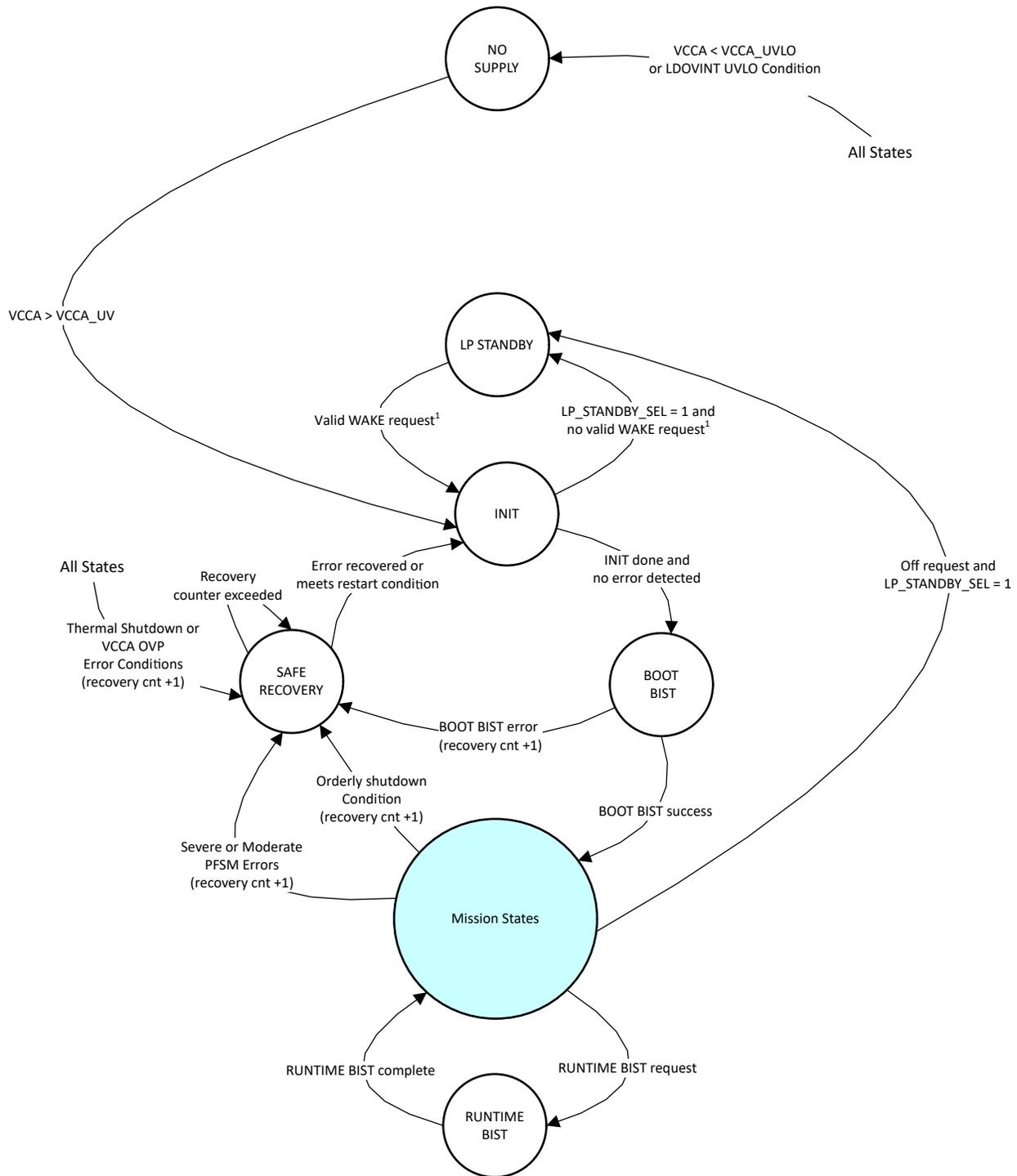
The device meets the qualified error condition for immediate or ordered shutdown request. If the error is recovered within the recovery time interval or meets the restart condition, the device increments the recovery counter, and returns to INIT state if the recovery counter value does not exceed the threshold value. Until a supply power cycle occurs, the device stays in the SAFE RECOVERY state if one of the following conditions occur:

- the recovery counter exceeds the threshold value
- the die temperature cannot be reduced to less than T_{WARN} level
- VCCA stays above OVP threshold

When multiple system conditions occur simultaneously that demand power state arbitration, the device goes to the higher priority state according to the following priority order:

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

[Figure 7-35](#) shows the power transition states of the FSM engine.



¹ A valid WAKE request consist of:

- ENABLE high level if the device arrived the LP_STANDBY state through a low level at the ENABLE pin, or
- LWKUP1 or WKUP2 detection if the device arrived the LP_STANDBY state through writing to a TRIGGER_I2C_0 bit, or
- SPMI wake-up event

Figure 7-35. State Diagram for Device Power States

7.4.1.1.1 Register Resets and EEPROM read at INIT state

When the device transitions from LP_STANDBY to INIT state, the registers are reset and EEPROM is read based on FIRST_STARTUP_DONE and SKIP_LP_STANDBY_EE_READ bits. At the transition from SAFE RECOVERY to INIT, the SKIP_LP_STANDBY_EE_READ bit is ignored as shown in the table.

Table 7-7. Register resets and EEPROM read at INIT state

| State transition | FIRST_STARTUP_DONE | SKIP_LP_STANDBY_EE_READ | Conf_registers | Other registers |
|----------------------|--------------------|-------------------------|-------------------------------------|-------------------------------------|
| LP_STANDBY → INIT | Don't care | 1 | No changes | No changes |
| LP_STANDBY → INIT | 1 | 0 | No changes | Reset and defaults read from EEPROM |
| LP_STANDBY → INIT | 0 | 0 | Reset and defaults read from EEPROM | Reset and defaults read from EEPROM |
| SAFE RECOVERY → INIT | 1 | Don't care | No changes | Reset and defaults read from EEPROM |
| SAFE RECOVERY → INIT | 0 | Don't care | Reset and defaults read from EEPROM | Reset and defaults read from EEPROM |

The Conf_registers are:

- ENABLE_POL bit in the ENABLE_CONF register
- FSD_MASK, ENABLE_MASK bits in the MASK_STARTUP register
- FIRST_STARTUP_DONE, STARTUP_DEST, FAST_BIST, LP_STANDBY_SEL, and SKIP_LP_STANDBY_EE_READ bits in the STARTUP_CTRL register
- SCRATCH_PAD_x bits in the SCRATCH_PAD_REG_x registers
- PFSM_DELAYx bits in the PFSM_DELAY_REG_x registers

7.4.1.2 Pre-Configurable Mission States

When the device arrives at 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 that together form the configurable sub state machine within the scope of mission states. This sub state machine can 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. 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. Unlike the user registers, the PFSM definition stored in the NVM cannot be modified during normal operation. When the PMIC determines that a transition to another operation state is necessary, it reads the configuration memory to determine what sequencing is needed for the state transition.

Table 7-11 shows how the trigger signals for each state transition can come from a variety of interface or GPIO inputs, or potential error sources. Figure 7-36 shows how the device processes all of the possible error sources inside the PFSM engine, a hierarchical mask system is applied to filter out the common errors that can be handled by interrupt only, and categorize the other error sources as Severe Global Error, Moderate Global Error, and so forth. The filtered and categorized triggers are sent into the PFSM engine, that then determines the entry and exit condition for each configured mission state.

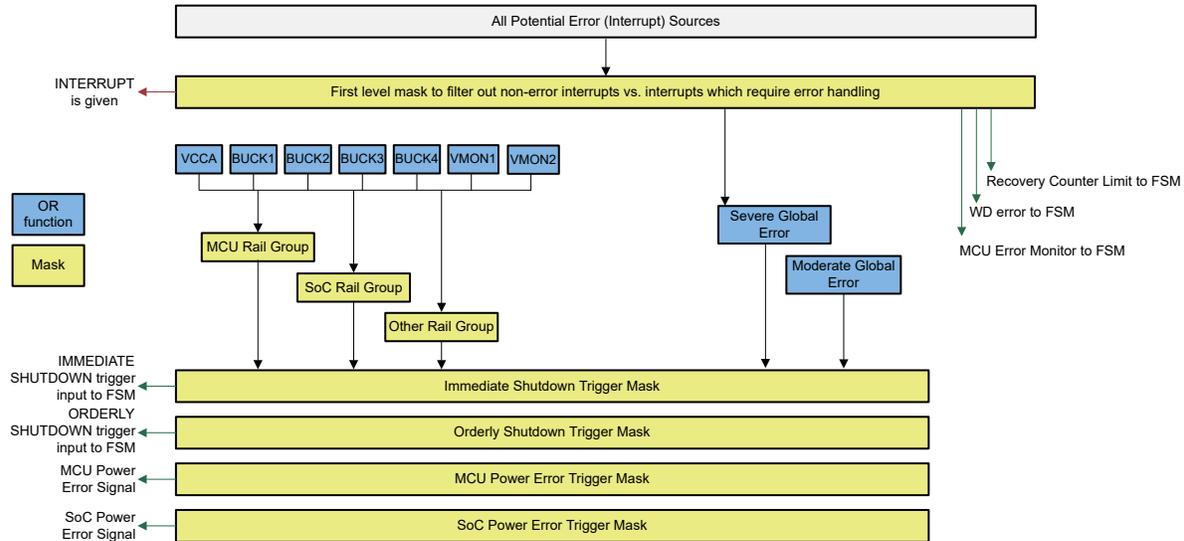


Figure 7-36. Error Source Hierarchical Mask System

Figure 7-38 shows an example of how the PFSM engine utilizes instructions to execute the configured device state and sequence transitions of the mission state-machine. Table 7-8 provides the instruction set and usage description of each instruction in the following sections. Section 7.4.1.2.2 describes how the instructions are stored in the NVM memory.

Table 7-8. PFSM Instruction set

| Command Opcode | Command | Command Description |
|----------------|--------------------------|--|
| "0000" | REG_WRITE_MASK_PAGE0_IMM | Write the specified data, except the masked bits, to the specified page 0 register address. |
| "0001" | REG_WRITE_IMM | Write the specified data to the specified register address. |
| "0010" | REG_WRITE_MASK_IMM | Write the specified data, except the masked bits, to the specified register address. |
| "0011" | REG_WRITE_VOUT_IMM | Write the target voltage of a specified regulator after a specified delay. |
| "0100" | REG_WRITE_VCTRL_IMM | Write the operation mode of a specified regulator after a specified delay. |
| "0101" | REG_WRITE_MASK_SREG | Write the data from a scratch register, except the masked bits, to the specified register address. |
| "0110" | SREG_READ_REG | Write scratch register (REG0-3) with data from a specified address. |
| "0111" | WAIT | Execution is paused until the specified type of the condition is met or timed out. |
| "1000" | DELAY_IMM | Delay the execution by a specified time. |
| "1001" | DELAY_SREG | Delay the execution by a time value stored in the specified scratch register. |
| "1010" | TRIG_SET | Set a trigger destination address for a given input signal or condition. |
| "1011" | TRIG_MASK | Sets a trigger mask that determines which triggers are active. |
| "1100" | END | Mark the final instruction in a sequential task. |
| "1101" | REG_WRITE_BIT_PAGE0_IMM | Write the specified data to the BIT_SEL location of the specified page 0 register address. |
| "1110" | REG_WRITE_WIN_PAGE0_IMM | Write the specified data to the SHIFT location of the specified page 0 register address. |
| "1111" | SREG_WRITE_IMM | Write the specified data to the scratch register (REG0-3). |

7.4.1.2.1 PFSM Commands

Following section describes each PFSM command in detail and provides example usage codes. More information on example NVM configuration, available device options and documentations can be found at [Fully Customizable Integrated Power](#).

7.4.1.2.1.1 REG_WRITE_IMM Command

Description: Write the specified data to the specified register address

Assembly command: **REG_WRITE_IMM [ADDR=]<Address> [DATA=]<Data>**

Address and Data can be in any literal integer format (decimal, hex, and so forth).

'ADDR=' and 'DATA=' are optional. When included, the parameters can be in any order.

Examples:

- **REG_WRITE_IMM 0x1D 0x55** — Write value 0x55 to address 0x1D
- **REG_WRITE_IMM ADDR=0x10 DATA=0xFF** — Write value 0xFF to address 0x10
- **REG_WRITE_IMM DATA=0xFF ADDR=0x10** — Write value 0xFF to address 0x10

7.4.1.2.1.2 REG_WRITE_MASK_IMM Command

Description: Write the specified data, except the masked bits, to the specified register address

Assembly command: **REG_WRITE_MASK_IMM [ADDR=]<Address> [DATA=]<Data> [MASK=]<Mask>**

Address, Data, and Mask can be in any literal integer format (decimal, hex, and so forth).

'ADDR=', 'DATA=', and 'MASK=' are optional. When included, the parameters can be in any order.

Examples:

- **REG_WRITE_MASK_IMM 0x1D 0x80 0xF0** — Write 0b1000 to the upper 4 bits of the register at address 0x1D
- **REG_WRITE_MASK_IMM ADDR=0x10 DATA=0x0F MASK=0xF0** — Write 0b1111 to the lower 4 bits of the register at address 0x10
- **REG_WRITE_MASK_IMM DATA=0x0F MASK=0xF0 ADDR=0x10** — Write 0b1111 to the lower 4 bits of the register at address 0x10

7.4.1.2.1.3 REG_WRITE_MASK_PAGE0_IMM Command

Description: Write the specified data, except the masked bits, to the specified page 0 register address

Assembly command: **REG_WRITE_MASK_PAGE0_IMM [ADDR=]<Address> [DATA=]<Data> [MASK=]<Mask>**

Address, Data, and Mask can be in any literal integer format (decimal, hex, and so forth).

'ADDR=', 'DATA=', and 'MASK=' are optional. When included, the parameters can be in any order.

Examples:

- **REG_WRITE_MASK_PAGE0_IMM 0x1D 0x80 0xF0** — Write 0b1000 to the upper 4 bits of the register at address 0x1D
- **REG_WRITE_MASK_PAGE0_IMM ADDR=0x10 DATA=0x0F MASK=0xF0** — Write 0b1111 to the lower 4 bits of the register at address 0x10
- **REG_WRITE_MASK_PAGE0_IMM DATA=0x0F MASK=0xF0 ADDR=0x10** — Write 0b1111 to the lower 4 bits of the register at address 0x10

7.4.1.2.1.4 REG_WRITE_BIT_PAGE0_IMM Command

Description: Write the specified data to the BIT_SEL location of the specified page 0 register address

Assembly command: **REG_WRITE_BIT_PAGE0_IMM [ADDR=]<Address> [BIT=]<Bit> [DATA=]<Data>**

Address, Bit, and Data can be in any literal integer format (decimal, hex, and so forth).

'ADDR=', 'BIT=', and 'DATA=' are optional. When included, the parameters can be in any order.

Examples:

- **REG_WRITE_BIT_PAGE0_IMM 0x1D 7 0** — Write '0' to bit 7 of the register at address 0x1D
- **REG_WRITE_BIT_PAGE0_IMM ADDR=0x10 BIT=3 DATA=1** — Write 0b1 to bit 3 of the register at address 0x10

7.4.1.2.1.5 REG_WRITE_WIN_PAGE0_IMM Command

Description: Write the specified data to the SHIFT location of the specified page 0 register address

Assembly command: **REG_WRITE_WIN_PAGE0_IMM [ADDR=]<Address> [DATA=]<Data> [MASK=]<Mask> [SHIFT=]<Shift>**

Address, Data, Mask, and Shift can be in any literal integer format (decimal, hex, and so forth).

'ADDR=', 'DATA=', 'MASK=', and 'SHIFT=' are optional. When included, the parameters can be in any order.

Examples:

- **REG_WRITE_WIN_PAGE0_IMM ADDR=0x1D DATA=0x8 MASK=0x13 SHIFT=2** — Write bits 5:4 to 0b10 to the register at address 0x1D. Data and mask give 5-bit value 0bx10xx. These two bits are then left shifted 2 bit-positions to give full byte value of 0bxx10xxxx, hence sets bits 5:4 to 0b10.

7.4.1.2.1.6 REG_WRITE_VOUT_IMM Command

Description: Write the target voltage of a specified regulator after a specified delay. This command is a spin-off of the REG_WRITE_IMM command with the intention to save instruction bits.

Assembly command: **REG_WRITE_VOUT_IMM [REGULATOR=]<Regulator ID> [SEL=]<VSEL> [VOUT=]<Vout> [DELAY=]<Delay>**

'REGULATOR=', 'SEL=', 'VOUT=', and 'DELAY=' are options. When included, the parameters can be in any order.

Regulator ID = BUCK1, BUCK2, BUCK3, BUCK4.

VSEL selects the BUCKn_VSET1 or BUCKn_VSET2 bits which the command writes to if Regulator ID is BUCK1-4. VSEL is defined as: '0': BUCKn_VSET1, '1': BUCKn_VSET2, '2': Currently Active BUCKn_VSET, '3': Currently Inactive BUCKn_VSET.

VOUT = output voltage in mV or V. Unit must be listed.

DELAY = delay time in ns, μ s, ms, or s. If no unit is entered, this field must be an integer value between 0-63, which becomes the threshold count for the counter running a step size specified in the register PFSM_DELAY_STEP. The delay value is rounded to the nearest achievable delay time based on the current step size. Current step size is based on the default NVM setting or a SET_DELAY value from a previous command in the same sequence. Assembler reports an error if the step size is too large or too small to meet the delay.

Examples:

- **REG_WRITE_VOUT_IMM BUCK3 2 1.05 V 100 μ s** — Sets BUCK3 to 1.05 V by updating the active BUCK3_VSET register after 100 μ s

7.4.1.2.1.7 REG_WRITE_VCTRL_IMM Command

Description: Write the operation mode of a specified regulator after a specified delay. This command is a spin-off of the REG_WRITE_IMM command with the intention to save instruction bits.

Assembly command: **REG_WRITE_VCTRL_IMM [REGULATOR=]<Regulator ID> [VCTRL=]<VCTRL> [MASK=]<Mask> [DELAY=]<Delay> [DELAY_MODE=]<Delay Mode>**

'REGULATOR=', 'VCTRL=', 'MASK=', and 'DELAY=' are options. When included, the parameters can be in any order.

Regulator ID = BUCK1, BUCK2, BUCK3, BUCK4.

VCTRL = 0-15, in hex, decimal, or binary format. Data to write to the following regulator control fields:

- BUCKs: BUCKn_PLDN, BUCKn_VMON_EN, BUCKn_VSEL, BUCKn_FPWM_MP, BUCKn_FPWM, and BUCKn_EN

DELAY = delay time in ns, μ s, ms, or s. If no unit is entered, this field must be an integer value between 0-63, which becomes the threshold count for the counter running a step size specified in the register PFSM_DELAY_STEP. Delay value is rounded to the nearest achievable delay time based on the current step size. Current step size is based on the default NVM setting or a SET_DELAY value from a previous command in the same sequence. Assembler reports an error if the step size is too large or too small to meet the delay.

Delay Mode must be one of the below options: MATCH_EN = 0 (Delay if VCTRL enable bit mismatches) MATCH_ALL = 1 (Delay if any VCTRL bits mismatch) ALWAYS = 2 (Delay always)

Examples:

- **REG_WRITE_VCTRL_IMM BUCK3 0x00 0xE0 100 μ s** — Sets BUCK3 to OFF (VCTRL bits = 0b00000) after 100 μ s

7.4.1.2.1.8 REG_WRITE_MASK_SREG Command

Description: Write the data from a scratch register, except the masked bits, to the specified register address

Assembly command: **REG_WRITE_MASK_SREG [REG=]<Scratch Register> [ADDR=]<Address> [MASK=]<Mask>**

'REG=', 'ADDR=', and 'MASK=' are options. When included, the parameters can be in any order.

Scratch Register can be R0, R1, R2, or R3.

Address and Mask can be in any literal integer format (decimal, hex, and so forth).

Examples:

- **REG_WRITE_MASK_SREG R2 0x22 0x00** — Write the content of scratch register 2 to address 0x22
- **REG_WRITE_MASK_SREG REG=R0 ADDR=0x054 MASK=0xF0** — Write the lower 4 bits of scratch register 0 to address 0x54

7.4.1.2.1.9 SREG_READ_REG Command

Description: Write scratch register (REG0-3) with data from a specified address

Assembly command: **SREG_READ_REG [REG=]<Scratch Register> [ADDR=]<Address>**

'REG=' and 'ADDR=' are options. When included, the parameters can be in any order.

Scratch Register can be R0, R1, R2, or R3.

Address can be in any literal integer format (decimal, hex, and so forth).

Examples:

- **SREG_READ_REG R2 0x15** — Read the content of address 0x15 and write the data to scratch register 2
- **SREG_READ_REG ADDR=0x077 REG=R3** — Read the content of address 0x77 and write the data to scratch register 3

7.4.1.2.1.10 SREG_WRITE_IMM Command

Description: Write the specified data to the scratch register (REG0-3)

Assembly command: **SREG_WRITE_IMM [REG=]<Register> [DATA=]<Data>**

Data can be in any literal integer format (decimal, hex, and so forth).

Register can be R0, R1, R2, or R3.

'REG=' and 'DATA=' are options. When included, the parameters can be in any order.

Examples:

- **SREG_WRITE_IMM R2 0x15** — Write 0x15 to scratch register 2
- **SREG_WRITE_IMM ADDR=0x077 REG=R3** — Read 0x77 to scratch register 3

7.4.1.2.1.11 WAIT Command

Description: Wait upon a condition of a given type. Execution is paused until the specified type of the condition is met or timed out

Assembly command: **WAIT [COND=]<Condition> [TYPE=]<Type> [TIMEOUT=]<Timeout> [DEST=]<Destination>**

Alternative assembly command: **JUMP [DEST=]<Destination>**

'COND=', 'TYPE=', 'TIMEOUT=', and 'DEST=' are options. When included, the parameters can be in any order.

Condition are listed in [Table 7-9](#). Examples: GPIO1, BUCK1_PG, I2C_1

Type = LOW, HIGH, RISE, or FALL

Timeout = timeout value in ns, μ s, ms, or s. If no unit is entered, this field must be an integer value between 0-63. Timeout value is be rounded to the nearest achievable time based on the current step size. Current step size is based on the default NVM setting or a SET_DELAY value from a previous command in the same sequence. Assembler reports an error if the step size is too large or too small to meet the delay.

Destination = Label to jump to if when timeout occurs. Destination must be after the WAIT statement in memory.

Memory space can be either '0' or '1'. '0' indicates the destination address is in the PFSM memory space. '1' indicates the destination address is external and represents a FSM state ID.

When using the jump command, the PFSM performs an unconditional jump. The command is be compiled as "WAIT COND=63 TYPE=LOW TIMEOUT=0 DEST=<Destination>". Condition 63 is a hardcoded 1, so the condition is never satisfied and hence always times out. Therefore this command always jumps to the destination.

Examples:

- **WAIT GPIO4 RISE 1 s <Destination> 0** — Wait to execute the command at the specified SRAM address when a rise edge is detected at GPIO4, or after 1 second
- **WAIT COND=BUCK1_PG TYPE=HIGH TIMEOUT=500 μ s DEST=<mcu2act_seq>** — Wait to execute the commands at <mcu2act_seq> address as soon as BUCK1 output is within power-good range, or after 500 μ s

Table 7-9. WAIT Command Conditions

| COND_SEL | Condition Name |
|----------|----------------|----------|----------------|----------|----------------|----------|----------------|
| 0 | GPIO1 | 16 | N/A | 32 | I2C_0 | 48 | LP_STANDBY_SEL |
| 1 | GPIO2 | 17 | N/A | 33 | I2C_1 | 49 | N/A |
| 2 | GPIO3 | 18 | N/A | 34 | I2C_2 | 50 | N/A |
| 3 | GPIO4 | 19 | N/A | 35 | I2C_3 | 51 | N/A |
| 4 | GPIO5 | 20 | PGOOD | 36 | I2C_4 | 52 | N/A |
| 5 | GPIO6 | 21 | TWARN_EVENT | 37 | I2C_5 | 53 | N/A |
| 6 | GPIO7 | 22 | INTERRUPT_PIN | 38 | I2C_6 | 54 | N/A |
| 7 | GPIO8 | 23 | N/A | 39 | I2C_7 | 55 | N/A |
| 8 | GPIO9 | 24 | N/A | 40 | SREG0_0 | 56 | N/A |
| 9 | GPIO10 | 25 | N/A | 41 | SREG0_1 | 57 | N/A |
| 10 | N/A | 26 | N/A | 42 | SREG0_2 | 58 | N/A |
| 11 | BUCK1_PG | 27 | N/A | 43 | SREG0_3 | 59 | N/A |
| 12 | BUCK2_PG | 28 | N/A | 44 | SREG0_4 | 60 | N/A |
| 13 | BUCK3_PG | 29 | N/A | 45 | SREG0_5 | 61 | N/A |
| 14 | BUCK4_PG | 30 | N/A | 46 | SREG0_6 | 62 | 0 |
| 15 | N/A | 31 | N/A | 47 | SREG0_7 | 63 | 1 |

7.4.1.2.1.12 DELAY_IMM Command

Description: Delay the execution by a specified time

Assembly command: **DELAY_IMM <Delay>**

Delay = delay time in ns, μ s, ms, or s. If no unit is entered, this field must be an integer value between 0-63. Delay value is rounded to the nearest achievable time based on the current step size. Current step size is based on the default NVM setting or a SET_DELAY value from a previous command in the same sequence. Assembler reports an error if the step size is too large or too small to meet the delay.

Examples:

- **DELAY_IMM 100 μ s** — Delay execution by 100 μ s
- **DELAY_IMM 10 ms** — Delay execution by 10 ms
- **DELAY_IMM 8** — Delay execution by 8 ticks of the current PFSM time step

7.4.1.2.1.13 DELAY_SREG Command

Description: Delay the execution by a time value stored in the specified scratch register.

Assembly command: **DELAY_SREG <Register>**

Register can be R0, R1, R2, or R3.

Examples:

- **DELAY_SREG R0** — Delay execution by the time value stored in scratch register0

7.4.1.2.1.14 TRIG_SET Command

Description: Set a trigger destination address for a given input signal or condition. These commands must be defined at the beginning of PFSM configuration memory.

Assembly command: **TRIG_SET [DEST=]<Destination> [ID=]<Trig_ID> [SEL=]<Trig_sel> [TYPE=]<Trig_type> [IMM=]<IMM> [EXT=]<Memory space>**

'DEST=', 'ID=', 'SEL=', 'TYPE=', 'IMM=', and 'EXT=' are options. When included, the parameters can be in any order.

Destination is the label where this trigger starts executing.

Trig_ID is the ID of the hardware trigger module to be configured (value range 0-27). They must be defined in numeric order based on the priority of the trigger.

Trig_Sel is the 'Trigger Name' from the [Table 7-11](#). This 'Trigger Name' is the trigger signal to be associated with the specified TRIG_ID.

Trig_type = LOW, HIGH, RISE, or FALL.

IMM can be either '0' or '1'. = '0' if the trigger is not activated until the END command of a given task; '1' if the trigger is activated immediately and can abort a sequence.

REENTRANT can be either '0' or '1'. '1' permits a trigger to return to the current state, that allows a self-branching trigger to execute the current sequence again.

Memory space can be either '0' or '1'. '0' indicates the destination address is in the PFSM memory space. '1' indicates the destination address is external and represents a FSM state ID.

Examples:

- **TRIG_SET seq1 20 GPIO_1 LOW 0 0** — Set trigger 20 to be GPIO_1=Low, not immediate. When triggered, start executing at 'seq1' label.
- **TRIG_SET DEST=seq2 ID=15 SEL=WD_ERROR TYPE=RISE IMM=0 EXT=0** — Set trigger 15 to be rising WD_ERROR trigger, not immediate. When triggered, start executing at 'seq2' label.

7.4.1.2.1.15 TRIG_MASK Command

Description: Sets a trigger mask that determines which triggers are active. Setting a '0' enables the trigger, setting a '1' disables (masks) the trigger.

Assembly command: **TRIG_MASK <Mask value>**

Mask Value = 28-bit mask in any literal integer format (decimal, hex, and so forth).

Examples:

- **TRIG_MASK 0x5FF82F0** — Set the trigger mask to 0x5FF82F0

7.4.1.2.1.16 END Command

Table 7-10 shows the format of the END commands.

Table 7-10. END Command Format

| Bit[3:0] |
|----------|
| CMD |
| 4 bits |

Description: Marks the final instruction in a sequential task

Fields:

- CMD: Command opcode (0xC)

Assembly command: **END**

7.4.1.2.2 Configuration Memory Organization and Sequence Execution

The configuration memory is loaded from NVM into an SRAM. Figure 7-37 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 0xFFFFF0
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 0xFE6EDC
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 7-37. Configuration Memory Script Example

As soon as the PMIC state reaches the mission states, it starts reading from the configuration memory until it hits the first END command. Setting up the triggers (1-28) must 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 that contains the starting address associated with each trigger in the configuration memory. If the trigger destination is an FFSSM 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 7-11](#).

When a trigger or multiple triggers are activated, the PFSM execution engine looks up the starting address associated with the highest priority unmasked trigger, and starts executing commands until it hits an END command. The last commands before END statement is generally the TRIG_MASK command, which directs the PFSM to a new set of unmasked trigger configurations, and the trigger with the highest priority in the new set is 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 is 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 sequence 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 FFSSM state machine takes over control from the execution engine each time an event occurs that requires a transition from the MISSION state of the PMIC to a fixed device state.

Table 7-11. 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 intended action for the activated trigger is to <i>immediate shutdown</i> 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 that causes ESM_MCU_RST_INT |
| WD_ERROR | An event that causes WD_RST_INT |
| SOC_POWER_ERROR | Output failure detection from a regulator which is assigned to the SOC rail group (x_GRP_SEL = '10') |
| A | NSLEEP2 and NSLEEP1 = '11'. More information regarding the NSLEEP1 and NSLEEP2 functions can be found under Section 7.4.1.2.4.3 |
| WKUP1 | A rising or falling edge detection on a GPIO pin that is configured as 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 7.4.1.2.4.3 |
| WKUP2 | A rising or falling edge detection on a GPIO pin that is configured as 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 7.4.1.2.4.3 |
| D | NSLEEP2 and NSLEEP1 = '00'. More information regarding the NSLEEP1 and NSLEEP2 functions can be found under Section 7.4.1.2.4.3 |

Table 7-11. PFSM Trigger Selections (continued)

| Trigger Name | Trigger Source |
|--------------|--|
| 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. More information regarding the WAIT command can be found under Section 7.4.1.2.1.11 |
| 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 |
| 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 |
| 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' |

7.4.1.2.3 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 Figure 7-38 is used as an example state machine that is defined through the configuration memory using the configuration FSM instructions.

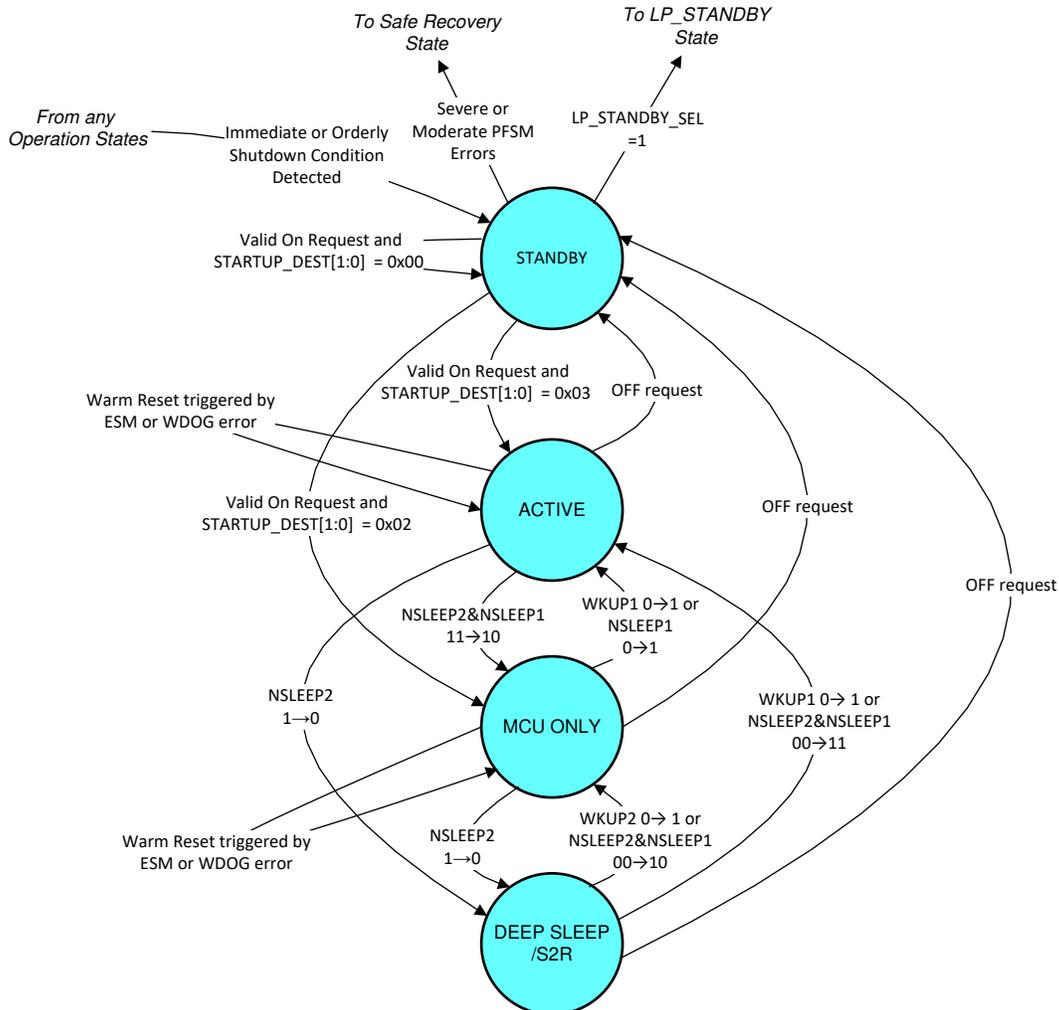


Figure 7-38. Example of a Mission State-Machine

Each power state (light blue bubbles in Figure 7-38) defines the ON or OFF state and the sequencing timing of the external regulators and GPIO outputs. This example defines four power states: STANDBY, ACTIVE, MCU ONLY, and DEEP_SLEEP/S2R three power states: STANDBY, ACTIVE and RETENTION states. The priority order of these states is as follows:

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 Table 7-11. 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 transitions during normal device operation, are then placed according to the priority order of the state the device is transitioning to. Table 7-12 list 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 7-38](#). This table also helps to determine which triggers must be masked by the TRIG_MASK command upon arriving a pre-defined power state to produce the desired PFSM behavior.

Table 7-12. 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 | | | | |
| 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 |
| 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 |

Table 7-12. 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 |
| 28-bit TRIG_MASK Value in Hex format: | | | 0xFFE4FF8 | 0xFF181C0 | 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 exits the mission states and the FFMSM state machine takes over control of the device power states once this trigger is executed.

7.4.1.2.4 Pre-Configured Hardware Transitions

There are some pre-defined trigger sources, such as on-requests and off-requests, that are constructed with the combination of hardware input signals and register bits settings. This section provides more detail to these pre-defined trigger sources and shows how they can be utilized in the PFMSM configuration to initiate state to state transitions.

7.4.1.2.4.1 ON Requests

ON requests are used to switch on the device, which then 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 stays or moves to the next state corresponding to the NSLEEP signals state assignment as specified in [Table 7-16](#).

[Table 7-13](#) lists the available ON requests.

Table 7-13. ON Requests

| EVENT | MASKABLE | COMMENT | DEBOUNCE |
|--|----------|---|-----------|
| ENABLE (pin) | Yes | Level sensitive | 8 μ s |
| First Supply Detection (FSD) | Yes | VCCA > VCCA_UV and FSD unmasked | N/A |
| WKUP1 or WKUP2 Detection | Yes | Edge sensitive | 8 μ s |
| Recovery from Immediate and Orderly Shutdown | No | Recover from system errors which caused immediate or orderly shutdown of the device | N/A |

If one of the events listed in [Table 7-13](#) occurs, then the event powers on the device unless one of the gating conditions listed in [Table 7-14](#) is present.

Table 7-14. ON Requests Gating Conditions

| EVENT | MASKABLE | COMMENT |
|-------------------|----------|---|
| VCCA_OVP (event) | No | VCCA > VCCA_OVP Device stays in SAFE RECOVERY until VCCA < VCCA_OVP |
| VCCA_UVLO (event) | No | VCCA < VCCA_UVLO |
| VINT_OVP (event) | No | LDOVINT > 1.98 V |
| VINT_UVLO (event) | No | LDOVINT < 1.62 V |
| TSD (event) | No | Device stays in SAFE RECOVERY until temperature decreases below TWARN level |

7.4.1.2.4.2 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 7-15](#) lists the conditions to generate the OFF requests and the corresponding destination state.

Table 7-15. OFF Requests

| EVENT | DEBOUNCE | LP_STANDBY_SEL BIT SETTING | DESTINATION STATE |
|---------------|-----------|----------------------------|-------------------|
| ENABLE (pin) | 8 μ s | 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 |

Using the I2C_TRIGGER_0 bit as the OFF request enables the device to wake up from the STANDBY or the LP_STANDBY states through the detection of WKUPn pins. 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 ENABLE pin remains active (if the ENABLE pin is used). Also the interrupt bits ENABLE_INT, FSD_INT and the GPIOx_INT bits (for the GPIOx pins assigned as WKUP1 or WKUP2) must be cleared before setting the I2C_TRIGGER_0 bit to '1'.

7.4.1.2.4.3 NSLEEP1 and NSLEEP2 Functions

The SLEEP requests are activated through the assertion of nSLEEP1 or nSLEEP2 pins that are the secondary functions of the 10 GPIO pins. These pins 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 \rightarrow 0 logic level transition of the NSLEEP signal generates a sleep request, while a 0 \rightarrow 1 logic level transition reverses the sleep request in the example PFSM from [Figure 7-38](#). When a NSLEEPn signal transitions from 1 \rightarrow 0, it generates a sleep request to go from a higher power state to a lower power state. When the signal transitions from 0 \rightarrow 1, it reverses the sleep request and returns the device to the higher power state.

The NSLEEP1 signal is designed to control the SoC supply rails. The NSLEEP2 signal is designed to control the MCU supply rails. When NSLEEP1 signal changes from 1 \rightarrow 0, depending on the state of NSLEEP2, the LP876242-Q1 device exits ACTIVE state and enters either the MCU ONLY or the S2R states. When NSLEEP2 signal changes from 1 \rightarrow 0, the device enters the S2R state regardless the state of NSLEEP1.

When the NSLEEP2 input signal changes from 0 \rightarrow 1, the MCU supply rails are enabled and the device exits S2R state. Depending on the state of NSLEEP1 signal, the device enters either the MCU ONLY or the ACTIVE state. In order for the system to function correctly, the MCU rails must be enabled when the NSLEEP1 input signal changes from 0 \rightarrow 1, which enables the SOC supply rails. NSLEEP1 0 \rightarrow 1 transition is ignored if NSLEEP2 is 0.

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 is ignored. [Table 7-16](#) shows how 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.

The states of the resources during ACTIVE, SLEEP, and DEEP SLEEP/S2R states are defined in the 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 7-16](#) shows the corresponding state assignment based on the state of the NSLEEPn and their corresponding mask signals using the example PFSM from [Figure 7-38](#).

Table 7-16. NSLEEPn Transitions and Mission State Assignments

| Current State | NSLEEP1 | NSLEEP2 | NSLEEP1 MASK | NSLEEP2 MASK | Trigger to FSM | Next State |
|----------------|-------------------|-------------------|--------------|--------------|----------------|------------|
| DEEP SLEEP/S2R | 0 | 0 \rightarrow 1 | 0 | 0 | TRIGGER B | MCU ONLY |
| DEEP SLEEP/S2R | 0 \rightarrow 1 | 0 \rightarrow 1 | 0 | 0 | TRIGGER A | ACTIVE |
| DEEP SLEEP/S2R | Don't care | 0 \rightarrow 1 | 1 | 0 | TRIGGER A | ACTIVE |

Table 7-16. NSLEEPn Transitions and Mission State Assignments (continued)

| Current State | NSLEEP1 | NSLEEP2 | NSLEEP1 MASK | NSLEEP2 MASK | Trigger to FSM | Next State |
|-------------------------------|------------|------------|--------------|--------------|----------------|----------------------|
| 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 |

7.4.1.2.4.4 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 GPIO_n_SEL bit to select the WKUP1 or WKUP2 functions. In the example PFSM depicted in [Figure 7-38](#), 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 GPIO_n_FALL_MASK and the GPIO_n_RISE_MASK bits) wakes up the device to the ACTIVE state. Likewise if a GPIO pin is configured as a WKUP2 pin, a detected edge wakes up the device to the MCU ONLY state. If multiple edge detections of WKUP signals occur simultaneous, the device goes to the state in the following priority order:

1. ACTIVE
2. MCU ONLY

When a valid edge is detected at a WKUP pin, the nINT pin generates an interrupt to signal the MCU of the wake-up event, and the GPIO_x_INT interrupt bit is set. The wake request remains active until the GPIO_x_INT bit is cleared by the MCU. While the wake request is executing, the device does not react to sleep requests to enter a lower power state until the corresponding GPIO_x_INT interrupt bit is cleared to cancel the wake request. After the wake request is canceled, the device returns to the state indicated by the NSLEEP1 and NSLEEP2 signals as shown in [Table 7-16](#).

The WKUP1 and WKUP2 functions can be used in all Mission States and also in LP_STANDBY state.

7.4.1.3 Error Handling Operations

The FSM engine of the LP876242-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

7.4.1.3.1 Power Rail Output Error

A power rail output error occurs when an error condition is detected on the output rails of the device that 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 exceeds the forward current limit.

The BUCKx_GRP_SEL, VMONn_GRP_SEL and VCCA_GRP_SEL registers are used to configure the rail group for all of the Bucks and the voltage monitors that are available for external rails. The selectable rail groups are MCU rail group, SoC rail group, or OTHER rail group. The LP876242-Q1 device is designed to react differently when an error is detected from a power resource assigned to the different rail groups.

Figure 7-36 shows how the SOC_RAIL_TRIG[1:0], MCU_RAIL_TRIG[1:0], and OTHER_RAIL_TRIG[1:0] register bits 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 that the assigned groups of rails perform in case of an 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] register bits, any failure detected in this group of rails causes 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 immediately resets the attached MCU and SoC by driving the nRSTOUT and nRSTOUT_SoC pins low. All of the power resources assigned to the MCU and SOC shut down immediately without a sequencing order. The nINT pin signals that an MCU_PWR_ERR_INT interrupt event has occurred and the EN_DRV pin is forced low. If the error is recoverable within the recovery time interval, the device increments the recovery count, returns to INIT state, and reattempts 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 stays 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. In this PFSM example depicted in Figure 7-38, when a power resource in this group is detected, the PFSM typically causes the device to execute the shutdown of all the resources assigned to the SoC rail group, and the device enters the MCU ONLY state. The device immediately resets the attached SoC by toggling the nRSTOUT_SoC pin. The reset output to the MCU and the resources assigned to the MCU rail group remain unchanged. The EN_DRV pin also remains unchanged, and the nINT pin signals that an 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 LP876242-Q1 that MCU has acknowledged 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 reattempt the SoC power up. Refer to Section 7.4.1.2.4.3 for information regarding the setting of the NSLEEP1 and NSLEEP2 signals.

The power resources used for peripheral devices in the system, for which no error-handling action is required, the rail mapping needs to be selected as "no group assigned".

7.4.1.3.2 Boot 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 causes 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 attempts to enter the INIT state again and reattempts the BOOT BIST until the recovery count reaches the maximum threshold. When this occurs, the device stays in the SAFE RECOVERY state until VCCA voltage is below the VCCA_UVLO threshold and the device is power cycled.

7.4.1.3.3 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, that causes the device to execute the immediate shutdown sequence and enter the SAFE RECOVERY state. The device immediately resets the attached MCU and SoC by driving the nRSTOUT and nRSTOUT_SoC pins low. All of the power resources assigned to the MCU and SOC are immediately shut down. The EN_DRV pin is forced low, and the nINT pin is driven low to signal an interrupt event has occurred.

7.4.1.3.4 Catastrophic Error

Catastrophic errors are errors that affect multiple power resources such as errors detected in supply voltage, LDOVINT supply for control logic, errors on internal clock signals, device temperature passing the thermal shutdown threshold, error detected on the SPMI bus, or an error detected in the PFSM sequence.

Following errors are grouped as severe errors:

- VCCA > OVP threshold
- Junction temperature > immediate shutdown level
- Error in PFSM Sequence

For these above listed errors, depending on the setting of bits SEVERE_ERR_TRIG[1:0], an immediate or orderly shutdown condition is created. The PFSM executes the corresponding sequence for the IMMEDIATE_SHUTDOWN trigger or the ORDERLY_SHUTDOWN trigger and enters the SAFE RECOVERY state.

Following errors are grouped as moderate errors:

- Junction temperature > orderly shutdown level
- BIST failure
- CRC error in register map
- Recovery counter exceeding the threshold value
- Error on SPMI bus
- Readback error on nRSTOUT pin or nINT pin

For these above listed errors, depending on the setting of bits MODERATE_ERR_TRIG[1:0], an immediate or orderly shutdown condition is created. The PFSM executes the corresponding sequence for the IMMEDIATE_SHUTDOWN trigger or the ORDERLY_SHUTDOWN trigger and enters the SAFE RECOVERY state.

For following errors, the device performs an immediate shutdown and resets all internal logic circuits:

- VCCA < UVLO threshold
- Error on LDOVINT supply
- Errors on internal clock signals
- Unrecoverable CRC error in the SRAM memory of the PFSM

For all of the above listed errors, the device resets the attached MCU and SoC by driving the GPIO pins used as nRSTOUT and nRSTOUT_SoC pins low. All of the power resources assigned to the attached MCU and SOC are shut down. The nINT pin is driven low to signal an interrupt event has occurred, and the GPIO pins used as EN_DRV pin is forced low.

7.4.1.3.5 Watchdog (WDOG) Error

Watchdog (WD) describes details about the Watchdog (WDOG) errors detection mechanisms.

7.4.1.3.6 Error Signal Monitor (ESM) Error

There is one Error Signal Monitor (ESM) available for the LP876242-Q1 device, designed to detect and handle the error signal received from the attached MCU. The error detection mechanisms for the monitor is described in detail under Error Signal Monitor (ESM).

7.4.1.3.7 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 current limit detection while the output voltage still maintains within the power good threshold. When these errors occur, the nINT pin is driven low to signal an interrupt event has occurred. The device remains in the operation state and the state of the EN_DRV pin, the power resources, and the reset outputs remain unchanged.

The power resource current limit detection can, by setting bit EN_ILIM_FSM_CTRL=1, be configured such that it is handled as a Power Rail Output Error as described in [Section 7.4.1.3.1](#).

7.4.1.4 Device Start-up Timing

[Figure 7-39](#) shows the timing diagram of the LP876242-Q1 after the first supply detection.

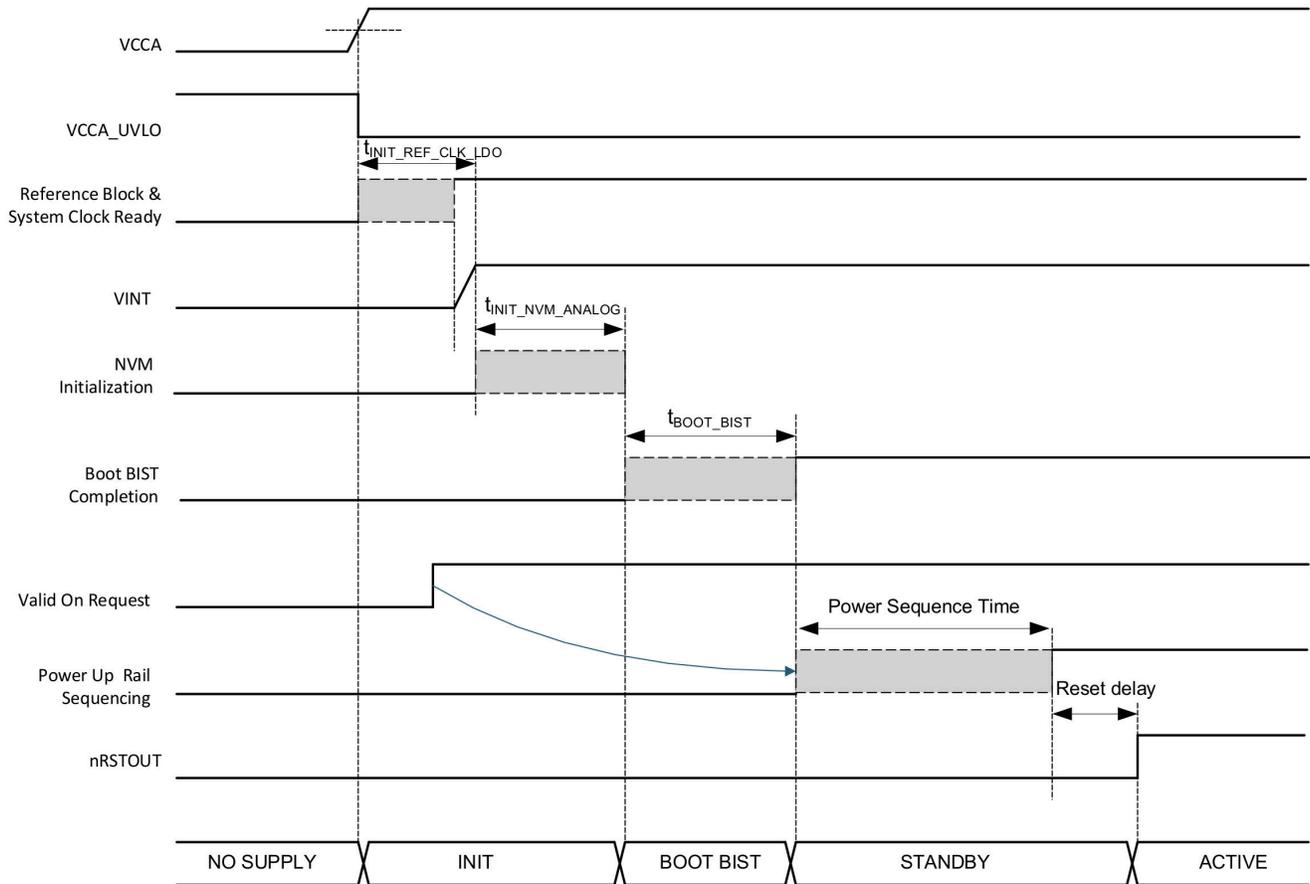


Figure 7-39. Device Start-up Timing Diagram

$t_{INIT_REF_CLK_LDO}$ is the start-up time for the reference block, LDO, VINT and internal oscillator. $t_{INIT_NVM_ANALOG}$ 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 $t_{INIT_REF_CLK_LDO}$ 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 7.4.1.5](#) for more details.

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

7.4.1.5 Power Sequences

A power sequence is an automatic preconfigured sequence the LP876242-Q1 device applies to its resources, which include the states of the BUCKs, the VMON1 and VMON2 and the GPIO output signals. For a detailed description of the GPIOs signals, please refer to General-Purpose I/Os (GPIO Pins).

[Figure 7-40](#) 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 triggered by a valid off request. The resources controlled (for this example) are: BUCK3, BUCK2, REGEN1, SYNCCLKOUT, and nRSTOUT. The time between each resource enable and disable (t_{instX}) is also part of the preconfigured sequence definition.

A resource not assigned to any power sequence remains in off mode during the power-up sequence. The attached MCU or SoC can enable and configure this resource independently when the power-up sequence completes. Also the GPIO pin used as EN_DRV remains low during the power-up sequence. The attached MCU

or SoC can release the EN_DRV pin by setting the ENABLE_DRV bit (requires running watchdog, ESM started and no pending interrupt bits).

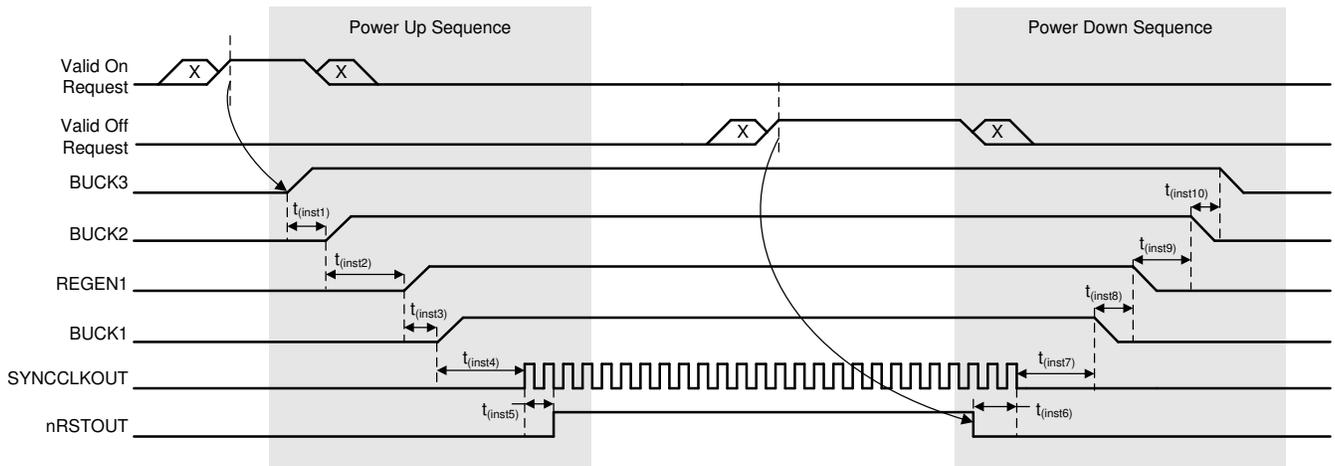


Figure 7-40. Power Sequence Example

As the power sequences of the LP876242-Q1 device are defined according to the processor requirements, the total time for the completion of the power sequence varies across various system definitions.

7.4.1.6 First Supply Detection

The LP876242-Q1 device can be configured to automatically start up from a first supply-detection (FSD) event detection. This feature is enabled by setting the FSD_MASK register bit to '0', and driving ENABLE pin active if ENABLE pin function is selected for GPIO. 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 immediately powers up from the NO SUPPLY state to an operation state, configured by the STARTUP_DEST[1:0] bits when $V_{CCA} > V_{CCA_UV}$, while V_{CCA_UV} gating is performed, and only when V_{CCA} voltage monitoring is enabled ($V_{CCA_VMON_EN} = 1$). 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 either stays in the current state or moves to the destination state according to the state of the NSLEEP1 and NSLEEP2 signals as specified in [Table 7-16](#).

7.4.2 Multi-PMIC Synchronization

A multi-PMIC synchronization scheme is implemented in the LP876242-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 microcontroller and multiple PMICs in the system. The control interface consists of an SPMI protocol that communicates the next power state information from the primary PMIC to up to 5 secondary PMICs, and receives feedback signal from the secondary PMICs to indicate any error condition or power state information. [Figure 7-41](#) is the block diagram of the power state synchronization scheme. The primary PMIC in this block diagram is responsible for broadcasting the synchronous system power state data, and processing the error feedback signals from the secondary PMICs. The primary PMIC is the *controller* device on the SPMI bus, and the secondary PMICs are the *target* devices on the SPMI bus.

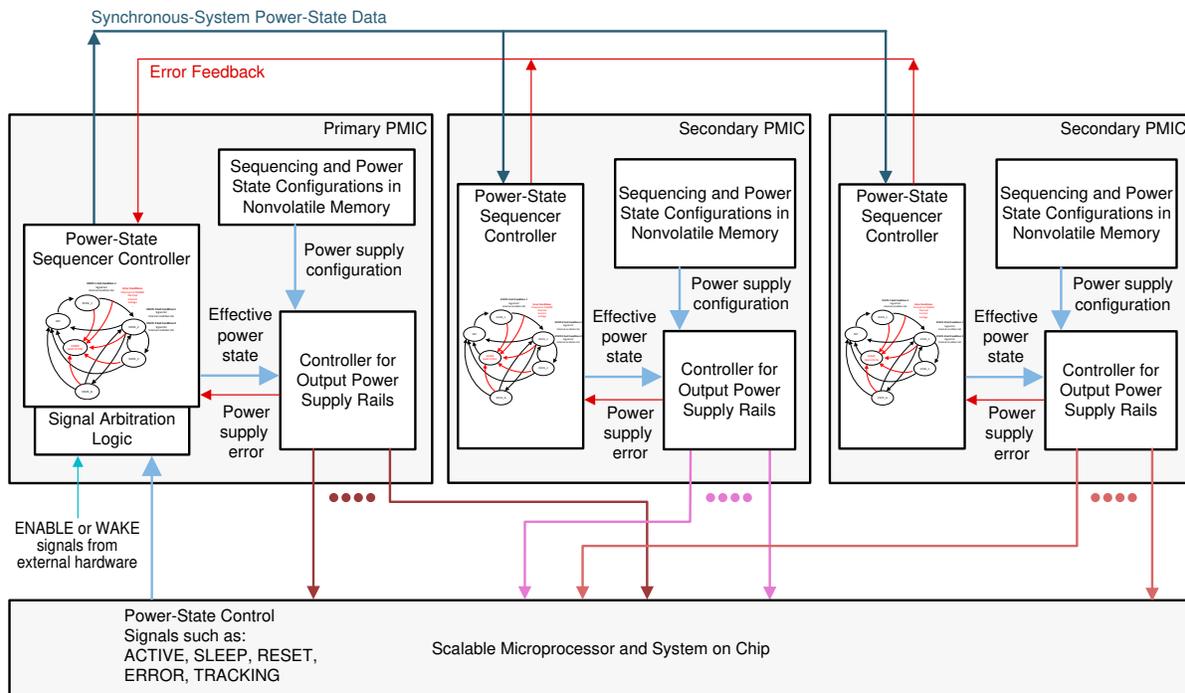


Figure 7-41. Multi-PMIC Power State Synchronization Block Diagram

In this scheme, each primary and secondary PMIC runs on its own system clock, and maintains its own register map. Each PMIC monitors its own activities and pulls down the open-drain output of nINT or PGOOD pin when errors are detected. The microprocessor must read the status bits from each PMIC device through the I2C or SPI interface to find out the source of the error that is reported.

[Figure 7-42](#) illustrates the pin connections between the primary, the secondary, and the application processor or the System-on-Chip.

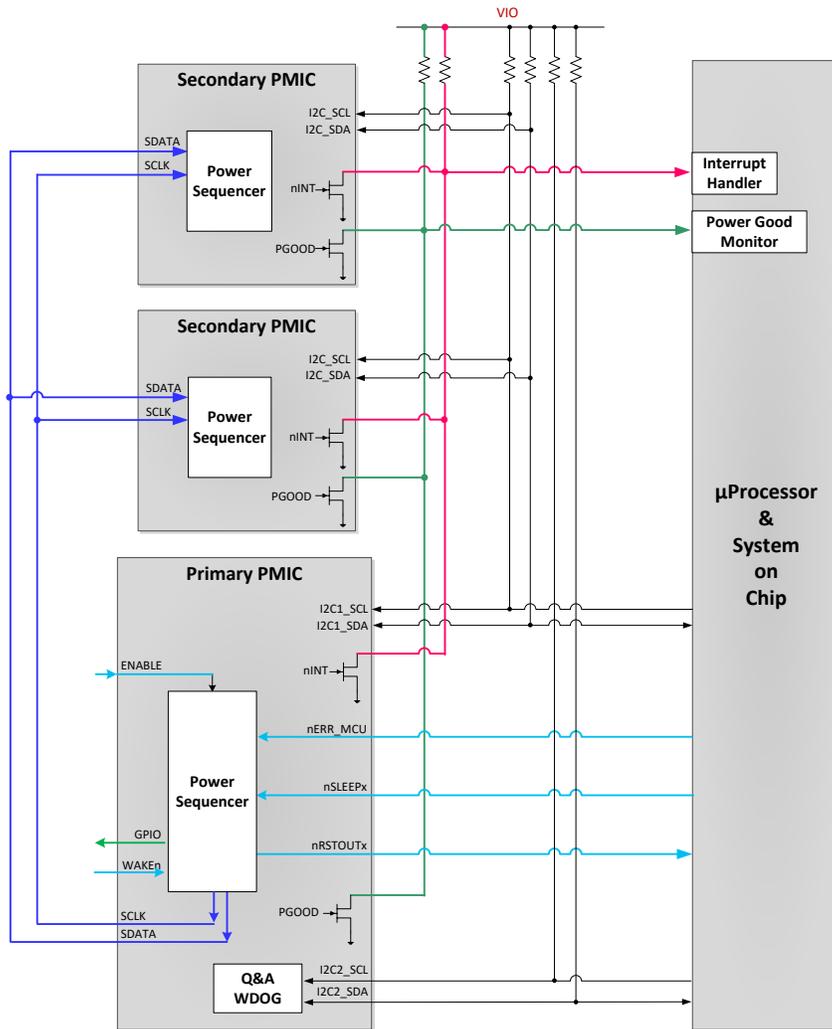


Figure 7-42. Multi-PMIC Pin Connections

The power sequencer of the multiple PMICs are synchronized at the beginning of each power up and power down sequence; a variation in the sequence timing, however, is still possible due to the $\pm 5\%$ clock accuracy of the independent system clocks on the primary and secondary PMICs. The worst-case sequence timing variation from different PMIC rails is up to $\pm 10\%$ of the target delay time. Figure 7-43 illustrates the creation of this timing variation between PMICs.

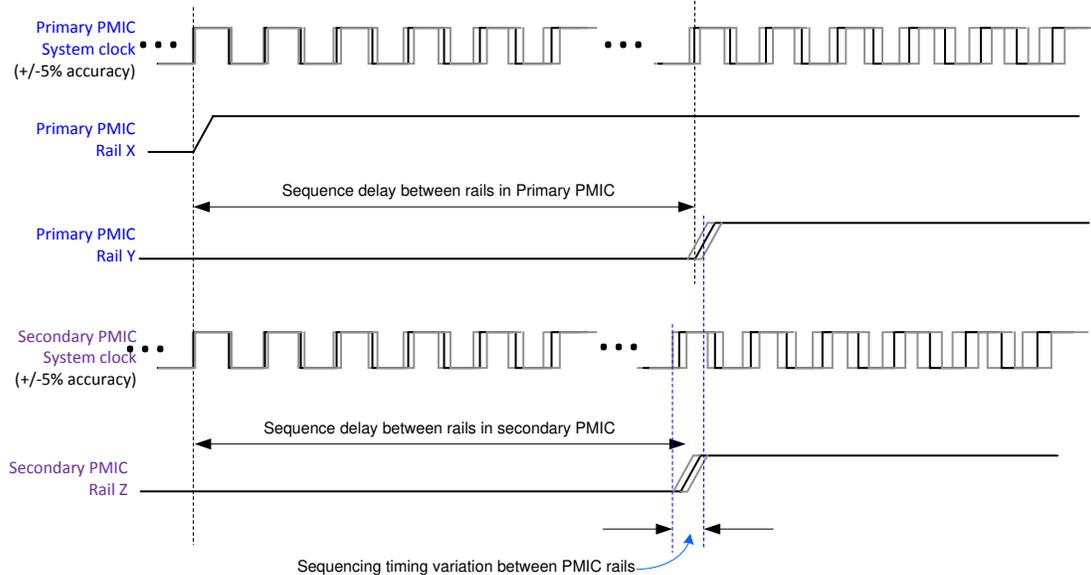


Figure 7-43. Multi-PMIC Rail Sequencing Timing Variation

7.4.2.1 SPMI Interface System Setup

An SPMI interface in the LP876242-Q1 device is utilized to communicate the power state transition across multiple PMICs in the system. The SPMI interface contains a *controller* block and a *target* block. There is only one PMIC, which is the primary PMIC, that acts as SPMI controller in any given system. As the SPMI controller it initiates SPMI interface BIST and executes periodic checking of the SPMI bus health.

The primary PMIC has a controller-ID (CID)= 1. The target block of SPMI interface in the primary PMIC device is activated as well, in order to receive SPMI communication messages from the secondary PMICs. The primary PMIC has a target-ID (TID) = 0101.

Each secondary PMIC on the SPMI network only has the target block of its SPMI interface enabled. There cannot be more than five secondary PMICs in the system. The target-IDs (TIDs) for the five secondary PMICs are:

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

All devices in the SPMI network listen to the group target-ID (GTID): 1111. This address is used to communicate all power state transition information in broadcast mode to all connected devices on the SPMI bus.

7.4.2.2 Transmission Protocol and CRC

The communication between the devices on the network utilizes Extended Register Write command to GTID 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 TID of one or more target 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 TID information, all 8 bits contain the TID of the target 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 and excludes the parity bits in these three frames).

Figure 7-44 shows the data format of the SPMI Extended Register Write Command.

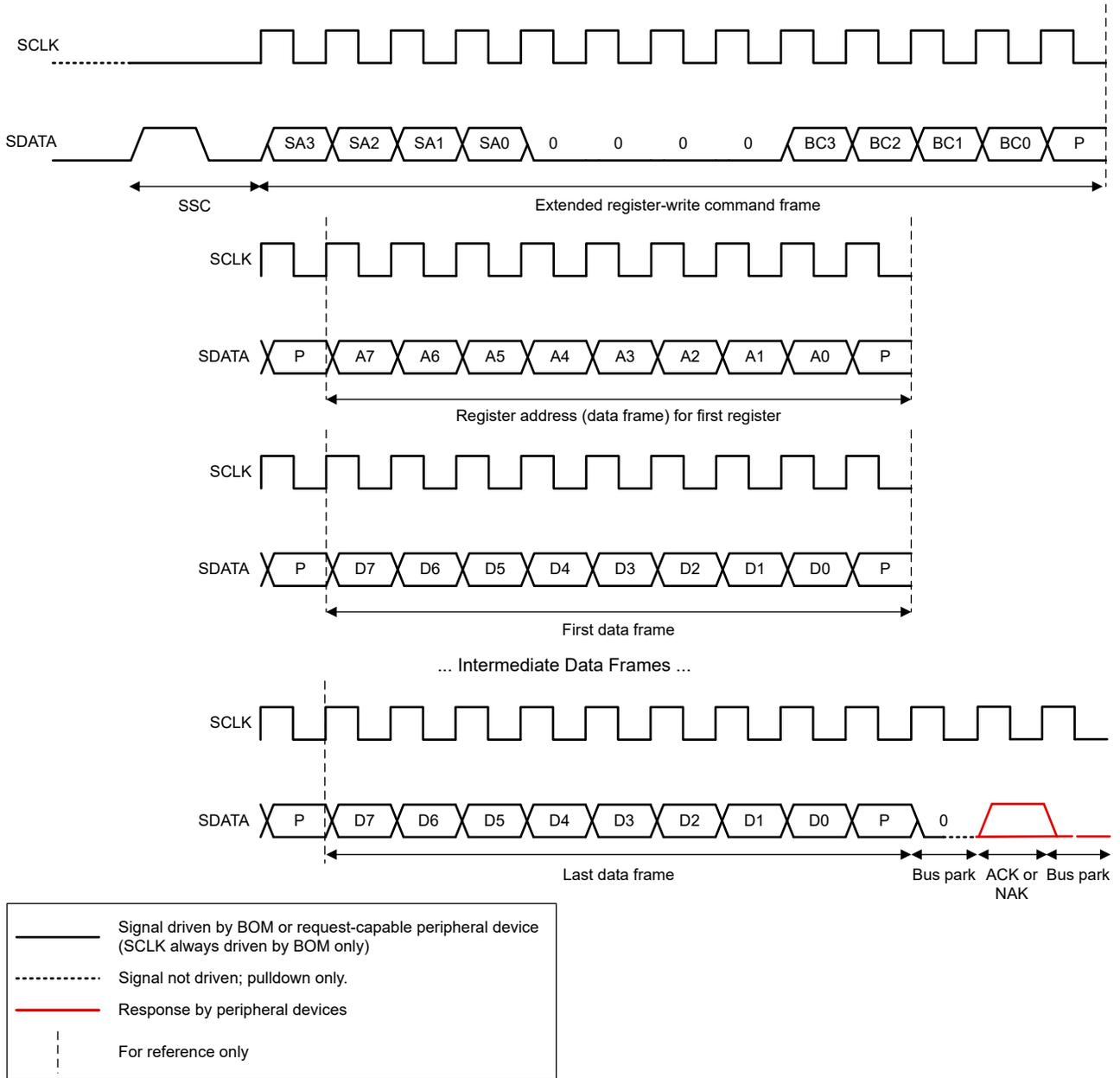


Figure 7-44. SPMI Extended Register Write Command

7.4.2.2.1 Operation with Transmission Errors

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

If the transmitting device sees NACK response, it tries to resend the message as many times as indicated by SPMI_RETRY_LIMIT register bits. After that it considers the 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 target device loses bus arbitration. SPMI_RETRY_LIMIT counter is reset after each successful transmission by the device.

If a target device has determined that SPMI does not work reliably it does not respond to any SPMI commands anymore until power-on-reset event has occurred. This "no-response" behavior is to prevent continued operation in a situation where SPMI is unreliable. If a target device does no longer respond to any SPMI command, the controller device on the SPMI bus detects a missing target device on the network during the periodic testing of SPMI bus. The target device then internally handles 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.

7.4.2.2.2 Transmitted Information

The SPMI bus is used to carry two types of information:

- PFSM Trigger ID between the SPMI controller and target devices
- TID from SPMI target devices to SPMI controller device

The SPMI controller device reads the TID of the target 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 explain reason for the power state transitions.

7.4.2.3 SPMI Target Device Communication to SPMI Controller Device

An SPMI target device communicates to the SPMI controller device and any other SPMI target devices, only if there is an internal error that is not SPMI related. The target device initiates the error communication using Arbitration Request with A-bit as defined in the SPMI 2.0 specification. SPMI 2.0 protocol manages the situation with multiple target devices requesting error communication at the same time, by using the target arbitration process as described in SPMI 2.0 specification. Once the SPMI target device wins the arbitration using the A-bit protocol, it performs an Extended Register Write command to Group Target ID (GTID) address 1111 by using the protocol described in [Section 7.4.2.2](#) for communicating PFSM trigger ID.

7.4.2.3.1 Incomplete Communication from SPMI Target Device to SPMI Controller Device

In case the SPMI controller device detects an arbitration request on the SPMI interface, but the received sequence has an error or is incomplete, the SPMI controller device immediately performs the SPMI Built-In Self-Test (SPMI-BIST). If this SPMI-BIST fails, the SPMI controller device executes the error handling for the SPMI error. If the SPMI-BIST passes successfully, the SPMI controller device resumes normal operation.

7.4.2.4 SPMI-BIST Overview

The SPMI-BIST is performed during BIST state and regularly during runtime operation. [Figure 7-45](#) below illustrates how the SPMI-BIST operates during device power-up.

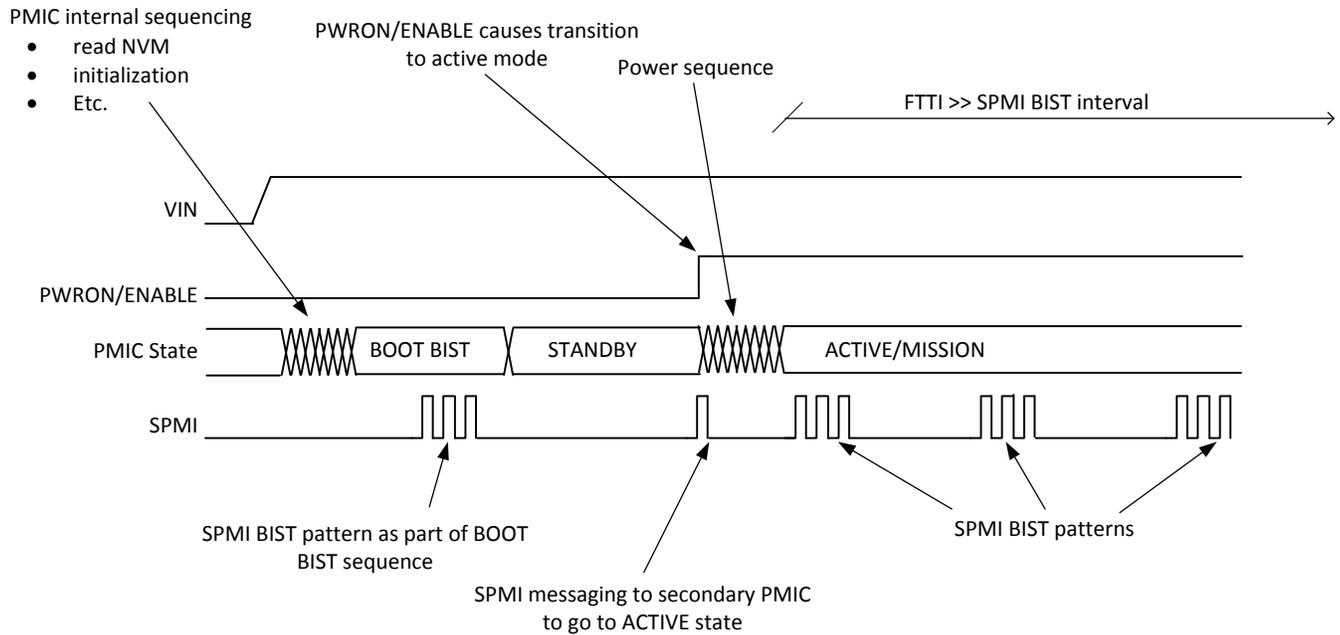


Figure 7-45. SPMI-BIST Operation

After the input power is detected and verified to be at the correct level, the LP876242-Q1 initializes itself by reading the NVM and performs all actions that are needed to prepare for operation. After this initialization, the LP876242-Q1 enters the BOOT BIST state, in which the internal logic performs a series of tests to verify that the LP876242-Q1 device is OK. As part of this test, the SPMI-BIST is performed. After it is completed successfully, the LP876242-Q1 device goes to the standby state and waits 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. The controller device communicates this event through the SPMI bus to all of the target devices. After that, the power-up sequence is executed and LP876242-Q1 enters the configured mission state.

7.4.2.4.1 SPMI Bus during Boot BIST and RUNTIME BIST

During Boot BIST and RUNTIME BIST, both the Logic BIST (LBIST) on the SPMI logic and the SPMI-BIST are performed to check correct operation of the SPMI bus. The LBIST is performed first before the SPMI-BIST during BOOT BIST and RUNTIME BIST. The SPMI-BIST is implemented by reading TID from each target device on the SPMI bus into the controller device, and ensuring they are unique and match the expected amount of target devices. This process of checking the TID of each target device ensures that:

- All SPMI target devices are present in the system as expected
- The SPMI logic blocks are working on the SPMI controller device and all of the SPMI target devices
- The pins and wires on the ICs and PCB are in working order

The SPMI-BIST is initiated by the SPMI controller block in the primary PMIC by writing a request to all SPMI target device(s) (using GTID) to send their TIDs to the SPMI controller block of the primary PMIC. Upon receiving this command from the SPMI controller device, the SPMI target devices request SPMI bus arbitration using the SR-bit protocol. Upon winning the bus arbitration the SPMI target devices transmit their TID into the SPMI target block of the primary PMIC.

The SPMI controller block of the primary PMIC contains a list of all SPMI target device(s) on the SPMI bus and their TIDs in the register set. The SPMI controller block of the primary PMIC reads the TID from each SPMI target device and compares the result with the stored TID for the corresponding SPMI target device. The SPMI controller device has to ensure that every non-zero TID on its list is returned, in order to support use cases in which there are two or more identical SPMI target devices, with same TID, in the system. In these cases, it is mandatory that the expected number of the same TIDs is returned. If no identical PMICs are to be used, then a return of the same TID multiple times is an error due to incorrect assembly of identical PMICs onto the PCB. An

all-zero TID stored in the list of the primary PMIC indicates that there are no SPMI target device(s) present on the SPMI Bus.

7.4.2.4.2 Periodic Checking of the SPMI

The SPMI controller block in the primary PMIC executes the SPMI-BIST periodically while device is operating. The time-period after the SPMI-BIST is repeated according factory-configured settings during the device boot time, and after the device reaches mission states. The factory-configured settings of this SPMI-BIST time period must be the same for all PMICs on the same SPMI network. The SPMI target devices on the SPMI bus expect a request for sending its TID from the SPMI controller device within 1.5x the factory-configured period . This factor 1.5x provides enough margin for clock uncertainty between the SPMI controller device and the SPMI target device.

During mission state operation, the SPMI controller device expects the SPMI target devices to respond to the TID request within the factory-configured polling time-out period . In other words, from the polling start command each SPMI target device must respond within this factory-configured time interval.

During boot time or when the device enters Safe Recovery state, to prevent the SPMI controller device from polling the SPMI target devices too often while one or more of these recovering from a system error such as a thermal shutdown event, the device sets a longer timeout period that allows the SPMI target devices to respond to the SPMI controller device before he SPMI controller device reports an error.

If one or more devices on the SPMI bus cause a violating of the polling time-out period either during start-up or during normal operation, the SPMI controller block in the affected PMIC(s) sets a SPMI error trigger signal to the PFSM of the affected PMIC(s), causing a complete shutdown of the affected PMIC(s). As a result, the affected PMIC(s) no longer respond on the SPMI bus, which in turn is detected by the SPMI controller block off the non-affected PMICs on the SPMI bus. The SPMI controller block in these PMICs sets an SPMI error to the PFSM in these PMICs, causing a complete shutdown of these PMICs. Therefore, all PMICs are finally shutdown if one or more devices on the SPMI bus cause a violating of the polling time-out period .

7.4.2.4.3 SPMI Message Priorities

The SPMI Bus uses the protocol priority levels listed in [Table 7-17](#) for each type of communication message.

Table 7-17. SPMI Message Types and Priorities

| SPMI protocol priority level | Name of priority level in SPMI standard | Message types |
|------------------------------|---|--|
| Highest | A-bit arbitration | State transition messages from target device(s) to controller device |
| | priority arbitration | State transition messages from controller device to target device(s) |
| | SR-bit arbitration | target device TID to controller device |
| Lowest | secondary arbitration | Controller device request of TIDs from target device(s) |

7.5 Control Interfaces

The device has two, exclusive selectable (from factory settings) interfaces. Please refer to the User's Guide of the orderable part number which option has been selected. The first selection is up to two high-speed I²C interfaces. The second selection is one SPI interface. The SPI and I²C1 interfaces are used to fully control and configure the device, and have access to all of the configuration registers and Watchdog registers. During normal operating mode, when the I²C configuration is selected, and the GPIO2 and GPIO3 pins are configured as the SCL_I2C2 and SDA_I2C2 pins, the I²C2 interface becomes the dedicated interface for the Q&A Watchdog communication channel, while I²C1 interface no longer has access to the Watchdog registers. .

7.5.1 CRC Calculation for I²C and SPI Interface Protocols

For safety applications, the LP876242-Q1 supports read and write protocols with embedded CRC data fields. The LP876242-Q1 uses 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 I²C Interface, the LP876242-Q1 uses the above mentioned CRC-8 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 LP876242-Q1 compares this calculated checksum with the R_CRC checksum value that it receives from the MCU. The LP876242-Q1 also uses the above mentioned CRC-8 polynomial to calculate the T_CRC checksum value during a read protocol. This T_CRC checksum value is based on every bit that the LP876242-Q1 receives, except the ACK and NACK bits and except the repeated I2C_ID bits, and the data that the LP876242-Q1 transmits to the MCU during a read protocol. The MCU must use this same CRC-8 polynomial to calculate the checksum value based on the bits that the MCU receives from the LP876242-Q1. The MCU must compare this calculated checksum with the T_CRC checksum value that it receives from the LP876242-Q1.

For the SPI interface, the LP876242-Q1 uses the above mentioned CRC-8 polynomial to calculate the checksum value on every bit it receives from the MCU during a write protocol. The LP876242-Q1 compares this calculated checksum with the R_CRC checksum value, that it receives from the MCU. During a read protocol, the device also uses the above mentioned CRC-8 polynomial to calculate the T_CRC checksum value based on the first 16 bits sent by the MCU, and the next 8 bits the LP876242-Q1 transmits to the MCU. The MCU must use this same CRC-8 polynomial to calculate the checksum value based on the bits which the MCU sends to and receives from the LP876242-Q1, and compare it with the T_CRC checksum value that it receives from the LP876242-Q1.

Figure 7-46 and Figure 7-47 are examples for the 8-bit R_CRC and the T_CRC calculation from 16-bit databus.

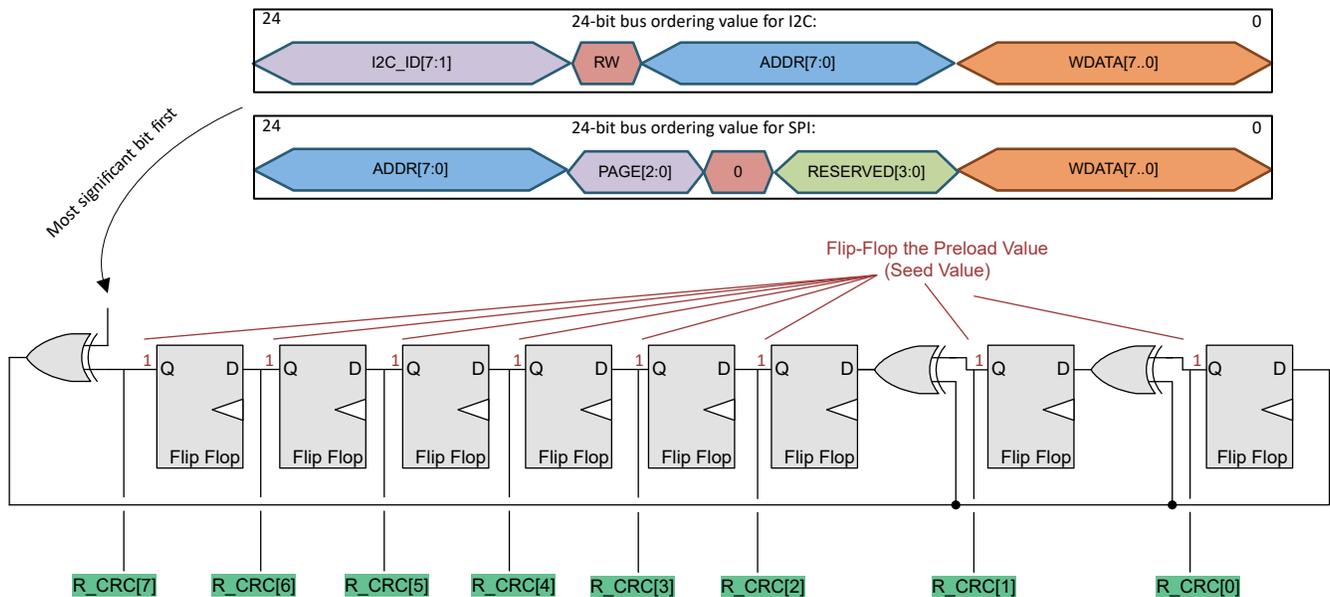


Figure 7-46. Calculation of 8-Bit CRC on Received Data (R_CRC)

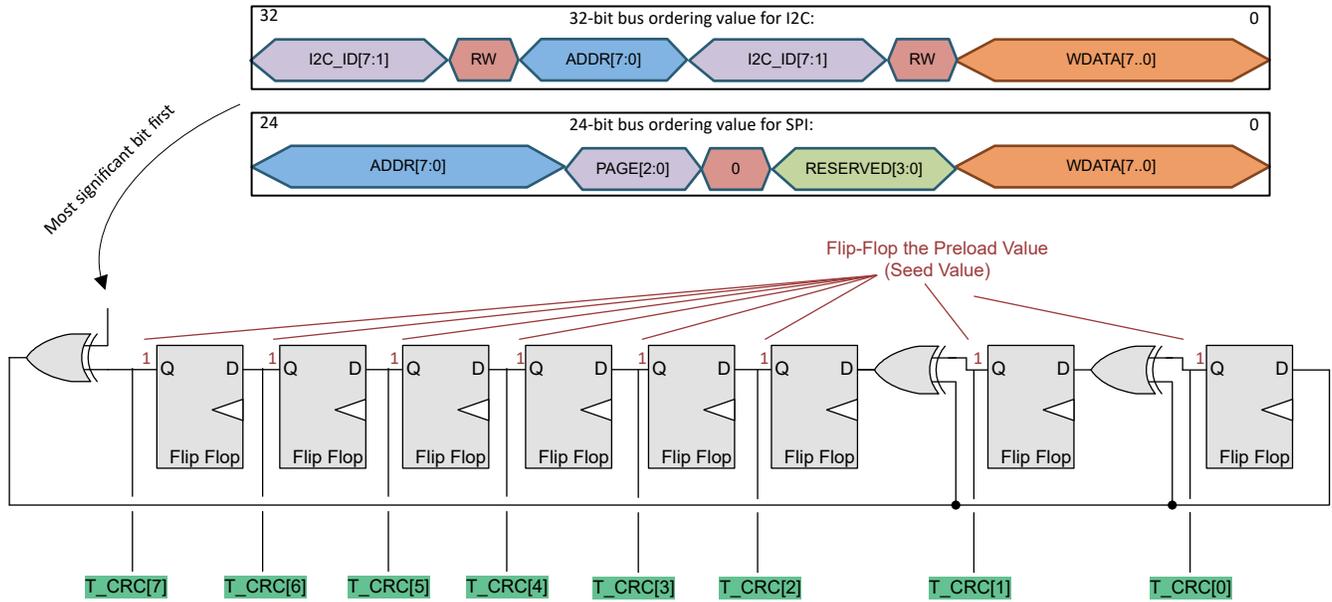


Figure 7-47. Calculation of 8-Bit CRC on Transmitted Data (T_CRC)

7.5.2 I²C-Compatible Interface

The default I²C1 7-bit device address of the LP876242-Q1 device is set to a binary value that is described in the User's Guide of the orderable part number of the LP876242-Q1 PMIC, while the two least-significant bits can be changed for alternative page selection listed under [Section 7.6.1](#). The default 7-bit device address for the I²C2 interface, for accessing the watchdog configuration registers and for operating the watchdog in Q&A mode, is described in the User's Guide of the orderable part number of the LP876242-Q1 PMIC.

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 controller or a target 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.

7.5.2.1 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.

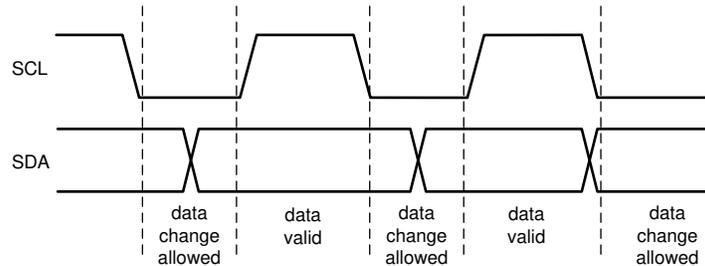


Figure 7-48. Data Validity Diagram

7.5.2.2 Start and Stop Conditions

The device is controlled through an I²C-compatible interface. START and STOP conditions classify the beginning and end of the I²C 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 I²C controller device always generates the START and STOP conditions.

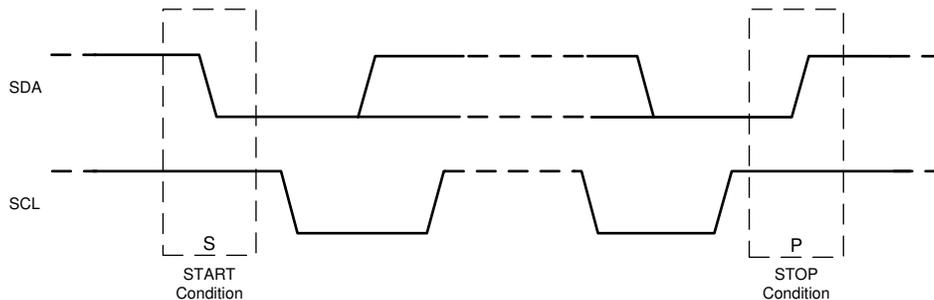


Figure 7-49. Start and Stop Sequences

The I²C bus is considered busy after a START condition and free after a STOP condition. The I²C controller device can generate repeated START conditions during data transmission. A START and a repeated START condition are equivalent function-wise. [Figure 7-50](#) shows the SDA and SCL signal timing for the I²C-compatible bus. For timing values, see the *Specification* section.

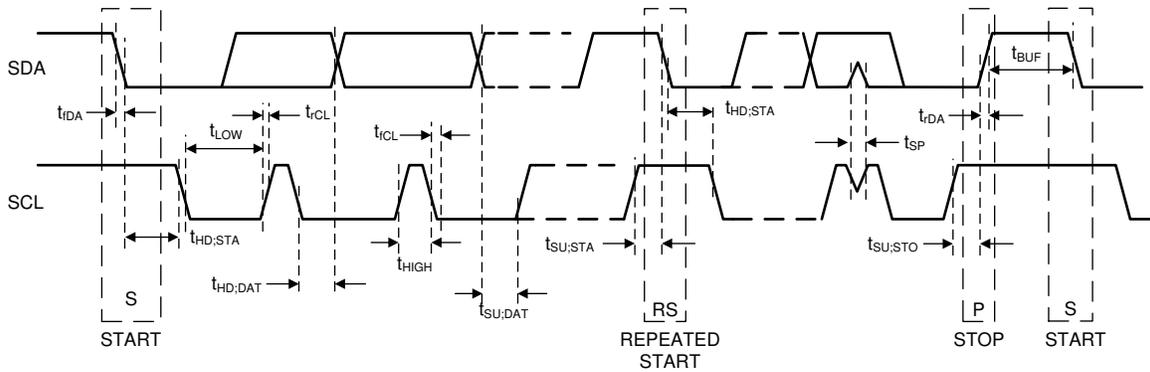


Figure 7-50. I²C-Compatible Timing

7.5.2.3 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 controller device. The controller device 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 controller device 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 target device. This *negative acknowledge* still includes the acknowledge clock pulse (generated by the controller device), but the SDA line is not pulled down.

After the START condition, the bus controller device 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 is written. The third byte contains data to write to the selected register. [Figure 7-51](#) shows an example bit format of device address 110000-Bin = 60Hex.

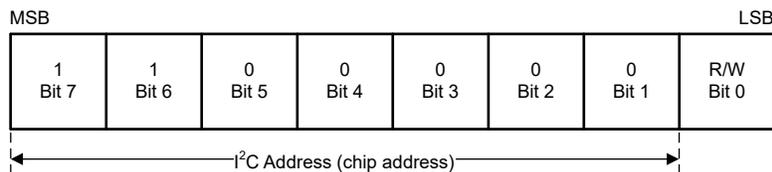


Figure 7-51. Example Device Address

For safety applications, the device supports read and write protocols with embedded CRC data fields. In a write cycle, the I²C controller device (i.e. the MCU) must provide the 8-bit CRC value after sending the write data bits and receiving the ACK from the target (i.e. the LP876242-Q1 PMIC). The CRC value must be calculated from every bit included in the write protocol except the ACK bits from the target. See [CRC Calculation for I2C and SPI Interface Protocols](#). In a read cycle, the I²C target must provide the 8-bit CRC value after sending the read data bits and the ACK bit, and expect to receive the NACK from the controller at the end of the protocol. The CRC value must be calculated from every bit included in the read protocol except the ACK and NACK bits. See [CRC Calculation for I2C and SPI Interface Protocols](#).

Note

If I2C CRC is enabled in the device and an I2C write without R_CRC bits is done, the device does not process the write request. The device does not set any interrupt bit and does not pull the nINT pin low.

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

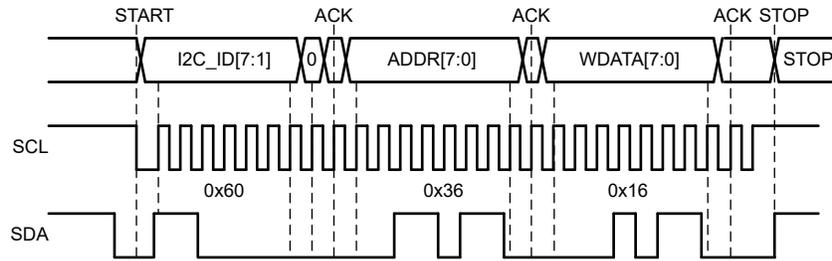
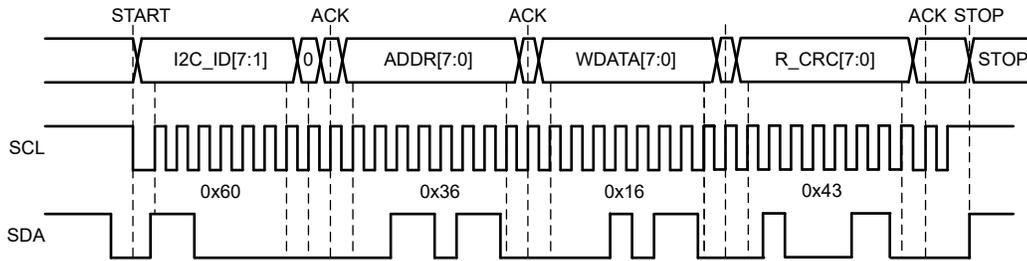
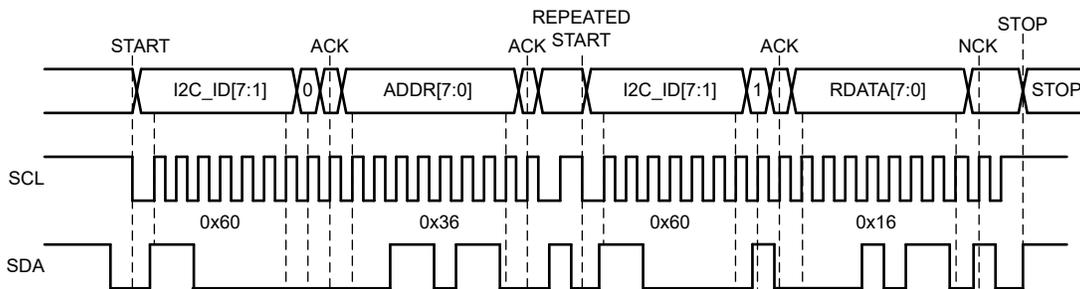


Figure 7-52. I²C Write Cycle without CRC



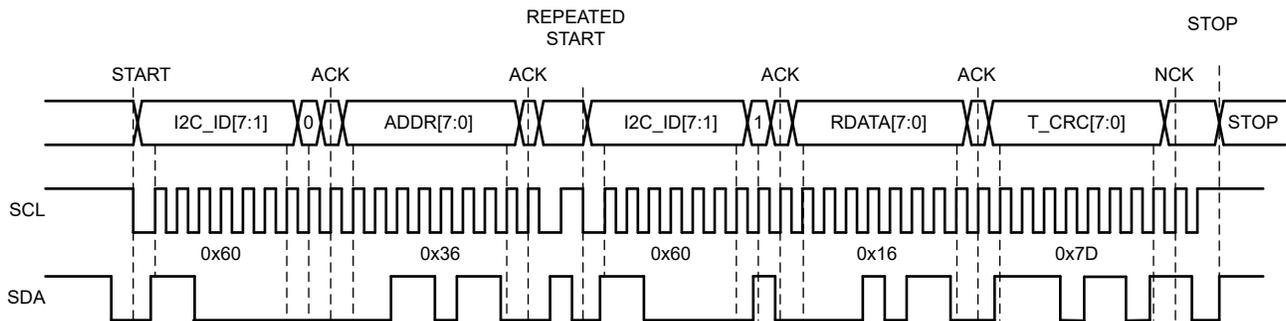
The I²C controller device (i.e. the MCU) provides R_CRC[7:0], which is calculated from the I2C_ID, R/W, ADDR, and the WDATA bits (24 bits). See CRC Calculation for I2C and SPI Interface Protocols.

Figure 7-53. I²C Write Cycle with CRC



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

Figure 7-54. I²C Read Cycle without CRC



The I²C target device (i.e. the LP876242-Q1) provides T_CRC[7:0], which is calculated from the I2C_ID, R/W, ADDR, I2C_ID, R/W, and the RDATA bits (32 bits). See CRC Calculation for I2C and SPI Interface Protocols.

Figure 7-55. I²C READ Cycle with CRC

7.5.2.4 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 7-18](#) lists the writing sequence to two consecutive registers. Note that auto increment feature does not support CRC protocol.

Table 7-18. Auto-Increment Example

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

7.5.3 Serial Peripheral Interface (SPI)

The device supports SPI serial-bus interface and it operates as a peripheral device. The MCU in the system acts as the controller device. A single read and write transmission consists 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: R_CRC[7:0], CRC error code calculated from bits 1-24 sent by the controller device (i.e. the MCU). See [Section 7.5.1](#).
- For Read: Bits 17-24: RDATA[7:0], read data
- For Read with CRC enabled: Bits 25-32: T_CRC[7:0], CRC error code calculated from bits 1-16 sent by the controller device (i.e. the MCU), and bits 17-24, sent by the peripheral device (i.e. the LP876242-Q1). See [Section 7.5.1](#).

In parallel with ADDR[7:0], PAGE[2:0], Read/Write definition and RESERVED[3:0] bits the device sends 16-bit interrupt status using SDO_SPI pin in the following order:

- Bit 1: always 0
- Bits 2-8: status of several interrupts and EN_DRV pin
- Bit 9: always 1
- Bits 10-16: status of several interrupts and EN_DRV pin, with opposite polarity

The status signals are in INT_SPI_STATUS register:

- Bit 8: always 0
- Bit 7: COMM_ADR_ERR_SWINT
- Bit 6: COMM_CRC_ERR_SWINT
- Bit 5: COMM_FRM_ERR_SWINT
- Bit 4: ESM_MCU_PIN_SWINT
- Bit 3: TWARN_SWINT
- Bit 2: WD_SWINT
- Bit 1: EN_DRV_STAT

Please refer to the bit descriptions of the INT_SPI_STATUS register in Register Map for the needed conditions in order to have an error-indication through these status bits.

EN_DRV_STAT bit is showing the live state of the EN_DRV pin, whereas all other status bits are latched in the same way as interrupts indicated with nINT pin. The latched status bits in INT_SPI_STATUS register are cleared by writing 1 to the latched bit. Bits 10-16 are sent for redundancy for bits 2-8 with opposite polarity. Bits 10-16 are always correlating with bits 2-8 and do not change during communication even when the status signal changes.

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

The address, page, data, and CRC are transmitted MSB first. The chip-select signal (CS_SPI) must be low during the cycle transmission. The CS_SPI 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 SCK_SPI clock signal and it is clocked out on the falling edge of SCK_SPI clock signal.

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

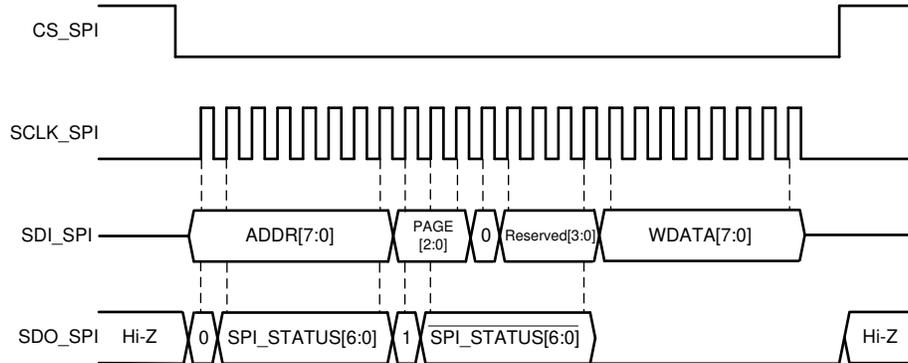


Figure 7-56. SPI Write Cycle

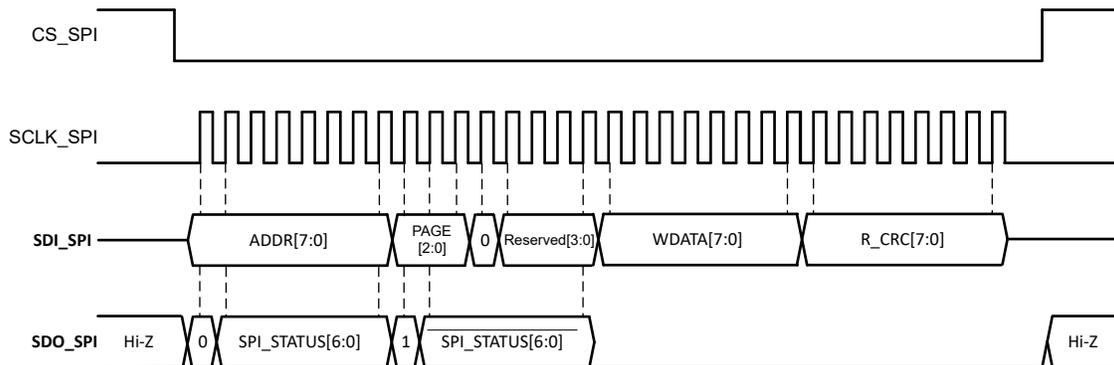


Figure 7-57. SPI Write Cycle with CRC

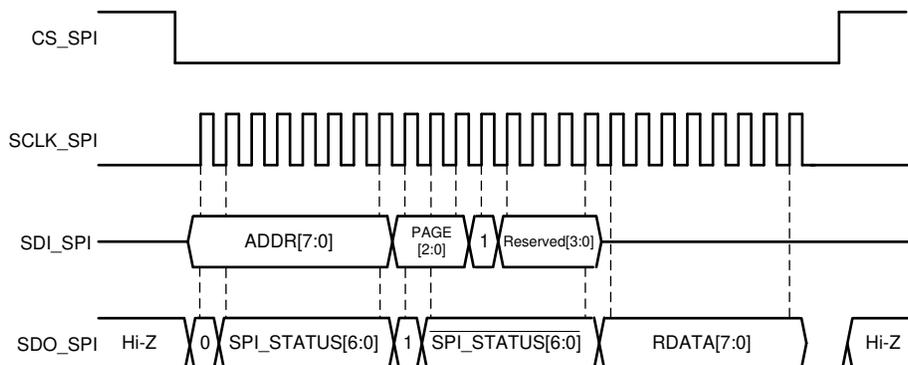


Figure 7-58. SPI Read Cycle

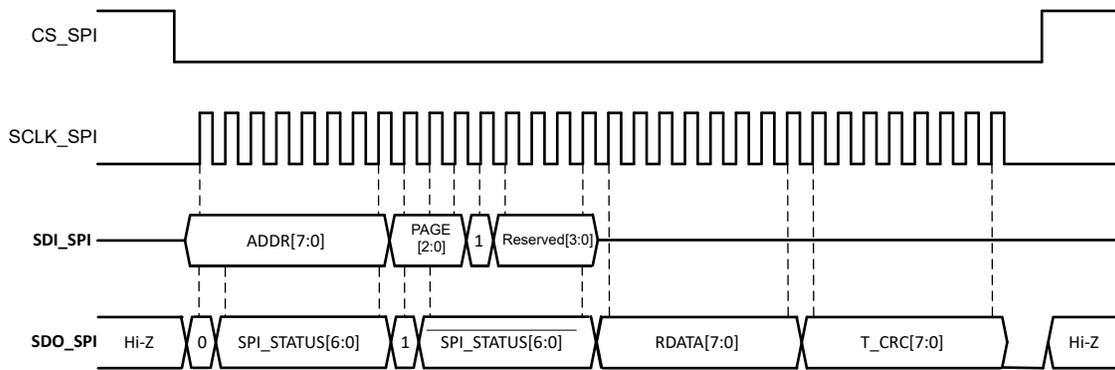


Figure 7-59. SPI Read Cycle with CRC

Note

Due to a digital control erratum in the device, the LP876242-Q1 pulls the nINT pin low and sets interrupt bit COMM_FRM_ERR_INT if the pin CS_SPI is low during device power-up and goes high after completion of the device power-up sequence. After system start-up, the MCU must clear this COMM_FRM_ERR_INT bit such that the LP876242-Q1 can release the nINT pin.

7.6 NVM Configurable Registers

7.6.1 Register Page Partitioning

The registers in the LP876242-Q1 device are organized into five internal pages. Each page represents a different type of register. The below list shows the pages with their register types:

- 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

Note

When I²C Interfaces are used, each of the above listed register pages has its own 6-bit I²C device address. In order to address Page 0 to 3, the two LSBs of the pre-configured I2C1_ID are replaced with 00 for Page 0, 01 for Page 1, 10 for Page 2 and 11 for Page 3. As an example, if I2C1_ID=0x26 (0100110b), Page 0 to 3 have following addresses:

- Page 0: **0100100**
- Page 1: **0100101**
- Page 2: **0100110**
- Page 3: **0100111**

For Page 4 the I²C device address is according register bits I2C2_ID. Therefore, in case both I²C1 and I²C2 Interfaces are used, each LP876242-Q1 device occupies four I²C device addresses (for Page 0, Page 1, Page 2 and Page 3) on the I²C1 bus and one I²C device address (for Page 4) on the I²C2 bus. And in case only I²C1 Interfaces is used, each LP876242-Q1 device occupies five I²C device addresses (for Page 0, Page 1, Page 2, Page 3 and Page 4) on the I²C1 bus. In case multiple devices are used on a common I²C bus, care must be taken to avoid overlapping I²C device addresses.

Note

When SPI Interface is used, the above listed register pages are addresses with the PAGE[2:0] bits: 0x0 addresses Page 0, 0x1 addresses Page 1, 0x2 addresses Page 2, 0x3 addresses Page 3

7.6.2 CRC Protection for Configuration, Control, and Test Registers

The LP876242-Q1 device includes a static CRC-16 engine to protect all the static registers of the device. 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. When the CRC-16 engine detects a mismatch between the calculated and expected CRC-16 values, the interrupt bit REG_CRC_ERR_INT is set and the device forces an 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 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 assumes a default 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 writable register, the CRC-16 engine detects a mismatch between the calculated and expected CRC-16 values, and hence the interrupt bit REG_CRC_ERR_INT is set and the device forces an orderly shutdown sequence to return to the SAFE RECOVERY state.

7.6.3 CRC Protection for User Registers

A dynamic CRC-8 engine exists to protect registers that have values that 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.

Note

If a RESERVED bit in a R/W configuration register gets set to 1h through a I2C/SPI write, the LP876242-Q1 detects a CRC error in the register map. Therefore, it is important that system software involved in the I2C/SPI write-access to the LP876242-Q1 keeps all RESERVED bits (i.e. all bits with the word RESERVED in the Register Field Description tables in the Register Map section at 0h).

7.6.4 Register Write Protection

For safety application, in order to prevent unintentional writes to the control registers, the LP876242-Q1 device implements locking and unlocking mechanisms to many of its configuration/control registers described in the following subsections.

7.6.4.1 ESM and Watchdog Configuration Registers

The configuration registers for the watchdog and the ESM are locked when their monitoring functions are in operation. The locking mechanism and the list of the locked watchdog register is described under [Section 7.3.8.2](#). The locking mechanism and the list of the locked ESM registers is described under

7.6.4.2 User Registers

User registers in page 0, except the ESM and the WDOG configuration registers described in [Section 7.6.4.1](#), 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' activates the lock again. To check the register lock status, user must 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 start-up sequence such as powering up for the first time, waking up from LP_STANDBY, or recovering from SAFE_RECOVERY, the user registers are 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 pre-configured NVM settings, unless mentioned otherwise in the User's Guide of the orderable part number.

7.7 Register Map

7.7.1 LP876242_map Registers

Table 7-19 lists the memory-mapped registers for the LP876242_map registers. All register offset addresses not listed in Table 7-19 should be considered as reserved locations and the register contents should not be modified.

Table 7-19. LP876242_MAP Registers

| Offset | Acronym | Register Name | Section |
|--------|-----------------|---------------|----------------------------------|
| 0x1 | DEV_REV | | Section 7.7.1.1 |
| 0x2 | NVM_CODE_1 | | Section 7.7.1.2 |
| 0x3 | NVM_CODE_2 | | Section 7.7.1.3 |
| 0x4 | BUCK1_CTRL | | Section 7.7.1.4 |
| 0x5 | BUCK1_CONF | | Section 7.7.1.5 |
| 0x6 | BUCK2_CTRL | | Section 7.7.1.6 |
| 0x7 | BUCK2_CONF | | Section 7.7.1.7 |
| 0x8 | BUCK3_CTRL | | Section 7.7.1.8 |
| 0x9 | BUCK3_CONF | | Section 7.7.1.9 |
| 0xA | BUCK4_CTRL | | Section 7.7.1.10 |
| 0xB | BUCK4_CONF | | Section 7.7.1.11 |
| 0xE | BUCK1_VOUT_1 | | Section 7.7.1.12 |
| 0xF | BUCK1_VOUT_2 | | Section 7.7.1.13 |
| 0x10 | BUCK2_VOUT_1 | | Section 7.7.1.14 |
| 0x11 | BUCK2_VOUT_2 | | Section 7.7.1.15 |
| 0x12 | BUCK3_VOUT_1 | | Section 7.7.1.16 |
| 0x13 | BUCK3_VOUT_2 | | Section 7.7.1.17 |
| 0x14 | BUCK4_VOUT_1 | | Section 7.7.1.18 |
| 0x15 | BUCK4_VOUT_2 | | Section 7.7.1.19 |
| 0x18 | BUCK1_PG_WINDOW | | Section 7.7.1.20 |
| 0x19 | BUCK2_PG_WINDOW | | Section 7.7.1.21 |
| 0x1A | BUCK3_PG_WINDOW | | Section 7.7.1.22 |
| 0x1B | BUCK4_PG_WINDOW | | Section 7.7.1.23 |
| 0x2B | VCCA_VMON_CTRL | | Section 7.7.1.24 |
| 0x2C | VCCA_PG_WINDOW | | Section 7.7.1.25 |
| 0x2D | VMON1_PG_WINDOW | | Section 7.7.1.26 |
| 0x2E | VMON1_PG_LEVEL | | Section 7.7.1.27 |
| 0x2F | VMON2_PG_WINDOW | | Section 7.7.1.28 |
| 0x30 | VMON2_PG_LEVEL | | Section 7.7.1.29 |
| 0x31 | GPIO1_CONF | | Section 7.7.1.30 |
| 0x32 | GPIO2_CONF | | Section 7.7.1.31 |
| 0x33 | GPIO3_CONF | | Section 7.7.1.32 |
| 0x34 | GPIO4_CONF | | Section 7.7.1.33 |
| 0x35 | GPIO5_CONF | | Section 7.7.1.34 |
| 0x36 | GPIO6_CONF | | Section 7.7.1.35 |
| 0x37 | GPIO7_CONF | | Section 7.7.1.36 |
| 0x38 | GPIO8_CONF | | Section 7.7.1.37 |
| 0x39 | GPIO9_CONF | | Section 7.7.1.38 |
| 0x3A | GPIO10_CONF | | Section 7.7.1.39 |
| 0x3C | ENABLE_CONF | | Section 7.7.1.40 |
| 0x3D | GPIO_OUT_1 | | Section 7.7.1.41 |
| 0x3E | GPIO_OUT_2 | | Section 7.7.1.42 |

Table 7-19. LP876242_MAP Registers (continued)

| Offset | Acronym | Register Name | Section |
|--------|-------------------|---------------|----------------------------------|
| 0x3F | GPIO_IN_1 | | Section 7.7.1.43 |
| 0x40 | GPIO_IN_2 | | Section 7.7.1.44 |
| 0x41 | RAIL_SEL_1 | | Section 7.7.1.45 |
| 0x43 | RAIL_SEL_3 | | Section 7.7.1.46 |
| 0x44 | FSM_TRIG_SEL_1 | | Section 7.7.1.47 |
| 0x45 | FSM_TRIG_SEL_2 | | Section 7.7.1.48 |
| 0x46 | FSM_TRIG_MASK_1 | | Section 7.7.1.49 |
| 0x47 | FSM_TRIG_MASK_2 | | Section 7.7.1.50 |
| 0x48 | FSM_TRIG_MASK_3 | | Section 7.7.1.51 |
| 0x49 | MASK_BUCK1_2 | | Section 7.7.1.52 |
| 0x4A | MASK_BUCK3_4 | | Section 7.7.1.53 |
| 0x4E | MASK_VMON | | Section 7.7.1.54 |
| 0x4F | MASK_GPIO1_8_FALL | | Section 7.7.1.55 |
| 0x50 | MASK_GPIO1_8_RISE | | Section 7.7.1.56 |
| 0x51 | MASK_GPIO9_10 | | Section 7.7.1.57 |
| 0x52 | MASK_STARTUP | | Section 7.7.1.58 |
| 0x53 | MASK_MISC | | Section 7.7.1.59 |
| 0x54 | MASK_MODERATE_ERR | | Section 7.7.1.60 |
| 0x56 | MASK_FSM_ERR | | Section 7.7.1.61 |
| 0x57 | MASK_COMM_ERR | | Section 7.7.1.62 |
| 0x58 | MASK_READBACK_ERR | | Section 7.7.1.63 |
| 0x59 | MASK_ESM | | Section 7.7.1.64 |
| 0x5A | INT_TOP | | Section 7.7.1.65 |
| 0x5B | INT_BUCK | | Section 7.7.1.66 |
| 0x5C | INT_BUCK1_2 | | Section 7.7.1.67 |
| 0x5D | INT_BUCK3_4 | | Section 7.7.1.68 |
| 0x62 | INT_VMON | | Section 7.7.1.69 |
| 0x63 | INT_GPIO | | Section 7.7.1.70 |
| 0x64 | INT_GPIO1_8 | | Section 7.7.1.71 |
| 0x65 | INT_STARTUP | | Section 7.7.1.72 |
| 0x66 | INT_MISC | | Section 7.7.1.73 |
| 0x67 | INT_MODERATE_ERR | | Section 7.7.1.74 |
| 0x68 | INT_SEVERE_ERR | | Section 7.7.1.75 |
| 0x69 | INT_FSM_ERR | | Section 7.7.1.76 |
| 0x6A | INT_COMM_ERR | | Section 7.7.1.77 |
| 0x6B | INT_READBACK_ERR | | Section 7.7.1.78 |
| 0x6C | INT_ESM | | Section 7.7.1.79 |
| 0x6D | STAT_BUCK1_2 | | Section 7.7.1.80 |
| 0x6E | STAT_BUCK3_4 | | Section 7.7.1.81 |
| 0x72 | STAT_VMON | | Section 7.7.1.82 |
| 0x73 | STAT_STARTUP | | Section 7.7.1.83 |
| 0x74 | STAT_MISC | | Section 7.7.1.84 |
| 0x75 | STAT_MODERATE_ERR | | Section 7.7.1.85 |
| 0x76 | STAT_SEVERE_ERR | | Section 7.7.1.86 |
| 0x77 | STAT_READBACK_ERR | | Section 7.7.1.87 |

Table 7-19. LP876242_MAP Registers (continued)

| Offset | Acronym | Register Name | Section |
|--------|----------------------|---------------|-----------------------------------|
| 0x78 | PGOOD_SEL_1 | | Section 7.7.1.88 |
| 0x7B | PGOOD_SEL_4 | | Section 7.7.1.89 |
| 0x7C | PLL_CTRL | | Section 7.7.1.90 |
| 0x7D | CONFIG_1 | | Section 7.7.1.91 |
| 0x80 | ENABLE_DRV_REG | | Section 7.7.1.92 |
| 0x81 | MISC_CTRL | | Section 7.7.1.93 |
| 0x82 | ENABLE_DRV_STAT | | Section 7.7.1.94 |
| 0x83 | RECOV_CNT_REG_1 | | Section 7.7.1.95 |
| 0x84 | RECOV_CNT_REG_2 | | Section 7.7.1.96 |
| 0x85 | FSM_I2C_TRIGGERS | | Section 7.7.1.97 |
| 0x86 | FSM_NSLEEP_TRIGGERS | | Section 7.7.1.98 |
| 0x87 | BUCK_RESET_REG | | Section 7.7.1.99 |
| 0x88 | SPREAD_SPECTRUM_1 | | Section 7.7.1.100 |
| 0x8B | FSM_STEP_SIZE | | Section 7.7.1.101 |
| 0x8E | USER_SPARE_REGS | | Section 7.7.1.102 |
| 0x8F | ESM_MCU_START_REG | | Section 7.7.1.103 |
| 0x90 | ESM_MCU_DELAY1_REG | | Section 7.7.1.104 |
| 0x91 | ESM_MCU_DELAY2_REG | | Section 7.7.1.105 |
| 0x92 | ESM_MCU_MODE_CFG | | Section 7.7.1.106 |
| 0x93 | ESM_MCU_HMAX_REG | | Section 7.7.1.107 |
| 0x94 | ESM_MCU_HMIN_REG | | Section 7.7.1.108 |
| 0x95 | ESM_MCU_LMAX_REG | | Section 7.7.1.109 |
| 0x96 | ESM_MCU_LMIN_REG | | Section 7.7.1.110 |
| 0x97 | ESM_MCU_ERR_CNT_REG | | Section 7.7.1.111 |
| 0xA1 | REGISTER_LOCK | | Section 7.7.1.112 |
| 0xA7 | CUSTOMER_NVM_ID_REG | | Section 7.7.1.113 |
| 0xA8 | VMON_CONF | | Section 7.7.1.114 |
| 0xA9 | INT_SPI_STATUS | | Section 7.7.1.115 |
| 0xC3 | STARTUP_CTRL | | Section 7.7.1.116 |
| 0xC9 | SCRATCH_PAD_REG_1 | | Section 7.7.1.117 |
| 0xCA | SCRATCH_PAD_REG_2 | | Section 7.7.1.118 |
| 0xCB | SCRATCH_PAD_REG_3 | | Section 7.7.1.119 |
| 0xCC | SCRATCH_PAD_REG_4 | | Section 7.7.1.120 |
| 0xCD | PFSM_DELAY_REG_1 | | Section 7.7.1.121 |
| 0xCE | PFSM_DELAY_REG_2 | | Section 7.7.1.122 |
| 0xCF | PFSM_DELAY_REG_3 | | Section 7.7.1.123 |
| 0xD0 | PFSM_DELAY_REG_4 | | Section 7.7.1.124 |
| 0x401 | WD_ANSWER_REG | | Section 7.7.1.125 |
| 0x402 | WD_QUESTION_ANSW_CNT | | Section 7.7.1.126 |
| 0x403 | WD_WIN1_CFG | | Section 7.7.1.127 |
| 0x404 | WD_WIN2_CFG | | Section 7.7.1.128 |
| 0x405 | WD_LONGWIN_CFG | | Section 7.7.1.129 |
| 0x406 | WD_MODE_REG | | Section 7.7.1.130 |
| 0x407 | WD_QA_CFG | | Section 7.7.1.131 |
| 0x408 | WD_ERR_STATUS | | Section 7.7.1.132 |

Table 7-19. LP876242_MAP Registers (continued)

| Offset | Acronym | Register Name | Section |
|--------|-----------------|---------------|-----------------------------------|
| 0x409 | WD_THR_CFG | | Section 7.7.1.133 |
| 0x40A | WD_FAIL_CNT_REG | | Section 7.7.1.134 |

Complex bit access types are encoded to fit into small table cells. [Table 7-20](#) shows the codes that are used for access types in this section.

Table 7-20. LP876242_map 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 |

7.7.1.1 DEV_REV Register (Offset = 0x1) [Reset = 0x00]

DEV_REV is shown in [Figure 7-60](#) and described in [Table 7-21](#).

Return to the [Table 7-19](#).

Figure 7-60. DEV_REV Register

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

Table 7-21. DEV_REV Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|--------------|------|-------|--|
| 7:0 | TI_DEVICE_ID | R/W | 0b | Refer to Technical Reference Manual / User's Guide for specific numbering. |

7.7.1.2 NVM_CODE_1 Register (Offset = 0x2) [Reset = 0x00]

NVM_CODE_1 is shown in [Figure 7-61](#) and described in [Table 7-22](#).

Return to the [Table 7-19](#).

Figure 7-61. NVM_CODE_1 Register

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

Table 7-22. NVM_CODE_1 Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|-----------|------|-------|---|
| 7:0 | TI_NVM_ID | R/W | 0b | NVM version of the IC This bit is Read-Only for I2C/SPI access. (Default from NVM memory) |

7.7.1.3 NVM_CODE_2 Register (Offset = 0x3) [Reset = 0x00]

 NVM_CODE_2 is shown in [Figure 7-62](#) and described in [Table 7-23](#).

 Return to the [Table 7-19](#).

Figure 7-62. NVM_CODE_2 Register

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

Table 7-23. NVM_CODE_2 Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|------------|------|-------|--|
| 7:0 | TI_NVM_REV | R/W | 0b | NVM revision of the IC This bit is Read-Only for I2C/SPI access. (Default from NVM memory) |

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

BUCK1_CTRL is shown in [Figure 7-63](#) and described in [Table 7-24](#).

Return to the [Table 7-19](#).

Figure 7-63. BUCK1_CTRL Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------------|----------|------------|-------------------|------------|----------|------------|----------|
| BUCK1_RV_SE L | RESERVED | BUCK1_PLDN | BUCK1_VMON _EN | BUCK1_VSEL | RESERVED | 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 7-24. 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 | RESERVED | R/W | 0b | |
| 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 |

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

 BUCK1_CONF is shown in [Figure 7-64](#) and described in [Table 7-25](#).

 Return to the [Table 7-19](#).

Figure 7-64. 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 7-25. 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. Maximum programmable current limit may be limited based on device settings. (Default from NVM memory) 0b = Reserved 1b = Reserved 10b = 2.5 A 11b = 3.5 A 100b = 4.5 A 101b = Reserved 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 |

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

BUCK2_CTRL is shown in [Figure 7-65](#) and described in [Table 7-26](#).

Return to the [Table 7-19](#).

Figure 7-65. BUCK2_CTRL Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------------|----------|------------|-------------------|------------|----------|------------|----------|
| BUCK2_RV_SE L | 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 7-26. 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: LP876242-Q1 device supports only forced PWM operation. (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 |

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

 BUCK2_CONF is shown in [Figure 7-66](#) and described in [Table 7-27](#).

 Return to the [Table 7-19](#).

Figure 7-66. 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 7-27. 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. Maximum programmable current limit may be limited based on device settings. (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 = 6.5 A 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 |

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

BUCK3_CTRL is shown in [Figure 7-67](#) and described in [Table 7-28](#).

Return to the [Table 7-19](#).

Figure 7-67. BUCK3_CTRL Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------------|----------|------------|-------------------|------------|----------|------------|----------|
| BUCK3_RV_SE L | RESERVED | BUCK3_PLDN | BUCK3_VMON _EN | BUCK3_VSEL | RESERVED | 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 7-28. 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 | RESERVED | R/W | 0b | |
| 1 | BUCK3_FPWM | R/W | 1b | Forces the BUCK3 regulator to operate in PWM mode: LP876242-Q1 device supports only forced PWM operation. (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 |

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

 BUCK3_CONF is shown in [Figure 7-68](#) and described in [Table 7-29](#).

 Return to the [Table 7-19](#).

Figure 7-68. 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 7-29. 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. Maximum programmable current limit may be limited based on device settings. (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 |

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

BUCK4_CTRL is shown in [Figure 7-69](#) and described in [Table 7-30](#).

Return to the [Table 7-19](#).

Figure 7-69. BUCK4_CTRL Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------------|----------|------------|-------------------|------------|----------|------------|----------|
| BUCK4_RV_SE L | 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 7-30. 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: LP876242-Q1 device supports only forced PWM operation. (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 |

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

 BUCK4_CONF is shown in [Figure 7-70](#) and described in [Table 7-31](#).

 Return to the [Table 7-19](#).

Figure 7-70. 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 7-31. 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. Maximum programmable current limit may be limited based on device settings. (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 |

7.7.1.12 BUCK1_VOUT_1 Register (Offset = 0xE) [Reset = 0x00]

BUCK1_VOUT_1 is shown in [Figure 7-71](#) and described in [Table 7-32](#).

Return to the [Table 7-19](#).

Figure 7-71. BUCK1_VOUT_1 Register

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

Table 7-32. 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) |

7.7.1.13 BUCK1_VOUT_2 Register (Offset = 0xF) [Reset = 0x00]

 BUCK1_VOUT_2 is shown in [Figure 7-72](#) and described in [Table 7-33](#).

 Return to the [Table 7-19](#).

Figure 7-72. BUCK1_VOUT_2 Register

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

Table 7-33. 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) |

7.7.1.14 BUCK2_VOUT_1 Register (Offset = 0x10) [Reset = 0x00]

BUCK2_VOUT_1 is shown in [Figure 7-73](#) and described in [Table 7-34](#).

Return to the [Table 7-19](#).

Figure 7-73. BUCK2_VOUT_1 Register

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

Table 7-34. 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) |

7.7.1.15 BUCK2_VOUT_2 Register (Offset = 0x11) [Reset = 0x00]

 BUCK2_VOUT_2 is shown in [Figure 7-74](#) and described in [Table 7-35](#).

 Return to the [Table 7-19](#).

Figure 7-74. BUCK2_VOUT_2 Register

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

Table 7-35. 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) |

7.7.1.16 BUCK3_VOUT_1 Register (Offset = 0x12) [Reset = 0x00]

BUCK3_VOUT_1 is shown in [Figure 7-75](#) and described in [Table 7-36](#).

Return to the [Table 7-19](#).

Figure 7-75. BUCK3_VOUT_1 Register

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

Table 7-36. 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) |

7.7.1.17 BUCK3_VOUT_2 Register (Offset = 0x13) [Reset = 0x00]

 BUCK3_VOUT_2 is shown in [Figure 7-76](#) and described in [Table 7-37](#).

 Return to the [Table 7-19](#).

Figure 7-76. BUCK3_VOUT_2 Register

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

Table 7-37. 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) |

7.7.1.18 BUCK4_VOUT_1 Register (Offset = 0x14) [Reset = 0x00]

BUCK4_VOUT_1 is shown in [Figure 7-77](#) and described in [Table 7-38](#).

Return to the [Table 7-19](#).

Figure 7-77. BUCK4_VOUT_1 Register

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

Table 7-38. 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) |

7.7.1.19 BUCK4_VOUT_2 Register (Offset = 0x15) [Reset = 0x00]

 BUCK4_VOUT_2 is shown in [Figure 7-78](#) and described in [Table 7-39](#).

 Return to the [Table 7-19](#).

Figure 7-78. BUCK4_VOUT_2 Register

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

Table 7-39. 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) |

7.7.1.20 BUCK1_PG_WINDOW Register (Offset = 0x18) [Reset = 0x00]

BUCK1_PG_WINDOW is shown in [Figure 7-79](#) and described in [Table 7-40](#).

Return to the [Table 7-19](#).

Figure 7-79. 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 7-40. 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 |

7.7.1.21 BUCK2_PG_WINDOW Register (Offset = 0x19) [Reset = 0x00]

 BUCK2_PG_WINDOW is shown in [Figure 7-80](#) and described in [Table 7-41](#).

 Return to the [Table 7-19](#).

Figure 7-80. 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 7-41. 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 |

7.7.1.22 BUCK3_PG_WINDOW Register (Offset = 0x1A) [Reset = 0x00]

BUCK3_PG_WINDOW is shown in [Figure 7-81](#) and described in [Table 7-42](#).

Return to the [Table 7-19](#).

Figure 7-81. 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 7-42. 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 |

7.7.1.23 BUCK4_PG_WINDOW Register (Offset = 0x1B) [Reset = 0x00]

 BUCK4_PG_WINDOW is shown in [Figure 7-82](#) and described in [Table 7-43](#).

 Return to the [Table 7-19](#).

Figure 7-82. 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 7-43. 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 |

7.7.1.24 VCCA_VMON_CTRL Register (Offset = 0x2B) [Reset = 0x00]

VCCA_VMON_CTRL is shown in [Figure 7-83](#) and described in [Table 7-44](#).

Return to the [Table 7-19](#).

Figure 7-83. VCCA_VMON_CTRL Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------------------|---|--------------|---|----------|--------------|----------|--------------|
| VMON_DEGLITCH_SEL | | VMON2_RV_SEL | | VMON2_EN | VMON1_RV_SEL | VMON1_EN | VCCA_VMON_EN |
| R/W-0b | | R/W-0b | | R/W-0b | R/W-0b | R/W-0b | R/W-0b |

Table 7-44. VCCA_VMON_CTRL Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|-------------------|------|-------|--|
| 7:5 | VMON_DEGLITCH_SEL | R/W | 0b | Deglintch time select for VCCA_VMON / BUCKx_VMON / VMONx voltage monitors. (Default from NVM memory) 0b = 4 us / 4 us / 4 us 1b = 20 us / 20 us / 20 us 10b = 0.5 us / 0.5 us / 0.5 us 11b = 4 us / 0.5 us / 0.5 us 100b = 20 us / 0.5 us / 0.5 us 101b = 0.5 us / 4 us / 4 us 110b = 4 us / 4 us / 4 us 111b = 20 us / 4 us / 4 us |
| 4 | VMON2_RV_SEL | R/W | 0b | Select residual voltage checking for VMON2 pin. (Default from NVM memory) 0b = Disabled 1b = Enabled |
| 3 | VMON2_EN | R/W | 0b | Enable VMON2 OV and UV comparators: (Default from NVM memory) 0b = OV and UV comparators are disabled 1b = OV and UV comparators are enabled |
| 2 | VMON1_RV_SEL | R/W | 0b | Select residual voltage checking for VMON1 pin. (Default from NVM memory) 0b = Disabled 1b = Enabled |
| 1 | VMON1_EN | R/W | 0b | Enable VMON1 OV and UV comparators: (Default from NVM memory) 0b = OV and UV comparators are disabled 1b = OV and UV comparators are enabled |
| 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. |

7.7.1.25 VCCA_PG_WINDOW Register (Offset = 0x2C) [Reset = 0x00]

 VCCA_PG_WINDOW is shown in [Figure 7-84](#) and described in [Table 7-45](#).

 Return to the [Table 7-19](#).

Figure 7-84. 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-0b | R/W-0b | | | R/W-0b | | |

Table 7-45. VCCA_PG_WINDOW Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|-------------|------|-------|--|
| 7 | RESERVED | R/W | 0b | |
| 6 | VCCA_PG_SET | R/W | 0b | 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% |

7.7.1.26 VMON1_PG_WINDOW Register (Offset = 0x2D) [Reset = 0x00]

VMON1_PG_WINDOW is shown in [Figure 7-85](#) and described in [Table 7-46](#).

Return to the [Table 7-19](#).

Figure 7-85. VMON1_PG_WINDOW Register

| | | | | | | | |
|----------|-------------|--------------|---|---|--------------|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RESERVED | VMON1_RANGE | VMON1_UV_THR | | | VMON1_OV_THR | | |
| R/W-0b | R/W-0b | R/W-0b | | | R/W-0b | | |

Table 7-46. VMON1_PG_WINDOW Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|--------------|------|-------|---|
| 7 | RESERVED | R/W | 0b | |
| 6 | VMON1_RANGE | R/W | 0b | Select OV/UV voltage monitoring range: (Default from NVM memory) 0b = 0.3 - 3.34 V 1b = 3.35 - 5.0 V |
| 5:3 | VMON1_UV_THR | R/W | 0b | Powergood low threshold level for VMON1. Threshold values in brackets are for extended voltage range (VMON1_RANGE = 1): (Default from NVM memory) 0b = -3% / -30 mV / (-150 mV) 1b = -3.5% / -35 mV / (-175 mV) 10b = -4% / -40 mV / (-200 mV) 11b = -5% / -50 mV / (-250 mV) 100b = -6% / -60 mV / (-300 mV) 101b = -7% / -70 mV / (-350 mV) 110b = -8% / -80 mV / (-400 mV) 111b = -10% / -100 mV / (-500 mV) |
| 2:0 | VMON1_OV_THR | R/W | 0b | Powergood high threshold level for VMON1. Threshold values in brackets are for extended voltage range (VMON1_RANGE = 1): (Default from NVM memory) 0b = +3% / +30 mV / (+150 mV) 1b = +3.5% / +35 mV / (+175 mV) 10b = +4% / +40 mV / (+200 mV) 11b = +5% / +50 mV / (+250 mV) 100b = +6% / +60 mV / (+300 mV) 101b = +7% / +70 mV / (+350 mV) 110b = +8% / +80 mV / (+400 mV) 111b = +10% / +100mV / (+500 mV) |

7.7.1.27 VMON1_PG_LEVEL Register (Offset = 0x2E) [Reset = 0x00]

 VMON1_PG_LEVEL is shown in [Figure 7-86](#) and described in [Table 7-47](#).

 Return to the [Table 7-19](#).

Figure 7-86. VMON1_PG_LEVEL Register

| | | | | | | | |
|--------------|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| VMON1_PG_SET | | | | | | | |
| R/W-0b | | | | | | | |

Table 7-47. VMON1_PG_LEVEL Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|--------------|------|-------|--|
| 7:0 | VMON1_PG_SET | R/W | 0b | Powergood voltage level of VMON1 pin, VMON1_OV_THR[2:0] and VMON1_UV_THR[2:0] defines the threshold levels. See Voltage monitoring chapter for voltage levels. (Default from NVM memory) |

7.7.1.28 VMON2_PG_WINDOW Register (Offset = 0x2F) [Reset = 0x00]

VMON2_PG_WINDOW is shown in [Figure 7-87](#) and described in [Table 7-48](#).

Return to the [Table 7-19](#).

Figure 7-87. VMON2_PG_WINDOW Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------|-------------|--------------|---|---|--------------|---|---|
| RESERVED | VMON2_RANGE | VMON2_UV_THR | | | VMON2_OV_THR | | |
| R/W-0b | R/W-0b | R/W-0b | | | R/W-0b | | |

Table 7-48. VMON2_PG_WINDOW Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|--------------|------|-------|--|
| 7 | RESERVED | R/W | 0b | |
| 6 | VMON2_RANGE | R/W | 0b | Select OV/UV voltage monitoring range: (Default from NVM memory) 0b = 0.3 - 3.34 V 1b = 3.35 - 5.0 V |
| 5:3 | VMON2_UV_THR | R/W | 0b | Powergood low threshold level for VMON2. Threshold values in brackets are for extended voltage range (VMON2_RANGE = 1): (Default from NVM memory) 0b = -3% / -30 mV / (-150 mV) 1b = -3.5% / -35 mV / (-175 mV) 10b = -4% / -40 mV / (-200 mV) 11b = -5% / -50 mV / (-250 mV) 100b = -6% / -60 mV / (-300 mV) 101b = -7% / -70 mV / (-350 mV) 110b = -8% / -80 mV / (-400 mV) 111b = -10% / -100 mV / (-500 mV) |
| 2:0 | VMON2_OV_THR | R/W | 0b | Powergood high threshold level for VMON2. Threshold values in brackets are for extended voltage range (VMON2_RANGE = 1): (Default from NVM memory) 0b = +3% / +30mV / (+150 mV) 1b = +3.5% / +35 mV / (+175 mV) 10b = +4% / +40 mV / (+200 mV) 11b = +5% / +50 mV / (+250 mV) 100b = +6% / +60 mV / (+300 mV) 101b = +7% / +70 mV / (+350 mV) 110b = +8% / +80 mV / (+400 mV) 111b = +10% / +100mV / (+500 mV) |

7.7.1.29 VMON2_PG_LEVEL Register (Offset = 0x30) [Reset = 0x00]

 VMON2_PG_LEVEL is shown in [Figure 7-88](#) and described in [Table 7-49](#).

 Return to the [Table 7-19](#).

Figure 7-88. VMON2_PG_LEVEL Register

| | | | | | | | |
|--------------|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| VMON2_PG_SET | | | | | | | |
| R/W-0b | | | | | | | |

Table 7-49. VMON2_PG_LEVEL Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|--------------|------|-------|--|
| 7:0 | VMON2_PG_SET | R/W | 0b | Powergood voltage level of VMON2 pin, VMON2_OV_THR[2:0] and VMON2_UV_THR[2:0] defines the threshold levels. See Voltage monitoring chapter for voltage levels. (Default from NVM memory) |

7.7.1.30 GPIO1_CONF Register (Offset = 0x31) [Reset = 0x2A]

GPIO1_CONF is shown in [Figure 7-89](#) and described in [Table 7-50](#).

Return to the [Table 7-19](#).

Figure 7-89. 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-1b | | R/W-0b | | R/W-1b | R/W-0b | R/W-1b | R/W-0b |

Table 7-50. GPIO1_CONF Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|-------------------|------|-------|--|
| 7:5 | GPIO1_SEL | R/W | 1b | GPIO1 signal function: (Default from NVM memory) 0b = GPIO1 1b = EN_DRV 10b = NRSTOUT_SOC 11b = PGOOD 100b = NSLEEP1 101b = NSLEEP2 110b = WKUP1 111b = WKUP2 |
| 4 | GPIO1_DEGLITCH_EN | R/W | 0b | GPIO1 signal glitch time when signal direction is input: (Default from NVM memory) 0b = No glitch, only synchronization. 1b = 8 us glitch 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 |

7.7.1.31 GPIO2_CONF Register (Offset = 0x32) [Reset = 0x0A]

 GPIO2_CONF is shown in [Figure 7-90](#) and described in [Table 7-51](#).

 Return to the [Table 7-19](#).

Figure 7-90. 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 7-51. 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 = SCL_I2C2 10b = CS_SPI 11b = TRIG_WDOG 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 |

7.7.1.32 GPIO3_CONF Register (Offset = 0x33) [Reset = 0x0A]

GPIO3_CONF is shown in [Figure 7-91](#) and described in [Table 7-52](#).

Return to the [Table 7-19](#).

Figure 7-91. GPIO3_CONF Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|---|-------------------|---|----------------|------------------|----------|-----------|
| GPIO3_SEL | | GPIO3_DEGLITCH_EN | | GPIO3_PU_PD_EN | GPIO3_PU_SE L | GPIO3_OD | GPIO3_DIR |
| R/W-0b | | R/W-0b | | R/W-1b | R/W-0b | R/W-1b | R/W-0b |

Table 7-52. 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 = SDA_I2C2 10b = SDO_SPI 11b = SDO_SPI 100b = NSLEEP1 101b = NSLEEP2 110b = WKUP1 111b = 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 |

7.7.1.33 GPIO4_CONF Register (Offset = 0x34) [Reset = 0x0A]

 GPIO4_CONF is shown in [Figure 7-92](#) and described in [Table 7-53](#).

 Return to the [Table 7-19](#).

Figure 7-92. 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 7-53. 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 = ENABLE 10b = TRIG_WDOG 11b = BUCK1_VMON. Buck1 voltage monitoring input is GPIO4 instead of FB_B1. 100b = NSLEEP1 101b = NSLEEP2 110b = WKUP1 111b = 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 |

7.7.1.34 GPIO5_CONF Register (Offset = 0x35) [Reset = 0x0A]

GPIO5_CONF is shown in [Figure 7-93](#) and described in [Table 7-54](#).

Return to the [Table 7-19](#).

Figure 7-93. 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 7-54. 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 = SYNCCLKIN 10b = SYNCCLKOUT 11b = nRSTOUT_SOC 100b = NSLEEP1 101b = NSLEEP2 110b = WKUP1 111b = WKUP2 |
| 4 | GPIO5_DEGLITCH_EN | R/W | 0b | GPIO5 signal glitch time when signal direction is input: (Default from NVM memory) 0b = No glitch, only synchronization. 1b = 8 us glitch 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 |

7.7.1.35 GPIO6_CONF Register (Offset = 0x36) [Reset = 0x0A]

 GPIO6_CONF is shown in [Figure 7-94](#) and described in [Table 7-55](#).

 Return to the [Table 7-19](#).

Figure 7-94. GPIO6_CONF Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|---|-------------------|---|----------------|------------------|----------|-----------|
| GPIO6_SEL | | GPIO6_DEGLITCH_EN | | GPIO6_PU_PD_EN | GPIO6_PU_SE L | GPIO6_OD | GPIO6_DIR |
| R/W-0b | | R/W-0b | | R/W-1b | R/W-0b | R/W-1b | R/W-0b |

Table 7-55. 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 = nERR_MCU 10b = SYNCCLKOUT 11b = PGOOD 100b = NSLEEP1 101b = NSLEEP2 110b = WKUP1 111b = WKUP2 |
| 4 | GPIO6_DEGLITCH_EN | R/W | 0b | GPIO6 signal glitch time when signal direction is input: (Default from NVM memory) 0b = No glitch, only synchronization. 1b = 8 us glitch 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 |

7.7.1.36 GPIO7_CONF Register (Offset = 0x37) [Reset = 0x0A]

GPIO7_CONF is shown in [Figure 7-95](#) and described in [Table 7-56](#).

Return to the [Table 7-19](#).

Figure 7-95. GPIO7_CONF Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|---|-------------------|---|----------------|------------------|----------|-----------|
| GPIO7_SEL | | GPIO7_DEGLITCH_EN | | GPIO7_PU_PD_EN | GPIO7_PU_SE L | GPIO7_OD | GPIO7_DIR |
| R/W-0b | | R/W-0b | | R/W-1b | R/W-0b | R/W-1b | R/W-0b |

Table 7-56. 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 = nERR_MCU 10b = REFOUT 11b = VMON1 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 |

7.7.1.37 GPIO8_CONF Register (Offset = 0x38) [Reset = 0x0A]

 GPIO8_CONF is shown in [Figure 7-96](#) and described in [Table 7-57](#).

 Return to the [Table 7-19](#).

Figure 7-96. 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 7-57. 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 = SCLK_SPMI 10b = VMON2 11b = VMON2 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 |

7.7.1.38 GPIO9_CONF Register (Offset = 0x39) [Reset = 0x0A]

GPIO9_CONF is shown in [Figure 7-97](#) and described in [Table 7-58](#).

Return to the [Table 7-19](#).

Figure 7-97. 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 7-58. 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 = SDATA_SPMI 10b = PGOOD 11b = SYNCCLKIN 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 |

7.7.1.39 GPIO10_CONF Register (Offset = 0x3A) [Reset = 0x0A]

 GPIO10_CONF is shown in [Figure 7-98](#) and described in [Table 7-59](#).

 Return to the [Table 7-19](#).

Figure 7-98. 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 7-59. 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 = nRSTOUT 10b = nRSTOUT_SOC 11b = nRSTOUT_SOC 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 |

7.7.1.40 ENABLE_CONF Register (Offset = 0x3C) [Reset = 0x88]

 ENABLE_CONF is shown in [Figure 7-99](#) and described in [Table 7-60](#).

 Return to the [Table 7-19](#).

Figure 7-99. ENABLE_CONF Register

| | | | | | | | |
|----------|---|------------|-----------|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RESERVED | | ENABLE_POL | RESERVED | | | | |
| R/W-10b | | R/W-0b | R/W-1000b | | | | |

Table 7-60. ENABLE_CONF Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|------------|------|-------|--|
| 7:6 | RESERVED | R/W | 10b | |
| 5 | ENABLE_POL | R/W | 0b | Control for ENABLE pin polarity: (Default from NVM memory) 0b = Active high 1b = Active low |
| 4:0 | RESERVED | R/W | 1000b | |

7.7.1.41 GPIO_OUT_1 Register (Offset = 0x3D) [Reset = 0x00]

 GPIO_OUT_1 is shown in [Figure 7-100](#) and described in [Table 7-61](#).

 Return to the [Table 7-19](#).

Figure 7-100. 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 |

Table 7-61. 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 |

7.7.1.42 GPIO_OUT_2 Register (Offset = 0x3E) [Reset = 0x00]

GPIO_OUT_2 is shown in [Figure 7-101](#) and described in [Table 7-62](#).

Return to the [Table 7-19](#).

Figure 7-101. GPIO_OUT_2 Register

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

Table 7-62. GPIO_OUT_2 Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|------------|------|-------|---|
| 7:2 | RESERVED | R/W | 0b | |
| 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 |

7.7.1.43 GPIO_IN_1 Register (Offset = 0x3F) [Reset = 0x00]

 GPIO_IN_1 is shown in [Figure 7-102](#) and described in [Table 7-63](#).

 Return to the [Table 7-19](#).

Figure 7-102. 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 |

Table 7-63. 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 |

7.7.1.44 GPIO_IN_2 Register (Offset = 0x40) [Reset = 0x00]

GPIO_IN_2 is shown in [Figure 7-103](#) and described in [Table 7-64](#).

Return to the [Table 7-19](#).

Figure 7-103. GPIO_IN_2 Register

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

Table 7-64. GPIO_IN_2 Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|-----------|------|-------|--|
| 7:2 | RESERVED | R | 0b | |
| 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 |

7.7.1.45 RAIL_SEL_1 Register (Offset = 0x41) [Reset = 0x00]

RAIL_SEL_1 is shown in [Figure 7-104](#) and described in [Table 7-65](#).

Return to the [Table 7-19](#).

Figure 7-104. 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 7-65. 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 |

7.7.1.46 RAIL_SEL_3 Register (Offset = 0x43) [Reset = 0x00]

RAIL_SEL_3 is shown in [Figure 7-105](#) and described in [Table 7-66](#).

Return to the [Table 7-19](#).

Figure 7-105. RAIL_SEL_3 Register

| | | | | | | | |
|---------------|---|---------------|---|--------------|---|----------|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| VMON2_GRP_SEL | | VMON1_GRP_SEL | | VCCA_GRP_SEL | | RESERVED | |
| R/W-0b | | R/W-0b | | R/W-0b | | R/W-0b | |

Table 7-66. RAIL_SEL_3 Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|---------------|------|-------|--|
| 7:6 | VMON2_GRP_SEL | R/W | 0b | Rail group selection for VMON2 monitoring: (Default from NVM memory) 0b = No group assigned 1b = MCU rail group 10b = SOC rail group 11b = OTHER rail group |
| 5:4 | VMON1_GRP_SEL | R/W | 0b | Rail group selection for VMON1 monitoring: (Default from NVM memory) 0b = No group assigned 1b = MCU rail group 10b = SOC rail group 11b = OTHER rail group |
| 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 | RESERVED | R/W | 0b | |

7.7.1.47 FSM_TRIG_SEL_1 Register (Offset = 0x44) [Reset = 0x00]

 FSM_TRIG_SEL_1 is shown in [Figure 7-106](#) and described in [Table 7-67](#).

 Return to the [Table 7-19](#).

Figure 7-106. 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 7-67. 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 |

7.7.1.48 FSM_TRIG_SEL_2 Register (Offset = 0x45) [Reset = 0x00]

FSM_TRIG_SEL_2 is shown in [Figure 7-107](#) and described in [Table 7-68](#).

Return to the [Table 7-19](#).

Figure 7-107. FSM_TRIG_SEL_2 Register

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

Table 7-68. 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 |

7.7.1.49 FSM_TRIG_MASK_1 Register (Offset = 0x46) [Reset = 0x00]

 FSM_TRIG_MASK_1 is shown in [Figure 7-108](#) and described in [Table 7-69](#).

 Return to the [Table 7-19](#).

Figure 7-108. FSM_TRIG_MASK_1 Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------------------|--------------------|------------------------|--------------------|------------------------|--------------------|------------------------|--------------------|
| GPIO4_FSM_M ASK_POL | GPIO4_FSM_M ASK | GPIO3_FSM_M ASK_POL | GPIO3_FSM_M ASK | GPIO2_FSM_M ASK_POL | GPIO2_FSM_M ASK | GPIO1_FSM_M ASK_POL | GPIO1_FSM_M ASK |
| 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 7-69. 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 |

7.7.1.50 FSM_TRIG_MASK_2 Register (Offset = 0x47) [Reset = 0x00]

FSM_TRIG_MASK_2 is shown in [Figure 7-109](#) and described in [Table 7-70](#).

Return to the [Table 7-19](#).

Figure 7-109. FSM_TRIG_MASK_2 Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------------------|--------------------|------------------------|--------------------|------------------------|--------------------|------------------------|--------------------|
| GPIO8_FSM_M ASK_POL | GPIO8_FSM_M ASK | GPIO7_FSM_M ASK_POL | GPIO7_FSM_M ASK | GPIO6_FSM_M ASK_POL | GPIO6_FSM_M ASK | GPIO5_FSM_M ASK_POL | GPIO5_FSM_M ASK |
| 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 7-70. 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 |

7.7.1.51 FSM_TRIG_MASK_3 Register (Offset = 0x48) [Reset = 0x00]

 FSM_TRIG_MASK_3 is shown in [Figure 7-110](#) and described in [Table 7-71](#).

 Return to the [Table 7-19](#).

Figure 7-110. FSM_TRIG_MASK_3 Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------|---|---|---|---------------------|-----------------|--------------------|----------------|
| RESERVED | | | | 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 |

Table 7-71. FSM_TRIG_MASK_3 Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|---------------------|------|-------|---|
| 7:4 | RESERVED | R/W | 0b | |
| 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 |

7.7.1.52 MASK_BUCK1_2 Register (Offset = 0x49) [Reset = 0x00]

MASK_BUCK1_2 is shown in [Figure 7-111](#) and described in [Table 7-72](#).

Return to the [Table 7-19](#).

Figure 7-111. MASK_BUCK1_2 Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---------------------|----------|-------------------|-------------------|---------------------|----------|-------------------|-------------------|
| BUCK2_ILIM_M ASK | RESERVED | BUCK2_UV_M ASK | BUCK2_OV_M ASK | BUCK1_ILIM_M ASK | RESERVED | BUCK1_UV_M ASK | BUCK1_OV_M ASK |
| 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 7-72. 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. |

7.7.1.53 MASK_BUCK3_4 Register (Offset = 0x4A) [Reset = 0x00]

 MASK_BUCK3_4 is shown in [Figure 7-112](#) and described in [Table 7-73](#).

 Return to the [Table 7-19](#).

Figure 7-112. MASK_BUCK3_4 Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---------------------|----------|-------------------|-------------------|---------------------|----------|-------------------|-------------------|
| BUCK4_ILIM_M ASK | RESERVED | BUCK4_UV_M ASK | BUCK4_OV_M ASK | BUCK3_ILIM_M ASK | RESERVED | BUCK3_UV_M ASK | BUCK3_OV_M ASK |
| 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 7-73. 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. |

7.7.1.54 MASK_VMON Register (Offset = 0x4E) [Reset = 0x00]

MASK_VMON is shown in [Figure 7-113](#) and described in [Table 7-74](#).

Return to the [Table 7-19](#).

Figure 7-113. MASK_VMON Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------|-------------------|-------------------|----------|-------------------|-------------------|------------------|------------------|
| RESERVED | VMON2_UV_M ASK | VMON2_OV_M ASK | RESERVED | VMON1_UV_M ASK | VMON1_OV_M ASK | VCCA_UV_MA SK | VCCA_OV_MA SK |
| 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 7-74. MASK_VMON Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|---------------|------|-------|--|
| 7 | RESERVED | R/W | 0b | |
| 6 | VMON2_UV_MASK | R/W | 0b | Masking of VMON2 under-voltage detection interrupt VMON2_UV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated. |
| 5 | VMON2_OV_MASK | R/W | 0b | Masking of VMON2 over-voltage detection interrupt VMON2_OV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated. |
| 4 | RESERVED | R/W | 0b | |
| 3 | VMON1_UV_MASK | R/W | 0b | Masking of VMON1 under-voltage detection interrupt VMON1_UV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated. |
| 2 | VMON1_OV_MASK | R/W | 0b | Masking of VMON1 over-voltage detection interrupt VMON1_OV_INT: (Default from NVM memory) 0b = Interrupt generated 1b = Interrupt not generated. |
| 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. |

7.7.1.55 MASK_GPIO1_8_FALL Register (Offset = 0x4F) [Reset = 0x00]

MASK_GPIO1_8_FALL is shown in [Figure 7-114](#) and described in [Table 7-75](#).

Return to the [Table 7-19](#).

Figure 7-114. 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 |

Table 7-75. 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. |

7.7.1.56 MASK_GPIO1_8_RISE Register (Offset = 0x50) [Reset = 0x00]

MASK_GPIO1_8_RISE is shown in [Figure 7-115](#) and described in [Table 7-76](#).

Return to the [Table 7-19](#).

Figure 7-115. 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 |

Table 7-76. 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. |

7.7.1.57 MASK_GPIO9_10 Register (Offset = 0x51) [Reset = 0x00]

 MASK_GPIO9_10 is shown in [Figure 7-116](#) and described in [Table 7-77](#).

 Return to the [Table 7-19](#).

Figure 7-116. MASK_GPIO9_10 Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------|---|---|------------------|-----------------|----------|------------------|-----------------|
| RESERVED | | | GPIO10_RISE_MASK | GPIO9_RISE_MASK | RESERVED | GPIO10_FALL_MASK | GPIO9_FALL_MASK |
| R/W-0b | | | R/W-0b | R/W-0b | R/W-0b | R/W-0b | R/W-0b |

Table 7-77. MASK_GPIO9_10 Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|------------------|------|-------|--|
| 7:5 | RESERVED | R/W | 0b | |
| 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 | RESERVED | R/W | 0b | |
| 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. |

7.7.1.58 MASK_STARTUP Register (Offset = 0x52) [Reset = 0x00]

 MASK_STARTUP is shown in [Figure 7-117](#) and described in [Table 7-78](#).

 Return to the [Table 7-19](#).

Figure 7-117. MASK_STARTUP Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------|---|------------------|----------|---|----------|-------------|----------|
| RESERVED | | SOFT_REBOOT_MASK | FSD_MASK | | RESERVED | ENABLE_MASK | RESERVED |
| R/W-0b | | R/W-0b | R/W-0b | | R/W-0b | R/W-0b | R/W-0b |

Table 7-78. 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_INT 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 | RESERVED | R/W | 0b | |

7.7.1.59 MASK_MISC Register (Offset = 0x53) [Reset = 0x00]

 MASK_MISC is shown in [Figure 7-118](#) and described in [Table 7-79](#).

 Return to the [Table 7-19](#).

Figure 7-118. MASK_MISC Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------|---|---|---|------------|----------|------------------|--------------------|
| RESERVED | | | | TWARN_MASK | RESERVED | EXT_CLK_MAS K | BIST_PASS_M ASK |
| R/W-0b | | | | R/W-0b | R/W-0b | R/W-0b | R/W-0b |

Table 7-79. 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. |

7.7.1.60 MASK_MODERATE_ERR Register (Offset = 0x54) [Reset = 0x00]

MASK_MODERATE_ERR is shown in [Figure 7-119](#) and described in [Table 7-80](#).

Return to the [Table 7-19](#).

Figure 7-119. MASK_MODERATE_ERR Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------------------|--------------------|----------|---------------|----------|------------------|----------------|----------|
| NRSTOUT_READBACK_MASK | NINT_READBACK_MASK | RESERVED | 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 7-80. 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 | RESERVED | R/W | 0b | |
| 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 | |

7.7.1.61 MASK_FSM_ERR Register (Offset = 0x56) [Reset = 0x00]

 MASK_FSM_ERR is shown in [Figure 7-120](#) and described in [Table 7-81](#).

 Return to the [Table 7-19](#).

Figure 7-120. MASK_FSM_ERR Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------|---|---|---|----------------------|----------------------|-----------------------|-----------------------|
| RESERVED | | | | SOC_PWR_ER R_MASK | MCU_PWR_ER R_MASK | ORD_SHUTDO WN_MASK | IMM_SHUTDO WN_MASK |
| R/W-0b | | | | R/W-0b | R/W-0b | R/W-0b | R/W-0b |

Table 7-81. 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. |

7.7.1.62 MASK_COMM_ERR Register (Offset = 0x57) [Reset = 0x00]

 MASK_COMM_ERR is shown in [Figure 7-121](#) and described in [Table 7-82](#).

 Return to the [Table 7-19](#).

Figure 7-121. 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 7-82. 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. |

7.7.1.63 MASK_READBACK_ERR Register (Offset = 0x58) [Reset = 0x00]

 MASK_READBACK_ERR is shown in [Figure 7-122](#) and described in [Table 7-83](#).

 Return to the [Table 7-19](#).

Figure 7-122. 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 7-83. 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. |

7.7.1.64 MASK_ESM Register (Offset = 0x59) [Reset = 0x00]

MASK_ESM is shown in [Figure 7-123](#) and described in [Table 7-84](#).

Return to the [Table 7-19](#).

Figure 7-123. MASK_ESM Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------|---|------------------|-------------------|------------------|----------|---|---|
| RESERVED | | ESM_MCU_RST_MASK | ESM_MCU_FAIL_MASK | ESM_MCU_PIN_MASK | RESERVED | | |
| R/W-0b | | R/W-0b | R/W-0b | R/W-0b | R/W-0b | | |

Table 7-84. 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. |
| 2:0 | RESERVED | R/W | 0b | |

7.7.1.65 INT_TOP Register (Offset = 0x5A) [Reset = 0x00]

 INT_TOP is shown in [Figure 7-124](#) and described in [Table 7-85](#).

 Return to the [Table 7-19](#).

Figure 7-124. 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 | VMON_INT | BUCK_INT |
| R-0b | R-0b | R-0b | R-0b | R-0b | R-0b | R-0b | R-0b |

Table 7-85. 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 | VMON_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. |
| 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. |

7.7.1.66 INT_BUCK Register (Offset = 0x5B) [Reset = 0x00]

 INT_BUCK is shown in [Figure 7-125](#) and described in [Table 7-86](#).

 Return to the [Table 7-19](#).

Figure 7-125. INT_BUCK Register

| | | | | | | | |
|----------|---|---|---|---|---|-------------|-------------|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RESERVED | | | | | | BUCK3_4_INT | BUCK1_2_INT |
| R-0b | | | | | | R-0b | R-0b |

Table 7-86. INT_BUCK Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|-------------|------|-------|--|
| 7:2 | RESERVED | R | 0b | |
| 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. |

7.7.1.67 INT_BUCK1_2 Register (Offset = 0x5C) [Reset = 0x00]

 INT_BUCK1_2 is shown in [Figure 7-126](#) and described in [Table 7-87](#).

 Return to the [Table 7-19](#).

Figure 7-126. 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 7-87. 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 following errors on BUCK2 output voltage: - BUCK2 output voltage has fallen below the short-circuit threshold level during operation, or - BUCK2 output did not exceed this short-circuit threshold level after expected ramp-up time, or - BUCK2 output exceeded this short-circuit threshold level before start-up of BUCK2 regulator 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 following errors on BUCK1 output voltage: - BUCK1 output voltage has fallen below the short-circuit threshold level during operation, or - BUCK1 output did not exceed this short-circuit threshold level after expected ramp-up time, or - BUCK1 output exceeded this short-circuit threshold level before start-up of BUCK1 regulator 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. |

7.7.1.68 INT_BUCK3_4 Register (Offset = 0x5D) [Reset = 0x00]

INT_BUCK3_4 is shown in [Figure 7-127](#) and described in [Table 7-88](#).

Return to the [Table 7-19](#).

Figure 7-127. 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 7-88. 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 following errors on BUCK4 output voltage: - BUCK4 output voltage has fallen below the short-circuit threshold level during operation, or - BUCK4 output did not exceed this short-circuit threshold level after expected ramp-up time, or - BUCK4 output exceeded this short-circuit threshold level before start-up of BUCK4 regulator 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 following errors on BUCK3 output voltage: - BUCK3 output voltage has fallen below the short-circuit threshold level during operation, or - BUCK3 output did not exceed this short-circuit threshold level after expected ramp-up time, or - BUCK3 output exceeded this short-circuit threshold level before start-up of BUCK3 regulator 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. |

7.7.1.69 INT_VMON Register (Offset = 0x62) [Reset = 0x00]

INT_VMON is shown in [Figure 7-128](#) and described in [Table 7-89](#).

Return to the [Table 7-19](#).

Figure 7-128. INT_VMON Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------------|------------------|------------------|------------------|------------------|------------------|-------------|-------------|
| VMON2_RV_IN T | VMON2_UV_IN T | VMON2_OV_IN T | VMON1_RV_IN T | VMON1_UV_IN T | VMON1_OV_IN T | VCCA_UV_INT | VCCA_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 7-89. INT_VMON Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|--------------|-------|-------|---|
| 7 | VMON2_RV_INT | R/W1C | 0b | Latched status bit indicating that the VMON2 voltage has been above residual voltage threshold level during voltage check. Write 1 to clear interrupt. |
| 6 | VMON2_UV_INT | R/W1C | 0b | Latched status bit indicating that the VMON2 input voltage has decreased below the under-voltage monitoring level. The actual status of the VMON2 under-voltage monitoring is indicated by VMON2_UV_STAT bit. Write 1 to clear interrupt. |
| 5 | VMON2_OV_INT | R/W1C | 0b | Latched status bit indicating that the VMON2 input voltage has exceeded the over-voltage detection level. The actual status of the over-voltage is indicated by VMON2_OV_STAT bit. Write 1 to clear interrupt. |
| 4 | VMON1_RV_INT | R/W1C | 0b | Latched status bit indicating that the VMON1 voltage has been above residual voltage threshold level during voltage check. Write 1 to clear interrupt. |
| 3 | VMON1_UV_INT | R/W1C | 0b | Latched status bit indicating that the VMON1 input voltage has decreased below the under-voltage monitoring level. The actual status of the VMON1 under-voltage monitoring is indicated by VMON1_UV_STAT bit. Write 1 to clear interrupt. |
| 2 | VMON1_OV_INT | R/W1C | 0b | Latched status bit indicating that the VMON1 input voltage has exceeded the over-voltage detection level. The actual status of the over-voltage is indicated by VMON1_OV_STAT bit. Write 1 to clear interrupt. |
| 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. |

7.7.1.70 INT_GPIO Register (Offset = 0x63) [Reset = 0x00]

 INT_GPIO is shown in [Figure 7-129](#) and described in [Table 7-90](#).

 Return to the [Table 7-19](#).

Figure 7-129. INT_GPIO Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------|---|---|---|-------------|----------|------------|-----------|
| RESERVED | | | | GPIO1_8_INT | RESERVED | GPIO10_INT | GPIO9_INT |
| R/W-0b | | | | R-0b | R/W-0b | R/W1C-0b | R/W1C-0b |

Table 7-90. 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 | RESERVED | R/W | 0b | |
| 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. |

7.7.1.71 INT_GPIO1_8 Register (Offset = 0x64) [Reset = 0x00]

 INT_GPIO1_8 is shown in [Figure 7-130](#) and described in [Table 7-91](#).

 Return to the [Table 7-19](#).

Figure 7-130. 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 |

Table 7-91. INT_GPIO1_8 Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|-----------|-------|-------|---|
| 7 | GPIO8_INT | R/W1C | 0b | Latched status bit indicating that GPIO8 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 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 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 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 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 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. |

7.7.1.72 INT_STARTUP Register (Offset = 0x65) [Reset = 0x00]

 INT_STARTUP is shown in [Figure 7-131](#) and described in [Table 7-92](#).

 Return to the [Table 7-19](#).

Figure 7-131. INT_STARTUP Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------|---|-----------------|----------|---|----------|------------|----------|
| RESERVED | | SOFT_REBOOT_INT | FSD_INT | | RESERVED | ENABLE_INT | RESERVED |
| R/W-0b | | R/W1C-0b | R/W1C-0b | | R/W-0b | R/W1C-0b | R/W-0b |

Table 7-92. 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 software reboot occurred. |
| 4 | FSD_INT | R/W1C | 0b | Latched status bit indicating that PMIC has started from NO_SUPPLY state (first supply detection). Write 1 to clear. |
| 3:2 | RESERVED | R/W | 0b | |
| 1 | ENABLE_INT | R/W1C | 0b | Latched status bit indicating that ENABLE pin active event has been detected. Write 1 to clear. |
| 0 | RESERVED | R/W | 0b | |

7.7.1.73 INT_MISC Register (Offset = 0x66) [Reset = 0x00]

 INT_MISC is shown in [Figure 7-132](#) and described in [Table 7-93](#).

 Return to the [Table 7-19](#).

Figure 7-132. 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 7-93. 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. |

7.7.1.74 INT_MODERATE_ERR Register (Offset = 0x67) [Reset = 0x00]

INT_MODERATE_ERR is shown in [Figure 7-133](#) and described in [Table 7-94](#).

Return to the [Table 7-19](#).

Figure 7-133. INT_MODERATE_ERR Register

| 7 | | 6 | | 5 | | 4 | | 3 | | 2 | | 1 | | 0 | |
|--------------------------|--|-----------------------|--|----------|--|------------------|--|-------------------|--|---------------------|--|---------------|--|-------------|--|
| NRSTOUT_RE ADBACK_INT | | NINT_READBA CK_INT | | RESERVED | | SPMI_ERR_IN T | | RECOV_CNT_I NT | | REG_CRC_ER R_INT | | BIST_FAIL_INT | | TSD_ORD_INT | |
| R/W1C-0b | | R/W1C-0b | | R/W-0b | | R/W1C-0b | | R/W1C-0b | | R/W1C-0b | | R/W1C-0b | | R/W1C-0b | |

Table 7-94. INT_MODERATE_ERR Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|--------------------------|-------|-------|---|
| 7 | NRSTOUT_READBACK_I NT | R/W1C | 0b | Latched status bit indicating that NRSTOUT read-back error has been detected. Write 1 to clear interrupt. |
| 6 | NINT_READBACK_INT | R/W1C | 0b | Latched status bit indicating that NINT read-back error has been detected. Write 1 to clear interrupt. |
| 5 | RESERVED | R/W | 0b | |
| 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. |

7.7.1.75 INT_SEVERE_ERR Register (Offset = 0x68) [Reset = 0x00]

 INT_SEVERE_ERR is shown in [Figure 7-134](#) and described in [Table 7-95](#).

 Return to the [Table 7-19](#).

Figure 7-134. INT_SEVERE_ERR Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------|---|---|---|---|------------------|------------------|-------------|
| RESERVED | | | | | PFSM_ERR_IN T | VCCA_OVP_IN T | TSD_IMM_INT |
| R/W-0b | | | | | R/W1C-0b | R/W1C-0b | R/W1C-0b |

Table 7-95. 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 STAT_SEVERE_ERR register. Write 1 to clear interrupt. |

7.7.1.76 INT_FSM_ERR Register (Offset = 0x69) [Reset = 0x00]

INT_FSM_ERR is shown in [Figure 7-135](#) and described in [Table 7-96](#).

Return to the [Table 7-19](#).

Figure 7-135. 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 7-96. 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. |

7.7.1.77 INT_COMM_ERR Register (Offset = 0x6A) [Reset = 0x00]

 INT_COMM_ERR is shown in [Figure 7-136](#) and described in [Table 7-97](#).

 Return to the [Table 7-19](#).

Figure 7-136. INT_COMM_ERR Register

| 7 | | 6 | | 5 | | 4 | | 3 | | 2 | | 1 | | 0 | |
|----------------------|----------|----------------------|----------|----------------------|----------|----------------------|----------------------|----------|--------|----------|--------|----------|--------|----------|--------|
| I2C2_ADR_ER R_INT | RESERVED | I2C2_CRC_ER R_INT | RESERVED | COMM_ADR_E RR_INT | RESERVED | COMM_CRC_E RR_INT | COMM_FRM_E RR_INT | | | | | | | | |
| R/W1C-0b | R/W-0b | R/W1C-0b | R/W-0b | R/W1C-0b | R/W-0b | R/W1C-0b | R/W-0b | R/W1C-0b | R/W-0b | R/W1C-0b | R/W-0b | R/W1C-0b | R/W-0b | R/W1C-0b | R/W-0b |

Table 7-97. 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. - Valid for I2C2 - CRC on I2C2 must be enabled (I2C2_CRC_EN=1 - NVM default bit) and I2C2_CRC_ERR_MASK=0 are required to generate nINT interrupt 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. - Valid for I2C2 - CRC on I2C2 must be enabled (I2C2_CRC_EN=1 - NVM default bit) and I2C2_CRC_ERR_MASK=0 are required to generate nINT interrupt 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. - Valid for SPI and I2C1 - CRC on I2C/SPI must be enabled (I2C1_SPI_CRC_EN=1 - NVM default bit) and COMM_CRC_ERR_MASK=0 are required to generate nINT interrupt 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. - Valid for SPI and I2C1 - CRC on I2C/SPI must be enabled (I2C1_SPI_CRC_EN=1 - NVM default bit) and COMM_CRC_ERR_MASK=0 are required to generate nINT interrupt 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. |

7.7.1.78 INT_READBACK_ERR Register (Offset = 0x6B) [Reset = 0x00]

INT_READBACK_ERR is shown in [Figure 7-137](#) and described in [Table 7-98](#).

Return to the [Table 7-19](#).

Figure 7-137. INT_READBACK_ERR Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------|---|---|---|----------------------------------|----------|---|-------------------------|
| RESERVED | | | | NRSTOUT_SO C_READBACK _INT | RESERVED | | EN_DRV_REA DBACK_INT |
| R/W-0b | | | | R/W1C-0b | R/W-0b | | R/W1C-0b |

Table 7-98. INT_READBACK_ERR Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|------------------------------|-------|-------|--|
| 7:4 | RESERVED | R/W | 0b | |
| 3 | NRSTOUT_SOC_READB ACK_INT | R/W1C | 0b | Latched status bit indicating that NRSTOUT_SOC read-back error has been detected. Write 1 to clear interrupt. |
| 2:1 | RESERVED | R/W | 0b | |
| 0 | EN_DRV_READBACK_IN T | R/W1C | 0b | Latched status bit indicating that EN_DRV read-back error has been detected. Write 1 to clear interrupt. |

7.7.1.79 INT_ESM Register (Offset = 0x6C) [Reset = 0x00]

 INT_ESM is shown in [Figure 7-138](#) and described in [Table 7-99](#).

 Return to the [Table 7-19](#).

Figure 7-138. INT_ESM Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------|---|-----------------|------------------|-----------------|----------|---|---|
| RESERVED | | ESM_MCU_RST_INT | ESM_MCU_FAIL_INT | ESM_MCU_PIN_INT | RESERVED | | |
| R/W-0b | | R/W1C-0b | R/W1C-0b | R/W1C-0b | R/W-0b | | |

Table 7-99. 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. |
| 2:0 | RESERVED | R/W | 0b | |

7.7.1.80 STAT_BUCK1_2 Register (Offset = 0x6D) [Reset = 0x00]

STAT_BUCK1_2 is shown in [Figure 7-139](#) and described in [Table 7-100](#).

Return to the [Table 7-19](#).

Figure 7-139. STAT_BUCK1_2 Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---------------------|----------|-------------------|-------------------|---------------------|----------|-------------------|-------------------|
| BUCK2_ILIM_S TAT | RESERVED | BUCK2_UV_ST AT | BUCK2_OV_ST AT | BUCK1_ILIM_S TAT | RESERVED | BUCK1_UV_ST AT | BUCK1_OV_ST AT |
| R-0b | R-0b | R-0b | R-0b | R-0b | R-0b | R-0b | R-0b |

Table 7-100. 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. |

7.7.1.81 STAT_BUCK3_4 Register (Offset = 0x6E) [Reset = 0x00]

 STAT_BUCK3_4 is shown in [Figure 7-140](#) and described in [Table 7-101](#).

 Return to the [Table 7-19](#).

Figure 7-140. STAT_BUCK3_4 Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---------------------|----------|-------------------|-------------------|---------------------|----------|-------------------|-------------------|
| BUCK4_ILIM_S TAT | RESERVED | BUCK4_UV_ST AT | BUCK4_OV_ST AT | BUCK3_ILIM_S TAT | RESERVED | BUCK3_UV_ST AT | BUCK3_OV_ST AT |
| R-0b | R-0b | R-0b | R-0b | R-0b | R-0b | R-0b | R-0b |

Table 7-101. 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. |

7.7.1.82 STAT_VMON Register (Offset = 0x72) [Reset = 0x00]

 STAT_VMON is shown in [Figure 7-141](#) and described in [Table 7-102](#).

 Return to the [Table 7-19](#).

Figure 7-141. STAT_VMON Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------|-------------------|-------------------|----------|-------------------|-------------------|------------------|------------------|
| RESERVED | VMON2_UV_S TAT | VMON2_OV_S TAT | RESERVED | VMON1_UV_S TAT | VMON1_OV_S TAT | VCCA_UV_STA T | VCCA_OV_STA T |
| R-0b | R-0b | R-0b | R-0b | R-0b | R-0b | R-0b | R-0b |

Table 7-102. STAT_VMON Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|---------------|------|-------|--|
| 7 | RESERVED | R | 0b | |
| 6 | VMON2_UV_STAT | R | 0b | Status bit indicating that VMON2 input voltage is below under-voltage level. |
| 5 | VMON2_OV_STAT | R | 0b | Status bit indicating that VMON2 input voltage is above over-voltage level. |
| 4 | RESERVED | R | 0b | |
| 3 | VMON1_UV_STAT | R | 0b | Status bit indicating that VMON1 input voltage is below under-voltage level. |
| 2 | VMON1_OV_STAT | R | 0b | Status bit indicating that VMON1 input voltage is above over-voltage level. |
| 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. |

7.7.1.83 STAT_STARTUP Register (Offset = 0x73) [Reset = 0x00]

 STAT_STARTUP is shown in [Figure 7-142](#) and described in [Table 7-103](#).

 Return to the [Table 7-19](#).

Figure 7-142. STAT_STARTUP Register

| | | | | | | | |
|----------|---|---|---|---|---|-------------|----------|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RESERVED | | | | | | ENABLE_STAT | RESERVED |
| R-0b | | | | | | R-0b | R-0b |

Table 7-103. STAT_STARTUP Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|-------------|------|-------|---|
| 7:2 | RESERVED | R | 0b | |
| 1 | ENABLE_STAT | R | 0b | Status bit indicating ENABLE pin status |
| 0 | RESERVED | R | 0b | |

7.7.1.84 STAT_MISC Register (Offset = 0x74) [Reset = 0x00]

STAT_MISC is shown in [Figure 7-143](#) and described in [Table 7-104](#).

Return to the [Table 7-19](#).

Figure 7-143. 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 | R-0b |

Table 7-104. 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 | |

7.7.1.85 STAT_MODERATE_ERR Register (Offset = 0x75) [Reset = 0x00]

 STAT_MODERATE_ERR is shown in [Figure 7-144](#) and described in [Table 7-105](#).

 Return to the [Table 7-19](#).

Figure 7-144. STAT_MODERATE_ERR Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------|---|---|---|---|---|---|--------------|
| RESERVED | | | | | | | TSD_ORD_STAT |
| R-0b | | | | | | | R-0b |

Table 7-105. 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. |

7.7.1.86 STAT_SEVERE_ERR Register (Offset = 0x76) [Reset = 0x00]

STAT_SEVERE_ERR is shown in [Figure 7-145](#) and described in [Table 7-106](#).

Return to the [Table 7-19](#).

Figure 7-145. STAT_SEVERE_ERR Register

| | | | | | | | |
|----------|---|---|---|---|---|-------------------|------------------|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RESERVED | | | | | | VCCA_OVP_S TAT | TSD_IMM_STA T |
| R-0b | | | | | | R-0b | R-0b |

Table 7-106. 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. |

7.7.1.87 STAT_READBACK_ERR Register (Offset = 0x77) [Reset = 0x00]

STAT_READBACK_ERR is shown in [Figure 7-146](#) and described in [Table 7-107](#).

Return to the [Table 7-19](#).

Figure 7-146. 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 7-107. 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. |

7.7.1.88 PGOOD_SEL_1 Register (Offset = 0x78) [Reset = 0x00]

PGOOD_SEL_1 is shown in [Figure 7-147](#) and described in [Table 7-108](#).

Return to the [Table 7-19](#).

Figure 7-147. 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 7-108. 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 |

7.7.1.89 PGOOD_SEL_4 Register (Offset = 0x7B) [Reset = 0x00]

 PGOOD_SEL_4 is shown in [Figure 7-148](#) and described in [Table 7-109](#).

 Return to the [Table 7-19](#).

Figure 7-148. PGOOD_SEL_4 Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------------|-----------|------------------------|--------------------|---------------------|-----------------|-----------------|----------------|
| PGOOD_WINDOW | PGOOD_POL | PGOOD_SEL_NIRSTOUT_SOC | PGOOD_SEL_NIRSTOUT | PGOOD_SEL_TDIE_WARN | PGOOD_SEL_VMON2 | PGOOD_SEL_VMON1 | PGOOD_SEL_VCCA |
| 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 7-109. 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_NIRSTOUT_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_NIRSTOUT | 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 | PGOOD_SEL_VMON2 | R/W | 0b | PGOOD signal source control from VMON2 monitoring (Default from NVM memory) 0b = Masked 1b = VMON2 OV/UV threshold affecting PGOOD signal |
| 1 | PGOOD_SEL_VMON1 | R/W | 0b | PGOOD signal source control from VMON1 monitoring (Default from NVM memory) 0b = Masked 1b = VMON1 OV/UV threshold affecting PGOOD signal |
| 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 |

7.7.1.90 PLL_CTRL Register (Offset = 0x7C) [Reset = 0x00]

PLL_CTRL is shown in [Figure 7-149](#) and described in [Table 7-110](#).

Return to the [Table 7-19](#).

Figure 7-149. PLL_CTRL Register

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

Table 7-110. 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 = 8.8 MHz |

7.7.1.91 CONFIG_1 Register (Offset = 0x7D) [Reset = 0xC0]

 CONFIG_1 is shown in [Figure 7-150](#) and described in [Table 7-111](#).

 Return to the [Table 7-19](#).

Figure 7-150. CONFIG_1 Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------------|------------------|----------------------|---------|---------|----------|-------------------|-----------------|
| NSLEEP2_MAS K | NSLEEP1_MAS K | EN_ILIM_FSM_ CTRL | I2C2_HS | I2C1_HS | RESERVED | TSD_ORD_LEV EL | TWARN_LEVE L |
| 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 7-111. 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 regulators ILIM interrupts do not affect FSM triggers. 1b = Buck 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 controller 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 controller 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 |

7.7.1.92 ENABLE_DRV_REG Register (Offset = 0x80) [Reset = 0x00]

ENABLE_DRV_REG is shown in [Figure 7-151](#) and described in [Table 7-112](#).

Return to the [Table 7-19](#).

Figure 7-151. ENABLE_DRV_REG Register

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

Table 7-112. 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 |

7.7.1.93 MISC_CTRL Register (Offset = 0x81) [Reset = 0x00]

 MISC_CTRL is shown in [Figure 7-152](#) and described in [Table 7-113](#).

 Return to the [Table 7-19](#).

Figure 7-152. MISC_CTRL Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---------------------|-------------|-----------|-----------|--------|-------------|---------|--------|
| SYNCCLKOUT_FREQ_SEL | SEL_EXT_CLK | REFOUT_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 | R/W-0b |

Table 7-113. 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 use 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 | REFOUT_EN | R/W | 0b | Control bandgap voltage to REFOUT 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 |

7.7.1.94 ENABLE_DRV_STAT Register (Offset = 0x82) [Reset = 0x08]

ENABLE_DRV_STAT is shown in [Figure 7-153](#) and described in [Table 7-114](#).

Return to the [Table 7-19](#).

Figure 7-153. ENABLE_DRV_STAT Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------|---|---|-------------|----------------------|--------------------|------------|-----------|
| RESERVED | | | SPMI_LPM_EN | FORCE_EN_D RV_LOW | NRSTOUT_SO C_IN | NRSTOUT_IN | EN_DRV_IN |
| R/W-0b | | | R/W-0b | R/W-1b | R-0b | R-0b | R-0b |

Table 7-114. 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 that stops SPMI WD (Bus heartbeat). The PMICs on the SPMI-bus must set SPMI_LPM_EN=1 synchronously to prevent SPMI WD failures. Therefore to mitigate clock variations, setting SPMI_LPM_EN=1 must be done early in the power-up 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 |

7.7.1.95 RECOV_CNT_REG_1 Register (Offset = 0x83) [Reset = 0x00]

 RECOV_CNT_REG_1 is shown in [Figure 7-154](#) and described in [Table 7-115](#).

 Return to the [Table 7-19](#).

Figure 7-154. RECOV_CNT_REG_1 Register

| | | | | | | | |
|----------|---|---|---|-----------|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RESERVED | | | | RECOV_CNT | | | |
| R-0b | | | | R-0b | | | |

Table 7-115. 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. |

7.7.1.96 RECOV_CNT_REG_2 Register (Offset = 0x84) [Reset = 0x00]

RECOV_CNT_REG_2 is shown in [Figure 7-155](#) and described in [Table 7-116](#).

Return to the [Table 7-19](#).

Figure 7-155. 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 7-116. 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) |

7.7.1.97 FSM_I2C_TRIGGERS Register (Offset = 0x85) [Reset = 0x00]

 FSM_I2C_TRIGGERS is shown in [Figure 7-156](#) and described in [Table 7-117](#).

 Return to the [Table 7-19](#).

Figure 7-156. 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 7-117. 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. |

7.7.1.98 FSM_NSLEEP_TRIGGERS Register (Offset = 0x86) [Reset = 0x00]

FSM_NSLEEP_TRIGGERS is shown in [Figure 7-157](#) and described in [Table 7-118](#).

Return to the [Table 7-19](#).

Figure 7-157. 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 7-118. 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 |

7.7.1.99 BUCK_RESET_REG Register (Offset = 0x87) [Reset = 0x00]

BUCK_RESET_REG is shown in [Figure 7-158](#) and described in [Table 7-119](#).

Return to the [Table 7-19](#).

Figure 7-158. BUCK_RESET_REG Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------|---|---|---|-------------|-------------|-------------|-------------|
| RESERVED | | | | BUCK4_RESET | BUCK3_RESET | BUCK2_RESET | BUCK1_RESET |
| R/W-0b | | | | R/W-0b | R/W-0b | R/W-0b | R/W-0b |

Table 7-119. BUCK_RESET_REG Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|-------------|------|-------|---|
| 7:4 | RESERVED | R/W | 0b | |
| 3 | BUCK4_RESET | R/W | 0b | Reset signal for Buck logic. (Default from NVM memory) |
| 2 | BUCK3_RESET | R/W | 0b | Reset signal for Buck logic. (Default from NVM memory) |
| 1 | BUCK2_RESET | R/W | 0b | Reset signal for Buck logic. (Default from NVM memory) |
| 0 | BUCK1_RESET | R/W | 0b | Reset signal for Buck logic. (Default from NVM memory) |

7.7.1.100 SPREAD_SPECTRUM_1 Register (Offset = 0x88) [Reset = 0x00]

SPREAD_SPECTRUM_1 is shown in [Figure 7-159](#) and described in [Table 7-120](#).

Return to the [Table 7-19](#).

Figure 7-159. 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 7-120. 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 |

7.7.1.101 FSM_STEP_SIZE Register (Offset = 0x8B) [Reset = 0x00]

 FSM_STEP_SIZE is shown in [Figure 7-160](#) and described in [Table 7-121](#).

 Return to the [Table 7-19](#).

Figure 7-160. FSM_STEP_SIZE Register

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

Table 7-121. 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 $50\text{ns} * 2^{\text{PFSM_DELAY_STEP}}$. (Default from NVM memory) |

7.7.1.102 USER_SPARE_REGS Register (Offset = 0x8E) [Reset = 0x00]

USER_SPARE_REGS is shown in [Figure 7-161](#) and described in [Table 7-122](#).

Return to the [Table 7-19](#).

Figure 7-161. 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 7-122. 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) |

7.7.1.103 ESM_MCU_START_REG Register (Offset = 0x8F) [Reset = 0x00]

 ESM_MCU_START_REG is shown in [Figure 7-162](#) and described in [Table 7-123](#).

 Return to the [Table 7-19](#).

Figure 7-162. ESM_MCU_START_REG Register

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

Table 7-123. 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. |

7.7.1.104 ESM_MCU_DELAY1_REG Register (Offset = 0x90) [Reset = 0x00]

ESM_MCU_DELAY1_REG is shown in [Figure 7-163](#) and described in [Table 7-124](#).

Return to the [Table 7-19](#).

Figure 7-163. ESM_MCU_DELAY1_REG Register

| | | | | | | | |
|----------------|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| ESM_MCU_DELAY1 | | | | | | | |
| R/W-0b | | | | | | | |

Table 7-124. 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. |

7.7.1.105 ESM_MCU_DELAY2_REG Register (Offset = 0x91) [Reset = 0x00]

ESM_MCU_DELAY2_REG is shown in [Figure 7-164](#) and described in [Table 7-125](#).

Return to the [Table 7-19](#).

Figure 7-164. ESM_MCU_DELAY2_REG Register

| | | | | | | | |
|----------------|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| ESM_MCU_DELAY2 | | | | | | | |
| R/W-0b | | | | | | | |

Table 7-125. 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. |

7.7.1.106 ESM_MCU_MODE_CFG Register (Offset = 0x92) [Reset = 0x00]

 ESM_MCU_MODE_CFG is shown in [Figure 7-165](#) and described in [Table 7-126](#).

 Return to the [Table 7-19](#).

Figure 7-165. ESM_MCU_MODE_CFG Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------------|------------|----------------|----------|--------------------|---|---|---|
| ESM_MCU_MODE | ESM_MCU_EN | ESM_MCU_EN_DRV | RESERVED | ESM_MCU_ERR_CNT_TH | | | |
| R/W-0b | R/W-0b | R/W-0b | R/W-0b | R/W-0b | | | |

Table 7-126. 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. (Default from NVM memory) 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 Section 4.17.1) 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. |

7.7.1.107 ESM_MCU_HMAX_REG Register (Offset = 0x93) [Reset = 0x00]

ESM_MCU_HMAX_REG is shown in [Figure 7-166](#) and described in [Table 7-127](#).

Return to the [Table 7-19](#).

Figure 7-166. ESM_MCU_HMAX_REG Register

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

Table 7-127. ESM_MCU_HMAX_REG Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|--------------|------|-------|---|
| 7:0 | ESM_MCU_HMAX | R/W | 0b | These bits configure 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. |

7.7.1.108 ESM_MCU_HMIN_REG Register (Offset = 0x94) [Reset = 0x00]

ESM_MCU_HMIN_REG is shown in [Figure 7-167](#) and described in [Table 7-128](#).

Return to the [Table 7-19](#).

Figure 7-167. ESM_MCU_HMIN_REG Register

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

Table 7-128. ESM_MCU_HMIN_REG Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|--------------|------|-------|---|
| 7:0 | ESM_MCU_HMIN | R/W | 0b | These bits configure 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. |

7.7.1.109 ESM_MCU_LMAX_REG Register (Offset = 0x95) [Reset = 0x00]

ESM_MCU_LMAX_REG is shown in [Figure 7-168](#) and described in [Table 7-129](#).

Return to the [Table 7-19](#).

Figure 7-168. ESM_MCU_LMAX_REG Register

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

Table 7-129. ESM_MCU_LMAX_REG Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|--------------|------|-------|---|
| 7:0 | ESM_MCU_LMAX | R/W | 0b | These bits configure 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. |

7.7.1.110 ESM_MCU_LMIN_REG Register (Offset = 0x96) [Reset = 0x00]

ESM_MCU_LMIN_REG is shown in [Figure 7-169](#) and described in [Table 7-130](#).

Return to the [Table 7-19](#).

Figure 7-169. ESM_MCU_LMIN_REG Register

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

Table 7-130. ESM_MCU_LMIN_REG Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|--------------|------|-------|---|
| 7:0 | ESM_MCU_LMIN | R/W | 0b | These bits configure 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. |

7.7.1.111 ESM_MCU_ERR_CNT_REG Register (Offset = 0x97) [Reset = 0x00]

 ESM_MCU_ERR_CNT_REG is shown in [Figure 7-170](#) and described in [Table 7-131](#).

 Return to the [Table 7-19](#).

Figure 7-170. ESM_MCU_ERR_CNT_REG Register

| | | | | | | | |
|----------|---|---|-----------------|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RESERVED | | | ESM_MCU_ERR_CNT | | | | |
| R-0b | | | R-0b | | | | |

Table 7-131. 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. |

7.7.1.112 REGISTER_LOCK Register (Offset = 0xA1) [Reset = 0x00]

REGISTER_LOCK is shown in [Figure 7-171](#) and described in [Table 7-132](#).

Return to the [Table 7-19](#).

Figure 7-171. REGISTER_LOCK Register

| | | | | | | | |
|----------|---|---|---|---|---|---|--------------------------|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RESERVED | | | | | | | REGISTER_LO CK_STATUS |
| R/W-0b | | | | | | | R/W-0b |

Table 7-132. REGISTER_LOCK Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|--------------------------|------|-------|--|
| 7:1 | RESERVED | R/W | 0b | |
| 0 | REGISTER_LOCK_STAT US | 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 is not stored, only lock status can be read. REGISTER_LOCK_STATUS bit shows the lock status: 0b = Registers are unlocked 1b = Registers are locked |

7.7.1.113 CUSTOMER_NVM_ID_REG Register (Offset = 0xA7) [Reset = 0x00]

 CUSTOMER_NVM_ID_REG is shown in [Figure 7-172](#) and described in [Table 7-133](#).

 Return to the [Table 7-19](#).

Figure 7-172. CUSTOMER_NVM_ID_REG Register

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

Table 7-133. CUSTOMER_NVM_ID_REG Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|-----------------|------|-------|--|
| 7:0 | CUSTOMER_NVM_ID | R/W | 0b | Customer NVM version of the IC (Default from NVM memory) |

7.7.1.114 VMON_CONF Register (Offset = 0xA8) [Reset = 0x00]

 VMON_CONF is shown in [Figure 7-173](#) and described in [Table 7-134](#).

 Return to the [Table 7-19](#).

Figure 7-173. VMON_CONF Register

| | | | | | | | |
|----------|---|-----------------|---|---|-----------------|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RESERVED | | VMON2_SLEW_RATE | | | VMON1_SLEW_RATE | | |
| R/W-0b | | R/W-0b | | | R/W-0b | | |

Table 7-134. VMON_CONF Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|-----------------|------|-------|---|
| 7:6 | RESERVED | R/W | 0b | |
| 5:3 | VMON2_SLEW_RATE | R/W | 0b | Voltage slew-rate for VMON2 pin. Setting is used to calculate OV/UV monitoring delays. 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 |
| 2:0 | VMON1_SLEW_RATE | R/W | 0b | Voltage slew-rate for VMON1 pin. Setting is used to calculate OV/UV monitoring delays. 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 |

7.7.1.115 INT_SPI_STATUS Register (Offset = 0xA9) [Reset = 0x00]

 INT_SPI_STATUS is shown in [Figure 7-174](#) and described in [Table 7-135](#).

 Return to the [Table 7-19](#).

Figure 7-174. INT_SPI_STATUS Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----------|--------------------|--------------------|--------------------|-------------------|-------------|----------|-------------|
| RESERVED | COMM_ADR_ERR_SWINT | COMM_CRC_ERR_SWINT | COMM_FRM_ERR_SWINT | ESM_MCU_PIN_SWINT | TWARN_SWINT | WD_SWINT | EN_DRV_STAT |
| R/W-0b | R/W1C-0b | R/W1C-0b | R/W1C-0b | R/W1C-0b | R/W1C-0b | R/W1C-0b | R-0b |

Table 7-135. INT_SPI_STATUS Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|--------------------|-------|-------|---|
| 7 | RESERVED | R/W | 0b | |
| 6 | COMM_ADR_ERR_SWINT | R/W1C | 0b | Latched status bit indicating that SPI (or I2C1) write to non-existing, protected or read-only register address, or read from non-existing register address has been detected. CRC on I2C/SPI must be enabled (I2C1_SPI_CRC_EN=1 - NVM default bit) is required to generate the error-indication with this status bit This interrupt bit cannot be masked with the COMM_ADR_ERR_MASK bit Write 1 to clear interrupt. |
| 5 | COMM_CRC_ERR_SWINT | R/W1C | 0b | Latched status bit indicating that SPI (or I2C1) CRC error has been detected. CRC on I2C/SPI must be enabled (I2C1_SPI_CRC_EN=1 - NVM default bit) is required to generate the error-indication with this status bit This interrupt bit cannot be masked with the COMM_CRC_ERR_MASK bit Write 1 to clear interrupt. |
| 4 | COMM_FRM_ERR_SWINT | R/W1C | 0b | Latched status bit indicating that SPI frame error has been detected. Write 1 to clear interrupt. |
| 3 | ESM_MCU_PIN_SWINT | R/W1C | 0b | Latched status bit indicating that MCU ESM fault has been detected. Write 1 to clear interrupt. |
| 2 | TWARN_SWINT | 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. |
| 1 | WD_SWINT | R/W1C | 0b | Latched status bit indicating that Watchdog error has been detected. This bit is cleared by writing 1 when WD_RST_INT, WD_FAIL_INT and WD_LONGWIN_TIMEOUT_INT are cleared. |
| 0 | EN_DRV_STAT | R | 0b | State of EN_DRV pin. |

7.7.1.116 STARTUP_CTRL Register (Offset = 0xC3) [Reset = 0x00]

STARTUP_CTRL is shown in [Figure 7-175](#) and described in [Table 7-136](#).

Return to the [Table 7-19](#).

Figure 7-175. STARTUP_CTRL Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------------------|--------------|---|-----------|----------------|-------------------------|----------|---|
| FIRST_STARTUP_DONE | STARTUP_DEST | | FAST_BIST | LP_STANDBY_SEL | SKIP_LP_STANDBY_EE_READ | RESERVED | |
| R/W-0b | R/W-0b | | R/W-0b | R/W-0b | R/W-0b | R/W-0b | |

Table 7-136. STARTUP_CTRL Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|-------------------------|------|-------|--|
| 7 | FIRST_STARTUP_DONE | R/W | 0b | Control for register reset and EEPROM read at INIT state. See "Register Resets and EEPROM read at INIT state" chapter for operation. Note that SKIP_LP_STANDBY_EE_READ affects the operation when transitioning from LP_STANDBY to INIT. 0b = Conf_registers are reset and default values are loaded from EEPROM 1b = Conf_registers stay unchanged |
| 6:5 | STARTUP_DEST | R/W | 0b | FSM start-up destination select. 0b = STANDBY/LP_STANDBY based on LP_STANDBY_SEL 1b = Reserved 10b = MCU_ONLY 11b = ACTIVE |
| 4 | FAST_BIST | R/W | 0b | FAST_BIST 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: 0b = Normal standby state is used. 1b = Low power standby state is used as standby state. |
| 2 | SKIP_LP_STANDBY_EE_READ | R/W | 0b | Control for regmap and regmap_rtc register resets and EEPROM read: 0b = register reset and EEPROM read are controlled by FIRST_STARTUP_DONE bit 1b = registers stay unchanged (no reset or EEPROM read) |
| 1:0 | RESERVED | R/W | 0b | |

7.7.1.117 SCRATCH_PAD_REG_1 Register (Offset = 0xC9) [Reset = 0x00]

SCRATCH_PAD_REG_1 is shown in [Figure 7-176](#) and described in [Table 7-137](#).

Return to the [Table 7-19](#).

Figure 7-176. SCRATCH_PAD_REG_1 Register

| | | | | | | | |
|---------------|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| SCRATCH_PAD_1 | | | | | | | |
| R/W-0b | | | | | | | |

Table 7-137. 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 when VINT is disabled. The data is maintained when VINT regulator is enabled, for example during STANDBY and LP_STANDBY states. |

7.7.1.118 SCRATCH_PAD_REG_2 Register (Offset = 0xCA) [Reset = 0x00]

SCRATCH_PAD_REG_2 is shown in [Figure 7-177](#) and described in [Table 7-138](#).

Return to the [Table 7-19](#).

Figure 7-177. SCRATCH_PAD_REG_2 Register

| | | | | | | | |
|---------------|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| SCRATCH_PAD_2 | | | | | | | |
| R/W-0b | | | | | | | |

Table 7-138. 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 when VINT is disabled. The data is maintained when VINT regulator is enabled, for example during STANDBY and LP_STANDBY states. |

7.7.1.119 SCRATCH_PAD_REG_3 Register (Offset = 0xCB) [Reset = 0x00]

 SCRATCH_PAD_REG_3 is shown in [Figure 7-178](#) and described in [Table 7-139](#).

 Return to the [Table 7-19](#).

Figure 7-178. SCRATCH_PAD_REG_3 Register

| | | | | | | | |
|---------------|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| SCRATCH_PAD_3 | | | | | | | |
| R/W-0b | | | | | | | |

Table 7-139. 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 when VINT is disabled. The data is maintained when VINT regulator is enabled, for example during STANDBY and LP_STANDBY states. |

7.7.1.120 SCRATCH_PAD_REG_4 Register (Offset = 0xCC) [Reset = 0x00]

SCRATCH_PAD_REG_4 is shown in [Figure 7-179](#) and described in [Table 7-140](#).

Return to the [Table 7-19](#).

Figure 7-179. SCRATCH_PAD_REG_4 Register

| | | | | | | | |
|---------------|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| SCRATCH_PAD_4 | | | | | | | |
| R/W-0b | | | | | | | |

Table 7-140. 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 when VINT is disabled. The data is maintained when VINT regulator is enabled, for example during STANDBY and LP_STANDBY states. |

7.7.1.121 PFSM_DELAY_REG_1 Register (Offset = 0xCD) [Reset = 0x00]

 PFSM_DELAY_REG_1 is shown in [Figure 7-180](#) and described in [Table 7-141](#).

 Return to the [Table 7-19](#).

Figure 7-180. PFSM_DELAY_REG_1 Register

| | | | | | | | |
|-------------|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PFSM_DELAY1 | | | | | | | |
| R/W-0b | | | | | | | |

Table 7-141. 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) |

7.7.1.122 PFSM_DELAY_REG_2 Register (Offset = 0xCE) [Reset = 0x00]

PFSM_DELAY_REG_2 is shown in [Figure 7-181](#) and described in [Table 7-142](#).

Return to the [Table 7-19](#).

Figure 7-181. PFSM_DELAY_REG_2 Register

| | | | | | | | |
|-------------|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PFSM_DELAY2 | | | | | | | |
| R/W-0b | | | | | | | |

Table 7-142. 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) |

7.7.1.123 PFSM_DELAY_REG_3 Register (Offset = 0xCF) [Reset = 0x00]

 PFSM_DELAY_REG_3 is shown in [Figure 7-182](#) and described in [Table 7-143](#).

 Return to the [Table 7-19](#).

Figure 7-182. PFSM_DELAY_REG_3 Register

| | | | | | | | |
|-------------|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PFSM_DELAY3 | | | | | | | |
| R/W-0b | | | | | | | |

Table 7-143. 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) |

7.7.1.124 PFSM_DELAY_REG_4 Register (Offset = 0xD0) [Reset = 0x00]

PFSM_DELAY_REG_4 is shown in [Figure 7-183](#) and described in [Table 7-144](#).

Return to the [Table 7-19](#).

Figure 7-183. PFSM_DELAY_REG_4 Register

| | | | | | | | |
|-------------|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PFSM_DELAY4 | | | | | | | |
| R/W-0b | | | | | | | |

Table 7-144. 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) |

7.7.1.125 WD_ANSWER_REG Register (Offset = 0x401) [Reset = 0x00]

 WD_ANSWER_REG is shown in [Figure 7-184](#) and described in [Table 7-145](#).

 Return to the [Table 7-19](#).

Figure 7-184. WD_ANSWER_REG Register

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

Table 7-145. 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. |

7.7.1.126 WD_QUESTION_ANSW_CNT Register (Offset = 0x402) [Reset = 0x3C]

WD_QUESTION_ANSW_CNT is shown in [Figure 7-185](#) and described in [Table 7-146](#).

Return to the [Table 7-19](#).

Figure 7-185. WD_QUESTION_ANSW_CNT Register

| | | | | | | | |
|----------|---|-------------|---|-------------|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RESERVED | | WD_ANSW_CNT | | WD_QUESTION | | | |
| R-0b | | R-11b | | R-1100b | | | |

Table 7-146. 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 | 1100b | 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. |

7.7.1.127 WD_WIN1_CFG Register (Offset = 0x403) [Reset = 0x7F]

 WD_WIN1_CFG is shown in [Figure 7-186](#) and described in [Table 7-147](#).

 Return to the [Table 7-19](#).

Figure 7-186. WD_WIN1_CFG Register

| | | | | | | | |
|----------|--------------|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RESERVED | WD_WIN1 | | | | | | |
| R/W-0b | R/W-1111111b | | | | | | |

Table 7-147. 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 Watchdog chapter). These bits can be only be written when the watchdog is in the Long Window. |

7.7.1.128 WD_WIN2_CFG Register (Offset = 0x404) [Reset = 0x7F]

WD_WIN2_CFG is shown in [Figure 7-187](#) and described in [Table 7-148](#).

Return to the [Table 7-19](#).

Figure 7-187. WD_WIN2_CFG Register

| | | | | | | | |
|----------|---------|--------------|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RESERVED | WD_WIN2 | | | | | | |
| R/W-0b | | R/W-1111111b | | | | | |

Table 7-148. 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. |

7.7.1.129 WD_LONGWIN_CFG Register (Offset = 0x405) [Reset = 0xFF]

 WD_LONGWIN_CFG is shown in [Figure 7-188](#) and described in [Table 7-149](#).

 Return to the [Table 7-19](#).

Figure 7-188. WD_LONGWIN_CFG Register

| | | | | | | | |
|---------------|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| WD_LONGWIN | | | | | | | |
| R/W-11111111b | | | | | | | |

Table 7-149. 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) |

7.7.1.130 WD_MODE_REG Register (Offset = 0x406) [Reset = 0x02]

WD_MODE_REG is shown in [Figure 7-189](#) and described in [Table 7-150](#).

Return to the [Table 7-19](#).

Figure 7-189. 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 7-150. WD_MODE_REG Register Field Descriptions

| Bit | Field | Type | Reset | Description |
|-----|-------------------|------|-------|---|
| 7:3 | RESERVED | R/W | 0b | |
| 2 | WD_PWRHOLD | R/W | 0b | Watchdog hold on. 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. |

7.7.1.131 WD_QA_CFG Register (Offset = 0x407) [Reset = 0x0A]

 WD_QA_CFG is shown in [Figure 7-190](#) and described in [Table 7-151](#).

 Return to the [Table 7-19](#).

Figure 7-190. 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 7-151. 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. |

7.7.1.132 WD_ERR_STATUS Register (Offset = 0x408) [Reset = 0x00]

WD_ERR_STATUS is shown in [Figure 7-191](#) and described in [Table 7-152](#).

Return to the [Table 7-19](#).

Figure 7-191. WD_ERR_STATUS Register

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|-------------|-----------------|------------|-------------------|-------------------|------------|----------------------------|
| WD_RST_INT | WD_FAIL_INT | WD_ANSW_ER R | WD_SEQ_ERR | WD_ANSW_EA RLY | WD_TRIG_EAR LY | 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 7-152. 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_TIMEOU T_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. |

7.7.1.133 WD_THR_CFG Register (Offset = 0x409) [Reset = 0xFF]

 WD_THR_CFG is shown in [Figure 7-192](#) and described in [Table 7-153](#).

 Return to the [Table 7-19](#).

Figure 7-192. 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 7-153. 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. |

7.7.1.134 WD_FAIL_CNT_REG Register (Offset = 0x40A) [Reset = 0x20]

WD_FAIL_CNT_REG is shown in [Figure 7-193](#) and described in [Table 7-154](#).

Return to the [Table 7-19](#).

Figure 7-193. 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 7-154. 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. |

8 Application and Implementation

Note

Information in the following applications sections is not part of the TI component specification, and TI does not warrant its accuracy or completeness. TI's customers are responsible for determining suitability of components for their purposes, as well as validating and testing their design implementation to confirm system functionality.

8.1 Application Information

The device is a multi-phase step-down converter with four switcher cores, which generate four single-phase regulator outputs.

8.2 Typical Applications

The configuration is shown in the figure below.

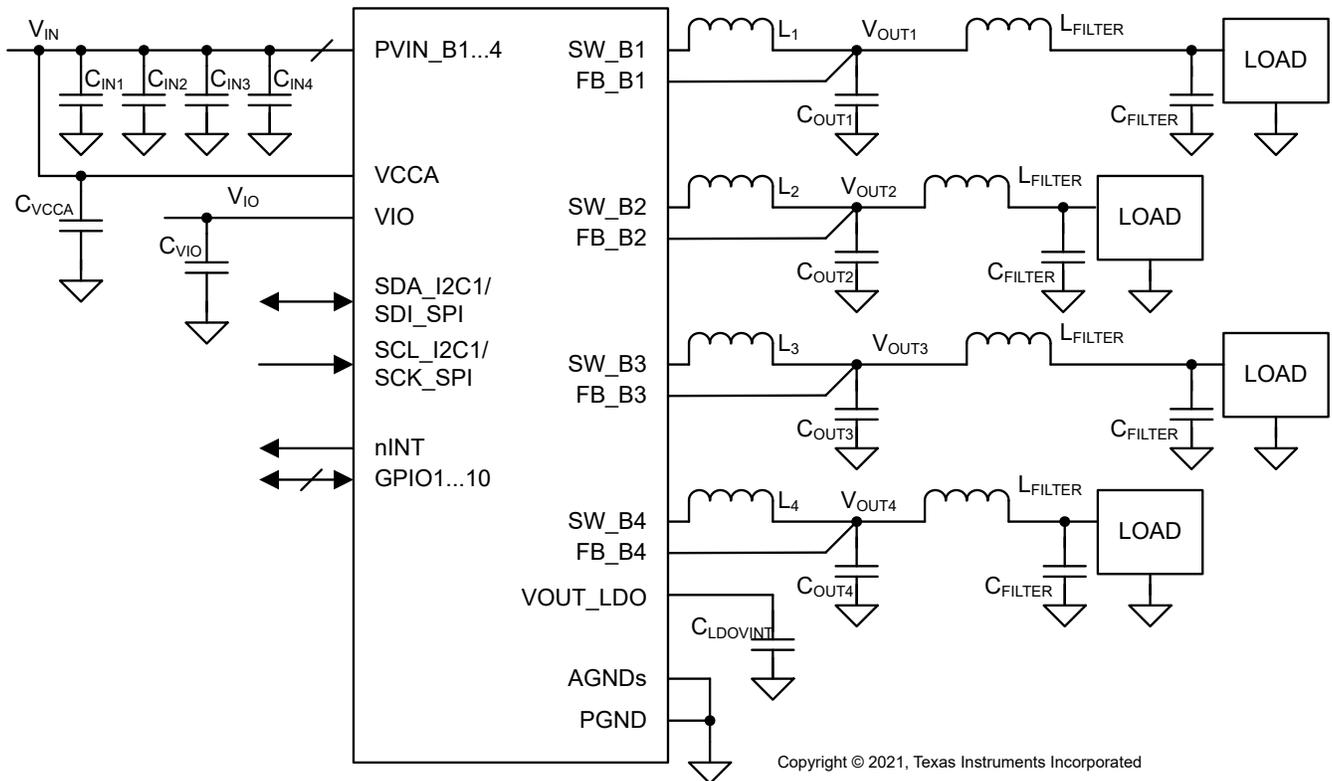


Figure 8-1. Four 1-Phase configuration

8.2.1 Design Requirements

8.2.1.1 Buck Inductor Selection

The buck inductors L_1 , L_2 , L_3 , and L_4 are shown in the [Section 8.2](#). The inductance and DCR of the inductor affects the control loop of the buck regulator. TI recommends using inductors similar to those listed in [Table 8-1](#). Pay attention to the saturation current and temperature rise current of the inductor. Check that the saturation current is higher than the peak current limit and the temperature rise current is higher than the maximum expected rms output current. DC resistance of the inductor must be minimized for good efficiency at high-current condition. The inductor AC loss (resistance) also affects conversion efficiency. Higher Q factor at switching frequency usually gives better efficiency at light load to middle load. Shielded inductors are preferred as they radiate less noise.

Table 8-1. Recommended Inductors

| MANUFACTURER | PART NUMBER | VALUE | DIMENSIONS L × W × H (mm) | RATED DC CURRENT, I_{SAT} typical (maximum) / I_{TEMP} typical (maximum) (A) | DCR typical (maximum) (mΩ) |
|--------------|--------------------------|---------------|------------------------------|--|----------------------------------|
| TDK | TFM322512ALMAR47 MTAA | 0.47 μH (20%) | 3.2 × 2.5 × 1.2 | 7.6 (6.9) / 6.1 (5.3) | 16 (21) |
| TDK | TFM252012ALMAR47 MTAA | 0.47 μH (20%) | 2.5 × 2.0 × 1.2 | 6.5 (5.8) / 5.6 (4.9) | 19 (24) |
| TDK | TFM201610ALMAR47 MTAA | 0.47 μH (20%) | 2.0 × 1.6 × 1.0 | 5.0 (4.5) / 4.5 (3.9) | 28 (39) |
| Murata | DFE252012PD-R47M | 0.47 μH (20%) | 2.5 × 2.0 × 1.2 | – (5.2) / – (4) ⁽¹⁾ | – (27) |
| Murata | DFE2HCAHR47MJ0 | 0.47 μH (20%) | 2.5 × 2.0 × 1.2 | – (5.1) / – (4.5) | 19 (25) |

(1) Operating temperature range is up to 125°C including self temperature rise.

8.2.1.2 Buck Input Capacitor Selection

The buck input capacitors C_{IN1} , C_{IN2} , C_{IN3} , and C_{IN4} are shown in the [Section 8.2](#). A ceramic input bypass capacitor of 10 μ F is required for each phase of the regulator. Place the input capacitor as close as possible to the PVIN_Bx pin and PGND pin of the device. A larger value or higher voltage rating improves the input voltage filtering. Use X7R type of capacitors, not Y5V or F. DC bias characteristics capacitors must be considered, minimum effective input capacitance to ensure good performance is 3 μ F per buck input at maximum input voltage including tolerances and ambient temperature range. See [Table 8-2](#).

The input filter capacitor supplies current to the high-side FET switch in the first half of each cycle and reduces voltage ripple imposed on the input power source. A ceramic capacitor with low ESR provides the best noise filtering of the input voltage spikes due to this rapidly changing current. Select an input filter capacitor with sufficient ripple current rating. In addition ferrite can be used in front of the input capacitor to reduce the EMI.

For optimal performance, additional 1 μ F 3-terminal input capacitors are required. Buck1 and buck2 can share one 3-T capacitor and buck3 and buck4 can share one 3-T capacitor. See [Table 8-2](#).

Table 8-2. Recommended Input Capacitors (X7R Dielectric)

| MANUFACTURER | PART NUMBER | VALUE | CASE SIZE | DIMENSIONS L x W x H (mm) | VOLTAGE RATING |
|--------------|--------------------------|------------------|-----------|---------------------------|----------------|
| Murata | GCM21BR71A106KE22 | 10 μ F (10%) | 0805 | 2 x 1.25 x 1.25 | 10 V |
| TDK | CGA4J1X7S1C106K125A B | 10 μ F (10%) | 0805 | 2 x 1.25 x 1.25 | 10 V |
| Murata | NFM18HC105C1C3 (3-T) | 1 μ F (20%) | 0603 | 1.6 x 0.8 x 0.7 | 16 V |
| TDK | YFF18AC0J105M (3-T) | 1 μ F (20%) | 0603 | 1.6 x 0.8 x 0.6 | 6.3 V |

8.2.1.3 Buck Output Capacitor Selection

The buck output capacitors C_{OUT1} , C_{OUT2} , C_{OUT3} , and C_{OUT4} are shown in [Section 8.2](#). Use ceramic local output capacitors, X7R or X7T types; do not use Y5V or F. DC bias voltage characteristics of ceramic capacitors must be considered. The output filter capacitor smooths out current flow from the inductor to the load, helps maintain a steady output voltage during transient load changes and reduces output voltage ripple. These capacitors must be selected with sufficient capacitance and sufficiently low ESR and ESL to perform these functions. Minimum effective output capacitance (including the DC voltage roll-off, tolerances, aging and temperature effects) is defined in Electrical Characteristics table for different buck configurations.

The output voltage ripple is caused by the charging and discharging of the output capacitor and also due to its R_{ESR} . The R_{ESR} is frequency dependent (as well as temperature dependent); make sure the value used for selection process is at the switching frequency of the part. See [Table 8-3](#).

POL capacitors (C_{POL1} , C_{POL2} , C_{POL3} , C_{POL4}) can be used to improve load transient performance and to decrease the ripple voltage. A higher output capacitance improves the load step behavior and reduces the output voltage ripple. Note that the output capacitor may be the limiting factor in the output voltage ramp and the maximum total output capacitance listed in electrical characteristics must not be exceeded. At shutdown the output voltage is discharged to 0.15 V level using forced-PWM operation. The discharging of the output capacitor can increase the input voltage if the load current is small and the output capacitor is large. Below 0.15 V level the output capacitor is discharged by the internal discharge resistor and with large capacitor more time is required to settle V_{OUT} down as a consequence of the increased time constant.

Table 8-3. Recommended Output Capacitors (X7R or X7T Dielectric)

| MANUFACTURER | PART NUMBER | VALUE | CASE SIZE | DIMENSIONS L × W × H (mm) | VOLTAGE RATING |
|--------------|--------------------------|-------------|-----------|---------------------------|----------------|
| Murata | GCM32EC71A476KE02 | 47 μF (10%) | 1210 | | 10 V |
| TDK | CGA6P1X7S1A476M250A C | 47 μF (20%) | 1210 | | 10 V |
| Murata | GCM32ER70J476ME19 | 47 μF (20%) | 1210 | 3.2 × 2.5 × 2.5 | 6.3 V |
| TDK | CGA6P1X7S0J476M250A C | 47 μF (20%) | 1210 | | 6.3 V |
| TDK | CGA5L1X7T0G476M | 47 μF (20%) | 1206 | | 4 V |
| Murata | GCM31CR71A226KE02 | 22 μF (10%) | 1206 | 3.2 × 1.6 × 1.6 | 10 V |
| TDK | CGA5L1X7S1A226M160A C | 22 μF (20%) | 1206 | | 10 V |
| Murata | GCM21BD70J226ME36 | 22 μF (20%) | 0805 | 2.0 × 1.25 × 1.25 | 6.3 V |
| TDK | CGA4J1X7T0J226M | 22 μF (20%) | 0805 | | 6.3 V |
| Murata | NFM18HC106D0G (3-T) | 10 μF (20%) | 0603 | 1.6 × 0.8 × 1.25 | 4 V |

Every buck output requires a local output capacitor to form the capacitive part of the LC output filter. These local output capacitors must be placed as close to the inductor as possible to minimize electromagnetic emissions. See [Section 8.3.1](#) for more information about component placement.

To achieve better ripple and transient performance, additional capacitors are recommended to compensate the parasitic impedances due to board routing and provide faster transient response to a load step. These caps are placed close to the point of load (POL). POL capacitor usage varies based on the application and generally follows the SoC or FPGA input capacitor requirements. Low ESL 3-terminal caps are recommended, as their high performance can help reduce the total number of capacitors required which simplifies board layout design and saves board area. They also help to reduce the total cost of the solution.

[Figure 8-2](#) is an example power distribution network (PDN) of local and POL caps at the output of a buck for optimal ripple and transient performance. [Table 8-4](#) lists the local and POL capacitors used to validate the buck transient and ripple performance specified in the parametric table. [Table 8-5](#) lists the actual capacitor part numbers used for the different use case tests. It is recommended to simulate and validate that the capacitor network chosen for a particular design meets the desired requirements as these are provided as guidelines.

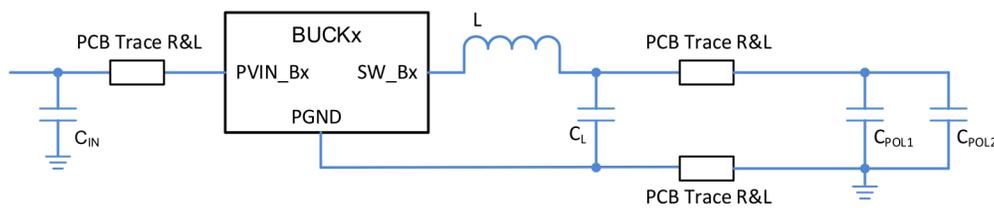


Figure 8-2. Buck Regulators Power Distribution Network (PDN)

Table 8-4. Local and POL Capacitors and Inductors used for Buck Use Case Simulations and Validations

| Use Case # | Use Case Test Condition | Fsw | Phase Config. | L | C _L per phase | R _{PCB} per phase1 | L _{PCB} per phase2 | C _{POL1} | C _{POL2} |
|------------|-------------------------|---------|---------------|--------|------------------------------|-----------------------------|-----------------------------|---|-------------------|
| 1 | 8.8 MHz Min C | 8.8 MHz | 1 PH | 470 nH | 22 μF × 2 | | | | |
| 2 | 8.8 MHz Max C | 8.8 MHz | 1 PH | 470 nH | 22 μF × 4 | | | | |
| 3 | 8.8 MHz Filter1 Min C | 8.8 MHz | 1 PH | 470 nH | 22 μF × 2, point of feedback | | | Additional inductor (100nH) and capacitors (20 μF total) for filtering | |
| 4 | 8.8 MHz Filter1 Max C | 8.8 MHz | 1 PH | 470 nH | 22 μF × 4, point of feedback | | | Additional inductor (100 nH) and capacitors (80 μF total) for filtering | |
| 5 | 8.8 MHz Filter2 Min C | 8.8 MHz | 1 PH | 470 nH | 22 μF × 2, point of feedback | | | Ferrite and capacitors (20 μF total) for filtering | |
| 6 | 8.8 MHz Filter2 Max C | 8.8 MHz | 1 PH | 470 nH | 22 μF × 4, point of feedback | | | Ferrite and capacitors (80 μF total) for filtering | |

1. R_{PCB} is the PCB wiring resistance between local and POL capacitors including both positive and negative paths. For multi-phase outputs the total resistance is divided by the number of phases.
2. L_{PCB} is the PCB wiring inductance between local and POL capacitors including both positive and negative paths. For multi-phase outputs the total inductance is divided by the number of phases.

Power input and output wiring parasitic resistance and inductance must be minimized.

Table 8-5. Capacitor and Inductor Part Numbers in Buck Use Case Simulations and Validations

| Component | Component Value | Component Part Number | Description |
|--------------------------------|-----------------|-------------------------------|---|
| C _{IN} ⁽¹⁾ | 1 μF | NFM18HC105C1C3 ⁽²⁾ | MuRata 3-T Cap: 1.0 μF ±20% 16 V, X7S, 0603, -55°C to 125°C |
| C _L | 22 μF | GCM21BD70J226ME36L | MuRata Cap: 22 μF±20%, 6.3 V, X7T, 0805 |
| C _{POL} | 1 μF | GCM188R71C105KA64J | MuRata Cap: 1 μF±10%, 16 V, X7R, 0603 |
| C _{POL} | 10 μF | GCM21BR71A106KE22L | MuRata Cap: 10 μF±10%, 10 V, X7R, 0805 |
| C _{POL} | 22 μF | GCM21BD70J226ME36L | MuRata Cap: 22 μF±20%, 6.3 V, X7T, 0805 |
| L | 470 nH | TFM252012ALMAR47MTAA | TDK Inductor: 0.47 μH, 20 V, 24 mΩ DCR, 5.8A Isat, 4.9A Itemp, -55°C to 150°C |
| L _{Filter1} | 100 nH | NLCV32T-R10M-EFRD | TDK Inductor: 100 nH, 2.85 A, 24 mΩ DCR |
| Ferrite _{Filter2} | | MPZ2012S300A | TDK Ferrite: 30 Ω at 100 MHz, 6 A |

- (1) One 3-T capacitor is shared with Buck1/Buck2 and Buck3/Buck4 inputs. Additional bulk capacitors are connected to PVIN_x power supplies.
- (2) Low ESL 3-terminal cap

8.2.1.4 LDO Output Capacitor Selection

A 2.2- μ F capacitor is recommended for internal LDO output. See [Table 8-6](#) for the specific part number of the recommended output capacitors.

Table 8-6. Recommended LDO Output Capacitors⁽¹⁾

| MANUFACTURER | PART NUMBER | VALUE | EIA size code | SIZE (mm) |
|--------------|----------------------|------------------------|---------------|------------------|
| TDK | CGA3E1X7S1C225M080AC | 2.2 μ F, 16 V, X7S | 0603 | 1.6 \times 0.8 |
| Murata | GCM188R70J225KE22 | 2.2 μ F, 16 V, X7R | 0603 | 1.6 \times 0.8 |

(1) Component minimum and maximum tolerance values are specified in the electrical parameters section of each IP.

8.2.1.5 VCCA Supply Filtering Components

The VCCA input is used to power LDOVINT internal regulator and other internal functions. The VCCA input pin is always connected in parallel with the buck input pins (PVIN_Bx pins). See [Table 8-7](#) for recommended components for VCCA input supply filtering.

Table 8-7. Recommended VCCA Supply Filtering Components

| MANUFACTURER | PART NUMBER | VALUE | CASE SIZE | DIMENSIONS L \times W \times H (mm) | VOLTAGE RATING (V) |
|--------------|----------------------|--------|-----------|---|--------------------|
| Murata | GCM155C71A474KE36 | 470 nF | 0402 | 1.0 \times 0.5 \times 0.5 | 10 |
| TDK | CGA2B3X7S1A474K050BB | 470 nF | 0402 | 1.0 \times 0.5 \times 0.5 | 10 |

Table 8-8. Recommended VIO Capacitor

| COMPONENT | MANUFACTURER | PART NUMBER | VALUE | EIA SIZE CODE | SIZE (mm) | USED FOR VALIDATION |
|-----------|--------------|----------------------|-------------------------|---------------|------------------|---------------------|
| Capacitor | Murata | GCM155C71A474KE36 | 0.47 μ F, 10 V, X7S | 0402 | 1.0 \times 0.5 | Yes |
| Capacitor | TDK | CGA2B3X7S1A474K050BB | 0.47 μ F, 10 V, X7S | 0402 | 1.0 \times 0.5 | — |

8.2.2 Detailed Design Procedure

The performance of the device depends greatly on the care taken in designing the printed circuit board (PCB). The use of low-inductance and low serial-resistance ceramic capacitors is strongly recommended, while correct grounding is crucial. Attention must be given to decoupling the power supplies. Decoupling capacitors must be connected close to the device and between the power and ground pins to support high peak currents being drawn from system power rail during turn on of the switching MOSFETs. Keep input and output traces as short as possible, because trace inductance, resistance, and capacitance can easily become the performance limiting items. The separate power pins PVIN_Bx are not connected together internally. Connect the PVIN_Bx power connections together outside the package using power plane construction.

8.2.3 Voltage Scaling Precautions

Voltage scaling behavior depends on the set slew rate and output capacitance. Therefore there may be limitations on maximum capacitance with certain slew rate settings. Worst case over- and undershoot also depends on the phase margin of the buck converter. Good phase margin mitigates the over- and undershoot. Reducing the output voltage slew rate reduces the slew rate current thus reducing the worst case over- and undershoot.

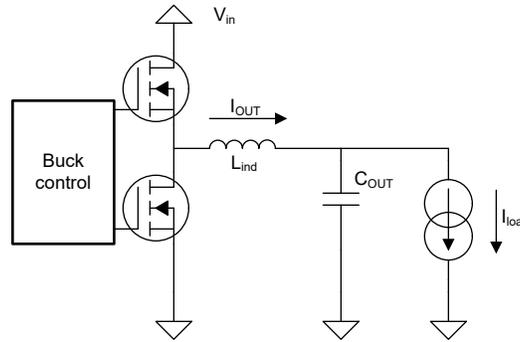


Figure 8-3. Power stage

The following equations must be met to ensure the output voltage scaling with regulator switching frequency:

$$I_{SR_vout} + I_{load_max} \leq I_{out_max} \tag{27}$$

$$I_{out_min} \leq I_{load_min} - I_{SR_vout} \tag{28}$$

Where $I_{SR_vout} = SR_{Vout}C_{out}$

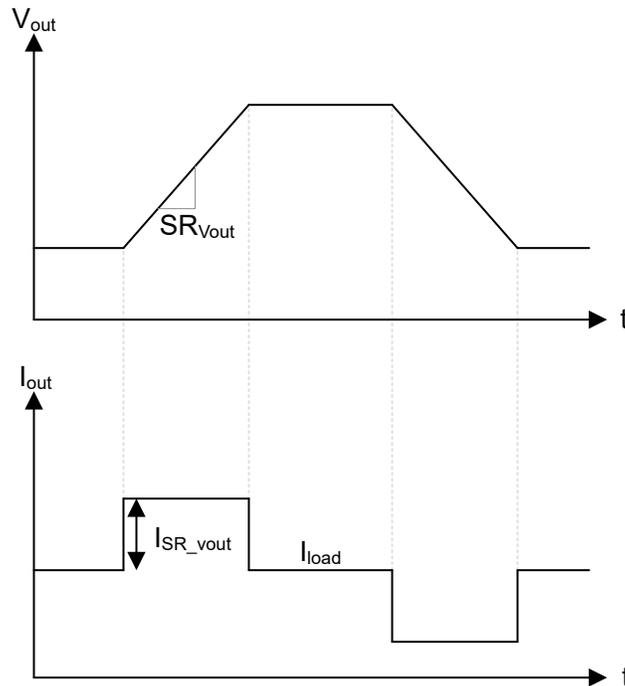


Figure 8-4. Voltage ramp

The worst case overshoot can be calculated with the following equations.

$$V_{overshoot} = \frac{t_{Iout_scale_down} I_{SR_vout}}{2C_{out}} \tag{29}$$

Where $t_{Iout_scale_down} = \frac{I_{SR_vout}}{SR_{Iout_scale_down}}$ and $SR_{Iout_scale_down} = \frac{V_{out}}{L_{ind}}$.

After substitutions

$$V_{\text{overshoot}} = \frac{SR_{\text{vout}}^2 C_{\text{out}} L_{\text{ind}}}{2V_{\text{out}}} \quad (30)$$

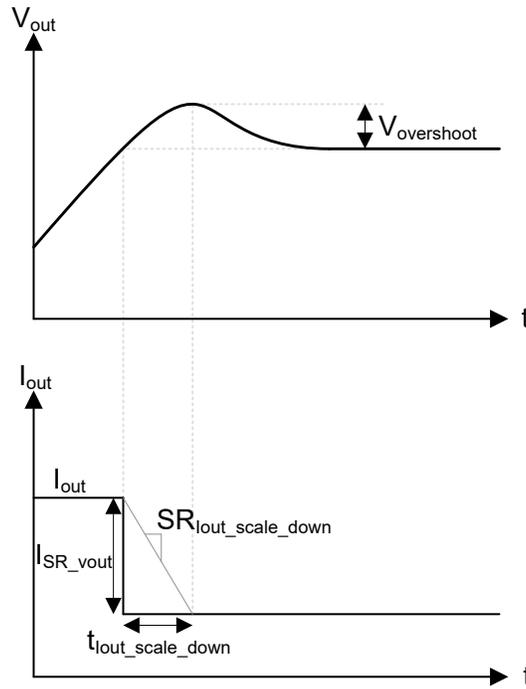


Figure 8-5. Overshoot

The worst case undershoot can be calculated with the following equations

$$V_{\text{undershoot}} = \frac{t_{\text{Iout_scale_up}} I_{\text{SR_vout}}}{2C_{\text{out}}} \quad (31)$$

Where $t_{\text{Iout_scale_up}} = \frac{I_{\text{SR_vout}}}{SR_{\text{Iout_scale_up}}}$ and $SR_{\text{Iout_scale_up}} = \frac{V_{\text{in}} - V_{\text{out}}}{L_{\text{ind}}}$.

After substitutions

$$V_{\text{undershoot}} = \frac{SR_{\text{vout}}^2 C_{\text{out}} L_{\text{ind}}}{2(V_{\text{in}} - V_{\text{out}})} \quad (32)$$

8.2.4 Application Curves

$V_{IN} = 3.3V$, $V_{OUT} = 1.0V$, $T_A = 25^\circ C$ unless otherwise noted.

Efficiency results for other use cases can be estimated using PEET-GUI,

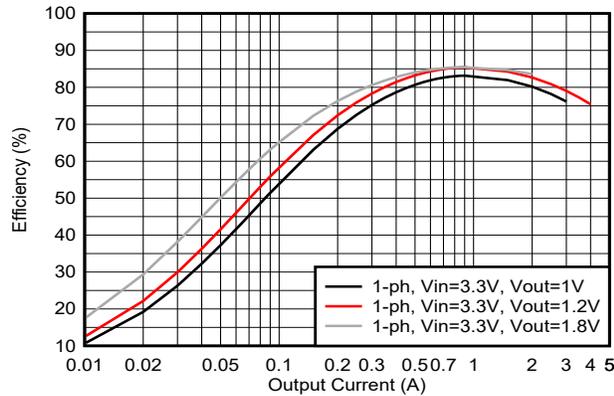


Figure 8-6. Efficiency in 8.8MHz 1-Phase, Force PWM Mode

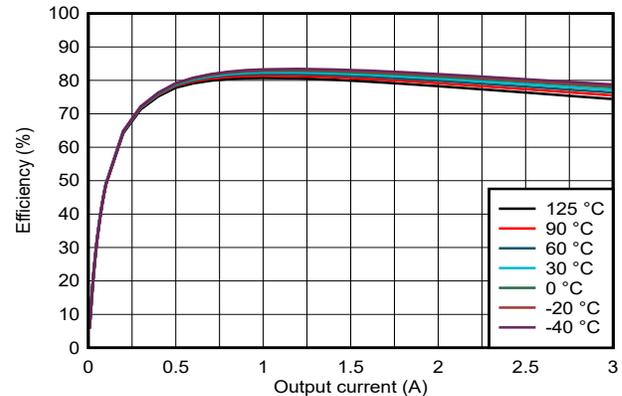
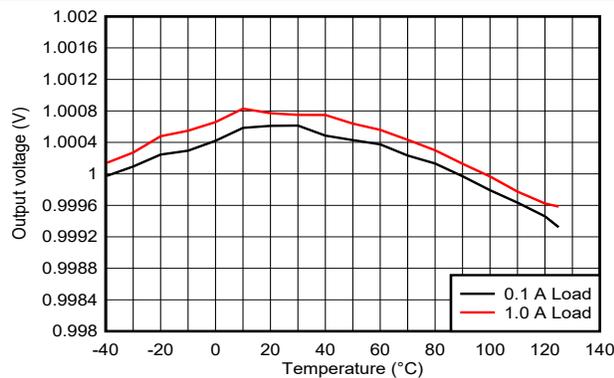


Figure 8-7. Efficiency in 8.8MHz 1-Phase, Force PWM Mode



$I_{LOAD} = 0.1A$ and
1A

Figure 8-8. Output Voltage Accuracy 8.8MHz Single Phase Mode

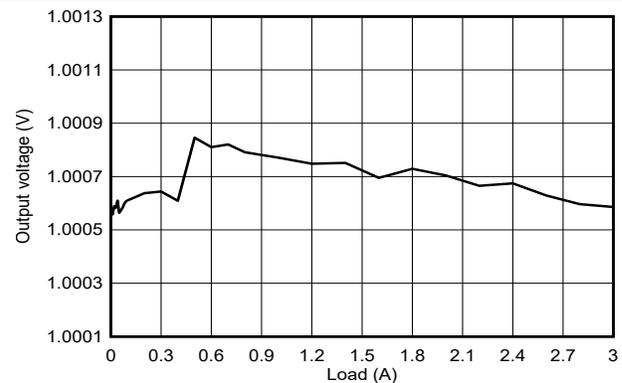


Figure 8-9. Load Regulation, FPWM Mode

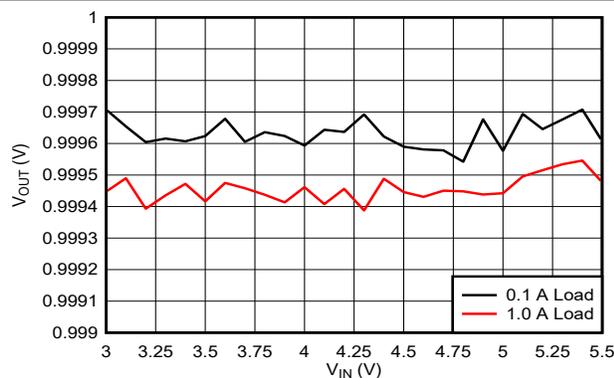
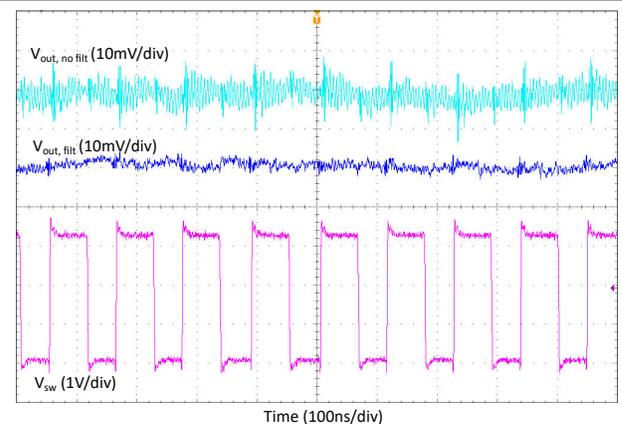
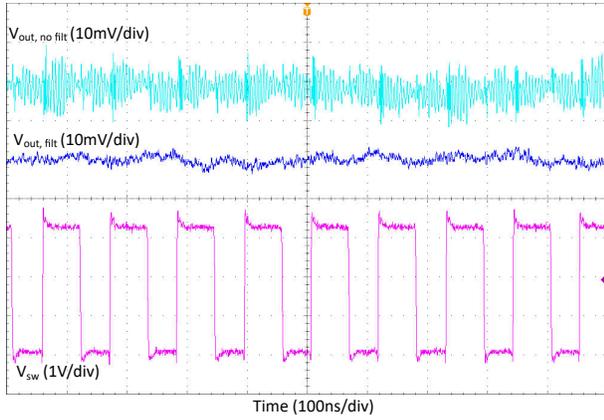


Figure 8-10. Line Regulation



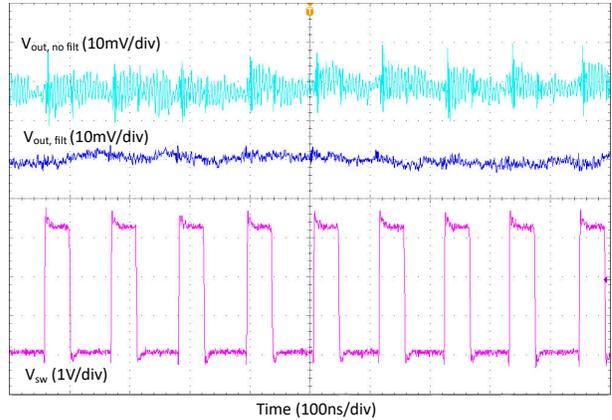
$I_{LOAD} = 200\text{ mA}$ $V_{OUT} = 1.8\text{ V}$

Figure 8-11. Output Voltage Ripple 8.8 MHz Single Phase. FPWM mode. L = 100 nH.



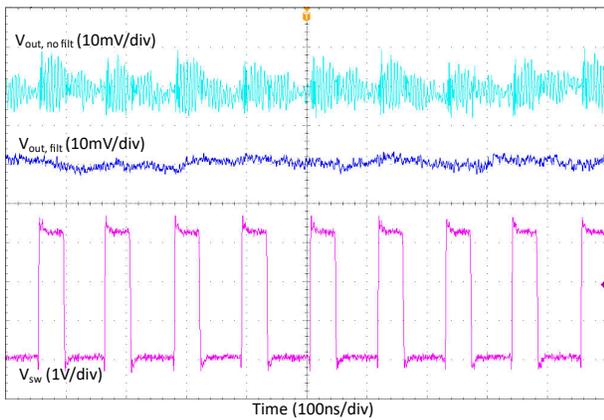
$I_{LOAD} = 200\text{ mA}$ $V_{OUT} = 1.8\text{ V}$

Figure 8-12. Output Voltage Ripple 8.8 MHz Single Phase. FPWM mode. L = 33 nH.



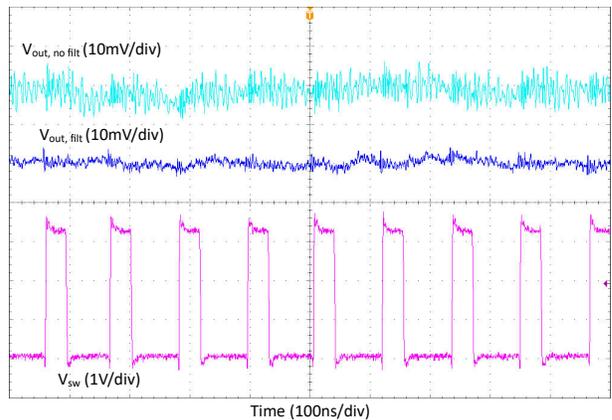
$I_{LOAD} = 200\text{ mA}$ $V_{OUT} = 1.2\text{ V}$

Figure 8-13. Output Voltage Ripple 8.8 MHz Single Phase. FPWM mode. L = 100 nH.



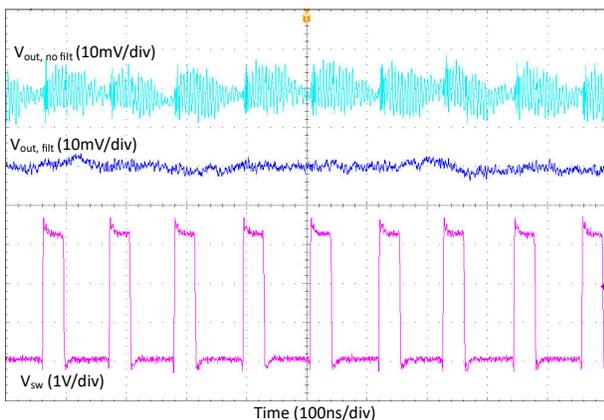
$I_{LOAD} = 200\text{ mA}$ $V_{OUT} = 1.2\text{ V}$

Figure 8-14. Output Voltage Ripple 8.8 MHz Single Phase. FPWM mode. L = 33 nH.



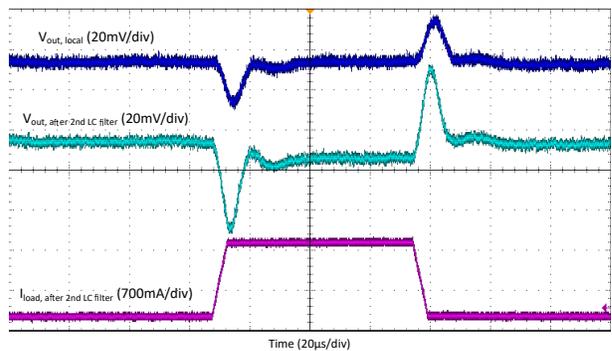
$I_{LOAD} = 200\text{ mA}$ $V_{OUT} = 1\text{ V}$

Figure 8-15. Output Voltage Ripple 8.8 MHz Single Phase. FPWM mode. L = 100 nH.



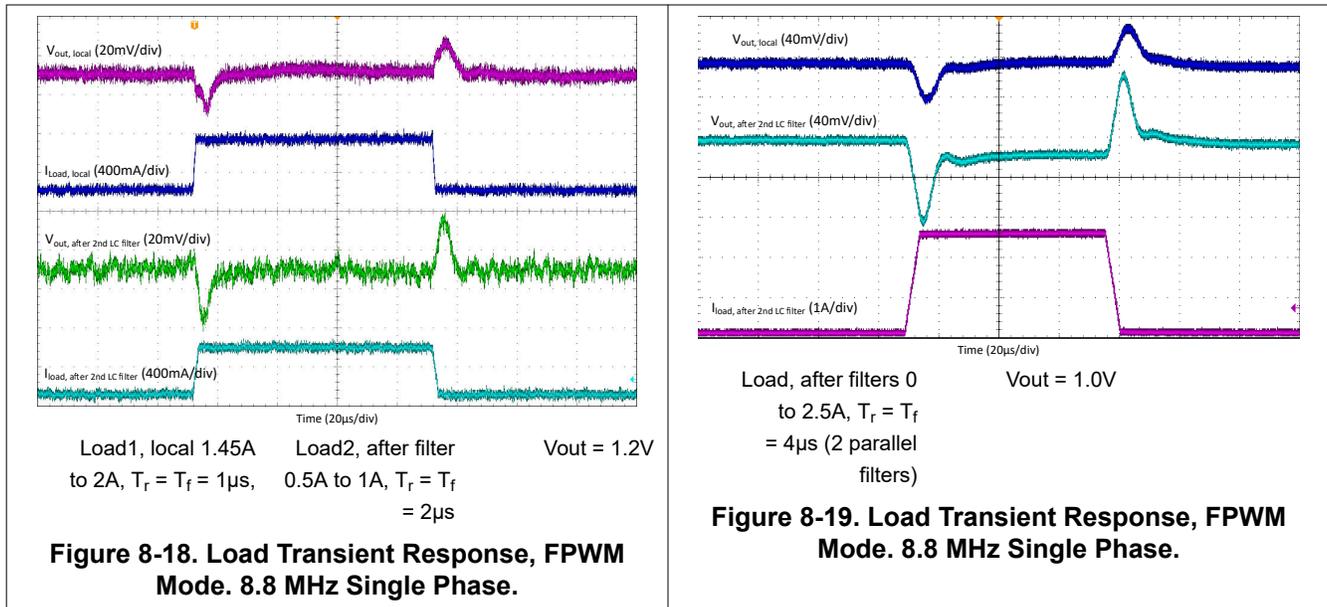
$I_{LOAD} = 200\text{ mA}$ $V_{OUT} = 1\text{ V}$

Figure 8-16. Output Voltage Ripple 8.8 MHz Single Phase. FPWM mode. L = 33 nH.



Load, after filters $V_{out} = 1.8\text{ V}$
0.5 to 1.8A, $T_r =$
 $T_f = 4\mu\text{s}$ (2 parallel
filters)

Figure 8-17. Load Transient Response, FPWM Mode. 8.8 MHz Single Phase.



8.3 Layout

8.3.1 Layout Guidelines

The high frequency and large switching currents of the device make the choice of layout important. Good power supply results only occur when care is given to correct design and layout. Layout affects noise pickup and generation and can cause a good design to perform with less-than-expected results. With a range of output currents from milliamps to 10 A and over, good power supply layout is much more difficult than most general PCB design. Use the following steps as a reference to ensure the device is stable and maintains correct voltage and current regulation across its intended operating voltage and current range.

1. Place C_{IN} as close as possible to the $PVIN_Bx$ pin and the $PGND$ pin. Route the V_{IN} trace wide and thick to avoid IR drops. The trace between the positive node of the input capacitor and the $PVIN_Bx$ pins of the device, as well as the trace between the negative node of the input capacitor and power $PGND$ pin, must be kept as short as possible. The input capacitance provides a low-impedance voltage source for the switching converter. The inductance of the connection is the most important parameter of a local decoupling capacitor — parasitic inductance on these traces must be kept as small as possible for correct device operation. The parasitic inductance can be reduced by using a ground plane as close as possible to top layer by using thin dielectric layer between top layer and ground plane.
2. The output filter, consisting of C_{OUT} and L , converts the switching signal at SW_Bx to the noiseless output voltage. The output filter must be placed as close as possible to the device keeping the switch node small, for best EMI behavior. Route the traces between the output capacitors of the device and the load direct and wide to avoid losses due to the IR drop.
3. Input for analog blocks ($VCCA$ and $AGND$) must be isolated from noisy signals. Connect $VCCA$ directly to a quiet system voltage node and $AGND$ to a quiet ground point where no IR drop occurs. Place the decoupling capacitor as close as possible to the $VCCA$ pin.
4. If the processor load supports remote voltage sensing, connect the feedback pins FB_Bx of the device to the respective sense pins on the processor. The sense lines are susceptible to noise. They must be kept away from noisy signals such as $PGND$, $PVIN_Bx$, and SW_Bx , as well as high bandwidth signals such as the I^2C . Avoid both capacitive and inductive coupling by keeping the sense lines short, direct, and close to each other. Run the lines in a quiet layer. Isolate them from noisy signals by a voltage or ground plane if possible. Running the signal as a differential pair is recommended. If series resistors are used for load current measurement, place them after connection of the voltage feedback.
5. $PGND$, $PVIN_Bx$, and SW_Bx must be routed on thick layers. They must not surround inner signal layers, that are not able to withstand interference from noisy $PGND$, $PVIN_Bx$ and SW_Bx .

Due to the small package of this converter and the overall small solution size, the thermal performance of the PCB layout is important. Many system-dependent parameters such as thermal coupling, airflow, added heat sinks and convection surfaces, and the presence of other heat-generating components affect the power dissipation limits of a given component. Proper PCB layout, focusing on thermal performance, results in lower die temperatures. Wide and thick power traces come with the ability to sink dissipated heat. The heat dissipation can be improved further on multi-layer PCB designs with vias to different planes, that results in reduced junction-to-ambient ($R_{\theta JA}$) and junction-to-board ($R_{\theta JB}$) thermal resistances and thereby reduces the device junction temperature, T_J . TI strongly recommends to perform of a careful system-level 2D or full 3D dynamic thermal analysis at the beginning product design process, by using a thermal modeling analysis software.

8.3.2 Layout Example

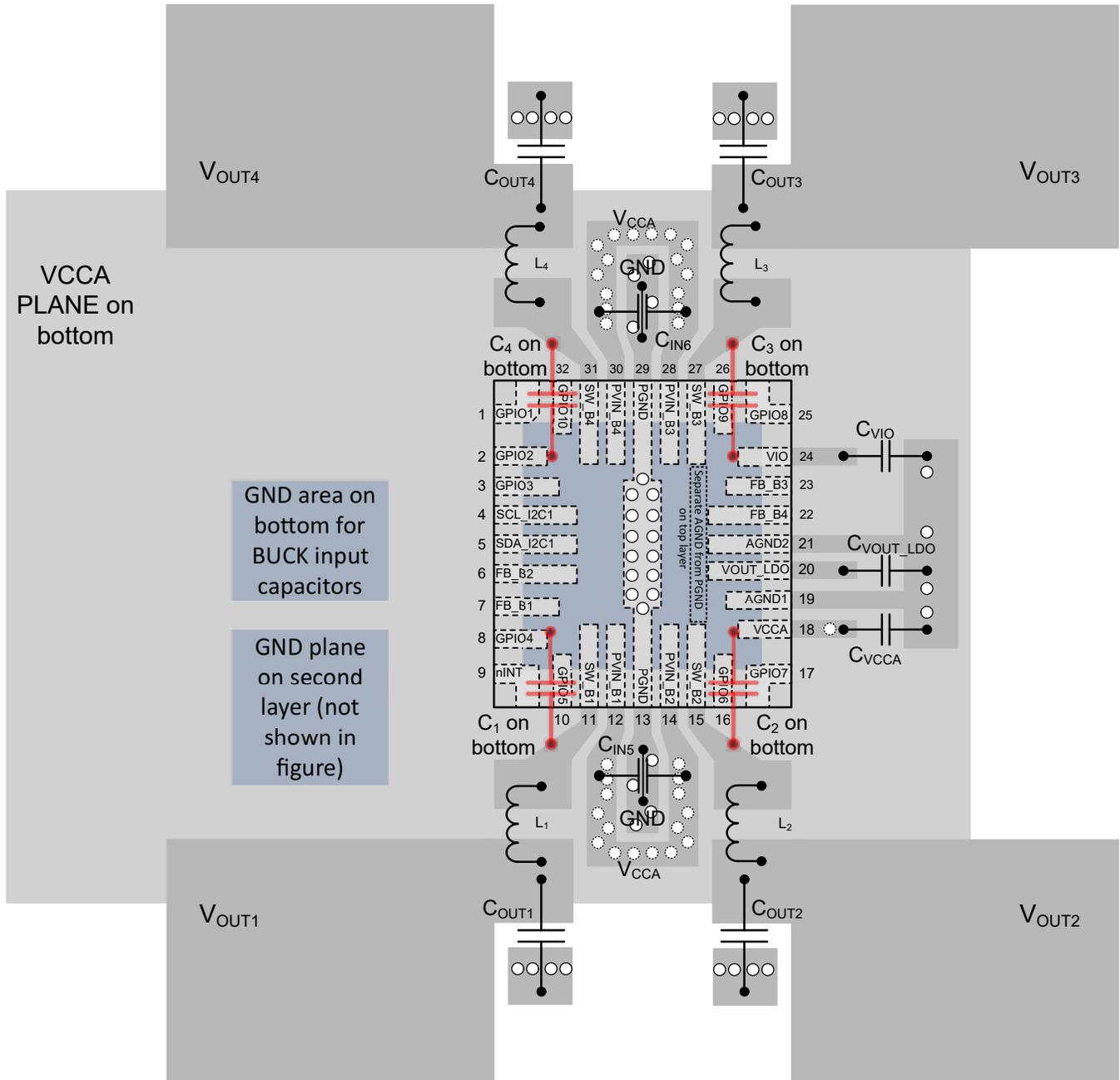
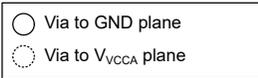


Figure 8-20. Device Board Layout

8.4 Power Supply Recommendations

The device is designed to operate from an input voltage supply range from 3.0 V and 5.5 V. This input supply must be well regulated and able to withstand maximum input current and maintain stable voltage without voltage drop even at load transition condition. The resistance of the input supply rail must be low enough that the input current transient does not cause too high drop in the device supply voltage that can cause false UVLO fault triggering. If the input supply is located more than a few inches from the additional bulk capacitance of the device may be required in addition to the ceramic bypass capacitors.

9 Device and Documentation Support

9.1 Receiving Notification of Documentation Updates

To receive notification of documentation updates, navigate to the device product folder on ti.com. Click on *Subscribe to updates* to register and receive a weekly digest of any product information that has changed. For change details, review the revision history included in any revised document.

9.2 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](#).

9.3 Trademarks

TI E2E™ is a trademark of Texas Instruments.

All trademarks are the property of their respective owners.

9.4 Electrostatic Discharge Caution



This integrated circuit can be damaged by ESD. Texas Instruments recommends that all integrated circuits be handled with appropriate precautions. Failure to observe proper handling and installation procedures can cause damage.

ESD damage can range from subtle performance degradation to complete device failure. Precision integrated circuits may be more susceptible to damage because very small parametric changes could cause the device not to meet its published specifications.

9.5 Glossary

[TI Glossary](#) This glossary lists and explains terms, acronyms, and definitions.

10 Mechanical, Packaging, and Orderable Information

The following pages include mechanical, packaging, and orderable information. This information is the most current data available for the designated devices. This data is subject to change without notice and revision of this document. For browser-based versions of this data sheet, refer to the left-hand navigation.

PACKAGING INFORMATION

| Orderable part number | Status (1) | Material type (2) | Package Pins | Package qty Carrier | RoHS (3) | Lead finish/ Ball material (4) | MSL rating/ Peak reflow (5) | Op temp (°C) | Part marking (6) |
|----------------------------------|---------------|----------------------|--------------------|-----------------------|-------------|--------------------------------------|-----------------------------------|--------------|---------------------|
| LP876242B0RQKRQ1 | Active | Production | VQFN-HR (RQK) 32 | 3000 LARGE T&R | Yes | SN | Level-2-260C-1 YEAR | -40 to 125 | LP8762 42B0-Q1 |
| LP876242B0RQKRQ1.A | Active | Production | VQFN-HR (RQK) 32 | 3000 LARGE T&R | Yes | SN | Level-2-260C-1 YEAR | -40 to 125 | LP8762 42B0-Q1 |

(1) **Status:** For more details on status, see our [product life cycle](#).

(2) **Material type:** When designated, preproduction parts are prototypes/experimental devices, and are not yet approved or released for full production. Testing and final process, including without limitation quality assurance, reliability performance testing, and/or process qualification, may not yet be complete, and this item is subject to further changes or possible discontinuation. If available for ordering, purchases will be subject to an additional waiver at checkout, and are intended for early internal evaluation purposes only. These items are sold without warranties of any kind.

(3) **RoHS values:** Yes, No, RoHS Exempt. See the [TI RoHS Statement](#) for additional information and value definition.

(4) **Lead finish/Ball material:** Parts may have multiple material finish options. Finish options are separated by a vertical ruled line. Lead finish/Ball material values may wrap to two lines if the finish value exceeds the maximum column width.

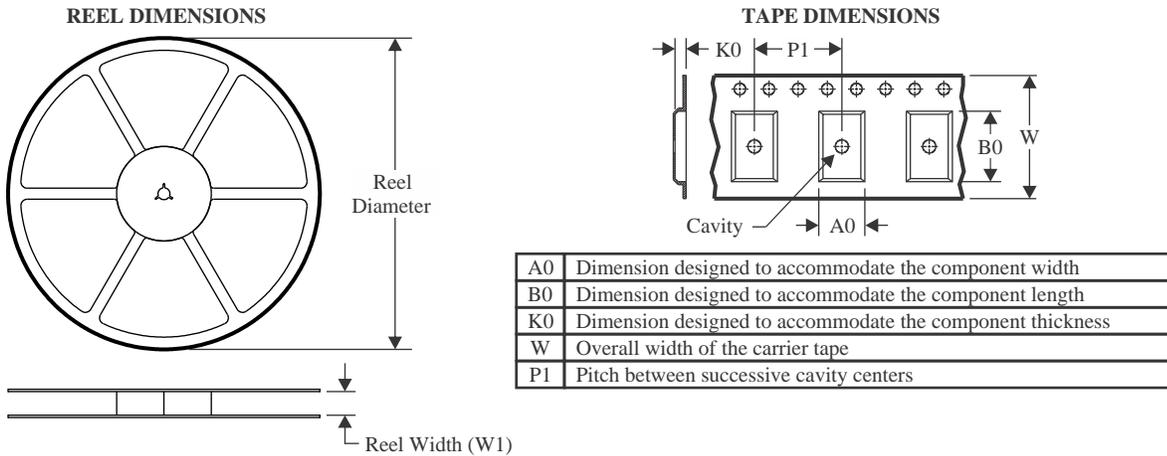
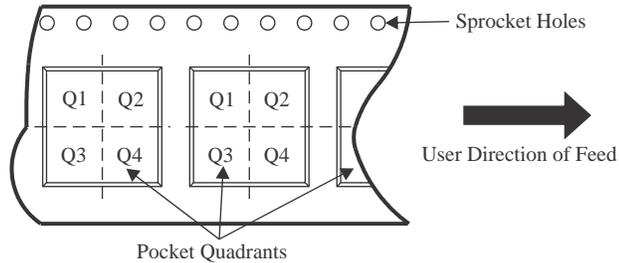
(5) **MSL rating/Peak reflow:** The moisture sensitivity level ratings and peak solder (reflow) temperatures. In the event that a part has multiple moisture sensitivity ratings, only the lowest level per JEDEC standards is shown. Refer to the shipping label for the actual reflow temperature that will be used to mount the part to the printed circuit board.

(6) **Part marking:** There may be an additional marking, which relates to the logo, the lot trace code information, or the environmental category of the part.

Multiple part markings will be inside parentheses. Only one part marking contained in parentheses and separated by a "-" will appear on a part. If a line is indented then it is a continuation of the previous line and the two combined represent the entire part marking for that device.

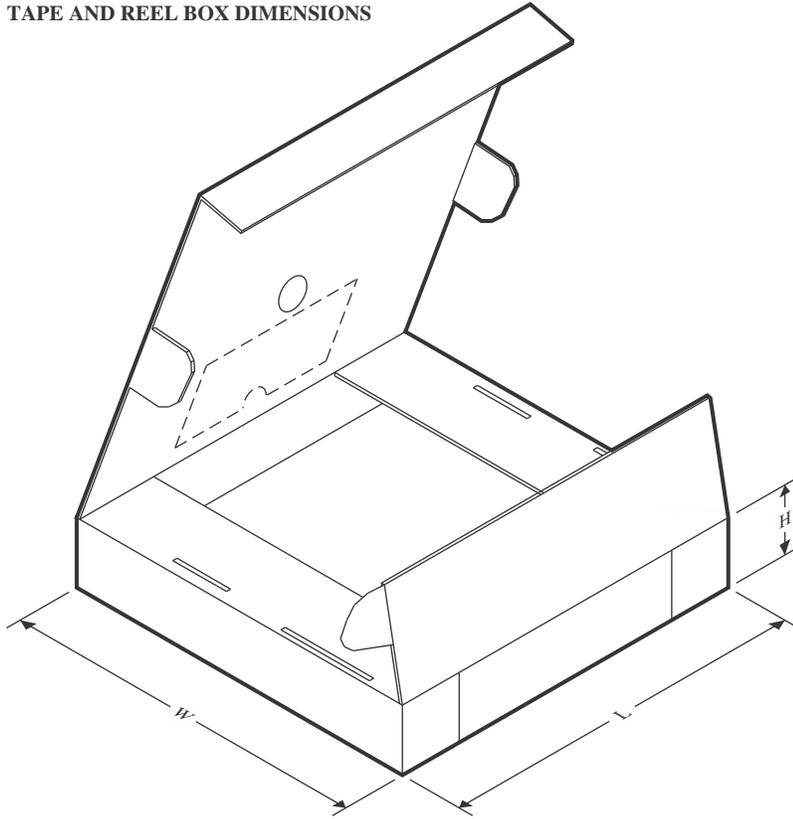
Important Information and Disclaimer: The information provided on this page represents TI's knowledge and belief as of the date that it is provided. TI bases its knowledge and belief on information provided by third parties, and makes no representation or warranty as to the accuracy of such information. Efforts are underway to better integrate information from third parties. TI has taken and continues to take reasonable steps to provide representative and accurate information but may not have conducted destructive testing or chemical analysis on incoming materials and chemicals. TI and TI suppliers consider certain information to be proprietary, and thus CAS numbers and other limited information may not be available for release.

In no event shall TI's liability arising out of such information exceed the total purchase price of the TI part(s) at issue in this document sold by TI to Customer on an annual basis.

TAPE AND REEL INFORMATION

QUADRANT ASSIGNMENTS FOR PIN 1 ORIENTATION IN TAPE


*All dimensions are nominal

| Device | Package Type | Package Drawing | Pins | SPQ | Reel Diameter (mm) | Reel Width W1 (mm) | A0 (mm) | B0 (mm) | K0 (mm) | P1 (mm) | W (mm) | Pin1 Quadrant |
|------------------|--------------|-----------------|------|------|--------------------|--------------------|---------|---------|---------|---------|--------|---------------|
| LP876242B0RQKRQ1 | VQFN-HR | RQK | 32 | 3000 | 330.0 | 12.4 | 5.25 | 5.75 | 1.05 | 8.0 | 12.0 | Q1 |

TAPE AND REEL BOX DIMENSIONS


*All dimensions are nominal

| Device | Package Type | Package Drawing | Pins | SPQ | Length (mm) | Width (mm) | Height (mm) |
|------------------|--------------|-----------------|------|------|-------------|------------|-------------|
| LP876242B0RQKRQ1 | VQFN-HR | RQK | 32 | 3000 | 367.0 | 367.0 | 38.0 |

GENERIC PACKAGE VIEW

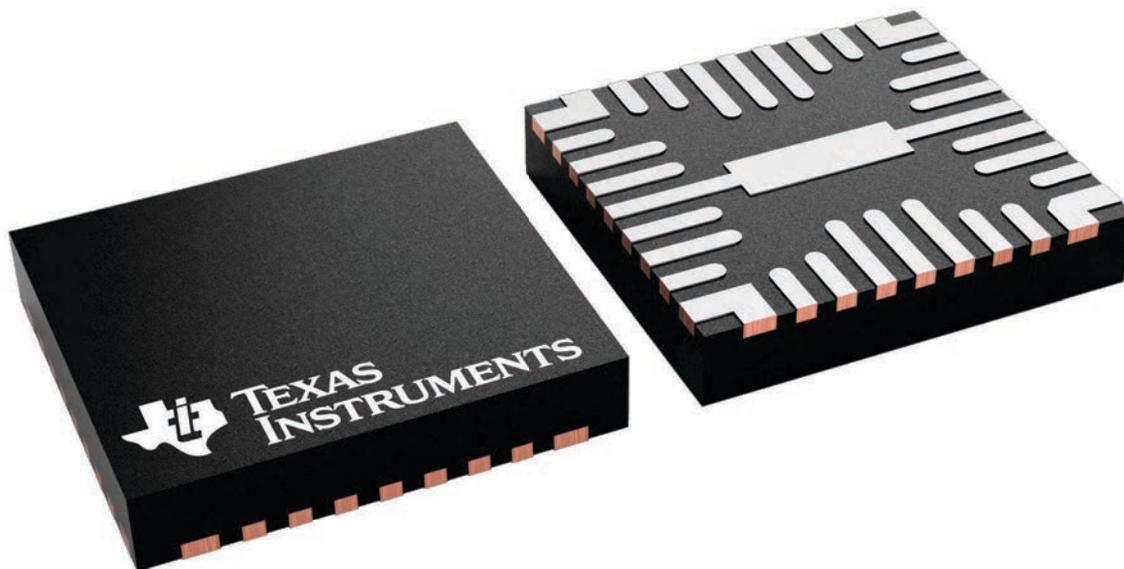
RQK 32

VQFN-HR - 1 mm max height

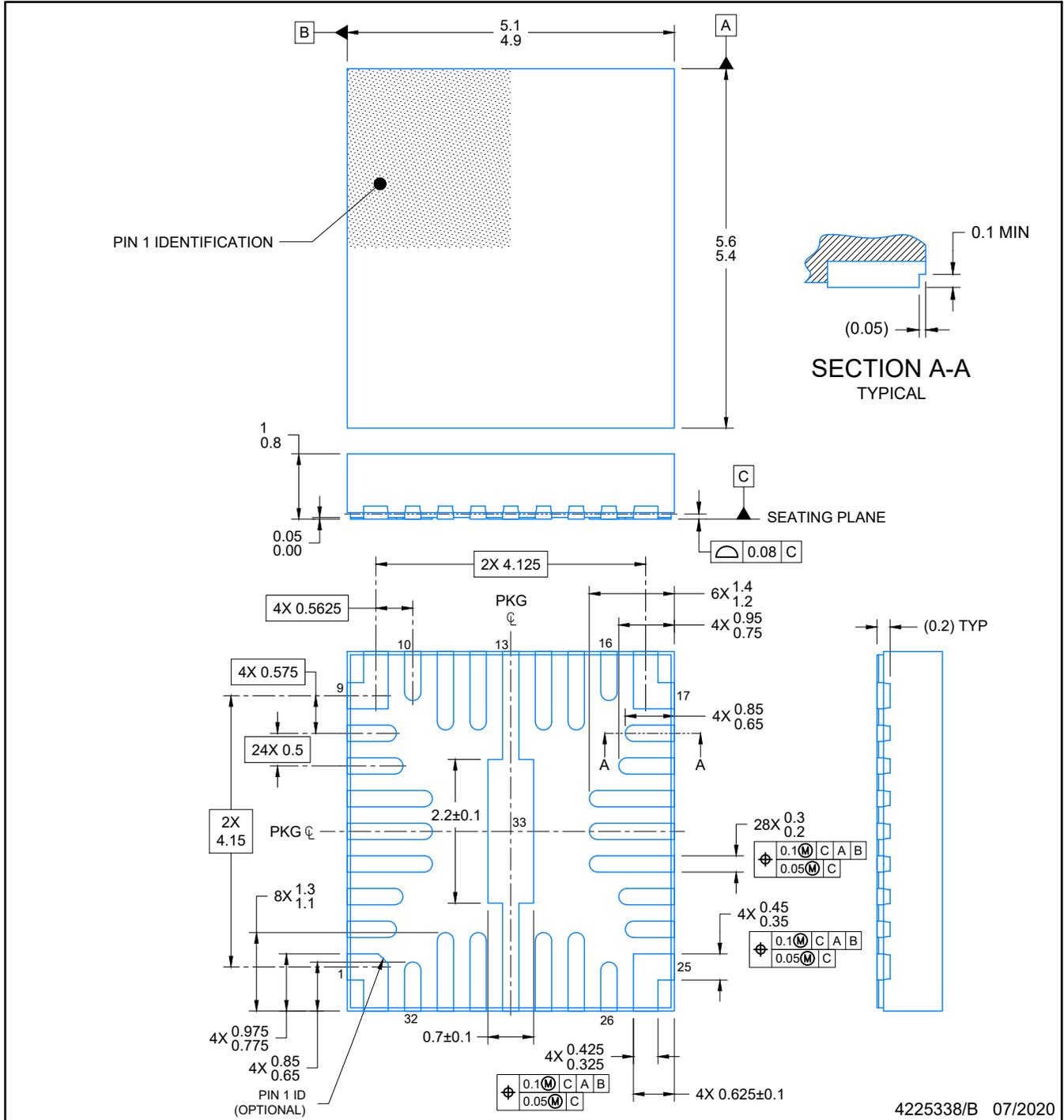
5 x 5.5, 0.5 mm pitch

PLASTIC QUAD FLATPACK - NO LEAD

This image is a representation of the package family, actual package may vary.
Refer to the product data sheet for package details.

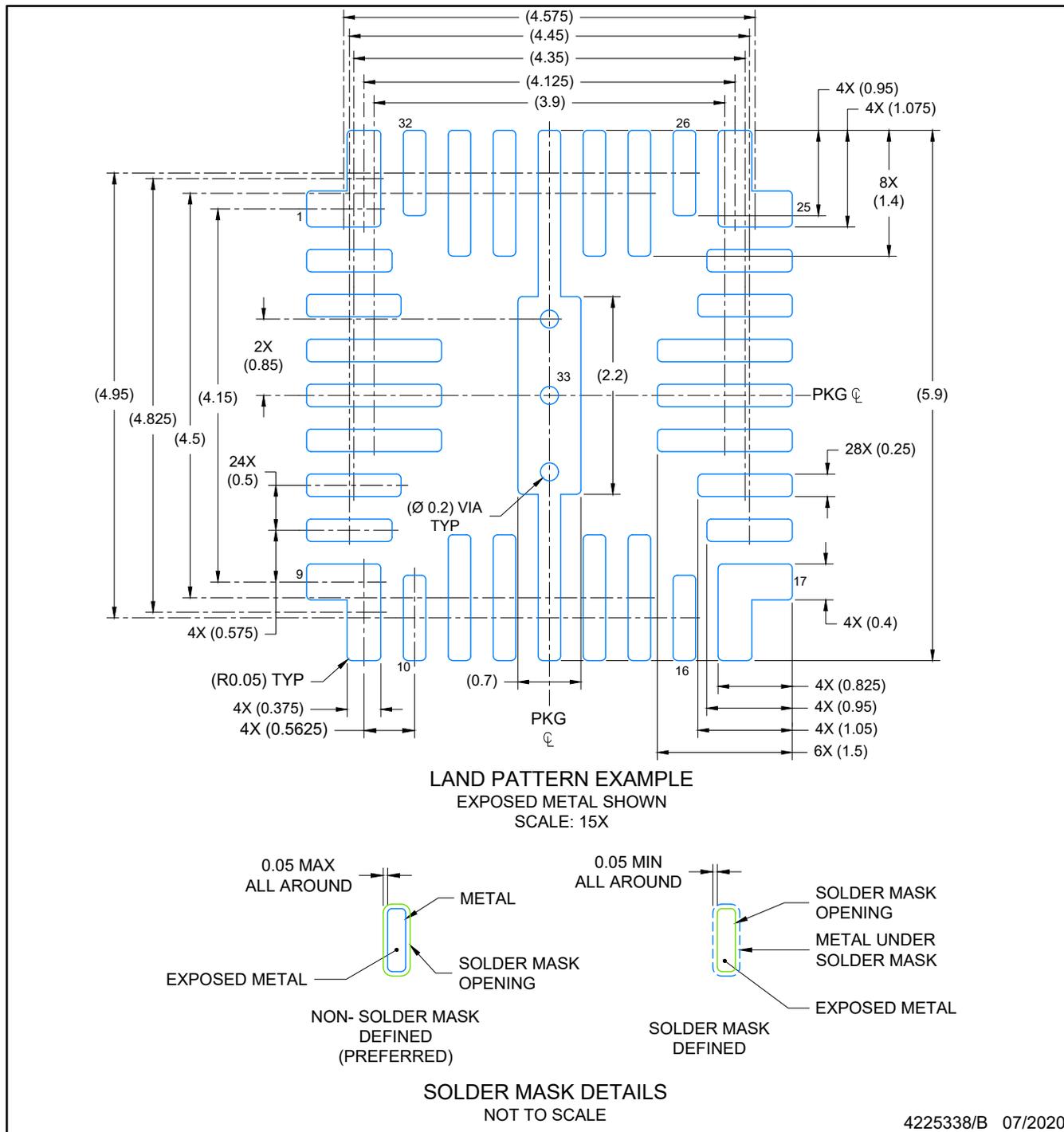


4225349/B



NOTES:

1. All linear dimensions are in millimeters. Any dimensions in parenthesis are for reference only. Dimensioning and tolerancing per ASME Y14.5M.
2. This drawing is subject to change without notice.



NOTES: (continued)

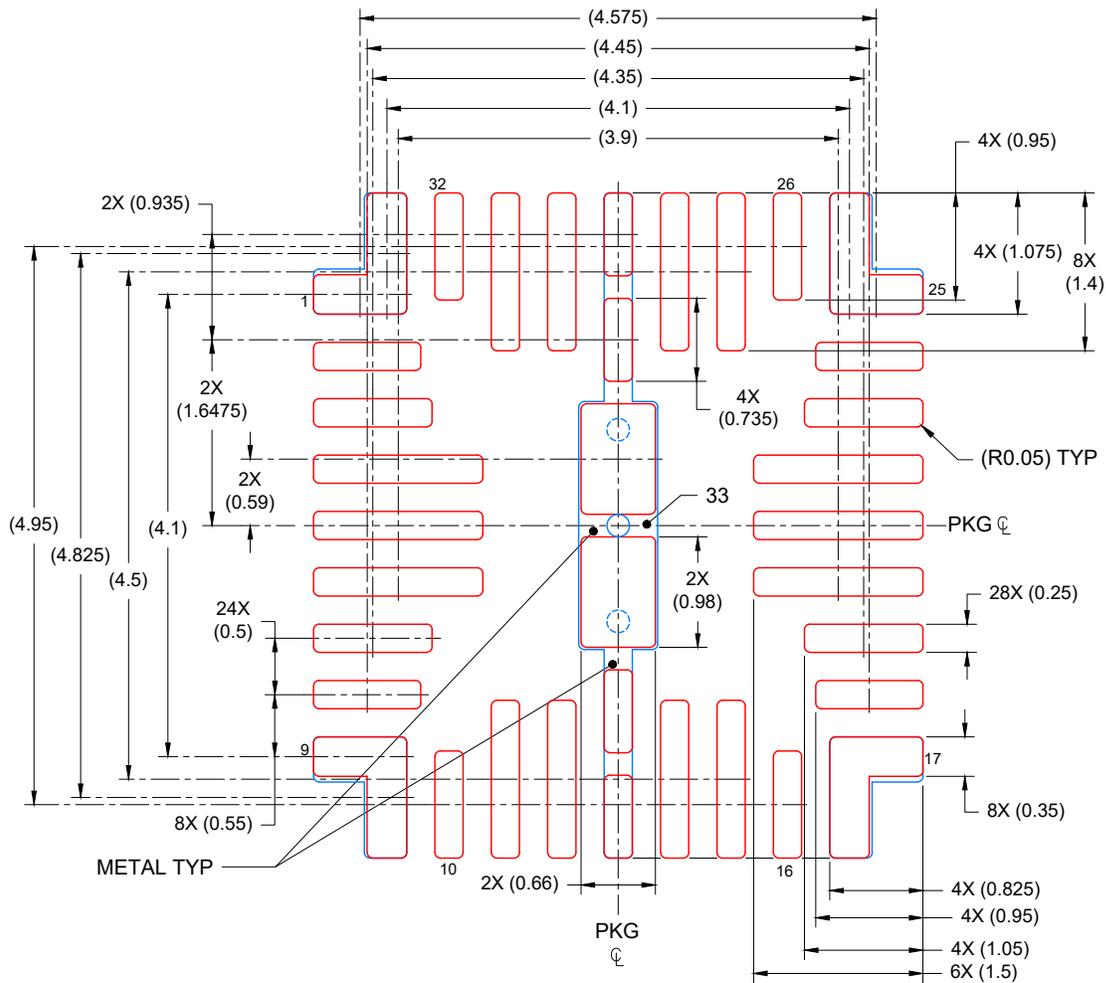
3. For more information, see Texas Instruments literature number SLUA271 (www.ti.com/lit/sluea271).
4. Solder mask tolerances between and around signal pads can vary based on board fabrication site.

EXAMPLE STENCIL DESIGN

RQK0032A

VQFN-HR - 1 mm max height

PLASTIC QUAD FLATPACK-NO LEAD



SOLDER PASTE EXAMPLE
BASED ON 0.1mm THICK STENCIL

PIN 1,9,16 & 25: 93%; PIN 13& 29: 79%; PIN 33: 84%
SCALE: 15X

4225338/B 07/2020

NOTES: (continued)

5. Laser cutting apertures with trapezoidal walls and rounded corners may offer better paste release. IPC-7525 may have alternate design recommendations.

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES "AS IS" AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you fully indemnify TI and its representatives against any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI's products are provided subject to [TI's Terms of Sale](#), [TI's General Quality Guidelines](#), or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI's provision of these resources does not expand or otherwise alter TI's applicable warranties or warranty disclaimers for TI products. Unless TI explicitly designates a product as custom or customer-specified, TI products are standard, catalog, general purpose devices.

TI objects to and rejects any additional or different terms you may propose.

Copyright © 2026, Texas Instruments Incorporated

Last updated 10/2025