

BQ79616 16-Series Battery Monitor, Balancer, and Integrated Hardware Protector

1 Features

- $\pm 1.5\text{mV}$ ADC accuracy
- Built-in redundancy path for voltage and temperature diagnostics
- Highly accurate cell voltage measurements within $128\mu\text{s}$ for all cell channels
- Integrated post-ADC configurable digital low-pass filters
- Bus bar connection and measurement support
- Built-in host-controlled hardware reset to emulate POR-like device reset
- Supports internal cell balancing
 - Balancing current at 240mA
 - Built-in balancing thermal management with automatic pause and resume control
- Isolated differential daisy chain communication with optional ring architecture
- Embedded fault signal and heartbeat through communication line
- UART/SPI host interface/communication bridge device BQ79600
- Built-in SPI controller

2 Applications

- [Battery Management System \(BMS\) in hybrid and electric powertrain systems](#)
- [Energy storage battery packs with Battery Management Systems](#)

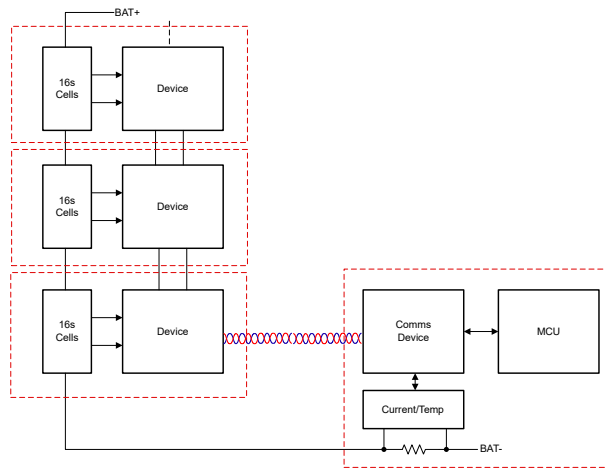
3 Description

The BQ79616 device provides high-accuracy cell voltage measurements in less than $200\mu\text{s}$ for 16-series battery modules in high-voltage battery management systems in HEV/EV. The monitor offers different channel options in the same package type, providing pin-to-pin compatibility and supporting high reuse of the established software and hardware across any platform. The integrated front-end filters enable the system to implement with simple, low voltage rating, differential RC filters on the cell input channels. The integrated, post-ADC, low-pass filters enable filtered, DC-like, voltage measurements for better state of charge (SOC) calculation. This device supports autonomous internal cell balancing with temperature monitoring to auto-pause and resume balancing to avoid an overtemperature condition.

Package Information

| PART NUMBER | PACKAGE ⁽¹⁾ | BODY SIZE (NOM) |
|-------------|------------------------|-------------------|
| BQ79616 | HTQFP (64-pin) | 10.00mm × 10.00mm |

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



Simplified System Diagram



Table of Contents

| | | | |
|--|-----------|--|------------|
| 1 Features | 1 | 7 Application and Implementation | 187 |
| 2 Applications | 1 | 7.1 Application Information..... | 187 |
| 3 Description | 1 | 7.2 Typical Applications..... | 187 |
| 4 Pin Configuration and Functions | 3 | 8 Power Supply Recommendations | 198 |
| 5 Specifications | 6 | 9 Layout | 199 |
| 5.1 Absolute Maximum Ratings | 6 | 9.1 Layout Guidelines..... | 199 |
| 5.2 ESD Ratings | 6 | 9.2 Layout Example..... | 202 |
| 5.3 Recommended Operating Conditions | 7 | 10 Device and Documentation Support | 206 |
| 5.4 Thermal Information | 7 | 10.1 Device Support..... | 206 |
| 5.5 Electrical Characteristics | 7 | 10.2 Receiving Notification of Documentation Updates..... | 206 |
| 5.6 Timing Requirements | 13 | 10.3 Support Resources..... | 206 |
| 5.7 Typical Characteristics..... | 16 | 10.4 Trademarks..... | 206 |
| 6 Detailed Description | 18 | 10.5 Electrostatic Discharge Caution..... | 206 |
| 6.1 Overview..... | 18 | 10.6 Glossary..... | 206 |
| 6.2 Functional Block Diagram..... | 19 | 11 Revision History | 206 |
| 6.3 Feature Description..... | 20 | 12 Mechanical, Packaging, and Orderable Information | 206 |
| 6.4 Device Functional Modes..... | 100 | | |
| 6.5 Register Maps..... | 109 | | |

4 Pin Configuration and Functions

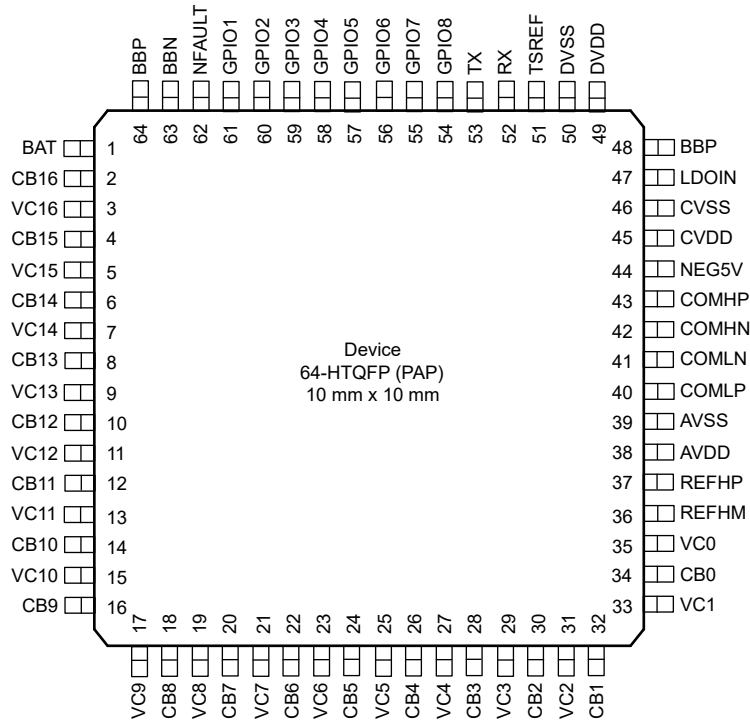


Table 4-1. Pin Functions

| PIN | | TYPE | DESCRIPTION |
|---------|-----|------|---|
| NAME | No. | | |
| BQ79616 | | | |
| BAT | 1 | P | Power supply input and top of module measurement input. Connect to the top cell of the battery module. |
| NPN | 48 | P | Connect to the base of an external NPN transistor. |
| LDOIN | 47 | P | 6-V preregulated analog power supply input/sense pin. Connect to the emitter of the external NPN transistor and connect a 0.1- μ F decoupling capacitor to CVSS. |
| AVDD | 38 | P | 5-V regulated output. AVDD supplies the internal analog circuits. Bypass AVDD with a capacitor to AVSS. |
| AVSS | 39 | GND | Analog ground. Ground connection for internal analog circuits. Connect DVSS, CVSS, REFHM, and AVSS externally. |
| NEG5V | 44 | P | Negative 5-V charge pump used for daisy chain and Main ADC. Connect with a capacitor to CVSS. |
| DVDD | 49 | P | 1.8-V regulated output. DVDD supplies the internal digital circuits. Bypass DVDD with a capacitor to DVSS. |
| DVSS | 50 | GND | Digital ground. Ground connection for internal digital logics. Connect DVSS, CVSS, REFHM, and AVSS externally. |
| CVDD | 45 | P | 5-V daisy chain communication and I/Os power supply. CVDD supplies the stack daisy chain communication transceiver circuit and the I/O pins. This power supply also supports an additional 10-mA external load in ACTIVE and SLEEP. |
| CVSS | 46 | GND | Daisy chain communication ground. Ground connection for internal daisy chain transceivers. Connect DVSS, CVSS, REFHM, and AVSS externally. |
| TSREF | 51 | P | 5-V bias voltage for NTC thermistor. Connect TSREF to the top of the NTC resistor divider network to the GPIOs when they are configured for NTC temperature monitoring. Bypass TSREF with a capacitor to CVSS. |
| REFHP | 37 | P | Precision reference output pin. Bypass with a capacitor to REFHM. |
| REFHM | 36 | GND | Precision reference ground. Ground connection for the internal precision reference. Connect DVSS, CVSS, REFHM, and AVSS externally. |
| VC16 | 3 | I | Cell voltage sense input. Connect to the positive terminal of cell 16. Connect a differential RC filter to VC15. Tie unused NC pins to BAT pin as explained in Cell Connections . |

Table 4-1. Pin Functions (continued)

| PIN | | TYPE | DESCRIPTION |
|---------|-----|------|---|
| NAME | No. | | |
| BQ79616 | | | |
| VC15 | 5 | I | Cell voltage sense input. Connect to the positive terminal of cell 15. Connect a differential RC filter to VC14. Tie unused NC pins to BAT pin as explained in Cell Connections . |
| VC14 | 7 | I | Cell voltage sense input. Connect to the positive terminal of cell 14. Connect a differential RC filter to VC13. Tie unused NC pins to BAT pin as explained in Cell Connections . |
| VC13 | 9 | I | Cell voltage sense input. Connect to the positive terminal of cell 13. Connect a differential RC filter to VC12. Tie unused NC pins to BAT pin as explained in Cell Connections . |
| VC12 | 11 | I | Cell voltage sense input. Connect to the positive terminal of cell 12. Connect a differential RC filter to VC11. |
| VC11 | 13 | I | Cell voltage sense input. Connect to the positive terminal of cell 11. Connect a differential RC filter to VC10. |
| VC10 | 15 | I | Cell voltage sense input. Connect to the positive terminal of cell 10. Connect a differential RC filter to VC9. |
| VC9 | 17 | I | Cell voltage sense input. Connect to the positive terminal of cell 9. Connect a differential RC filter to VC8. |
| VC8 | 19 | I | Cell voltage sense input. Connect to the positive terminal of cell 8. Connect a differential RC filter to VC7. |
| VC7 | 21 | I | Cell voltage sense input. Connect to the positive terminal of cell 7. Connect a differential RC filter to VC6. |
| VC6 | 23 | I | Cell voltage sense input. Connect to the positive terminal of cell 6. Connect a differential RC filter to VC5. |
| VC5 | 25 | I | Cell voltage sense input. Connect to the positive terminal of cell 5. Connect a differential RC filter to VC4. |
| VC4 | 27 | I | Cell voltage sense input. Connect to the positive terminal of cell 4. Connect a differential RC filter to VC3. |
| VC3 | 29 | I | Cell voltage sense input. Connect to the positive terminal of cell 3. Connect a differential RC filter to VC2. |
| VC2 | 31 | I | Cell voltage sense input. Connect to the positive terminal of cell 2. Connect a differential RC filter to VC1. |
| VC1 | 33 | I | Cell voltage sense input. Connect to the positive terminal of cell 1. Connect a differential RC filter to VC0. |
| VC0 | 35 | I | Cell voltage sense input. Connect to the negative terminal of cell 1. Connect a differential RC filter to AVSS. |
| CB16 | 2 | I/O | Cell balance connection. This pin is connected to the internal cell balancing FET. Connect this pin to the positive terminal of cell 16 with a differential RC filter to CB15. The filter resistor also sets the internal balance current. Tie unused CB16 pin via RC to BAT pin and tie unused NC pins to BAT pin as explained in Cell Connections . |
| CB15 | 4 | I/O | Cell balance connection. This pin is connected to the internal cell balancing FET. Connect this pin to the positive terminal of cell 15 with a differential RC filter to CB14. The filter resistor also sets the internal balance current. Tie unused NC pins to BAT pin as explained in Cell Connections . |
| CB14 | 6 | I/O | Cell balance connection. This pin is connected to the internal cell balancing FET. Connect this pin to the positive terminal of cell 14 with a differential RC filter to CB13. The filter resistor also sets the internal balance current. Tie unused NC pins to BAT pin as explained in Cell Connections . |
| CB13 | 8 | I/O | Cell balance connection. This pin is connected to the internal cell balancing FET. Connect this pin to the positive terminal of cell 13 with a differential RC filter to CB12. The filter resistor also sets the internal balance current. Tie unused NC pins to BAT pin as explained in Cell Connections . |
| CB12 | 10 | I/O | Cell balance connection. This pin is connected to the internal cell balancing FET. Connect this pin to the positive terminal of cell 12 with a differential RC filter to CB11. The filter resistor also sets the internal balance current. |
| CB11 | 12 | I/O | Cell balance connection. This pin is connected to the internal cell balancing FET. Connect this pin to the positive terminal of cell 11 with a differential RC filter to CB10. The filter resistor also sets the internal balance current. |
| CB10 | 14 | I/O | Cell balance connection. This pin is connected to the internal cell balancing FET. Connect this pin to the positive terminal of cell 10 with a differential RC filter to CB9. The filter resistor also sets the internal balance current. |
| CB9 | 16 | I/O | Cell balance connection. This pin is connected to the internal cell balancing FET. Connect this pin to the positive terminal of cell 9 with a differential RC filter to CB8. The filter resistor also sets the internal balance current. |

Table 4-1. Pin Functions (continued)

| PIN | | TYPE | DESCRIPTION |
|---------|-----|------|---|
| NAME | No. | | |
| BQ79616 | | | |
| CB8 | 18 | I/O | Cell balance connection. This pin is connected to the internal cell balancing FET. Connect this pin to the positive terminal of cell 8 with a differential RC filter to CB7. The filter resistor also sets the internal balance current. |
| CB7 | 20 | I/O | Cell balance connection. This pin is connected to the internal cell balancing FET. Connect this pin to the positive terminal of cell 7 with a differential RC filter to CB6. The filter resistor also sets the internal balance current. |
| CB6 | 22 | I/O | Cell balance connection. This pin is connected to the internal cell balancing FET. Connect this pin to the positive terminal of cell 6 with a differential RC filter to CB5. The filter resistor also sets the internal balance current. |
| CB5 | 24 | I/O | Cell balance connection. This pin is connected to the internal cell balancing FET. Connect this pin to the positive terminal of cell 5 with a differential RC filter to CB4. The filter resistor also sets the internal balance current. |
| CB4 | 26 | I/O | Cell balance connection. This pin is connected to the internal cell balancing FET. Connect this pin to the positive terminal of cell 4 with a differential RC filter to CB3. The filter resistor also sets the internal balance current. |
| CB3 | 28 | I/O | Cell balance connection. This pin is connected to the internal cell balancing FET. Connect this pin to the positive terminal of cell 3 with a differential RC filter to CB2. The filter resistor also sets the internal balance current. |
| CB2 | 30 | I/O | Cell balance connection. This pin is connected to the internal cell balancing FET. Connect this pin to the positive terminal of cell 2 with a differential RC filter to CB1. The filter resistor also sets the internal balance current. |
| CB1 | 32 | I/O | Cell balance connection. This pin is connected to the internal cell balancing FET. Connect this pin to the positive terminal of cell 1 with a differential RC filter to CB0. The filter resistor also sets the internal balance current. |
| CB0 | 34 | I/O | Cell balance connection. This pin is connected to the internal cell balancing FET. Connect to the negative terminal of cell 1 with differential RC filter to AVSS. The filter resistor also sets the internal balance current. |
| BBP | 64 | I | Bus bar connection. With BBP and BBN connecting to each end of a bus bar, this channel provides a differential input to the ADC measurement with a 5x gain. |
| BBN | 63 | I | Bus bar connection. With BBP and BBN connecting to each end of a bus bar, this channel provides a differential input to the ADC measurement with a 5x gain. |
| RX | 52 | I | UART receiver input. Pull up to CVDD with an external resistor and connect the device RX to the TX output of the host MCU. If unused (for example, for stack devices), connect RX to CVDD. |
| TX | 53 | O | UART transmitter output. Connect device TX to RX input of the host MCU and will be pulled up from the host side. If unused (for example, for stack devices), leave it floating. |
| COMHP | 43 | I/O | Vertical bidirectional communication interface for daisy chain connection. High side (north side) differential I/O. Will connect to the low side (south side) COMLP and COMLN of the lower adjacent device in the daisy chain configuration. If unused, connect COMHP and COMHN with a 1-k Ω resistor. |
| COMHN | 42 | I/O | |
| COMLP | 40 | I/O | Vertical bidirectional communication interface for daisy chain connection. Low side (south side) differential I/O. Will connect to the high side (north side) COMHP and COMHN of the upper adjacent device in the daisy chain configuration. If unused, connect COMLP and COMLN with a 1-k Ω resistor. |
| COMLN | 41 | I/O | |
| NFAULT | 62 | O | Fault indication output. Active low. If used on the base device, pull up NFAULT to CVDD with a pullup resistor and connect NFAULT to host MCU GPIO. If unused, leave it unconnected. |
| GPIO1 | 61 | I/O | General purpose input/output, configuration options are: <ul style="list-style-type: none"> • For external NTC thermistor connection, connect NTC thermistor to the pin and pull up to TSREF. Used as input to ADC and OT and UT hardware comparators • For external DC voltage measurement, configured as input to ADC • Generic digital input/output • Use as I/O for SPI controller. |
| GPIO2 | 60 | I/O | |
| GPIO3 | 59 | I/O | |
| GPIO4 | 58 | I/O | |
| GPIO5 | 57 | I/O | |
| GPIO6 | 56 | I/O | |
| GPIO7 | 55 | I/O | |
| GPIO8 | 54 | I/O | |

5 Specifications

5.1 Absolute Maximum Ratings

over operating free-air temperature range (unless otherwise noted)⁽¹⁾

| | | MIN | MAX | UNIT |
|-----------------------|--|------|------|------|
| Input Voltage | BAT, VC* (except VC0), CB* (except CB0), NFAULT, BBP, BBN to AVSS ^{(2) (3)} | -0.3 | 100 | V |
| | CB0, VC0 to AVSS | -0.3 | 5.5 | V |
| | VCn to VCn-1, n = 1 to 16 ⁽²⁾ | -80 | 80 | V |
| | CBn to CBn-1, n = 1 to 16 ⁽³⁾ | -0.3 | 16 | V |
| | BBP to BBN | -80 | 80 | V |
| | LDOIN to AVSS | -0.3 | 9 | V |
| | NPNB to AVSS | -0.3 | 10 | V |
| | AVDD to AVSS | -0.3 | 5.5 | V |
| | DVDD to DVSS | -0.3 | 1.98 | V |
| | CVDD to CVSS | -0.3 | 6 | V |
| | TSREF to AVSS | -0.3 | 5.5 | V |
| | REFHP to REFHM | -0.3 | 5.5 | V |
| | NEG5V to AVSS | -5.5 | 0 | V |
| | TX, RX to AVSS | -0.3 | 6 | V |
| | COMHP, COMHN, COMLP, COMLN to CVSS | -20 | 20 | V |
| | COMHP to COMHN, COMLP to COMLN | -5.5 | 5.5 | V |
| | GPIO* to AVSS | -0.3 | 5.5 | V |
| CB* current | Max of 8 cell in balancing at 75°C ambient | | 240 | mA |
| I/O current | GPIO*, RX, TX current | | 10 | mA |
| T _{OTP_PROG} | Device will not start OTP programming above this temperature | | 55 | °C |
| T _A | Ambient temperature | -40 | 130 | °C |
| T _J | Junction temperature | -40 | 150 | °C |
| T _{stg} | Storage temperature | -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. This disclaimer is new as of 1Q2021. All newly released data sheets m
- (2) VC pin voltage has to meet criteria of both VCn to AVSS as well as VCn to VCn-1.
- (3) CB pin voltage has to meet criteria of both CBn to AVSS as well as CBn to CBn-1.

5.2 ESD Ratings

| | | | VALUE | UNIT | |
|--------------------|-------------------------|---|--|------|------|
| V _(ESD) | Electrostatic discharge | Human body model (HBM) ⁽¹⁾ | ±2000 | V | |
| | | Charged device model (CDM) ⁽²⁾ | All Pins | | ±500 |
| | | | Other pins (1, 16, 17, 32, 33, 48, 49, 64) | | ±750 |

- (1) JEDEC document JEP155 states that 500V HBM allows safe manufacturing with a standard ESD control process.
- (2) JEDEC document JEP157 states that 250V CDM allows safe manufacturing with a standard ESD control process.

5.3 Recommended Operating Conditions

over operating free-air temperature range (unless otherwise noted)

| | | MIN | NOM | MAX | UNIT |
|----------------------------|---|------|-----|------|------|
| V _{BAT_RANGE} | Total module voltage, full functionality, no OTP programming | 9 | | 80 | V |
| V _{BAT_OTP_RANGE} | Total module voltage, full functionality, OTP programming allow | 11 | | 80 | V |
| V _{CELL_RANGE} | VC _n – VC _{n-1} , where n = 2 to 16 | –1 | | 5 | V |
| | VC1 – VC0 | 0 | | 5 | V |
| | VC0, CB0 to AVSS | –0.3 | | 5 | V |
| | VC1, VC2, CB1, CB2 to AVSS | –0.3 | | 80 | V |
| | VC _n , CB _n to AVSS, where n = 3 to 16 | 3 | | 80 | V |
| V _{BB_RANGE} | V _{BBP} – V _{BBN} | –600 | | 800 | mV |
| V _{CB_RANGE} | CB _n – CB _{n-1} , where n = 1 to 16 | 0 | | 5 | V |
| V _{IO_RANGE} | RX, TX, NFAULT | 0 | | CVDD | V |
| V _{GPIO_RANGE} | GPIO _n input, where n = 1 to 8 | 0.2 | | 4.8 | V |
| I _{IO} | GPIO _n , RX, TX, where n = 1 to 8 | | | 5 | mA |
| T _A | Operation temperature | –40 | | 105 | °C |

5.4 Thermal Information

| THERMAL METRIC ⁽¹⁾ | | BQ79616 | | UNIT |
|-------------------------------|--|-------------|--|------|
| | | PAP (HTQFP) | | |
| | | 64 PINS | | |
| R _{θJA} | Junction-to-ambient thermal resistance | 21.6 | | °C/W |
| R _{θJC(top)} | Junction-to-case (top) thermal resistance | 8.7 | | °C/W |
| R _{θJB} | Junction-to-board thermal resistance | 7.9 | | °C/W |
| ψ _{JT} | Junction-to-top characterization parameter | 0.1 | | °C/W |
| ψ _{JB} | Junction-to-board characterization parameter | 7.8 | | °C/W |
| R _{θJC(bot)} | Junction-to-case (bottom) thermal resistance | 2.1 | | °C/W |

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

5.5 Electrical Characteristics

over operating –40°C to 125°C free-air temperature range, VBAT = 9V to 80V (unless otherwise noted)

| PARAMETER | | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|-------------------------|--|-----------------|-----|-----|-----|------|
| THERMAL SHUTDOWN | | | | | | |
| T _{SHUT} | Thermal shutdown (rising direction) | | 130 | 137 | 152 | °C |
| T _{SHUT_FALL} | Thermal shutdown (falling direction) | | 112 | | 129 | °C |
| T _{SHUT_HYS} | Thermal shutdown (rising - falling direction) | | | 20 | | °C |
| T _{WARN_RANGE} | Thermal warning Threshold (rising direction) | | 85 | | 115 | °C |
| T _{WARN_HYS} | Thermal warning hysteresis (falling direction) | | | 10 | | °C |
| T _{WARN_ACC} | Thermal warning accuracy (±) | | | 5 | | °C |

5.5 Electrical Characteristics (continued)

over operating -40°C to 125°C free-air temperature range, VBAT = 9V to 80V (unless otherwise noted)

| PARAMETER | | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|-------------------------|---|---|------|------|------|---------------|
| SUPPLY CURRENTS | | | | | | |
| I_{SHDN} | Supply current in SHUTDOWN mode | Sum of both I_{BAT} and I_{LDOIN} | | 16 | 23 | μA |
| $I_{\text{SLP(IDLE)}}$ | Baseline supply current in SLEEP mode. No fault, no protector comparator, no cell balancing | Sum of both I_{BAT} and I_{LDOIN} $T_{\text{A}} = -20^{\circ}\text{C}$ to 65°C | | 120 | 160 | μA |
| | | Sum of both I_{BAT} and I_{LDOIN} $T_{\text{A}} = -40^{\circ}\text{C}$ to 125°C | | | 220 | μA |
| $I_{\text{ACT(IDLE)}}$ | Baseline supply current in ACTIVE mode | Sum of both I_{BAT} and I_{LDOIN} No fault, no communication, no protector comparator, no cell balancing | | 10.4 | 11.6 | mA |
| $I_{\text{CB_EN}}$ | Additional supply current when cell balancing is on | At least 1 cell balancing FET is on, OT_{CB} is enabled. Other functions are inactive | | 1 | 1.5 | mA |
| I_{PROTCOMP} | Additional supply current when protector comparator is on | Either OV/UV/OT/UT protector is enabled. Other functions are inactive | | 20 | 60 | μA |
| I_{ADC} | Additional supply current when ADC is enabled | One ADC on, and conversion is in progress. Other functions are inactive. | | 0.4 | 0.6 | mA |
| | | 2 ADCs on, and conversion is in progress. Other functions are inactive. | | 0.6 | 0.9 | mA |
| I_{BAT} | Supply current goes into BAT pin | ACTIVE mode | | 150 | | μA |
| | | SLEEP mode | | 25 | | μA |
| | | SHUTDOWN mode | | 5 | | μA |
| I_{COMT} | Additional supply current during daisy-chain broadcast read of 128-byte data | Use transformer isolation for daisy-chain interface. | | 10 | | mA |
| I_{COMC} | Additional supply current during daisy-chain broadcast read of 128-byte data | Use capacitor or capacitor and choke isolation for daisy-chain interface. | | 10 | | mA |
| $I_{\text{OW_SINK}}$ | Sink current for open wire test, applies to VC1 to VC16 and CB1 to CB 16 | | 380 | 500 | 600 | μA |
| $I_{\text{OW_SOURCE}}$ | Source current for open wire test, applies to VC0 and CB0 | | 380 | 500 | 600 | μA |
| I_{LEAK} | Leakage current on VC, CB pins | VC, CB pins with ADC off. | | | 0.1 | μA |
| Supplies (LDOIN) | | | | | | |
| V_{LDOIN} | LDOIN voltage | No OTP programming | 5.9 | 6 | 6.1 | V |
| | | OTP programming | 7.9 | 8 | 8.1 | V |
| Supplies (CVDD) | | | | | | |
| V_{CVDD} | CVDD output voltage | ACTIVE and SLEEP mode | 4.9 | 5 | 5.1 | V |
| | | SHUTDOWN mode, no external load | 3.95 | | 6 | V |
| | | SHUTDOWN mode, max external load = 5mA | 3.4 | | 5.5 | V |
| $V_{\text{CVDD_LDRG}}$ | CVDD load regulation | ACTIVE/SLEEP mode, max external load = 10mA | -30 | | 30 | mV |
| $V_{\text{CVDD_OV}}$ | CVDD OV threshold | ACTIVE/SLEEP mode, max external load = 10mA | 5.3 | 5.5 | 5.7 | V |

5.5 Electrical Characteristics (continued)

over operating -40°C to 125°C free-air temperature range, VBAT = 9V to 80V (unless otherwise noted)

| PARAMETER | | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|-------------------------------------|---|--|-------|------|-------|------|
| V _{CVDD_OVHYS} | CVDD OV Hysteresis | ACTIVE/SLEEP mode, max external Iload = 10mA | 130 | 150 | 170 | mV |
| V _{CVDD_UV} | CVDD UV threshold | SHUTDOWN mode | | 3.5 | | V |
| | | ACTIVE/SLEEP mode, max external Iload = 10mA | 4.3 | 4.45 | 4.65 | V |
| V _{CVDD_UVHYS} | CVDD UV Hysteresis | | | 260 | | mV |
| V _{CVDD_ILIMIT} | CVDD current limit | ACTIVE, SLEEP | 35 | 60 | 85 | mA |
| Supplies (AVDD) | | | | | | |
| V _{AVDD} | AVDD output voltage | C _{SUPPLIES} = 1μF, ACTIVE mode | 4.85 | 5 | 5.21 | V |
| V _{AVDD_OV} | AVDD OV threshold | C _{SUPPLIES} = 1μF, ACTIVE mode | 5.25 | 5.5 | 5.7 | V |
| V _{AVDD_OVHYS} | AVDD OV Hysteresis | C _{SUPPLIES} = 1μF, ACTIVE mode | 135 | 155 | 165 | mV |
| V _{AVDD_UV} | AVDD UV threshold | C _{SUPPLIES} = 1μF, ACTIVE mode | 4.25 | 4.45 | 4.6 | V |
| V _{AVDD_UVHYS} | AVDD UV Hysteresis | C _{SUPPLIES} = 1μF, ACTIVE mode | 235 | 340 | 430 | mV |
| V _{AVDD_ILIMIT} | AVDD current limit | C _{SUPPLIES} = 1μF | 10 | 30 | 50 | mA |
| Supplies (DVDD) | | | | | | |
| V _{DVDD} | DVDD output voltage | C _{SUPPLIES} = 1μF, ACTIVE mode | 1.65 | 1.8 | 1.95 | V |
| V _{DVDD_OV} | DVDD OV threshold | C _{SUPPLIES} = 1μF, ACTIVE mode | 1.95 | 2.1 | 2.3 | V |
| V _{DVDD_OVHYS} | DVDD OV Hysteresis | C _{SUPPLIES} = 1μF, ACTIVE mode | 40 | 65 | 120 | mV |
| V _{DVDD_UV} | DVDD UV threshold | C _{SUPPLIES} = 1μF, ACTIVE mode | 1.623 | 1.65 | 1.71 | V |
| V _{DVDD_UVHYS} | DVDD UV Hysteresis | C _{SUPPLIES} = 1μF, ACTIVE mode | 15 | 50 | 73 | mV |
| V _{DVDD_ILIMIT} | DVDD current limit | | 13 | 30 | 53 | mA |
| Supplies (TSREF) | | | | | | |
| V _{TSREF} | TSREF output voltage | C _{SUPPLIES} = 1μF, ACTIVE mode | 4.975 | 5 | 5.025 | V |
| V _{TSREF_LDRG} | TSREF load regulation | I _{load} = 4mA, C _{SUPPLIES} = 1μF, ACTIVE mode | -30 | | 30 | mV |
| V _{TSREF_OV} | TSREF OV threshold | I _{load} = 4mA, C _{SUPPLIES} = 1μF, ACTIVE mode | 5.2 | 5.6 | 5.8 | V |
| V _{TSREF_OVHYS} | TSREF OV Hysteresis | I _{load} = 4mA, C _{SUPPLIES} = 1μF, ACTIVE mode | 98 | 110 | 120 | mV |
| V _{TSREF_UV} | TSREF UV threshold | I _{load} = 4mA, C _{SUPPLIES} = 1μF, ACTIVE mode | 4.0 | 4.2 | 4.4 | V |
| V _{TSREF_UVHYS} | TSREF UV Hysteresis | I _{load} = 4mA, C _{SUPPLIES} = 1μF, ACTIVE mode | 300 | 350 | 400 | mV |
| V _{TSREF_ILIMIT} | TSREF current limit | Device in ACTIVE Mode | 15 | 30 | 52 | mA |
| Negative Charge Pump (NEG5V) | | | | | | |
| V _{NEG5V} | NEG5V pin voltage | C _{NEG5V} = 0.1μF | -5.3 | -4.6 | -4.0 | V |
| V _{NEG5V_UV} | NEG5V UV threshold (rising) | C _{NEG5V} = 0.1μF | -4.1 | -3.5 | -3.0 | V |
| V _{NEG5V_UVRECOV} | NEG5V UV Recovery | C _{NEG5V} = 0.1μF | -4.3 | -3.8 | -3.3 | V |
| CELL BALANCE | | | | | | |
| R _{DSON} | Internal cell balance FET R _{dson} | VC _n > 2.8V, where n = 1 to 16; $-40^{\circ}\text{C} < T_A < 125^{\circ}\text{C}$ | 1.45 | | 4.6 | Ω |
| V _{CB_DONE} | VCB_DONE detection threshold setting range (not accuracy) | Step of 25mV | 2.45 | | 4 | V |
| V _{MB_DONE} | VMB_DONE detection threshold setting range (not accuracy) | Step of 1V | 18 | | 65 | V |
| T _{OTCB} | OTCB threshold setting range (not accuracy) | Step of 2% | 10 | | 24 | % |

5.5 Electrical Characteristics (continued)

over operating -40°C to 125°C free-air temperature range, VBAT = 9V to 80V (unless otherwise noted)

| PARAMETER | | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|----------------------------------|---|--|------|--------|-----|-------------------------------|
| T_{COOLOFF} | COOLOFF threshold setting range (not accuracy) | Step of 2% | 4 | | 14 | % |
| $T_{\text{CB_WARN}}$ | CB TWARN threshold | | | 105 | | $^{\circ}\text{C}$ |
| $T_{\text{CB_WARN_HYS}}$ | CB TWARN hysteresis | | | 10 | | $^{\circ}\text{C}$ |
| ADC Resolution | | | | | | |
| $\text{ENOB}_{\text{MAIN}}$ | Main ADC Effective number of bits | | | 16 | | bits |
| ENOB_{AUX} | AUX ADC Effective number of bits | | | 14 | | bits |
| $V_{\text{LSB_ADC}}$ | Main and AUX ADC Resolution for VCELL measurement | | | 190.73 | | $\mu\text{V}/\text{LSB}$ |
| $V_{\text{LSB_BB}}$ | Main and AUX ADC Resolution for (BBP-BBN) measurement | | | 30.52 | | $\mu\text{V}/\text{LSB}$ |
| $V_{\text{LSB_MAIN_DIETEMP1}}$ | DieTemp1 resolution (Main ADC) | ADC measurement is centered with 0x000 = 0°C | | 0.025 | | $^{\circ}\text{C}/\text{LSB}$ |
| $V_{\text{LSB_AUX_DIETEMP2}}$ | DieTemp2 resolution (AUX ADC) | ADC measurement is centered with 0x000 = 0°C | | 0.025 | | $^{\circ}\text{C}/\text{LSB}$ |
| $V_{\text{LSB_AUX_BAT}}$ | BAT resolution (AUX ADC) | Applies to BAT voltage measurement from AUX ADC | | 3.05 | | mV/LSB |
| $V_{\text{LSB_GPIO}}$ | GPIO resolution (Main and AUX ADC) | | | 152.59 | | $\mu\text{V}/\text{LSB}$ |
| $V_{\text{LSB_TSREF}}$ | TSREF resolution (Main ADC) | | | 169.54 | | $\mu\text{V}/\text{LSB}$ |
| $V_{\text{LSB_DIAG}}$ | Diagnostic measurements resolution | REFL, VBG2, LPBG5, VCM, AVAO_REF, AVDD_REF, all the HW protector DAC | | 152.59 | | $\mu\text{V}/\text{LSB}$ |
| ADC Accuracy | | | | | | |
| $I_{\text{VC_DELTA}}$ | VCn to VCn-1 input current delta (when Main ADC is on) | $T_A = -20^{\circ}\text{C}$ to 65°C | | | 1.8 | μA |
| | | $T_A = -40^{\circ}\text{C}$ to 105°C | | | 2 | μA |
| I_{VC} | VCn input current (when Main ADC is on) | | | 8 | 12 | μA |
| $R_{\text{CB_INPUT}}$ | CB pin input impedance (when AUX ADC is on) | | | 16 | | $\text{M}\Omega$ |
| $V_{\text{ACC_MAIN_CELL}}$ | Total channel accuracy for main ADC VCELL measurement, LPF_VCELL[2:0] = 0x03 setting; | $2\text{V} < V_{\text{CELL}} < 4.5\text{V}; T_A = 25^{\circ}\text{C}$ | -2.2 | | 1.5 | mV |
| | | $2\text{V} < V_{\text{CELL}} < 4.5\text{V}; -20^{\circ}\text{C} < T_A < 65^{\circ}\text{C}$ | -3.0 | | 2.4 | mV |
| | | $2\text{V} < V_{\text{CELL}} < 4.5\text{V}; -40^{\circ}\text{C} < T_A < 105^{\circ}\text{C}$ | -3.5 | | 2.6 | mV |
| | | $2\text{V} < V_{\text{CELL}} < 4.5\text{V}; -40^{\circ}\text{C} < T_A < 125^{\circ}\text{C}$ | -3.5 | | 2.6 | mV |
| | | $1\text{V} < V_{\text{CELL}} < 5\text{V}; -40^{\circ}\text{C} < T_A < 125^{\circ}\text{C}$ | -3.7 | | 2.8 | mV |
| | | $-2\text{V} < V_{\text{CELL}} < 5\text{V}; -40^{\circ}\text{C} < T_A < 125^{\circ}\text{C}$ | -4.5 | | 3.2 | mV |
| $V_{\text{ACC_AUX_CELL}}$ | Total channel accuracy for AUX ADC measurement (excluding BAT and GPIO accuracy); | $2\text{V} < V_{\text{CELL}} < 4.5\text{V}; T_A = 25^{\circ}\text{C}$ | -7.5 | | 5.4 | mV |
| | | $2\text{V} < V_{\text{CELL}} < 4.5\text{V}; -20^{\circ}\text{C} < T_A < 65^{\circ}\text{C}$ | -8.0 | | 6.3 | mV |
| | | $2\text{V} < V_{\text{CELL}} < 4.5\text{V}; -40^{\circ}\text{C} < T_A < 105^{\circ}\text{C}$ | -9.0 | | 6.3 | mV |
| | | $2\text{V} < V_{\text{CELL}} < 4.5\text{V}; -40^{\circ}\text{C} < T_A < 125^{\circ}\text{C}$ | -9.0 | | 6.5 | mV |
| | | $1\text{V} < V_{\text{CELL}} < 5\text{V}; -40^{\circ}\text{C} < T_A < 125^{\circ}\text{C}$ | -9.0 | | 6.6 | mV |
| | | $0\text{V} < V_{\text{CELL}} < 5\text{V}; -40^{\circ}\text{C} < T_A < 125^{\circ}\text{C}$ | -9.0 | | 6.6 | mV |

5.5 Electrical Characteristics (continued)

 over operating -40°C to 125°C free-air temperature range, VBAT = 9V to 80V (unless otherwise noted)

| PARAMETER | | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|-------------------------------------|---|---|-------|------|-------|------|
| $V_{(\text{MAIN-AUX})}$ | Main - AUX measurement during VCELL and OVDAC Reference diagnostic. Same input voltage to both ADC under same T_A | $2\text{V} < V_{\text{CELL}} < 4.5\text{V}$; $T_A = 25^{\circ}\text{C}$ | -7.1 | | 6.1 | mV |
| | | $2\text{V} < V_{\text{CELL}} < 4.5\text{V}$; $-20^{\circ}\text{C} < T_A < 65^{\circ}\text{C}$ | -7.8 | | 6.6 | mV |
| | | $2\text{V} < V_{\text{CELL}} < 4.5\text{V}$; $-40^{\circ}\text{C} < T_A < 105^{\circ}\text{C}$ | -7.8 | | 6.6 | mV |
| | | $2\text{V} < V_{\text{CELL}} < 4.5\text{V}$; $-40^{\circ}\text{C} < T_A < 125^{\circ}\text{C}$ | -7.8 | | 6.7 | mV |
| | | $1\text{V} < V_{\text{CELL}} < 5\text{V}$; $-40^{\circ}\text{C} < T_A < 125^{\circ}\text{C}$ | -7.9 | | 6.9 | mV |
| | | $0\text{V} < V_{\text{CELL}} < 5\text{V}$; $-40^{\circ}\text{C} < T_A < 125^{\circ}\text{C}$ | -7.9 | | 6.9 | mV |
| $V_{\text{ACC_MAIN_GPIO_RATIO}}$ | Measured GPIO from Main ADC/ measured TSREF from Main ADC | $0.08\text{V} < V_{\text{IN}} < 0.2\text{V}$, $85^{\circ}\text{C} < T_A < 125^{\circ}\text{C}$ | -0.20 | | 0.20 | % |
| | | $0.2\text{V} < V_{\text{IN}} < 4.6\text{V}$, $-40^{\circ}\text{C} < T_A < 105^{\circ}\text{C}$ | -0.20 | | 0.20 | % |
| | | $4.6\text{V} < V_{\text{IN}} < 4.8\text{V}$, $-40^{\circ}\text{C} < T_A < -20^{\circ}\text{C}$ | -0.30 | | 0.30 | % |
| $V_{\text{ACC_AUX_GPIO_RATIO}}$ | Measured GPIO from AUX ADC/ measured TSREF from AUX ADC | $0.08\text{V} < V_{\text{IN}} < 0.2\text{V}$, $85^{\circ}\text{C} < T_A < 125^{\circ}\text{C}$ | -0.20 | | 0.20 | % |
| | | $0.2\text{V} < V_{\text{IN}} < 4.6\text{V}$, $-40^{\circ}\text{C} < T_A < 105^{\circ}\text{C}$ | -0.20 | | 0.20 | % |
| | | $4.6\text{V} < V_{\text{IN}} < 4.8\text{V}$, $-40^{\circ}\text{C} < T_A < -20^{\circ}\text{C}$ | -0.30 | | 0.30 | % |
| $V_{\text{ACC_MAIN_GPIO_ABS}}$ | Total channel accuracy for GPIO measurement (Main ADC) | $0.08\text{V} < V_{\text{IN}} < 0.2\text{V}$, $85^{\circ}\text{C} < T_A < 125^{\circ}\text{C}$ | -4.00 | | 4.00 | mV |
| | | $0.2\text{V} < V_{\text{IN}} < 4.6\text{V}$, $-40^{\circ}\text{C} < T_A < 105^{\circ}\text{C}$ | -5.00 | | 3.00 | mV |
| | | $4.6\text{V} < V_{\text{IN}} < 4.8\text{V}$, $-40^{\circ}\text{C} < T_A < -20^{\circ}\text{C}$ | -4.00 | | 4.00 | mV |
| $V_{\text{ACC_AUX_GPIO_ABS}}$ | Accuracy from AUX ADC on GPIO | $0.08\text{V} < V_{\text{IN}} < 0.2\text{V}$, $85^{\circ}\text{C} < T_A < 125^{\circ}\text{C}$ | -6.00 | | 6.00 | mV |
| | | $0.2\text{V} < V_{\text{IN}} < 4.6\text{V}$, $-40^{\circ}\text{C} < T_A < 105^{\circ}\text{C}$ | -6.00 | | 6.00 | mV |
| | | $4.6\text{V} < V_{\text{IN}} < 4.8\text{V}$, $-40^{\circ}\text{C} < T_A < -20^{\circ}\text{C}$ | -6.00 | | 6.00 | mV |
| $V_{\text{ACC_MAIN_BB}}$ | Total channel accuracy for (BBP- BBN) from Main ADC | LPF_BB[2:0] = 0x00 | -1.1 | | 1.1 | mV |
| $V_{\text{ACC_AUX_BB}}$ | Total channel accuracy for (BBP- BBN) from AUXADC | | -4 | | 4 | mV |
| $V_{\text{ACC_AUX_BAT}}$ | AUX ADC measurement accuracy for BAT pin | Vbat pack range: 32V to 72V, $T_A = -40^{\circ}\text{C}$ to 125°C | -225 | | 135 | mV |
| $V_{\text{ACC_AUX_REFL}}$ | AUX ADC measurement result | | 1.092 | 1.1 | 1.106 | V |
| $V_{\text{ACC_AUX_VBG2}}$ | AUX ADC measurement result | | 1.092 | 1.1 | 1.106 | V |
| $V_{\text{ACC_AUX_VCM}}$ | AUX ADC measurement result | | 2.400 | 2.5 | 2.550 | V |
| $V_{\text{ACC_AUX_AVAO_REF}}$ | AUX ADC measurement result | | 2.400 | 2.47 | 2.550 | V |
| $V_{\text{ACC_AUX_AVDD_REF}}$ | AUX ADC measurement result | | 2.400 | 2.47 | 2.550 | V |
| $V_{\text{ACC_AUX_OVDAC}}$ | AUX ADC measurement result | Setting at 4.475V; $T_A = -20^{\circ}\text{C}$ to 65°C | 4.450 | | 4.500 | V |
| $V_{\text{ACC_AUX_OVDAC}}$ | AUX ADC measurement result | Setting at 4.475V; $T_A = -40^{\circ}\text{C}$ to 105°C | 4.445 | | 4.500 | V |
| $V_{\text{ACC_AUX_OVDAC}}$ | AUX ADC measurement result | Setting at 4.475V; $T_A = -40^{\circ}\text{C}$ to 125°C | 4.445 | | 4.500 | V |
| $V_{\text{ACC_AUX_OVDAC}}$ | AUX ADC measurement result | Setting at 3.8V | 3.770 | | 3.825 | V |
| $V_{\text{ACC_AUX_OVDAC}}$ | AUX ADC measurement result | Setting at 3V | 2.970 | | 3.030 | V |
| $V_{\text{ACC_AUX_UVDAC}}$ | AUX ADC measurement result | Setting at 3.1V | 3.095 | 3.1 | 3.150 | V |
| $V_{\text{ACC_AUX_VCBDONEDAC}}$ | AUX ADC measurement result | Setting at 4V | 3.950 | 4 | 4.050 | V |

5.5 Electrical Characteristics (continued)

over operating -40°C to 125°C free-air temperature range, VBAT = 9V to 80V (unless otherwise noted)

| PARAMETER | | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|--|--|---|--------------------------|------|--------------------------|--------------------|
| V _{ACC_AUX_OTDAC} | AUX ADC measurement result | Setting at 39% | 1.900 | 1.95 | 2.000 | V |
| V _{ACC_AUX_UTDAC} | AUX ADC measurement result | Setting at 80% | 3.950 | 4 | 4.050 | V |
| V _{ACC_MAIN_TSREF} | Main ADC measurement result | | 4.975 | 5 | 5.025 | V |
| V _{ACC_MAIN_DIETEMP} | Total channel accuracy for Die Temp1 measurement (\pm) | | | 3 | | $^{\circ}\text{C}$ |
| V _{ACC_AUX_DIETEMP} | Total channel accuracy for Die Temp2 measurement (\pm) | | | 6 | | $^{\circ}\text{C}$ |
| Reference Voltages | | | | | | |
| V _{REFH} | REFHP to REFHM voltage | | 4.975 | 5 | 5.025 | V |
| HW Voltage Comparator/Protector (CELL OV/UV) | | | | | | |
| V _{OV_COMP_RANGE} | OV comparator detection threshold setting range (not accuracy) | Step of 25mV | 2700 | | 3000 | mV |
| | | Step of 25mV | 3600 | | 3800 | mV |
| | | Step of 25mV | 4175 | | 4500 | mV |
| V _{OV_COMP_HYS} | OV comparator hysteresis after detection | | | 50 | | mV |
| V _{OV_COMP_ACC} | OV comparator accuracy | T _A = -20°C to 65°C | -24 | | 24 | mV |
| | | T _A = -40°C to 105°C | -28 | | 28 | mV |
| V _{UV_COMP_RANGE} | UV comparator detection threshold setting range (not accuracy) | Step of 50mV | 1200 | | 3100 | mV |
| V _{UV_COMP_HYS} | UV comparator hysteresis after detection | | | 50 | | mV |
| V _{UV_COMP_ACC} | UV comparator accuracy | T _A = -20°C to 65°C | -35 | | 35 | mV |
| | | T _A = -40°C to 105°C | -50 | | 50 | mV |
| HW Temperature Comparator/Protector (NTC OT/UT) | | | | | | |
| V _{OT_COMP_RANGE} | OT comparator detection threshold setting range (not accuracy) | Step of 1%, ratiometric with respect to TSREF | 10 | | 39 | % |
| V _{OT_COMP_HYS} | OT comparator hysteresis after detection | | | 2 | | % |
| V _{OT_COMP_ACC} | OT comparator accuracy | | -0.5 | | 0.5 | % |
| V _{UT_COMP_RANGE} | UT comparator detection threshold range | Step of 2%, ratiometric with respect to TSREF | 66 | | 80 | % |
| V _{UT_COMP_HYS} | UT comparator hysteresis after detection | | | 2 | | % |
| V _{UT_COMP_ACC} | UT comparator accuracy | | -0.5 | | 0.5 | % |
| Digital I/Os (TX, RX, GPIO, SPI controller) | | | | | | |
| V _{OH} | Output as logic level high (TX, GPIO as output) | GPIO is configured as output. I _{OUT} = 1mA | V _{CVDD} - 0.3 | | | V |
| V _{OL} | Output as logic level low (TX, NFAULT, GPIO as output) | GPIO is configured as output. I _{OUT} = 1mA | | | 0.3 | V |
| V _{IH} | Input as logic level high (RX, GPIO as fault input) | GPIO is configured as input. I _{OUT} = 1mA | 0.75 × V _{CVDD} | | | V |
| V _{IL} | Input as logic level low (RX, GPIO as fault input) | GPIO is configured as input. I _{OUT} = 1mA | | | 0.25 × V _{CVDD} | V |
| R _{WK_PU} | GPIO weak pullup resistance | | 20 | 37 | 60 | K Ω |
| R _{WK_PD} | GPIO weak pulldown resistance | | 20 | 40 | 60 | K Ω |
| COML and COMH | | | | | | |

5.5 Electrical Characteristics (continued)

over operating -40°C to 125°C free-air temperature range, VBAT = 9V to 80V (unless otherwise noted)

| PARAMETER | | TEST CONDITIONS | MIN | TYP | MAX | UNIT |
|-------------------------|--|-----------------|------|-----|------|------------|
| R _{DCTX} | Transmitter output impedance (COML and COMH) | | | 18 | | Ω |
| R _{DCCM} | Common mode impedance (COML and COMH) | | | 45 | | k Ω |
| V _{DCCM} | Common mode voltage (COML and COMH) | | 2.21 | 2.5 | 2.76 | V |
| V _{COMM_DATA1} | Receiver threshold range (V _{COMP} – V _{COML}) form communication | CODE:0 | 0.4 | | 1.2 | V |
| V _{COMM_TONE1} | Receiver threshold range (V _{COMP} – V _{COML}) form Tone | CODE:0 | 0.4 | | 1.2 | V |

5.6 Timing Requirements

over operating -40°C to 125°C free-air temperature range, VBAT = 9V to 80V (unless otherwise noted)

| PARAMETER | | TEST CONDITIONS | MIN | NOM | MAX | UNIT |
|----------------------------|--|--|-----|-----|-----|---------------|
| POWER STATE TIMING | | | | | | |
| t _{SU(WAKE_SHUT)} | Startup from SHUTDOWN to ACTIVE mode | Base device: From the end of WAKE ping to the start of a forwarding WAKE tone | | 6 | 10 | ms |
| | | Stack device: From the end of a received WAKE tone to the start of a forwarding WAKE tone | | 6 | 10 | ms |
| t _{SU(SLP2ACT)} | Startup from SLEEP to ACTIVE mode (with SLEEP2ACTIVE ping/tone) | Base device: From the end of SLEEP2ACTIVE ping to the start of the forwarding SLEEP2ACTIVE tone | | | 230 | μs |
| | | Stack device: From the end of SLEEP2ACTIVE tone to the start of the forwarding SLEEP2ACTIVE tone | | | 230 | μs |
| t _{SU(WAKE_SLP)} | Startup from SLEEP to ACTIVE mode (with WAKE ping/tone) | Base device: From the end of WAKE ping to the start of a forwarding WAKE tone | | | 1 | ms |
| | | Stack device: From the end of a received WAKE tone to the start of a forwarding WAKE tone | | | 1 | ms |
| t _{SLP} | From ACTIVE to SLEEP mode | From receiving SLEEP entry condition to enter in SLEEP mode | | | 100 | μs |
| t _{SHTDN} | From ACTIVE to SHUTDOWN mode | From receiving SHUTDOWN entry condition to enter in SHUTDOWN mode (all LDOs in 10% of their nominal value) | | 20 | | ms |
| t _{RST} | Reset time during ACTIVE mode | CONTROL1[SOFT_RST] = 1 is sent to a completion of the digital reset | | | 1 | ms |
| t _{HWRST} | The time device will be in HW reset, after HW reset ping/tone issued | | | | 75 | ms |
| SUPPLIES TIMING | | | | | | |
| t _{TSREF_ON} | TSREF ramp up time (10%–90%) | C _{TSREF} = 1 μF | 6 | | | ms |
| t _{TSREF_OFF} | TSREF ramp down time (90%–10%) | C _{TSREF} = 1 μF | | | 8 | ms |
| PING SIGNAL TIMING | | | | | | |
| t _{HLD_WAKE} | WAKE ping low time on RX pin; no external load on CVDD | | 2 | | 2.5 | ms |
| t _{HLD_SD} | SHUTDOWN ping low time on RX pin; no external load on CVDD | | 7 | | 10 | ms |

5.6 Timing Requirements (continued)

over operating -40°C to 125°C free-air temperature range, VBAT = 9V to 80V (unless otherwise noted)

| PARAMETER | | TEST CONDITIONS | MIN | NOM | MAX | UNIT |
|--|--|-----------------|------|------|------|--------|
| t _{UART(SIA)} | SLEEPtoACTIVE ping low time on RX pin | | 250 | | 300 | μs |
| t _{HLD_HWRST} | HW_RESET ping low time on RX pin | | 36 | | | ms |
| COML and COMH (PULSE and TONE TIMING) | | | | | | |
| t _{PW_DC} | COMM: Pulse width of data (half bit time) for communication | | | 250 | | ns |
| t _{RECLK_DC} | COMM: data reclocking delay per device from COMH to COML or vice versa | | | 4 | 5 | μs |
| t _{COMTONE} | Time between pulses of comm tones (HFO based). Comm Tones are WAKE, SLEEPtoACTIVE, SHUTDOWN, HWRST tones | | | 11 | 15 | μs |
| t _{COMMTONE_HI} | The HIGH time of each comms pulse (HFO base) | | 0.92 | 1 | 1.08 | μs |
| t _{COMMTONE_LO} | The LOW time of each comms pulse (HFO base) | | 0.92 | 1 | 1.08 | μs |
| t _{FLTTONE} | Time between pulses of FAULT Tone (LFO based). Applies to FAULT Tone and HEARTBEAT | | | 11.5 | | μs |
| t _{FLTTONE_HI} | The HIGH time of each pulse of the tone couplet | | | 1 | | μs |
| t _{FLTTONE_LO} | The LOW time of each pulse of the tone couplet | | | 1 | | μs |
| n _{WAKEDET} | Number of pulses to detect as a WAKE tone | | | 60 | | pulses |
| n _{WAKE} | Number of pulses to transit for a WAKE tone | | | 90 | | pulses |
| n _{SHDNDET} | Number of pulses to detect as a SHUTDOWN tone | | | 180 | | pulses |
| n _{SHDN} | Number of pulses to transit for a SHUTDOWN tone | | | 270 | | pulses |
| n _{SLEPtoACTDET} | Number of pulses to detect as a SLEEPtoACTIVE tone | | | 20 | | pulses |
| n _{SLEPtoACT} | Number of pulses to transit for a SLEEPtoACTIVE tone | | | 30 | | pulses |
| n _{HWRSTDET} | Number of pulses to detect as a HW_RESET tone | | | 540 | | pulses |
| n _{HWRST} | Number of pulses to transit for a HW_RESET tone | | | 810 | | pulses |
| n _{HBDET} | HEARTBEAT: Number of pulses to detect as a valid tone | | | 20 | | pulses |
| n _{HB} | HEARTBEAT: Number of pulses to transit for a tone | | | 30 | | pulses |
| t _{HB_PERIOD} | HEARTBEAT: Period between HEARTBEAT burst (from the beginning of a HEARTBEAT to the beginning of the next HEARTBEAT) | | 360 | 400 | 440 | ms |
| t _{HB_TIMEOUT} | HEARTBEAT: Timeout to considered as not receiving HEARTBEAT | | 0.9 | 1 | 1.1 | s |

5.6 Timing Requirements (continued)

over operating -40°C to 125°C free-air temperature range, VBAT = 9V to 80V (unless otherwise noted)

| PARAMETER | | TEST CONDITIONS | MIN | NOM | MAX | UNIT |
|--|--|---|------|------|------|--------|
| t _{HB_FAST} | HEARTBEAT: If HEARTBEAT is received within this time, it is considered receiving HEARTBEAT too fast | | | 200 | | ms |
| n _{FTONEDET} | FAULT TONE: Number of pulses to detect as a valid tone | | | 60 | | pulses |
| n _{FTONE} | FAULT TONE: Number of pulses to transit for a tone | | | 90 | | pulses |
| t _{FTONE_PERIOD} | FAULT TONE: Period between FAULT TONE Burst (from the beginning of a FAULT TONE to the beginning of the next FAULT TONE) | | | 50 | | ms |
| t _{F_{TS}_LATENCY} | Fault tone latency in stack device | From time a device receive the tone to the time the same device detects and generate its fault tone | | 48 | | µs |
| t _{F_{TB}_LATENCY} | Fault tone latency in base device | From the time a device receive the tone to the time the same device detects and asserts NFAULT | | 24 | | µs |
| MAIN and AUX ADC TIMING | | | | | | |
| t _{SAR_CONV} | Single conversion time (both Main and AUX ADCs) | | | 8 | | µs |
| t _{MAIN_ADC_CYCLE} | Single round robin cycle (Main ADC) | | | 192 | | µs |
| t _{AUX_ADC_CYCLE} | Single round robin cycle (AUX ADC) | | | 192 | | µs |
| t _{A_{FE}_SETTLE} | Analog front end (level shifters) settling time whenever device enter ACTIVE mode from SLEEP or SHUTDOWN | | | 4 | | ms |
| t _{ADC_ACC} | This includes mux round robin, ADC conversions, and digital filters. | | -1.5 | | 1.5 | % |
| BALANCING TIMING | | | | | | |
| t _{BAL_ACC} | Balancing timer accuracy | | -5 | | 5 | % |
| HW COMPARATORS/PROTECTORS TIMING | | | | | | |
| t _{OV_CYCLE} | OV round robin cycle | | | 8 | | ms |
| t _{UV_CYCLE} | UV round robin cycle | | | 8 | | ms |
| t _{OVUV_BIST_CYCLE} | OV and UV BIST cycle | | 21.8 | 23 | 24.2 | ms |
| t _{OT_CYCLE} | OT round robin cycle | | | 4 | | ms |
| t _{UT_CYCLE} | UT round robin cycle | | | 4 | | ms |
| t _{PWR_BIST_CYCLE} | Time needed for the power supply BIST to complete after the power BIST go command | | 10.9 | 11.5 | 12.1 | ms |
| t _{OTUT_BIST_CYCLE} | OT and UT BIST cycle | | 19 | 20 | 21 | ms |
| t _{HW_COMP_ACC} | OV,UV,OT,UT comparators timing accuracy | | -5 | | 5 | % |
| I/O TIMING (TX, RX, GPIO, NFAULT) | | | | | | |
| t _{RISE} | Rise Time | V _{CVDD} > MIN V _{CVDD} , C _{LOAD} = 150pF, GPIO in output mode | | 12 | | ns |
| t _{FALL} | Fall Time (exclude NFAULT) | V _{CVDD} > MIN V _{CVDD} , C _{LOAD} = 150pF, GPIO in output mode | | 7 | | ns |
| t _{FALL_NFAULT} | Fall Time on NFAULT | V _{CVDD} > MIN V _{CVDD} , C _{LOAD} = 150pF, R _{PULLUP} = 10kΩ | | 100 | | ns |

5.6 Timing Requirements (continued)

over operating -40°C to 125°C free-air temperature range, VBAT = 9V to 80V (unless otherwise noted)

| PARAMETER | | TEST CONDITIONS | MIN | NOM | MAX | UNIT |
|--------------------------------------|--|------------------------------------|-------|-----|-------|------------|
| UART TIMING | | | | | | |
| UART _{BAUD} | UART TX/RX Baud Rate | | | 1 | | Mbps |
| UART _{ERR_BAUD(RX)} | UART RX baud rate error - requirement on the external host | | -1 | | 1 | % |
| UART _{ERR_BAUD(TX)} | UART TX baud rate error | | -1.5 | | 1.5 | % |
| t _{UART(CLR)} | UART Clear low time | | 15 | | 20 | bit period |
| t _{UART(RX_HIGH)} | After COMM CLEAR, wait this time before sending new frame | | 1 | | | bit period |
| OTP NVM TIMING | | | | | | |
| t _{CRC_CUST} | Time to complete a single cycle of CRC check on the customer OTP space | | | 175 | | μs |
| t _{CRC_FACT} | Time to complete a single cycle of CRC check on the factory OTP space | | | 1.6 | | ms |
| SPI CONTROLLER TIMING | | | | | | |
| f _{SCLK} | SCLK frequency | | 450 | 500 | 550 | kHz |
| t _{HIGH} , t _{LOW} | SCLK duty cycle | | | 50 | | % |
| t _{SS(HIGH)} | SS HIGH latency time. Time from register write high to SS pin high | | | 4 | | μs |
| t _{SS(LOW)} | SS LOW latency time. Time from register write low to SS pin low | | | 4 | | μs |
| t _{SU(MISO)} | MISO input data setup time-requirement for target device | MISO stable before SCLK transition | 100 | | | ns |
| t _{HD(MISO)} | MISO input data hold time | MISO stable after SCLK transition | | 0 | | ns |
| OSCILLATOR | | | | | | |
| f _{HFO} | High frequency oscillator | | 31.52 | 32 | 32.48 | MHz |
| f _{LFO} | Low frequency oscillator | | 248.9 | 262 | 275.1 | kHz |

5.7 Typical Characteristics

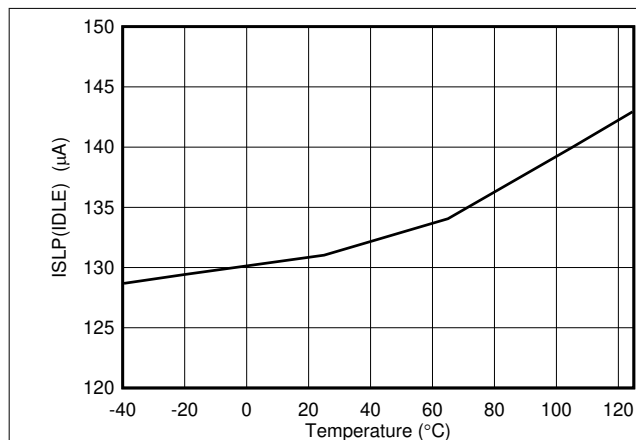


Figure 5-1. ISLP (IDLE) vs Temperature

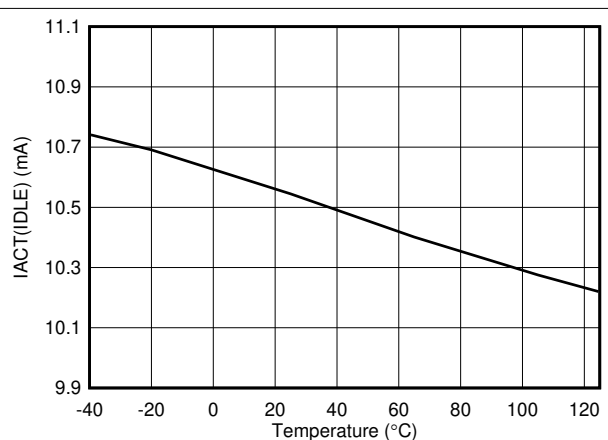


Figure 5-2. IACT (IDLE) vs Temperature

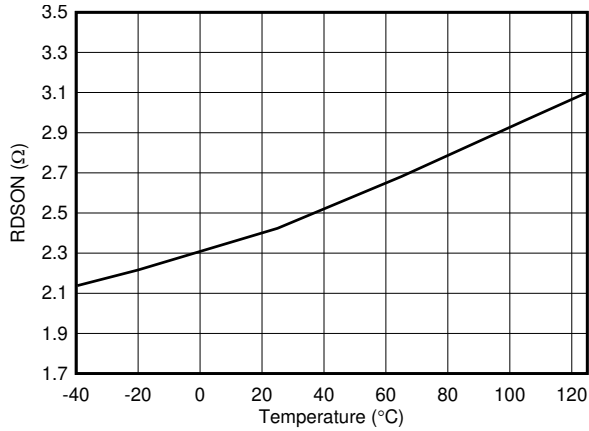


Figure 5-3. CBFET RDSON vs Temperature

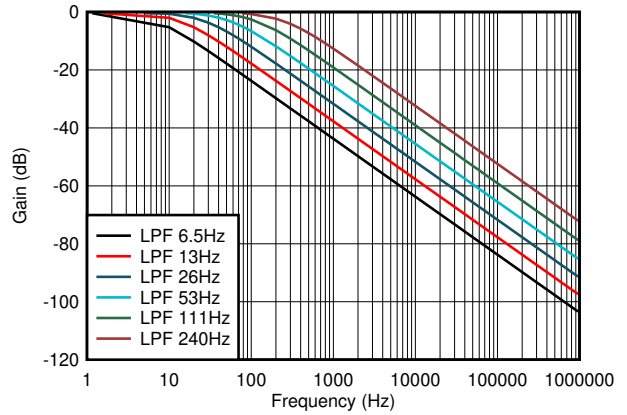


Figure 5-4. Digital Low Pass Filter

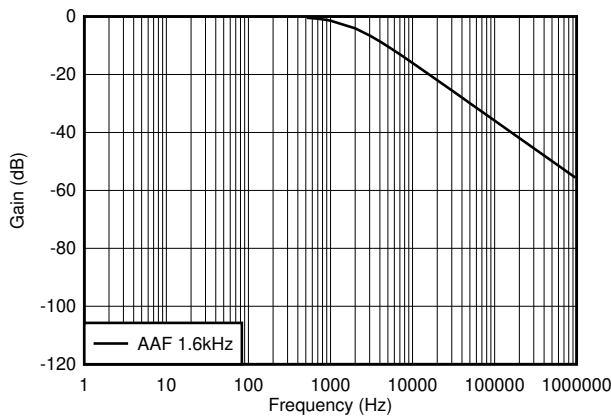


Figure 5-5. MAIN ADC Filter

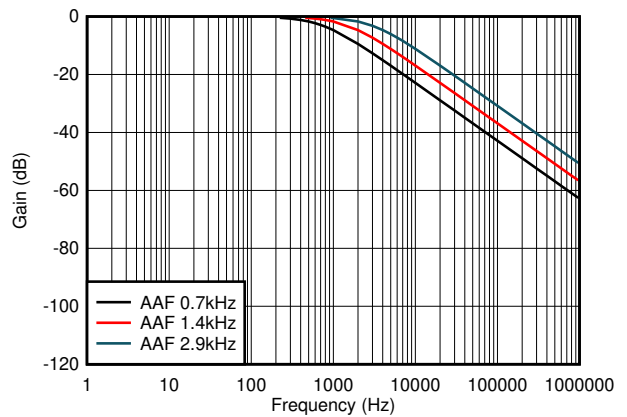


Figure 5-6. AUX ADC Filter

6 Detailed Description

6.1 Overview

The BQ79616 device is a stackable battery monitor that measures cell voltages and temperature. The device supports 6-series to 16-series connected (6S to 16S) battery cells. It allows up to three bus bar connections and measurements using cell sensing input channels or a dedicated bus bar channel maximizing the device flexibility to support various battery module sizes.

Multiple devices can be connected in a daisy chain. Each device has a pair of high (north) and low (south) vertical differential communication ports, requiring only one twisted pair cable. The device supports either capacitive only, capacitive and choke, or transformer isolation. Communication is relocked on each daisy-chained device, ensuring communication integrity for long distances. An optional RING connection is supported to reverse the daisy chain communication direction in case of cable failure. Each device includes a SPI controller configured through the GPIOs.

The ADCs in the daisy-chained devices can be configured to align the start of cell voltage measurements and all cell voltages can be measured within 128 μ s. Each cell sensing channel includes with a post-ADC digital low-pass filter (LPF) for noise reduction, as well as provides moving average measurement results. The device has eight GPIOs, all of which are configurable for NTC thermistor connections or use as general purpose I/O. All eight GPIOs can be measured within 1.6 ms.

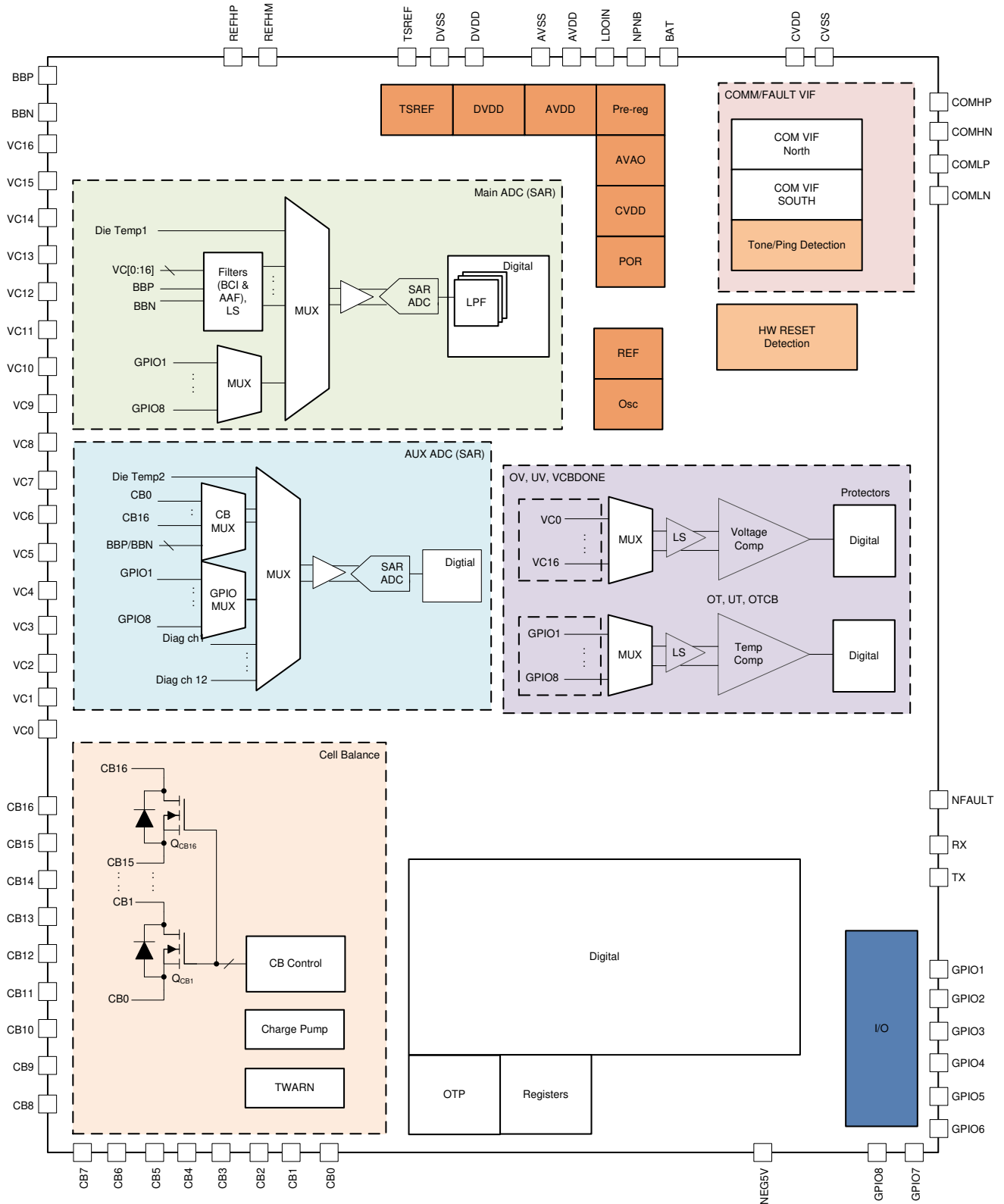
The device supports passive balancing through an internal cell balancing MOSFET (CBFET) for each cell. The balancing function runs autonomously without microcontroller (MCU) interaction. The BQ79616 includes an option to pause and then resume balancing based on a programmable threshold detected by the external thermistor or if the die temperature is too high (greater than 105°C). Once balancing starts, the device tracks the balancing time on each cell. MCU can read out the remaining balancing time at any time.

The device includes a hardware OVUV comparator and an OTUT comparator with user configurable thresholds. These can be used as a second-level protector for cell over- and undervoltage and thermistor over- and undertemperature detections independent of ADC measurements.

The device provides an option to embed fault status information to the communication frame. When a device in the daisy chain detects a fault condition, this information is embedded and travels along the communication response frame to the bottom device which can be configured to trigger an NFAULT pin as an interrupt signal to the system. This provides a way to reduce communication overhead without adding an additional twisted pair cable and isolation for faster fault detection.

The device has SLEEP and SHUTDOWN modes for lower power consumption. All functions work in ACTIVE mode, and balancing and hardware comparators for OVUV and OTUT also work in SLEEP mode. While in SHUTDOWN, all active functions are turned off. A HW reset function is available and can be activated by the host MCU. The HW reset provides a POR-like event to the device without actual battery removal. This provides a reliable, low cost, and recoverable option to improve overall system robustness.

6.2 Functional Block Diagram



6.3 Feature Description

6.3.1 Power Supplies

The device generates directly from the battery stack all required supplies for its operation. The following subsections provide an overview of each internal supply block. See [Section 7](#) for the recommended component connection. See [Section 6.3.6.4](#) for diagnostic control and fault detection on the power supplies block.

6.3.1.1 AVAO_REF and AVDD_REF

The AVAO_REF block (analog voltage always on) is powered from the BAT pin. It powers the always-on low-current circuits that are required for all power modes. This block also generates a preregulated reference, AVAO_REF. The AVAO_REF voltage passes through a load switch controlled by the SHUTDOWN mode. The reference voltage after the load switch is AVDD_REF.

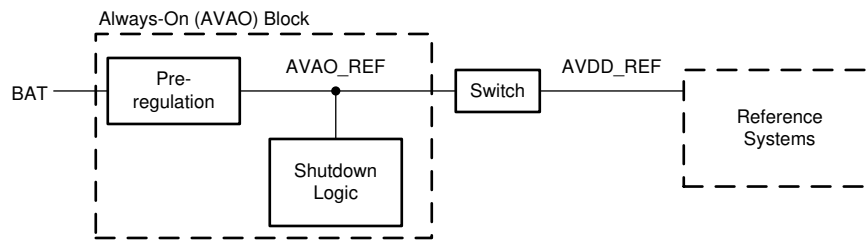


Figure 6-1. AVAO Block

6.3.1.2 LDOIN

The device is powered from the battery module in which the current draw for each cell is the same. From the top of the battery module, the device generates a 6-V regulated voltage (nominal) on the LDOIN pin through the internal linear regulator and an external NPN transistor. The NPNB pin controls the external NPN transistor of the regulator. The LDOIN output is the preregulated input to the rest of the internal low-dropout regulators (LDOs). During OTP (One-Time Programmable) memory programming, the LDOIN pin will be regulated to 8 V (nominal) to supply the programming voltage internally to the OTP programming. The LDOIN is turned off only during HW reset or a POR event.

6.3.1.3 AVDD

The AVDD LDO is the supply for the analog circuits. It takes the input voltage from LDOIN and generates a nominal 5 V. It will not be used to power any external circuit. This LDO is powered down in SHUTDOWN mode, during HW reset, or a POR event.

6.3.1.4 DVDD

The DVDD LDO is the supply for the digital circuits. It takes the input voltage from LDOIN and generates a nominal 1.8 V. It will not be used to power any external circuit. This LDO is powered down in SHUTDOWN mode, during HW reset, or a POR event.

6.3.1.5 CVDD and NEG5V

The CVDD LDO is the supply for the daisy chain interface (or vertical interface, VIF) and the I/O pins (RX, TX, NFAULT, and GPIOs). It takes the input voltage from LDOIN and generates a nominal 5V. Besides providing power for internal usage, this LDO can support an extra 10mA external load in ACTIVE and SLEEP mode, whereas extra 5mA external load in SHUTDOWN mode.

There is a $-5V$ charge pump used for the daisy chain interface (or vertical interface, VIF) and Main ADC blocks. The NEG5V pin has a $-4.6V$ output (nominal). It will be in a low-power burst mode when the device is in SLEEP or SHUTDOWN mode.

6.3.1.6 TSREF

The TSREF is a 5-V buffered reference that can bias the external thermistor circuits, allowing the ADCs to measure temperature and the OTUT protector to detect temperature faults. This reference is measurable by

the Main ADC. Both TSREF and GPIO measured by the Main ADC give a ratiometric measurement for best temperature measurement.

The TSREF is capable of supplying up to I_{TSREF_ILIMIT} and will not be used to power any external circuit other than the thermistor bias. The TSREF is off by default and can be enabled or disabled through the *CONTROL2[TSREF_EN]* bit. The startup time of TSREF is determined by the external capacitance. The MCU ensures TSREF is stable before making any GPIO measurement or OTUT protector detection. After enabling TSREF LDO, user shall wait 1.35ms before sending the next command.

6.3.2 Measurement System

There are two SAR ADCs in the device, a 16-bit Main ADC and a 14-bit AUX ADC; both use a precision reference (REFH) for high-accuracy measurement. Each ADC has its own independent control and can be enabled or disabled separately. The Main ADC is the main measurement for cell voltages (VCELL) and temperature through thermistors connecting to the GPIOs. It also provides TSREF and die temperature measurements. The AUX ADC is mainly used during diagnostic procedures such as providing measurements on internal reference voltages or DAC output of the OVUV and OTUT comparators. It serves as a redundancy measurement for cell voltage inputs and thermistor temperature input through the GPIOs.

The subsections below provide an overview of the Main and AUX ADCs measurement paths. See [Section 7](#) for the recommended external component connection. See [Section 6.3.6.4](#) for the diagnostic control function and status of this block.

6.3.2.1 Main ADC

There are total of 24 inputs (slots) multiplexed to the Main ADC ([Figure 6-2](#)). All inputs are measured in round robin fashion ([Figure 6-3](#)). Each input takes 8 μ s (nominal) to measure and a single round robin cycle completes in 192 μ s (nominal). The inputs to the Main ADC are:

- Die temperature 1
- TSREF
- Cell1 to Cell16 voltages through differential VC_{n-1} to VC_n , where $n = 1$ to 16
- Bus bar input through differential BBP–BBN pins
- Multiplexed GPIO1 through GPIO8
- Spares (RSVD)

All measurements are reported in 16-bit hexadecimal in 2s complement. Results are reported to the corresponding *_HI (high-byte) and *_LO (low-byte) registers. First, convert the hexadecimal results to decimal values. Follow the equations in [Table 6-1](#) to translate the result to μ V or $^{\circ}$ C.

When the Main ADC is enabled, all Main ADC-related result registers shown in [Table 6-1](#) are reset to the default value 0x8000. The measured result is populated to the result registers as the main ADC makes its conversion along the round robin cycle. When MCU reads the *_HI register, the device will pause the data refresh to the associated *_LO register until that *_LO register is read.

Table 6-1. Main ADC Measurement Conversion Equations

| Main ADC Inputs | Result Registers | Conversion Equations |
|-------------------|---------------------------------|--|
| Die Temperature 1 | DIETEMP1_HI/LO | Result in $^{\circ}$ C = $V_{LSB_MAIN_DIETEMP1}$ * Result in decimal 0x0000h is centered to 0 $^{\circ}$ C. |
| TSREF | TSREF_HI/LO | Result in μ V = V_{LSB_TSREF} * Result in decimal |
| Cell1 to Cell16 | VCELL*_HI/LO, where * = 1 to 16 | Result in μ V = V_{LSB_ADC} * Result in decimal |
| Bus Bar | BUSBAR_HI/LO | Result in μ V = V_{LSB_BB} * Result in decimal |
| GPIO1 to GPIO8 | GPIO*_HI/LO, where * = 1 to 8 | Result in μ V = V_{LSB_GPIO} * Result in decimal |

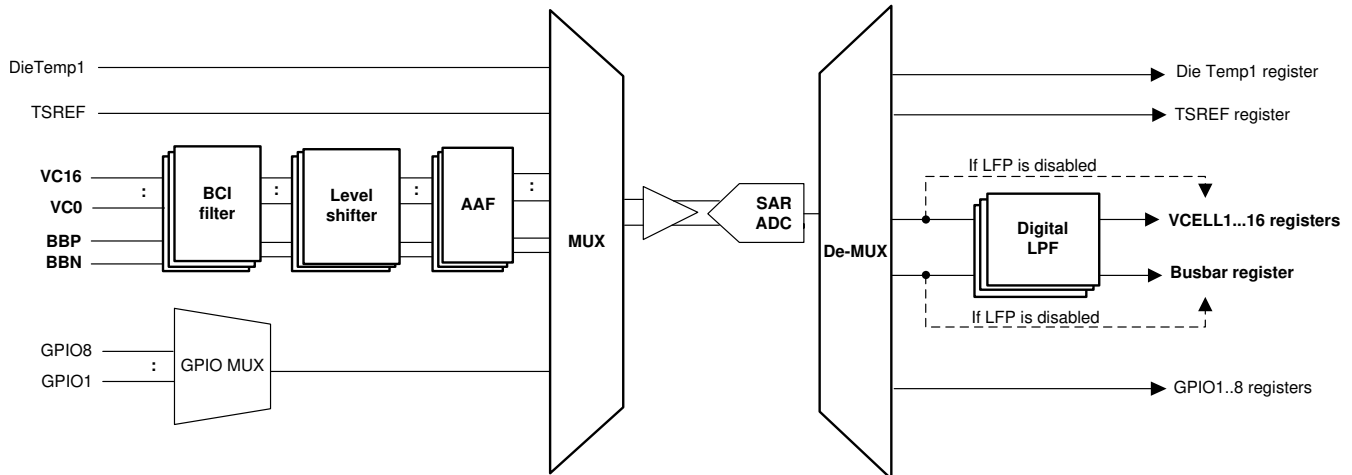


Figure 6-2. Main ADC Measurement Path

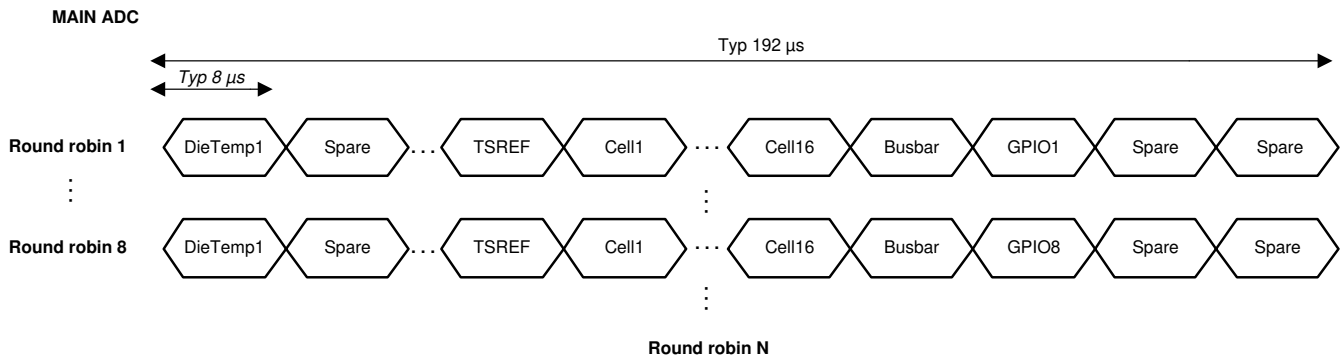


Figure 6-3. Main ADC Round Robin Measurements

6.3.2.1.1 Cell Voltage Measurements

6.3.2.1.1.1 Analog Front End

The cell voltage measurements of the Main ADC are taken from the VC0 through VC16 pins. The device allows a minimum of 6 cells to a maximum of 16 cells to be measured. The VC0 through VC16 pins are connected to the analog front end which consists of a BCI filter, level shifter, and an anti-aliasing filter (AAF) on each VC input channel. The BCI filter has a cutoff frequency (f_{cutoff}) of 100 kHz and the AAF has f_{cutoff} of 1.6 kHz. This filters out high-frequency noise on the VC input before going to the high-voltage multiplexer and measured by the Main ADC. The level shifter block is turned off to save power in SLEEP and SHUTDOWN modes.

6.3.2.1.1.2 VC Channel Measurements

The VC pins are the input channels for cell voltage measurements from the Main ADC measured in the Cell1 to Cell16 slots of the round robin. The round robin timing is always the same even if fewer than 16 cells are connected to the device (Figure 6-4). That is, for the inactive (or unused) VC channel, the device ignores the respective cell slot, but it does not remove the slot from the round robin cycle. This keeps a consistent measurement timing regardless of the cell number configuration. It also provides a consistent sampling time to the post-ADC digital LFP input.

To determine the number of active VCELL channels for ADC measurement, the `ACTIVE_CELL[NUM_CELL3:0]` parameter sets the highest active channel number. The device assumes any VC channel below the setting is also active. For example, when a 14S is connected to the device, the MCU sets the `[NUM_CELL3:0]` to 14S, the Main ADC ignores channel 15 and channel 16 measurements and takes measurements on channels 1 through 14.

The measurement results are reported in the corresponding $VCELL_*_HI$ (high-byte) and $VCELL_*_LO$ (low-byte) registers, where * = 1 to 16. If the digital LPFs are disabled, the result registers are reported with the single ADC conversion values; otherwise, the result registers are reported with filtered measurement values. For an inactive VC channel, the respective $_HI$ and $_LO$ registers remain with the default value 0x8000.

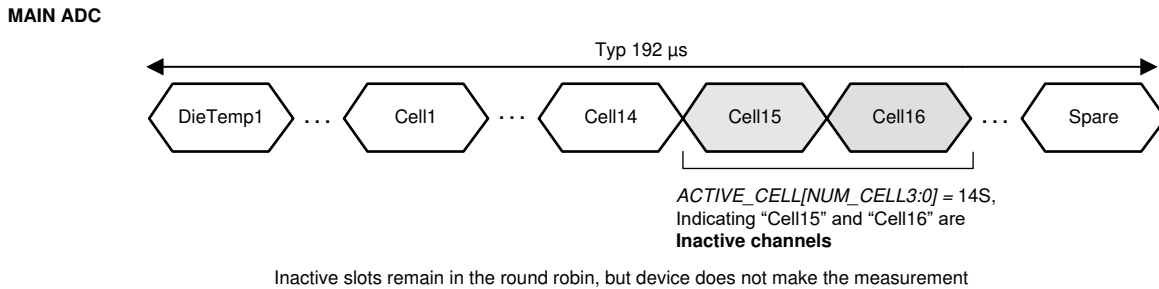


Figure 6-4. Same Round Robin Timing for 6S Through 16S

6.3.2.1.1.3 Post-ADC Digital LPF

Each differential VC channel measurement is equipped with a post-ADC LPF. The LPFs have much lower cutoff frequency (f_{cutoff}). There are 7 f_{cutoff} options: 6.5 Hz, 13 Hz, 26 Hz, 53 Hz, 111 Hz, 240 Hz, and 600 Hz, configurable through the $ADC_CONF1[LPF_VCELL2:0]$ setting. Once an f_{cutoff} value is selected and the LPFs are enabled by setting $ADC_CTRL1[LPF_VCELL_EN] = 1$, the same f_{cutoff} setting applies to all VC channel measurements.

The digital LPF is implemented as single-pole filter which responds very similarly as an analog RC circuit. This means the Main ADC will be running in continuous mode for the digital LPFs to produce effective filtered results.

The MCU should take into account the digital filter settling time when there is a step change in the input DC voltage level. Equation below gives a typical estimate of digital filter settling time to hit settling accuracy threshold for a step in VC voltage.

Digital Filter Settling Time $\sim [\{ \log_{10} (\text{Settling Accuracy Threshold [mV]} / \text{Voltage Step in Input Voltage [mV]}) \} / \{ \log_{10}(1 - \text{Filter Coefficient}) \} - 1] \times 0.192 \text{ ms}$

| Fcutoff (Hz) | 600 | 240 | 111 | 53 | 26 | 13 | 6.5 |
|--------------------|-----|------|-------|--------|---------|----------|----------|
| Filter Coefficient | 0.5 | 0.25 | 0.125 | 0.0625 | 0.03125 | 0.015625 | 0.007813 |

For example: If VC step by 15mV, and user has to accommodate ~27ms settling time to within 1 LSB of input step for 26Hz LPF setting.

When the LPF starts, from disabled to enabled state, it jumps to its first input value and starts the filtering from that point. As compared to starting from 0 V or some mid-level voltage, this implementation allows a fast settling time for Main ADC and LFP is just starting.

6.3.2.1.1.4 BBP and BBN Measurements

The BBP and BBN pins are the inputs for bus bar measurement from the Main ADC. The intent of the BB channel is to enable the system to share a bus bar with a cell to a single VC channel, as the example shows in Figure 6-5. Hence, similar to the VC inputs, the BBP/N inputs also have the BCI, Level-Shifters, and AAF filters in the front end. The differential BB channel measurement also has an option to pass-through a post-ADC digital LPF. With the same f_{cutoff} option as for the VC channel by using different configuration and enable control, $ADC_CONF1[LPF_BB2:0]$ and $ADC_CTRL1[LPF_BB_EN]$.

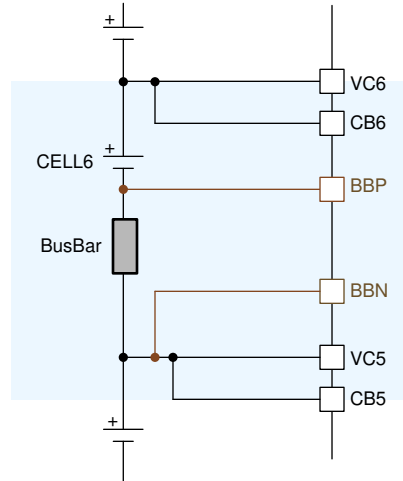


Figure 6-5. Simplified BBP and BBN Connections

The BB channel measurement is reported in the *BUSBAR_HI* (high-byte) and *BUSBAR_LO* (low-byte) registers. If the digital LPF is disabled, the result registers are reported with the single ADC conversion value; otherwise, the result is reported in the filtered measurement value. In [Figure 6-5](#), to obtain the actual Cell6 measurement, the MCU takes the difference of (*VCELL6_HI/LO* measurement – *BUSBAR_HI/LO* measurement). If the BBP and BBN pins are not used (floating), the *BUSBAR_HI/LO* register values are meaningless. The MCU will ignore these register values.

6.3.2.1.2 Temperature Measurements

6.3.2.1.2.1 DieTemp1 Measurement

There are 2 die temperature sensors, DieTemp1 and DieTemp2. The DieTemp1 is routed to the Main ADC and it is also used for the Main ADC gain and offset correction internally. The measurement is reported in the *DIETEMP1_HI* (high-byte) and *DIETEMP1_LO* (low-byte) registers. The 0°C measurement is centered to hex value 0x0000h, so a positive value represents a positive temperature and a negative value represents a negative temperature. The measurement is also capped off to +200°C and –100°C.

6.3.2.1.2.2 GPIOs and TSREF Measurements

There are eight GPIOs. All GPIO inputs are available to be used for thermistor connections for temperature measurements and be used as a simple, single-ended, voltage input measurement.

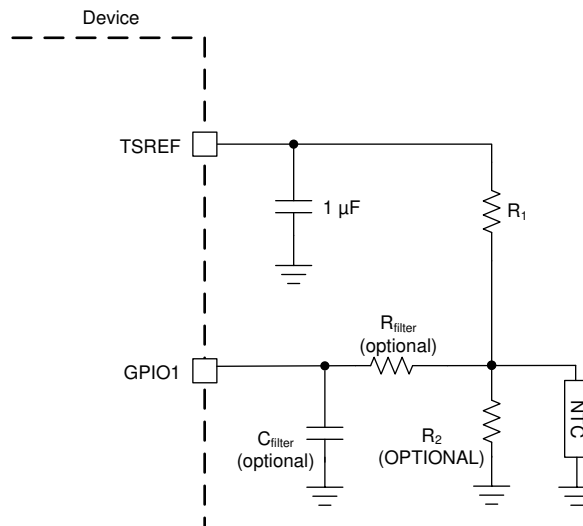


Figure 6-6. Thermistor Connection

Figure 6-6 shows the thermistor circuit when GPIO is enabled for thermistor measurements. MCU ensures TSREF is enabled by setting $CONTROL2[TSREF_EN] = 1$ and settled before taking the measurement value.

The GPIOs are multiplexed to one of the Main ADC MUX inputs. That is, in a single round robin cycle, only one GPIO is measured. To complete all eight GPIO measurements, it takes eight round robin cycles.

To enable the GPIO for ADC measurement, the corresponding $GPIO_CONF_n[GPIO*2:0]$ (where $n = 1$ to 4 , $*$ = 1 to 8 for the corresponding GPIO) register is configured to ADC input or ADC and OTUT input. For example, to enable GPIO1 for ADC measurement only, set $GPIO_CONF_1[GPIO12:0]$ to ADC input. See Section 6.3.5 for more details. If a GPIO is not configured for any ADC measurement, the device will ignore the corresponding GPIO slot but does not remove the slot from the round robin cycle. See Figure 6-7 for an example when GPIO2 is configured for non-ADC measurement.

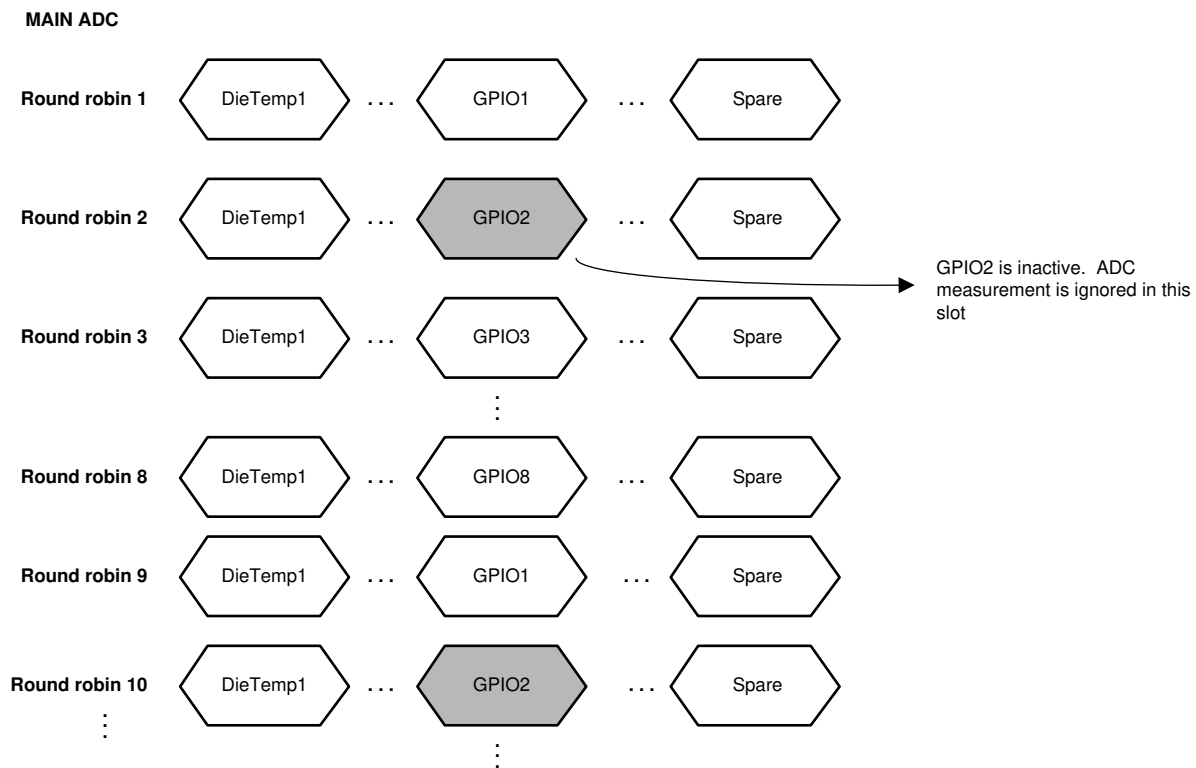


Figure 6-7. GPIO2 Not Configured for ADC Measurement

The measurements are reported in the corresponding $GPIO_HI$ (high-byte) and $GPIO_LO$ (low-byte) registers, where $*$ = 1 to 8 . The measurement result is in μV . To achieve better temperature accuracy, the MCU can use a ratiometric measurement by using both TSREF and GPIO measurement with the following formula: $(GPIO_ADC / TSREF_ADC) = RNTC / (RNTC + R1)$, where

- $GPIO_ADC$ = ADC measurement on GPIO
- $TSREF_ADC$ = ADC measurement on TSREF
- $RNTC$ = NTC thermistor resistance
- $ACTIVE_CELL$ register: Determine the inactive VC channel(s) and keep the result registers to default value 0x8000.
- $R1$ is the pull-up resistor as shown in Figure 6-6 with the assumption the $R2$ is not used

For an inactive GPIO channel, the respective $_HI$ and $_LO$ registers remain with the default value 0x8000.

6.3.2.1.3 Main ADC Operation Control

6.3.2.1.3.1 Operation Modes and Status

To start the Main ADC, the host MCU sets $ADC_CTRL1[MAIN_GO] = 1$. When the device receives the GO command, it first samples the following settings to determine Main ADC configuration and then operates the Main ADC accordingly. Any change of the settings below requires the MCU to resend another GO command to implement the new settings.

- $ADC_CTRL1[MAIN_MODE1:0]$: three run modes. See [Table 6-2](#) for details.
- $ADC_CTRL1[LFP_VCELL_EN]$: LPF for VC channels. Set to $ADC_CONF1[LFP_VCELL2:0] f_{cutoff}$ if enabled.
- $ADC_CTRL1[LFP_BB_EN]$: LPF for BB channel. Set to $ADC_CONF1[LFP_BB2:0] f_{cutoff}$ if enabled.
- $ADC_CONF2[ADC_DLY5:0]$: Delay the start of the Main ADC. Use to align the ADC start time among the daisy-chained devices.
- $ACTIVE_CELL$ register: Determine the inactive VC channel(s) and keep the result registers to default value 0x8000.
- $GPIO_CONF1$ to $GPIO_CONF4$: Determine the inactive GPIO channel(s) and keep the result registers to default value 0x8000.
- $MAIN_ADC_CAL1$, $MAIN_ADC_CAL2$, CS_ADC_CAL1 , CS_ADC_CAL2 , $ADC_CTRL1[CS_DR]$ register

Note

When using the MAIN ADC with the LPF Filter enabled and an ADC reset is desired, it is important that the LFP_VCELL_EN bit, LFP_BB_EN bit and $MAIN_GO$ bit is set to 0 and again set to 1 before running the MAIN ADC again, due to needed re-initialization of the internal LPF buffer. If this procedure is omitted then an LFP_FAIL status bit can occur on the following MAIN ADC activation.

There are two status bits to indicate the Main ADC status:

- $DEV_STAT[MAIN_RUN]$: indicates if the Main ADC is running or not.
- $ADC_STAT1[DRDY_MAIN_ADC]$: set when at least eight round robin cycles have completed indicating all active GPIO channels and all other Main ADC inputs have at least one measurement completed.

Table 6-2. Summary of Main ADC Run Modes

| $[MAIN_MODE1:0]$ | Run Mode | Description |
|-------------------|-------------------------------------|--|
| 0b00 | Stop Main ADC | Stop the Main ADC |
| 0b01 | 8 RR Run (eight round robin cycles) | Main ADC runs for eight round robin cycles then stops. This gives a single measurement on all cell voltages and all GPIO inputs to the system. Filtered measurements are not effective under run mode. For example, use as a quick burst read when MCU is periodically awake during system idle state. |
| 0b10 | Continuous Run | Main ADC runs in continuous mode and stops if $[MAIN_MODE1:0] = 0b00$ and a GO is sent. For example, must use this mode if LPF is enabled. Also use in diagnostic operation. |

The level shifter is enabled for the number of channels specified in the $ACTIVE_CELL[NUM_CELL3:0]$ when device enters ACTIVE mode. MCU shall wait for t_{AFE_SETTLE} time before starting the Main ADC whenever the device enters ACTIVE mode or when $[NUM_CELL3:0]$ setting is changed.

The Main ADC operates in ACTIVE mode only. If the ADC is running while the device goes into SLEEP, the Main ADC will be “frozen” (that is, ADC is stopped but device still remembers the operational state). When the device returns to ACTIVE mode without any digital reset event, the Main ADC will restart and continues from its “pre-frozen” state. In this condition, the cell voltage measurements are off during the t_{AFE_SETTLE} time because input voltage to the ADC is not settled yet. MCU can ignore these measurements or send a new GO command to restart the Main ADC after t_{AFE_SETTLE} .

6.3.2.2 AUX ADC

There are a total of 24 inputs (slots) multiplexed to the AUX ADC ([Figure 6-8](#)). All inputs are measured in round robin fashion ([Figure 6-9](#)). Each input takes 8 μ s (nominal) to measure and a single round robin cycle completes in 192 μ s (nominal). The inputs to AUX ADC are:

- Die temperature 2
- Multiplexed differential CB_{n-1} to CB_n (AUXCELL1 to AUXCELL16), where $n = 1$ to 16 and differential bus bar input through the BBP to BBN pins.
- MISC measurements:
 - BAT pin
 - REFL, internal reference
 - VBG2, internal bandgap
 - VCM, common voltage on Main ADC
 - AVAO_REF, always-on block reference
 - AVDD_REF
 - OV DAC from OV protector
 - UV DAC from UV protector
 - VCBDONE DAC from UV protector
 - OT or OTCB DAC from OT protector
 - UT DAC from UT protector
- Multiplexed GPIO1 to GPIO8
- Spares (RSVD)

All measurements are reported in 16-bit hexadecimal in 2s complement. Results are reported to the corresponding *_HI (high-byte) and *_LO (low-byte) registers. It first converts the hexadecimal results to decimal values. Follow the equations in [Table 6-3](#) to translate the result to μV or $^{\circ}\text{C}$.

When the AUX ADC is enabled, all AUX ADC related result registers shown in [Table 6-3](#) are reset to the default value 0x8000. The measured result is populated to the result registers as the AUX ADC makes its conversion along the round robin cycle. When MCU reads the *_HI register, the device will pause the data refresh to the associated *_LO register until that *_LO register is read.

Table 6-3. AUX ADC Measurement Conversion Equations

| AUX ADC inputs | Result Registers | Conversion Equations |
|--|---|---|
| Die Temperature 2 | DIETEMP2_HI/LO | Result in $^{\circ}\text{C} = V_{\text{LSB_AUX_DIETEMP2}} * \text{Result in decimal}$ Note: 0x0000h is centered to 0°C . |
| Multiplexed AUXCELL1 to AUXCELL16 and BB channel | AUX_CELL_HI/LO, when CB MUX is locked to a single channel | Result in $\mu\text{V} = V_{\text{LSB_ADC}} * \text{Result in decimal}$ |
| BAT | AUX_BAT_HI/LO | Result in $\mu\text{V} = V_{\text{LSB_AUX_BAT}} * \text{Result in decimal}$ |
| REFL | AUX_REFL_HI/LO | Result in $\mu\text{V} = V_{\text{LSB_AUX_DIAG}} * \text{Result in decimal}$ |
| VBG2 | AUX_VBG2_HI/LO | |
| VCM | AUX_VCM_HI/LO | |
| AVAO_REF | AUX_AVAO_REF_HI/LO | |
| AVDD_REF | AUX_AVDD_REF_HI/LO | |
| OV DAC | AUX_OV_DAC_HI/LO | |
| UV_DAC | AUX_UV_DAC_HI/LO | |
| VCBDONE DAC | AUX_VCBDONE_DAC_HI/LO | |
| OT or OTCB DAC | AUX_OT_OTCB_DAC_HI/LO | |
| UT DAC | AUX_UT_DAC_HI/LO | |
| Multiplexed GPIO1 to GPIO8 | AUX_GPIO_HI/LO | Result in $\mu\text{V} = V_{\text{LSB_GPIO}} * \text{Result in decimal}$ |

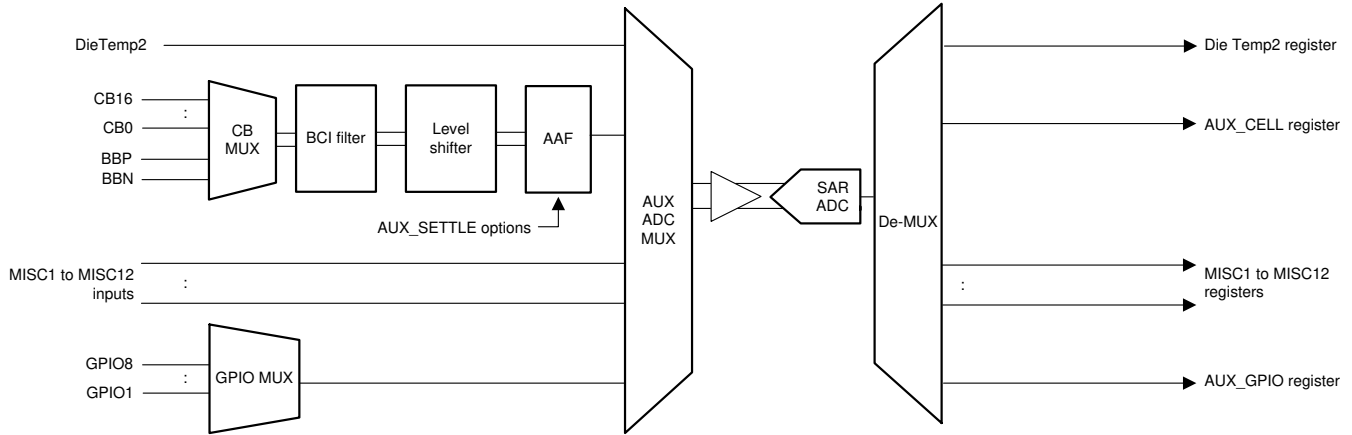


Figure 6-8. AUX ADC Measurement Path

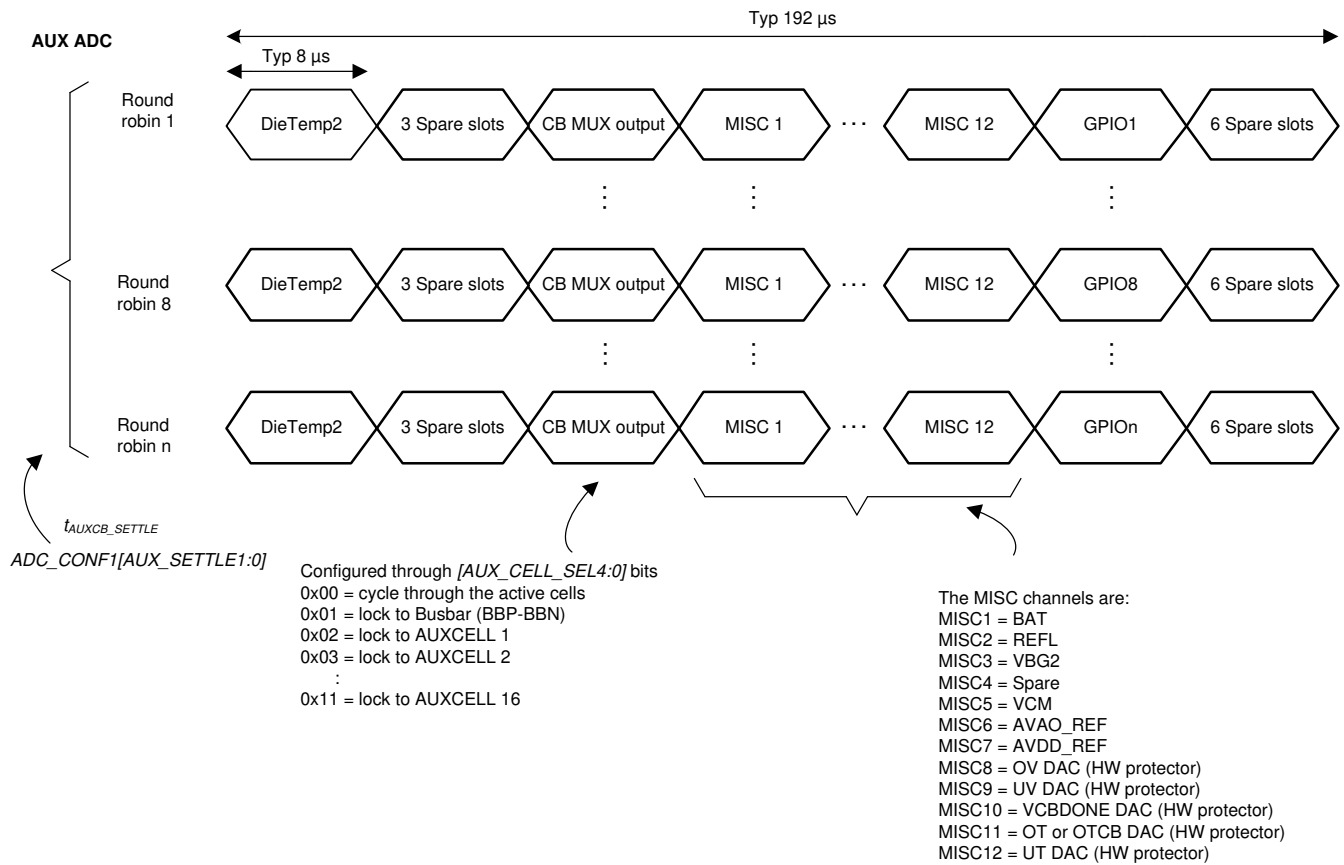


Figure 6-9. AUX ADC Round Robin Measurements

6.3.2.2.1 AUX Cell Voltage Measurements

6.3.2.2.1.1 AUX Analog Front End

The AUX ADC path serves as a redundancy path to the Main ADC measurement on cell voltage measurements and bus bar measurements. It also has the front end filters of a BCI filter and an AAF filter in the AUX ADC path. The AUXCELL channel and differential BB channel (taken from BBP and BBN pins) in the AUX path are multiplexed (shown as the CB MUX in Figure 6-8) to share a single BCI filter and AAF filter. The CB MUX output after the front end filters is then going into one of the AUX ADC MUX and to the AUX ADC for measurement.

Because the front end filters are shared, the device has to wait for the AAF filter to settle before making any valid CB channel (AUXCELL) or BBP and BBN channel measurement. The default AAF f_{cutoff} is 1.6 kHz as in the Main ADC path, which translates to additional 4.3ms settling time to complete a single CB or BB channel measurement. The device provides 3 AAF settling time options, 4.3ms (default), 2.3ms, and 1.3ms, configured by the `ADC_CONF1[AUX_SETTLE1:0]` bits. The BCI filter f_{cutoff} is 100 kHz as in the Main ADC path.

Note

In order to achieve best measurement accuracy through the AUX ADC it is recommended to reset the ADC every time a new CB channel is locked through the `AUX_CELL_SEL` bits. This will ensure that the common mode error calibration routine is re-run and the measured result is compensated for common mode error.

6.3.2.2.1.2 CB and BB Channel Measurements

One slot, the CB MUX output slot, is assigned in the AUX ADC round robin cycle for the CB channels (differential $CB_{n-1} - CB_n$, where $n = 1$ to 16) and BB (differential BBP – BBN) channel measurement because these channels are multiplexed to a single input to the AUX ADC multiplexer. For a single CB or BB channel measurement, it takes multiple round robin cycles because the device has to wait for the AAF settling time as well.

Because of the need to wait for the AAF to settle, the AUX ADC would only measure CB and BB channels that are active and are selected by the MCU; inactive or unselected channels are skipped.

Active CB channels are determined by the `ACTIVE_CELL[NUM_CELL3:0]` setting. These bits set the highest active channel number. For example, when a 14S is connected to the device, the MCU sets the `ACTIVE_CELL[NUM_CELL3:0]` to 14S, the device assumes CB channels 1 through 14 are active; CB channels 15 and 16 are inactive and will be skipped by the AUX ADC.

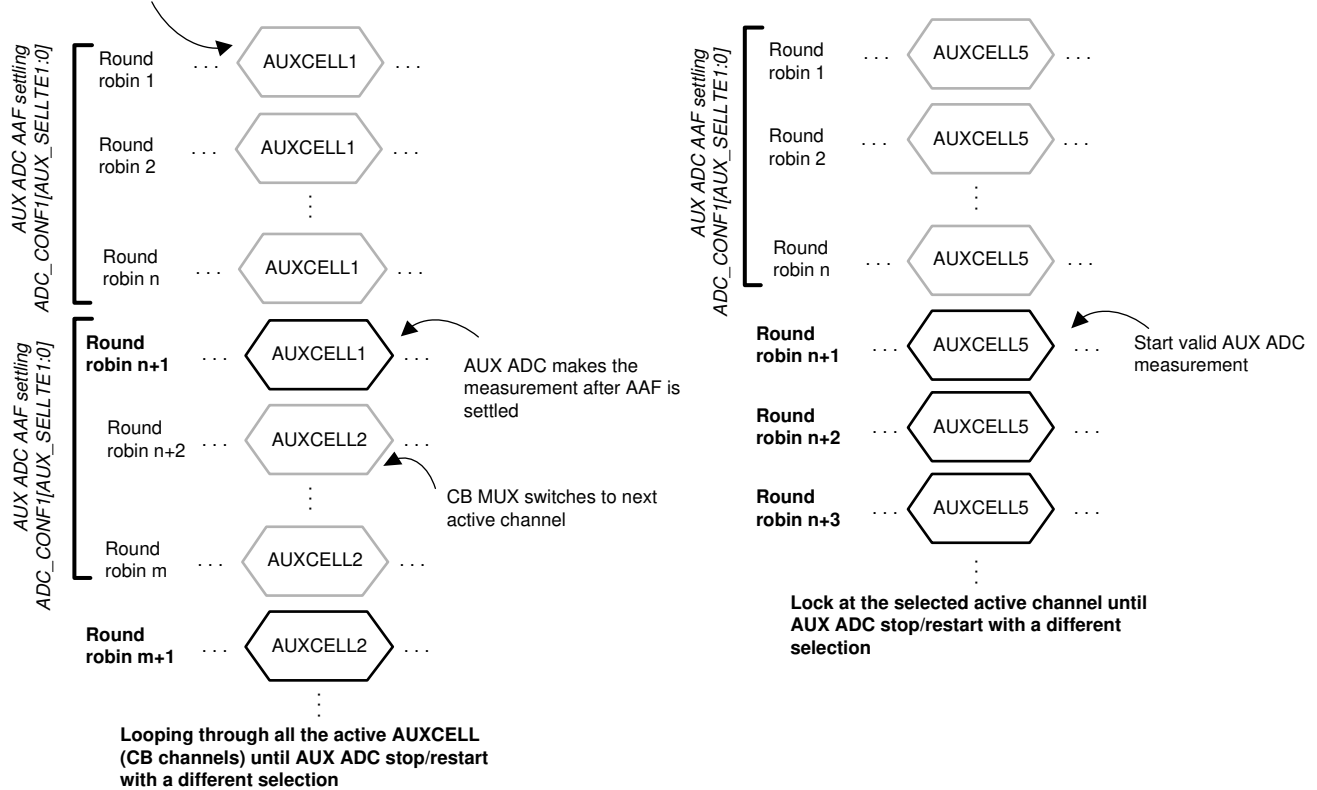
MCU can control which CB and BB channels to be measured through the AUX ADC. The `ADC_CTRL2[AUX_CELL_SEL4:0]` gives the options to run through all the active CB channel and BB channels or to lock to a single CB channels or lock to the BB channel. [Figure 6-10](#) shows the example of how the AUXCELL slot is implemented with different `[AUX_CELL_SEL4:0]` setting.

It is recommended to run AUX ADC in continuous mode and all AUX ADC to measure through all the active CB channel once. This enables the device to reduce the common mode error in AUX ADC measurement. MCU shall perform this procedure before running ADC comparison related diagnostic or locking to a single CB or BB channel measurement.

There is no post-ADC LPF in the AUX ADC path. When the AUX ADC measurements are used during diagnostics, the AUX CELL (CB channel) measurements are compared against the Main ADC prefiltered measurements. While the device performs VCELL (from Main ADC) to AUX CELL (from AUX ADC) measurement comparison internally, the AUX BB comparison is performed by the host instead. See [Section 6.3.6.4](#) for more details.

The device makes the CB or BB channel measurement available to read only when the `[AUX_CELL_SEL4:0]` bits are set to lock on a single CB (must be active) or BB channel. The measurement is reported in the `AUX_CELL_HI` (high-byte) and `AUX_CELL_LO` (low-byte) registers. The result registers will be updated after the AAF settling time is passed. For any other conditions, including lock to an inactive CB channel, the result registers remain with the default value 0x8000.

CB MUX stays at the selected channel for the AUX ADC AAF settling time, but the measurement during this time is discarded



(a) `[AUX_CELL_SEL4:0]` = loop through all active CB channels

(b) `[AUX_CELL_SEL4:0]` = Lock to CB channel 5 (AUXCELL5)

Figure 6-10. CB MUX Output Slot with Different `[AUX_CELL_SEL4:0]` Setting

6.3.2.2.2 AUX Temperature Measurements

6.3.2.2.2.1 DieTemp2 Measurement

There are 2 die temperature sensors, DieTemp1 and DieTemp2. The DieTemp2 is routed to the AUX ADC and is also used for the AUX ADC gain and offset correction internally. The measurement is reported in the `DIETEMP2_HI` (high-byte) and `DIETEMP2_LO` (low-byte) registers. The 0°C measurement is centered to hex value 0x00, so a positive value represents positive temperature and a negative value represents negative temperature. The measurement is also capped off to +200°C and -100°C.

6.3.2.2.2.2 AUX GPIO Measurements

The AUX GPIO path is the same as the main GPIO path. All eight GPIOs are multiplexed to a single AUX ADC MUX input. There is only one GPIO slot in the AUX ADC round robin cycle. That is, in a single AUX ADC round robin cycle, only one GPIO will be measured. To complete all eight GPIO measurements, it takes eight round robin cycles. If GPIO is connected to the thermistor network, the MCU enables TSREF by setting `CONTROL2[TSREF_EN] = 1` and ensures TSREF is stable before starting the AUX ADC measurement.

When AUX ADC is enabled, the GPIO slot in the 1st round robin cycle is GPIO1, 2nd round robin cycle is GPIO3, and so on. For the AUX ADC to make a measurement on a GPIO, the GPIO must be configured as ADC input or ADC and OTUT input in the corresponding `GPIO_CONFn[GPIO*2:0]` bits, where $n = 1$ to 4, $*$ = 1 to 8 for the respective GPIO channel. See Section 6.3.5 for more details. If the GPIO is inactive for the ADC measurement, the device ignores the corresponding GPIO slot but does not remove the slot from the AUX ADC round robin cycle.

By default, the AUX ADC loops through all GPIO channels and the measurements do not report out to the result registers. However, if MCU locks to a single GPIO channel, the locked GPIO measurement is

reported to the *AUX_GPIO*_HI* (high-byte) and *AUX_GPIO*_LO* (low-byte) registers. This channel lock can be set by the *ADC_CTRL3[AUX_GPIO_SEL3:0]* bits. The result registers will report a GPIO measurement if *[AUX_GPIO_SEL3:0]* is locked to single GPIO channel, any other condition will show default value 0x8000.

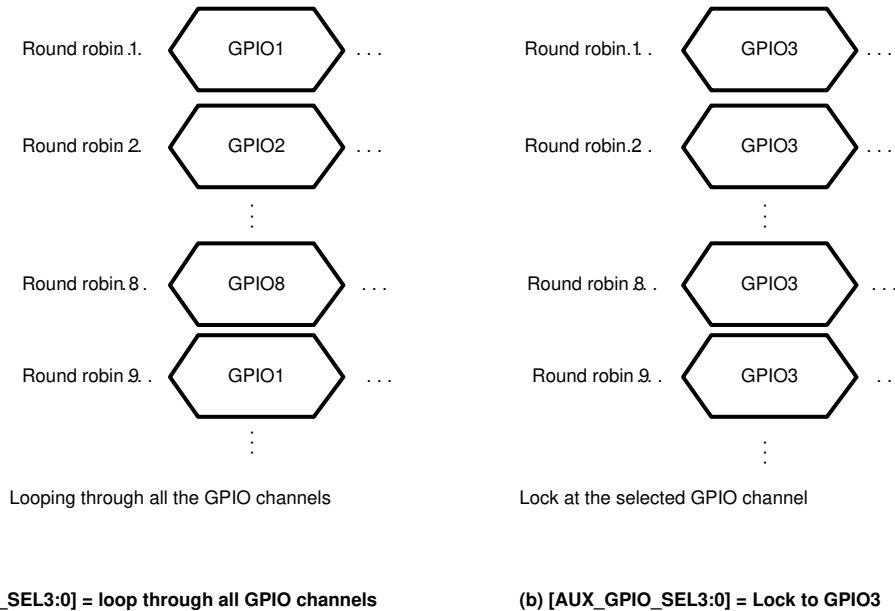


Figure 6-11. GPIO Slot with Different *[AUX_GPIO_SEL3:0]* Setting

6.3.2.2.3 MISC Measurements

There are 12 MISC measurements listed at the beginning of the AUX ADC section. When the AUX ADC is enabled, these inputs are measured in every round robin cycle. [Table 6-3](#) shows the corresponding result registers.

The DAC inputs of the OVUV and OTUT protectors reflect the real-time DAC values of the device which shows the OVUV and OTUT detection or recovery threshold currently in use in the protectors. It is normal to observe a change of the DAC measurements if there are unused channels or if any cell or GPIO channels detect a fault. See [Section 6.3.4](#) for description of the protector architecture and see [Section 6.3.6.4](#) for the protector DAC measurement configuration.

6.3.2.2.4 AUX ADC Operation Control

To start the AUX ADC, the host MCU sets *ADC_CTRL3[AUX_GO]* = 1. When the device receives the GO command, it first samples the following settings to determine the AUX ADC configuration, then operates the AUX ADC accordingly. Any change to the settings below requires the MCU to send another GO command to implement the new settings.

- *ADC_CTRL3[AUX_MODE1:0]*: Four run modes. See [Table 6-4](#) for details.
- *ADC_CTRL2[AUX_CELL_SEL4:0]*: Selects which CB channels are measured by AUX ADC.
- *ADC_CONF1[AUX_SETTLE1:0]*: Configures the AUX ADC AAF settling time.
- *ADC_CTRL3[AUX_GPIO_SEL3:0]*: Selects which GPIO channels are measured by AUX ADC.
- *ACTIVE_CELL* register: Determines the inactive CB channel(s).
- *GPIO_CONF1* to *GPIO_CONF4*: Determines the inactive GPIO channel(s).

There are four status bits to indicate the AUX ADC status:

- *DEV_STAT[AUX_RUN]*: indicates if the AUX ADC is running or not.
- *ADC_STAT1[DRDY_AUX_MISC]*: set when all MISC inputs are measured at least once.
- *ADC_STAT1[DRDY_AUX_CELL]*: set when the CB or BB channels selected by *[AUX_CELL_SEL4:0]* are measured at least once.

- *ADC_STAT1[DRDY_AUX_GPIO]*: set when all GPIO channels (active or inactive) have been measured once. Inactive channel measurements will be ignored by the device.

Table 6-4. Summary of AUX ADC Run Modes

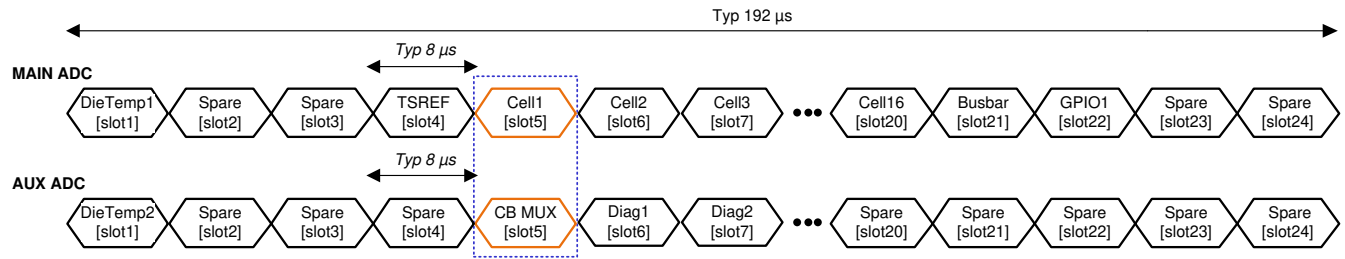
| [AUX_MODE1:0] | Run Mode | Description |
|---------------|-------------------------------------|---|
| 0b00 | Stop AUX ADC | Stop the AUX ADC |
| 0b01 | Single Run (1 round robin cycle) | AUX ADC runs for one round robin cycle then stops. This gives a single measurement on all MISC inputs. For example, use as a quick burst read for just the MISC inputs without the need to issue a stop command to the AUX ADC. |
| 0b10 | Continuous Run | AUX ADC runs in continuous mode and stops if [AUX_MODE1:0] = 0b00 and a GO command is sent. For example, must use this mode when ADC diagnostic comparison operation is used. See Section 6.3.6.4 for details. |
| 0b11 | 8 RR Run (eight round robin cycles) | AUX ADC runs for eight round robin cycles then stops. This gives a single measurement on all active GPIO inputs. |

The AUX ADC operates in ACTIVE mode only. If the ADC is running while the device goes into SLEEP mode, the AUX ADC will be “frozen”; that is, the ADC stops but the device still remembers the operational state. When the device returns to ACTIVE mode without any digital reset event, the AUX ADC will restart and continue from its “pre-frozen” state.

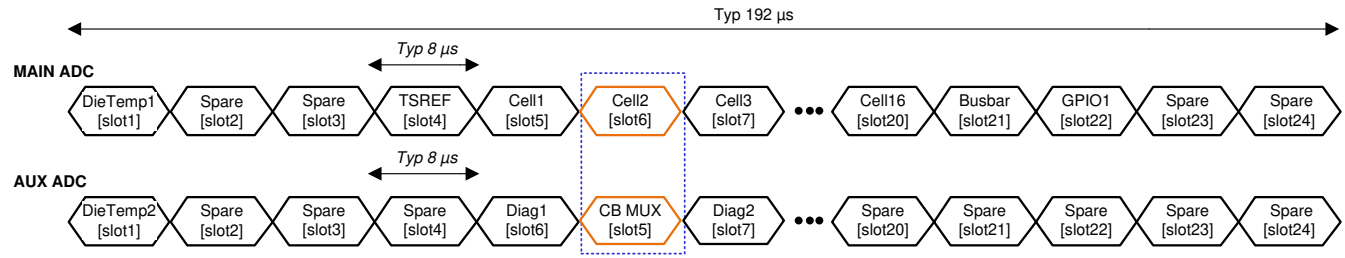
6.3.2.3 Synchronization Between MAIN and AUX ADC Measurements

When *AUX_CELL_ALIGN* = 0x0 in the *ADC_CTRL2* register, the device aligns AUX Cell Measurement (CB MUX - Slot 5) with the target VC channel slot on MAIN cell. DieTemp2 starts without any delay, and AUX cell CB MUX slot #5 moves dynamically to match the selected MAIN cell, and the remaining AUX ADC slots adjust accordingly. This ensures that there is no time skew between MAIN VC and AUX CB ADCs sampling.

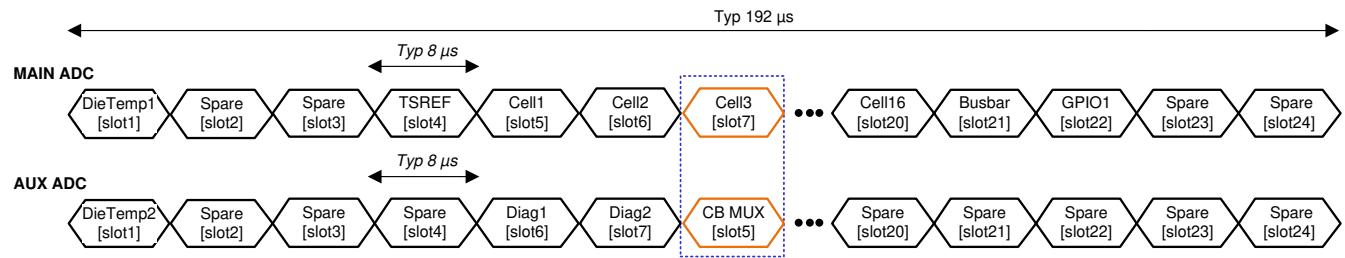
When *AUX_CELL_ALIGN* = 0x1, then the dynamic alignment is disabled and the AUX Cell Measurement (CB MUX Slot #5) is always aligned to Main ADC Cell 8 Measurement (MAIN ADC Slot #12).



a) [AUX_CELL_SEL] = 00h – Running all active cell channels set by ACTIVE_CELL_CONF register. Ch1 conversion.



b) [AUX_CELL_SEL] = 00h – Running all active cell channels set by ACTIVE_CELL_CONF register. Ch2 conversion.



c) [AUX_CELL_SEL] = 04h – Lock to AUX CELL 3. Ch3 conversion.

Figure 6-12. Synchronization Between MAIN and AUX ADC Sampling

6.3.3 Cell Balancing

The device integrates internal cell balancing MOSFET (CBFET) across each CB channel to enable passive cell balancing. The balancing current is determined by the cell voltage, the external resistor in series with the CB pin, and the internal CBFET R_{dson} , R_{DSON} parameter. The following equations calculate the effective balancing current with or without adjacent CBFETs being on. Cell balancing can run in ACTIVE or SLEEP mode.

- Balancing with no consecutive CBFET on (Figure 6-13 (a)): $I_{CB} = V_{Cell} / ((2 \times R_{CB}) + R_{dson_{QCB}})$
- Balancing with two consecutive CBFETs on (Figure 6-13 (b)): $I_{CB} = (\text{Sum of two } V_{CELL}) / ((2 \times R_{CB}) + R_{dson_{QCBn}} + R_{dson_{(QCBn-1)}})$

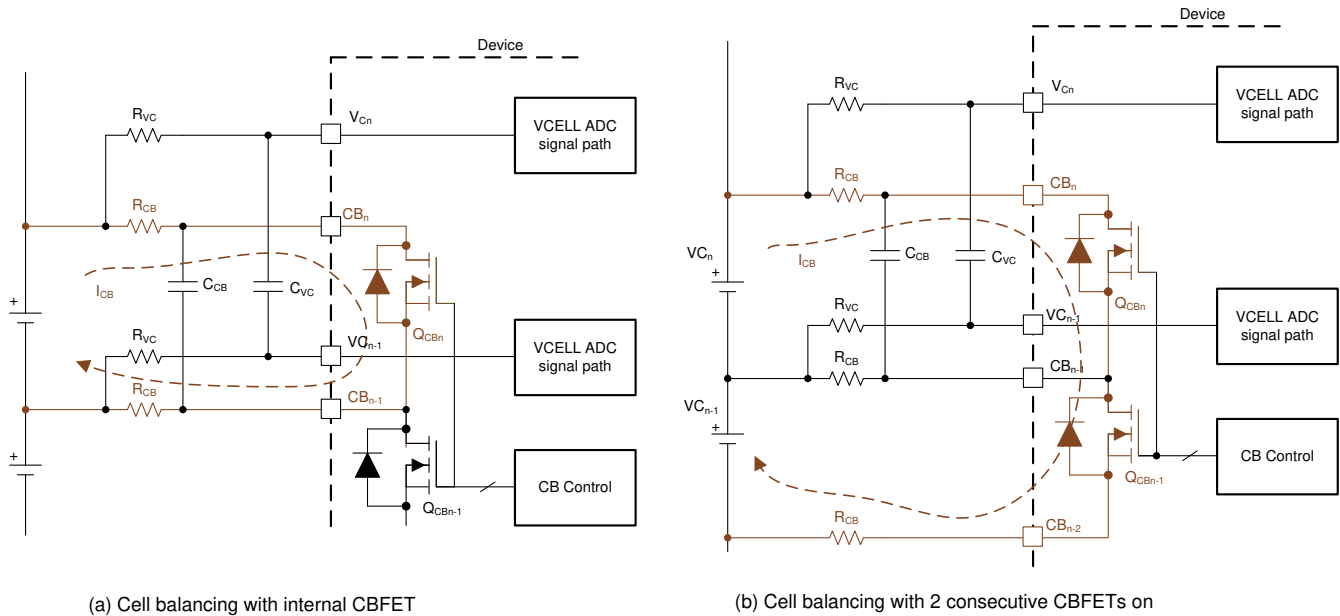


Figure 6-13. Internal Cell Balancing and the Flow of Balancing Current

6.3.3.1 Set Up Cell Balancing

There are three steps to set up cell balancing. Each step is described in detail in the following subsections. The host MCU follows the steps to configure the balancing control before starting cell balancing. Balancing starts by setting $BAL_CTRL2[BAL_GO] = 1$. The $BAL_STAT[CB_RUN] = 1$ indicates the cell balancing is actively running. Note that channels not selected by $ACTIVE_CELL[NUM_CELL3:0]$ are bypassed during cell balancing.

1. Determine which channel to enable for cell balancing.
2. Select the cell balancing control methods, auto or manual balancing control.
3. Decide the additional control configuration:
 - a. Will the thermal management based on thermistor measurement be enabled?
 - b. Is cell balancing stop based on cell voltage?
 - c. Will cell balancing terminate if any unmasked fault is detected?

6.3.3.1.1 Step 1: Determine Balancing Channels

The device provides an individual balancing timer for each channel. The balancing timer is the primary control setting to start and stop the cell balancing on a channel. The balancing timer is configured by $CB_CELL^*_CTRL$ registers, where $*$ = 1 to 16 corresponding to CBFET 1 (CB channel 1) to CBFET 16 (CB channel 16). A non-zero value in these registers sets up the corresponding channels for balancing, but the CBFETs will not turn on until MCU issues the $BAL_CTRL2[BAL_GO] = 1$. When a channel balancing timer expires, cell balancing on that channel stops. Cell balancing can also stop with other conditions, like cell voltage below a certain threshold, unmasked fault is detected, or a forced stop by the host. [Section 6.3.3.3](#) summarizes the cell balancing stop conditions.

6.3.3.1.2 Step 2: Select Balancing Control Methods

The cell balancing runs autonomously once it is configured. The cell balancing control can be configured in two ways using the $BAL_CTRL2[AUTO_BAL]$ bit.

- Auto balancing control ($[AUTO_BAL] = 1$): With this method, host MCU can enable balancing on any channel. Once the host sends a $[BAL_GO] = 1$, balancing starts and the device will automatically duty cycle all enabled CBFETs in an odd and even manner. The duty cycle is configured by $BAL_CTRL1[DUTY2:0]$ bits.
 - Example 1: MCU sets up all 16 channels for cell balancing.

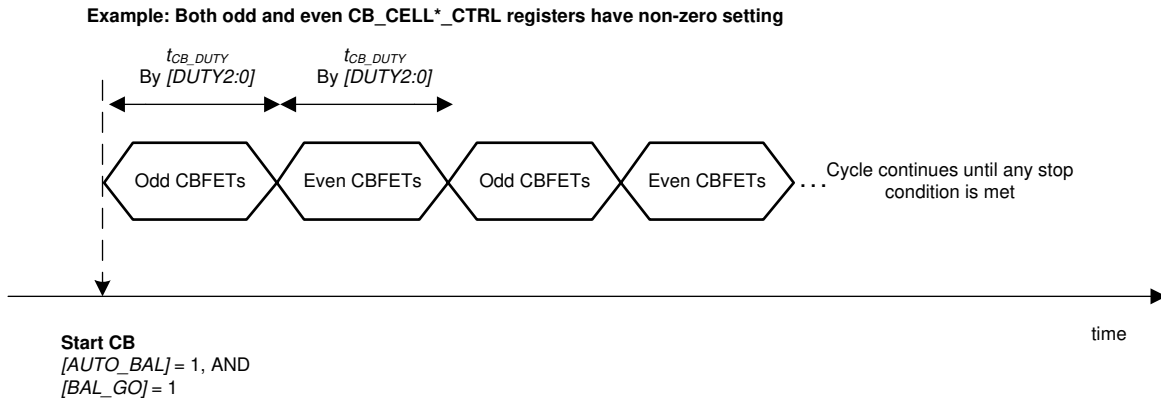


Figure 6-14. Auto Balancing Control, Example 1

- Example 2: MCU sets up odd or even channels only for cell balancing. The $BAL_CTRL1[DUTY2:0]$ bits setting is ineffective because the device is not switched between odd or even channels.

Example: Odd CB_CELL*_CTRL registers have non-zero value
Even CB_CELL*_CTRL registers are all zero

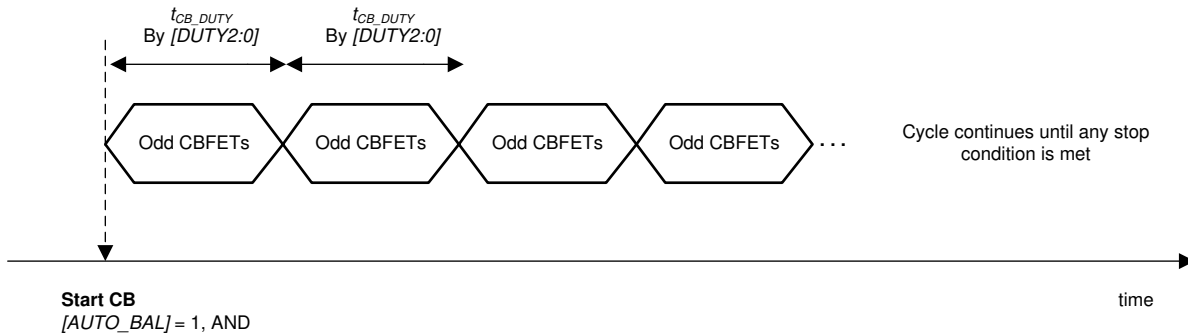


Figure 6-15. Auto Balancing Control, Example 2

- Manual balancing control ($[AUTO_BAL] = 0$): With this method, the device will turn on the CBFETs that have non-zero balancing timer settings once $[BAL_GO] = 1$ is received. There is no odd and even channel switching during the cell balancing and the $BAL_CTRL1[DUTY2:0]$ setting does not apply under this control. Host MCU can enable two consecutive CBFETs with this method and a maximum of eight CBFETs can be enabled. When two consecutive CBFETs are enabled with both channels connected to battery cells, the balancing current is significantly different compared to no adjacent CBFET being on (Figure 6-13). The $DEV_CONF[NO_ADJ_CB]$ bit is provided to avoid inadvertent enabling of an adjacent CBFET for a system that is not intended to have an adjacent channel on for balancing. In this control method, the device is relying on the MCU to enable the proper channels. If the MCU sends $[BAL_GO] = 1$ but the CBFETs are enabled with an invalid condition, the device does not start balancing and sets $BAL_STAT[INVALID_CBCONF] = 1$. Invalid configurations are either:
 - More than eight channels are enabled for balancing (that is, more than eight CB_CELL*_CTRL registers have non-zero settings),
 - $DEV_CONF[NO_ADJ_CB] = 1$, but adjacent channels are enabled for balancing,
 - $DEV_CONF[NO_ADJ_CB] = 0$, but more than two consecutive channels are enabled for balancing:
 - Example: Enabling CBFET 1, 2, 4, 5, 7, 10, 12, and 14 is valid.
 - Example: Enabling CBFET 1, 2, and 3 is invalid.

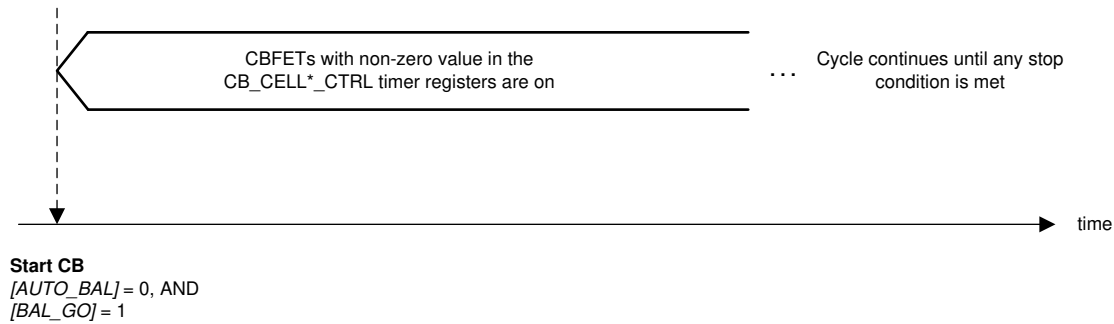


Figure 6-16. Manual Balancing Control

6.3.3.1.3 Step 3a: Balancing Thermal Management

With passive balancing, heat is generated through the internal CBFETs and the external balancing resistors. This creates two hotspots on the PCB, the device, and the balancing resistors area. The device is designed to support up to 240 mA at 75°C ambient. Higher balancing current can be supported with lower ambient temperature.

The device provides two thermal management functions to avoid overheating the die as well as managing the PCB temperature. Both functions monitor temperature, either die temperature or thermistor temperature, to automatically pause balancing if temperature exceeds a pause threshold. When temperature falls below a recovery threshold, balancing resumes automatically. In the cell balancing pause state, all balancing timers and balancing settings are “frozen”, balancing will resume with the same configuration when the device is out of the pause state.

- CB TWARN Balancing Pause: There are die temperature sensors built near the internal CBFETs. When [BAL_GO] = 1 is sent, these temperature sensors are enabled. If any of the sensors detect a die temperature > than the T_{CB_TWARN} threshold (105°C nominal), balancing on all channels is paused. The device sets the $BAL_STAT[CB_INPAUSE] = 1$ and $BAL_STAT[OT_PAUSE_DET] = 1$. When all sensors detect die temperature < ($T_{CB_TWARN} - T_{CB_HYS}$), cell balancing will resume on the balancing enabled channels.
- Thermistor OTCB Balancing Pause: To manage thermal increases due to external balancing resistors, the device has an option to pause cell balancing on all channels if any of the active thermistors connected to GPIOs detects a temperature greater than a threshold set by $OTCB_THRESH[OTCB_THR3:0]$. Once a OTCB detection is triggered, the $BAL_STAT[CB_INPAUSE] = 1$ and $BAL_STAT[OT_PAUSE_DET] = 1$. The balancing on all enabled channels will resume once all active thermistors detect a temperature less than a recovery threshold set by ($OTCB_THRESH[OTCB_THR3:0] + OTCB_THRESH[COOLOFF2:0]$). The OTCB detection is performed through the integrated OT protector. The protector must be turned on and running in round robin mode before cell balancing starts. See [Section 6.3.4](#) for the protector control details. To use the OTCB function, MCU follows the setup sequence state below:
 - Before enabling OT protector:
 - GPIO used for this function will be configured to ADC and OTUT inputs.
 - [OTCB_THR3:0] and [COOLOFF2:0] are configured.
 - Enable the OT protector in round robin mode.
 - Set [OTCB_EN] and [BAL_GO] to 1.

Failure to do so may result in no OTCB pausing action or pausing at the wrong temperature. If a different OTCB or COOLOFF threshold is needed, MCU configures the new threshold values and then re-starts the OT protector to latch in the new setting. It is not required to resend the [BAL_GO] = 1.

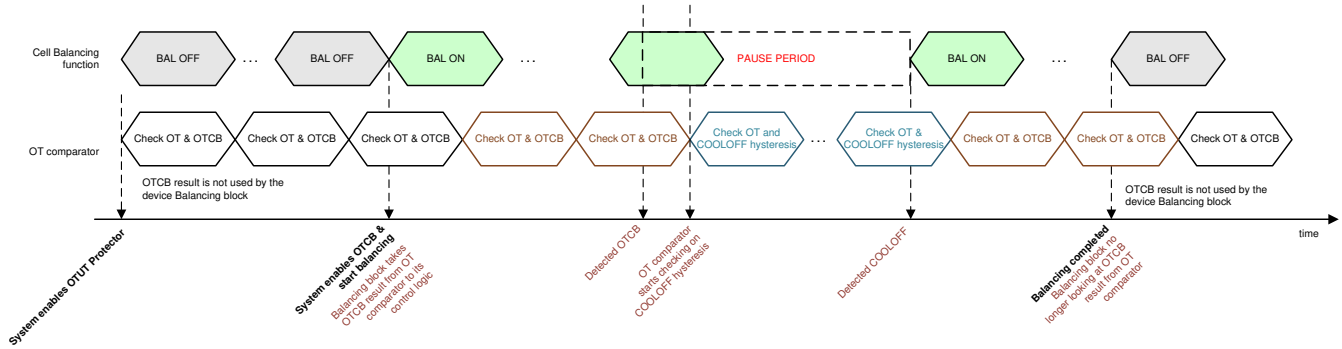


Figure 6-17. Cell Balancing Pause and Resume by OTCB Detection

6.3.3.1.4 Step 3b: Option to Stop On Cell Voltage Threshold

Besides the balancing timers, cell balancing can stop if the channel voltage is less than a threshold set by the `VCB_DONE_THRESH` register with a non-zero value. This stop voltage threshold applies to all channels. When this stop option is used, a channel will stop its balancing either if its balancing timer expires or its voltage level is less than `VCB_DONE_THRESH` setting.

The detection of the `VCB_DONE_THRESH` setting is performed by the integrated UV protector. The protector must be turned on and running in round robin mode before cell balancing starts. See [Section 6.3.4](#) for the protector control details.

When using the `VCB_DONE` detection function, the MCU follows the setup sequence state below:

- Configure the `VCB_DONE_THRESH` register
- Enable the UV protector in round robin mode
- Send `[BAL_GO]` to 1

Failure to do so may result in no `VCB_DONE` detection or cell balancing stops at a wrong channel voltage.

If different `VCB_DONE` thresholds are needed, MCU configures the new threshold values and then re-starts the UV protector to latch in the new setting. It is not required to resend the `[BAL_GO] = 1`.

6.3.3.1.5 Step 3c: Option to Stop at Fault

The device provides an option to abort cell balancing if an unmasked fault is detected. To enable this option, MCU sets `BAL_CTRL2[FLTSTOP_EN] = 1` before starting cell balancing. If cell balancing is aborted under this condition, the `BAL_STAT[ABORTFLT] = 1`.

6.3.3.2 Cell Balancing in SLEEP Mode

Cell balancing can be operated in both ACTIVE and SLEEP modes. To run cell balancing in SLEEP mode, simply configure and start cell balancing in ACTIVE mode first. Once cell balancing is running, put the device in SLEEP mode. Cell balancing will continue autonomously in SLEEP mode. See [Section 6.4](#) for description of putting device in SLEEP mode.

When cell balancing is completed with `BAL_STAT[CB_DONE] = 1`, there is an option to put the device in a different power mode by using the `BAL_CTRL2[BAL_ACT1:0]`. For example, setting `[BAL_ACT1:0]` to 0b10 (SHUTDOWN mode) and start cell balancing. When cell balancing is completed in all balancing enabled channels, the device will automatically enter SHUTDOWN mode without MCU interaction. When multiple devices are connected in daisy chain structure, one device may complete its balancing but another may not. Using this option can result with devices in different power states for some period of time. See [Section 6.3.3.3](#) for details about the `BAL_STAT[CB_DONE]` bit set conditions.

6.3.3.3 Pause and Stop Cell Balancing

6.3.3.3.1 Cell Balancing Pause

Cell balancing can be paused by one of three methods:

- If die temperature during balancing $> T_{CB_TWARN}$.

- If $[OTCB_EN] = 1$ when any thermistor detects a temperature greater than $OTCB_THR$.
- MCU sets $BAL_CTRL2[CB_PAUSE] = 1$.

The first two conditions are described in [Section 6.3.3.1.3](#). The third pause condition is a MCU-controlled pause action usually used during a diagnostic check that involves the CB path. MCU can pause cell balancing through the $[CB_PAUSE]$ bit at any given time once balancing starts.

When the cell balancing is paused due to any of the pause methods, the pause activity is the same:

- Turn off CBFETs on all channels.
- All balancing timers are in hold or “freeze” state.
- $BAL_STAT[CB_INPAUSE] = 1$.
- Any unmasked fault detected during the pause state does not terminate cell balancing. This is because the pause event can be used during diagnostic and fault insertion can be part of the diagnostic.

Once the device exits the cell balancing pause state, the cell balancing resumes. Cell balancing timers will continue the count down. CB channels with non-zero values in their timers will continue with the balancing.

6.3.3.3.2 Cell Balancing Stop

Cell balancing stops in one of three conditions summarized in [Table 6-5](#).

Table 6-5. Cell Balancing Stop Conditions

| Stop Condition | Apply to Individual Channel? | Set $BAL_STAT[CB_DONE] = 1$? |
|---|---|--|
| Cell balancing timer expires | Yes, this stop condition is monitored per channel | Yes, when all channels meet either stop condition 1 or stop condition 2. |
| CB channel voltage < VCB_DONE_THRESH register value | Yes, this stop condition is monitored per channel | |
| $[FLTSTOP_EN] = 1$ and unmasked fault is detected | No, this stops cell balancing on all channels | No, instead set $BAL_STAT[ABORTFLT] = 1$ |

Additionally, MCU can also force stop cell balancing on any particular channel or on all channels by either:

- Zeroing out the balancing timer setting and issuing $[BAL_GO] = 1$.
- Setting a voltage greater than the CB channel voltage in the VCB_DONE_THRESH register and issuing $[BAL_GO] = 1$.

Because the cell balancing timer is the primary control to start cell balancing, if the MCU resets all balancing timers to 0 with $[BAL_GO] = 1$, the device does not start balancing and $BAL_STAT[CB_DONE]$ remains 0.

On the other hand, if any of the cell balancing timers is non-zero but the VCB_DONE_THRESH register is set to a threshold greater than all CB channel voltages with $[BAL_GO] = 1$, the device starts cell balancing because of non-zero values on the balancing timers, but immediately stops because of the VCB_DONE_THRESH stop condition. The $BAL_STAT[CB_DONE]$ is set to 1 for this condition.

6.3.3.3.3 Remaining CB Time

Each channel has a balancing timer, when balancing starts, the timers start counting down from the configured balancing time set by CB_CELLn_CTRL registers, where $n = 1$ to 16. When balancing is pause, these timers are paused.

To read the remaining CB time, MCU set $[BAL_TIME_SEL3:0]$ to select a single channel, then issue $[BAL_TIME_GO] = 1$ which latch the remaining CB time of the selected channel to BAL_TIME register. Repeat the steps to read other remaining CB time on other channels. This timer information is only valid if CB is running, in pause state or in a valid CB stop condition.

If BAL_TIME register reports 0x7F or 0xFF, which is not a valid value. This indicates the balancing configuration is keeping the balancing in a stop state, such as $[BAL_GO] = 1$ with all balancing timer set to 0, or MCU never issue $[BAL_GO] = 1$.

Table 6-6. BAL_TIME Register Status

| CB Stop Condition | BAL_TIME Register |
|---|---|
| Cell balancing timer expires | The selected CB channel reports 0-s |
| CB channel voltage < VCB_DONE_THRESH register value | The selected CB channel reports the remaining CB time |
| [FLTSTOP_EN] = 1 and unmasked fault is detected | |

6.3.3.4 Module Balancing

A small current can sink through GPIO as a way to balance the module voltage. The host can connect a loading resistor on the GPIO and configure the GPIO as digital output high which then loading current through CVDD that is regulated from the module stack. Such control can be turned on or off manually by the host.

Alternatively, the device can control this loading path through a function called module balancing (MB). The concept remains the same as depleting module voltage through a loading resistor connected to a GPIO (MB takes over GPIO3 for this function). However, host can set a module balancing timer and a stop threshold to automatically turn off the loading path through the module balancing function.

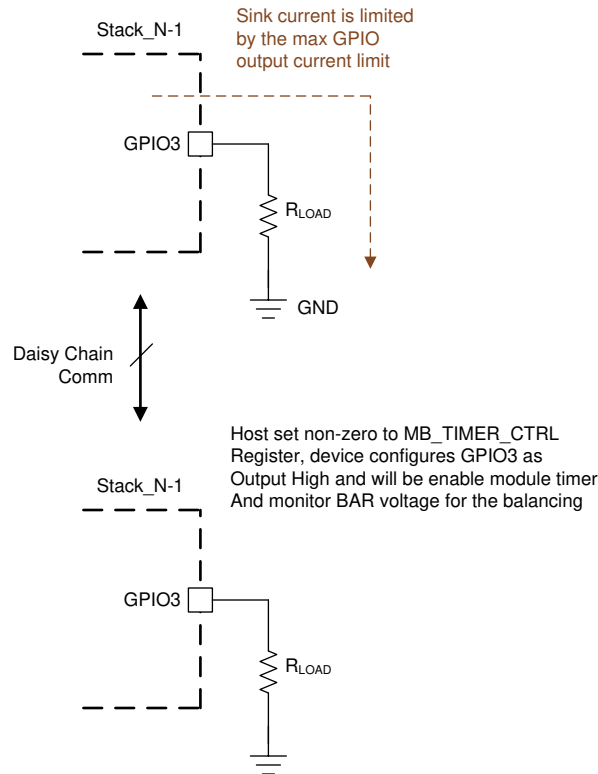


Figure 6-18. Module Balancing

6.3.3.4.1 Start Module Balancing

1. Set a non-zero value to the *MB_TIMER_CTRL* register.
2. Configure the stop MB voltage threshold in the *VMB_DONE_THRESH* register.
3. The stack module monitoring is performed through AUX ADC. So host starts AUX ADC in continuous.
4. Send *BAL_CTRL2[BAL_GO] = 1*.
5. GPIO3 will be taken over for this function and is set to digital output port and starts sinking current through a loading resistor. *BAL_STAT[MB_RUN] = 1*

6.3.3.4.2 Stop Module Balancing

Once started, MB stops if one of the following occurs:

- Balancing timer reaches *MB_TIMER_CTRL* setting, device will set *BAL_STAT[MB_DONE]* = 1.
- BAT voltage is less than *VMB_DONE_THRESH* setting (AUX ADC must be on), device will set *BAL_STAT[MB_DONE]* = 1.
- Host stops the balancing by setting *MB_TIMER_CTRL* = 0 and sends *BAL_CTRL2[BAL_GO]* = 1. *BAL_STAT[MB_DONE]* = 0.
- If *BAL_CTRL[FLTSTOP_EN]* = 1 and an unmasked fault is detected, *BAL_STAT[MB_DONE]* = 0

Note

- There is no pause control (manual pause or thermal pause) to module balancing.
- If AUX ADC stops (either stopped by the host or device enters SLEEP mode) during module balancing, the device will not stop module balancing based on BAT voltage. Module balancing will stop by the MB timer expiration condition.
- *BAL_CTRL2[BAL_ACT1:0]* setting applies to the module balancing function as well. When the *[MB_DONE]* = 1 (and *[CB_DONE]* = 1 if CB is enabled), device can enter the power mode set by the *[BAL_ACT1:0]* setting.

6.3.4 Integrated Hardware Protectors

The device integrates cell OV and UV protectors and thermistor OT and UT protectors with programmable thresholds independent of the ADC functionality or the ADC measurements path. The OVUV and OTUT protectors can operate in ACTIVE or SLEEP mode. The subsections below provide an overview of the protectors. See [Section 6.3.6.4](#) for diagnostic control function and status of this block.

6.3.4.1 OVUV Protectors

A set window comparator provides cell voltage monitoring for all VC channels. This comparator function is entirely separate from the ADC function and as such, even if the ADC function fails, the analog comparators still flag the crossing of the overvoltage (OV) and undervoltage (UV) comparator thresholds. The programmed thresholds are translated through DACs to the comparators.

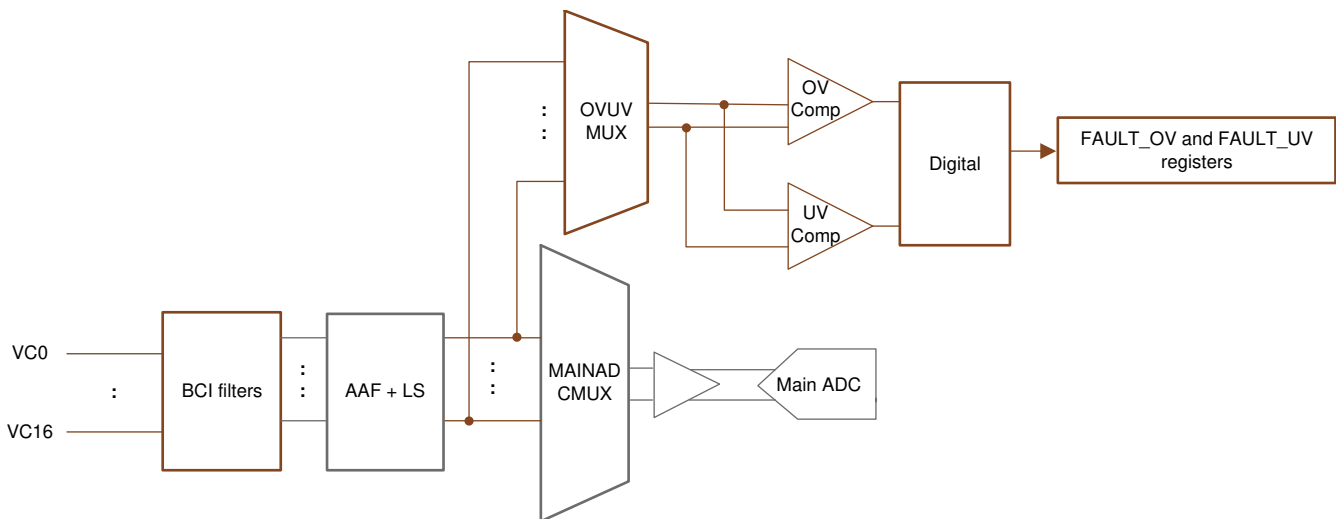


Figure 6-19. OV and UV Protectors

The OV and UV thresholds set by *OV_THRESH* and *UV_THRESH* registers are the same for all VC channels. The active channels are defined by the *ACTIVE_CELL[NUM_CELL3:0]* bits. These bits set the highest active channel number and the device assumes any lower channels are also active.

The *UV_DISABLE1* and *UV_DISABLE2* registers setting disable any individual channel for UV detection, such as channel is connected to bus bar.

Otherwise, the OV protector detects an OV fault on a particular channel if the VC channel voltage is greater than the OV_THRESH setting. The UV protector detects a UV fault on a particular channel if the VC channel voltage is less than the UV_THRESH setting.

6.3.4.1.1 OVUV Operation Modes

The OV and UV protectors have several operation modes controlled by *OVUV_CTRL[OVUV_MODE1:0]* and is summarized in Table 6-7. To start the OVUV protectors, MCU sets *OVUV_CTRL[OVUV_GO] = 1*.

Table 6-7. OVUV Protector Operation Modes

| [OVUV_MOD1:0] | Operation Mode | Description |
|---------------|---|--|
| 0b00 | Stop OV and UV protectors | Stop OV and UV protectors |
| 0b01 | Round robin run | The OV and UV protectors are looping through all VC inputs. The active channels are checked against the OV and UV thresholds (Figure 6-19). The round robin cycle timing is always the same regardless of the number of the active channels. For the inactive VC channels, the digital logic simply ignores the detection outcome. The UV protector detects both <i>UV_THRESH</i> and <i>VCB_DONE_THRESH</i> . |
| 0b10 | OV and UV BIST run (diagnostic use, see Section 6.3.6.4 for details) | A BIST (built-in self-test) cycle on the OV and UV comparators and the detection paths. VCELL (VC channels) ADC measurement from the Main ADC and the OV and UV detections through the OVUV protectors are not available during this run. MCU shall stop ADC measurement when performing OVUV BIST. |
| 0b11 | Single channel run (diagnostic use, see Section 6.3.6.4 for details) | Use for checking the OV and UV DACs. The OV and UV comparator is locked to a single VC input channel in this mode. Channel is locked by <i>OVUV_CTRL[OVUV_LOCK3:0]</i> . |

If OVUV BIST run is in progress, but MCU start ADC, the ADC result registers will be held at 0x8000. ADC measurements will resume once OVUV BIST is completed and after *tAFE_SETTLE* time pass.

If ADC is running, but MCU start OVUV BIST, the ADC result registers will be held at its last measurement. ADC measurement update resumes once OVUV BIST is completed and after *tAFE_SETTLE* time pass

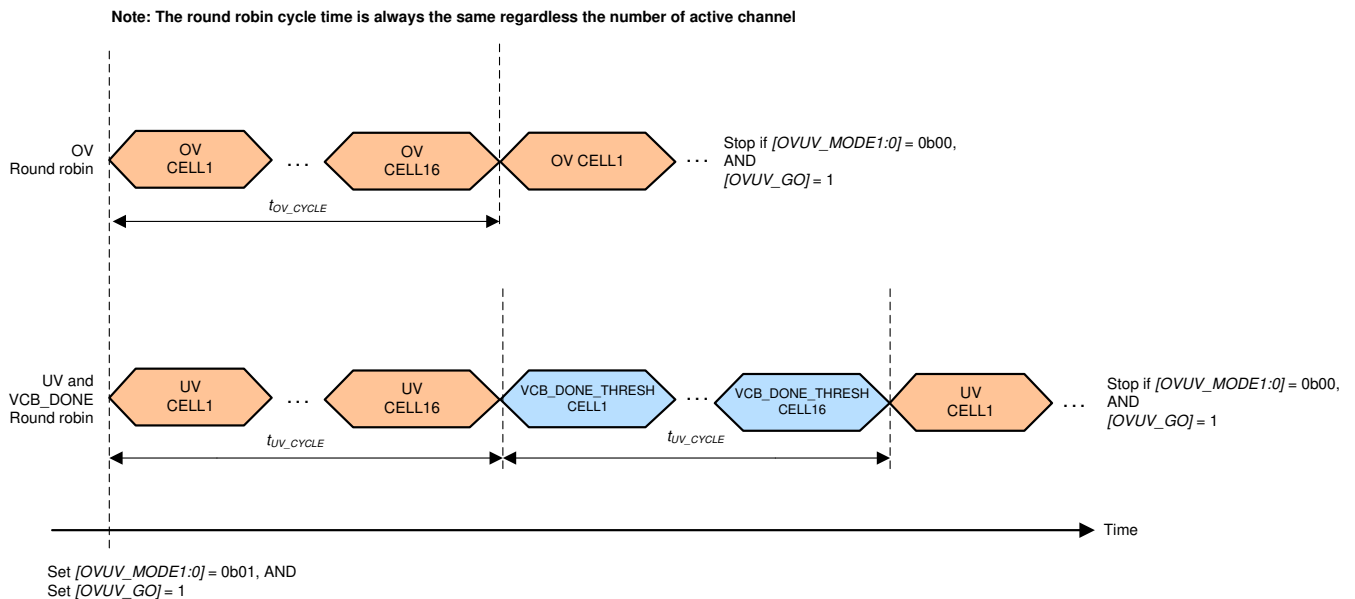


Figure 6-20. OV and UV Round Robin Mode

6.3.4.1.2 OVUV Control and Status

6.3.4.1.2.1 OVUV Control

To start the OV and UV protectors, MCU sets $OVUV_CTRL[OVUV_GO] = 1$. When the device receives the GO command, it samples the following register settings and then starts the OVUV protectors accordingly. Any change of the settings below requires the MCU to resend another GO command to implement the new settings.

- OV_THRESH register: Sets the OV threshold for all VC channels
- UV_THRESH register: Sets the UV threshold for all VC channels
- VCB_DONE_THRESH register: Sets the VCB_DONE threshold for cell balancing stop condition (if enabled)
- $OVUV_CTRL[OVUV_MODE1:0]$: OVUV operation mode selection
- $ACTIVE_CELL$ register: Determines the inactive VC channel(s) and ignores the detection result accordingly
- $UV_DISABLE1$ and $UV_DISABLE2$ registers: Determines the inactive VC channel(s) and ignores the detection result accordingly.

The OVUV protectors can also operate in SLEEP mode. MCU first starts the protector in ACTIVE mode, then puts the device in SLEEP mode. The OVUV protectors will continue the operation until the MCU commands to stop or if the device shuts down.

6.3.4.1.2.2 OVUV Status

The $DEV_STAT[OVUV_RUN] = 1$ indicates the OVUV protectors are running. The OV detection result is reflected in the $FAULT_OV1$ and $FAULT_OV2$ registers; the UV detection result is reflected in the $FAULT_UV1$ and $FAULT_UV2$ registers.

The VCB_DONE detection is not a fault but a cell balancing stop condition. The result is reflected in a particular channel stopping cell balancing. See [Section 6.3.3](#) for details.

6.3.4.2 OTUT Protector

A set window comparator provides temperature monitoring for all GPIO inputs with the external thermistor network pulled up to TSREF. This comparator function is entirely separate from the ADC function and, as such, even if the ADC function fails, the analog comparators still flag the crossing of the overtemperature (OT) and undertemperature (UT) comparator thresholds. The programmed thresholds are translated through DACs to the comparators.

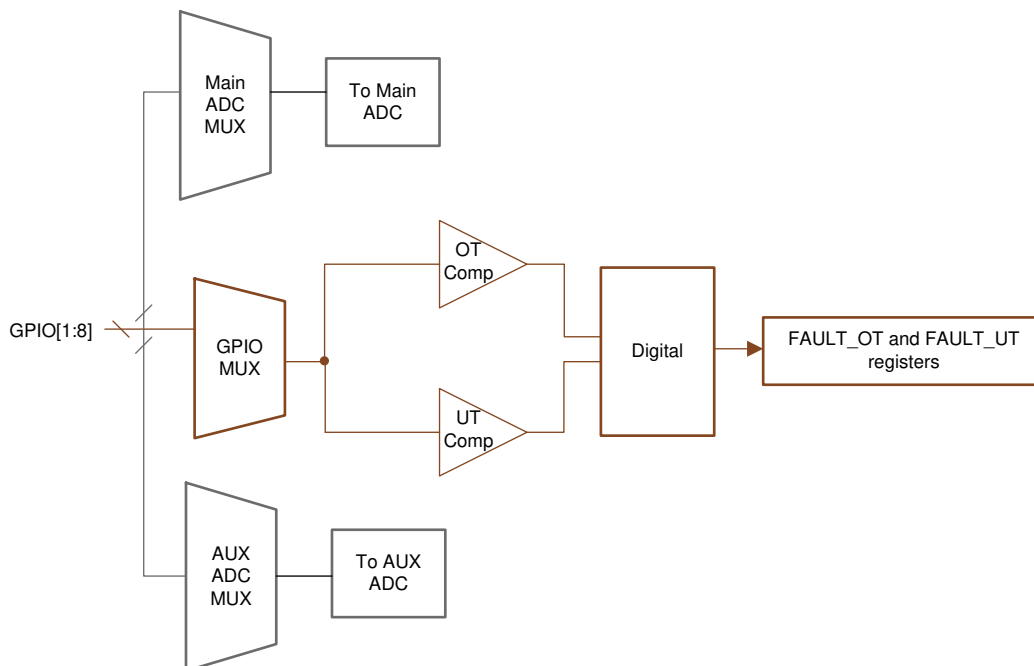


Figure 6-21. OT and UT Protectors

The OT and UT thresholds set by $OTUT_THRESH[OT_THR4:0]$ and $OTUT_THRESH[UT_THR2:0]$ bits are the same for all active GPIO inputs. The active GPIO inputs are defined by the $GPIO_CONFn[GPIO*2:0]$ (where n = 1 to 4, * = 1 to 8 for the corresponding GPIO input). The GPIO has to be configured as ADC and OTUT inputs to be considered as active GPIO inputs for the OTUT protectors.

The OTUT comparators use TSREF as reference, and so the detection is in ratiometric form. The OT protector detects an OT fault on a particular GPIO if the (GPIO voltage/TSREF) is less than the $OTUT_THRESH[OT_THR4:0]$ setting. The UT protector detects a UT fault on a particular GPIO if the (GPIO voltage/TSREF) is more than the $OTUT_THRESH[UT_THR2:0]$ setting. The OTUT protectors assume the NTC thermistor is used for temperature monitoring.

MCU ensures TSREF is enabled before starting the OTUT protectors. Failing to do so, the OTUT protectors will flag all OT and UT faults on all GPIO inputs as an indication of abnormal detection.

6.3.4.2.1 OTUT Operation Modes

The OT and UT protectors have several operation modes controlled by $OTUT_CTRL[OTUT_MODE1:0]$ and are summarized in Table 6-8. To start the OTUT protectors, the MCU sets $OTUT_CTRL[OTUT_GO] = 1$.

Table 6-8. OTUT Protector Operation Modes

| [OTUT_MOD1:0] | Operation Mode | Description |
|---------------|---|---|
| 0b00 | Stop OT and UT protectors | Stop OT and UT protectors |
| 0b01 | Round robin run | The OT and UT protectors are looping through all GPIO inputs. The active GPIO inputs are checked against the OT and UT thresholds (Figure 6-22). The round robin cycle timing is always the same regardless of the number of the active GPIOs. For the inactive GPIO inputs, the digital logic simply ignores the detection outcome. The OT protector detects both OT threshold and OTCB threshold. |
| 0b10 | OT and UT BIST run (diagnostic use, see Section 6.3.6.4 for details) | A BIST (built-in self-test) cycle on the OT and UT comparators and the detection paths. Temperature (GPIO channels) ADC measurement from the main or AUX ADC and the OT and UT detections through the OTUT protectors are not available during this run. |
| 0b11 | Single channel run (diagnostic use, see Section 6.3.6.4 for details) | Used for checking the OT and UT DACs. The OT and UT comparator is locked to a single GPIO input channel in this mode. Channel is locked by $OTUT_CTRL[OTUT_LOCK2:0]$. |

Note: The round robin cycle time is always the same regardless of the number of active GPIO inputs

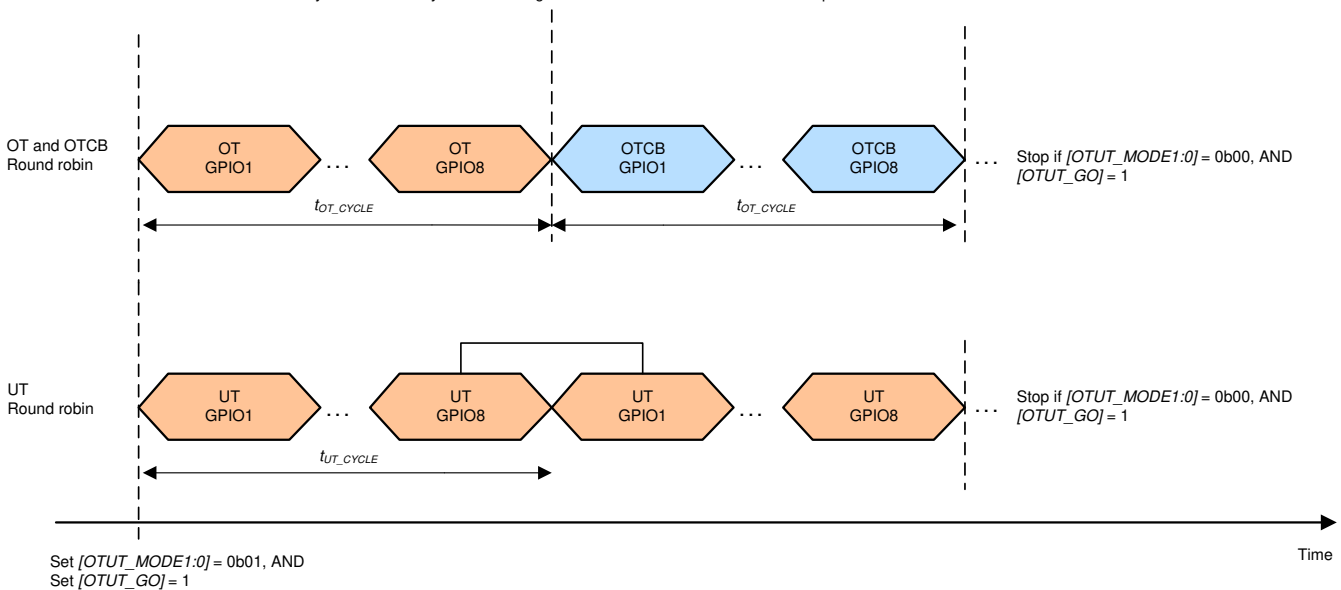


Figure 6-22. OT and UT Round Robin Modes

6.3.4.2.2 OTUT Control and Status

6.3.4.2.2.1 OTUT Control

Ensure TSREF is enabled. To start the OT and UT protectors, host MCU sets $OTUT_CTRL[OTUT_GO] = 1$. When the device receives the GO command, it samples the following register settings and then starts the OTUT protectors accordingly. Any change of the settings below requires the MCU to send another GO command to implement the new settings.

- $OTUT_THRESH[OT_THR4:0]$: Sets the OT threshold for all active GPIO inputs
- $OTUT_THRESH[UT_THR2:0]$: Sets the UT threshold for all active GPIO inputs
- $OTCB_THRESH$ register: Sets the OTCB threshold and COOLOFF hysteresis (if enabled)
- $OTUT_CTRL[OTUT_MODE1:0]$: OTUT operation mode selection
- $GPIO_CONF1$ to $GPIO_CONF4$: Determines the inactive GPIO channel(s) and ignores the detection result.

The OTUT protectors can also operate in SLEEP mode. MCU first starts the protector in ACTIVE mode, then puts the device in SLEEP mode. The OTUT protectors will continue the operation until the MCU commands them to stop or if device shuts down.

6.3.4.2.2.2 OTUT Status

The $DEV_STAT[OTUT_RUN] = 1$ indicates the OTUT protectors are running. The OT detection result is reflected in the $FAULT_OT$ register; the UT detection result is reflected in the $FAULT_UT$ register.

The OTCB detection is not a fault but a cell balancing pause condition. The result is reflected in a particular channel pausing cell balancing. See [Section 6.3.3](#) for details.

6.3.5 GPIO Configuration

The device has eight GPIOs. Each GPIO can be programmed to be one of the configurations below through the $GPIO_CONF1$ to $GPIO_CONF4$ registers.

| GPIO | DISABLE | INPUT | | | OUTPUT | | WEAK PULL-UP/DOWN | | SPECIAL | | |
|-------|---------|---------|------------|----------|--------|-----|--------------------|----------------------|--|-----------------------------------|--------------------------------------|
| | High-Z | Digital | ADC & OTUT | ADC Only | High | Low | ADC & weak pull-up | ADC & weak pull-down | Module Balancing MB_TIMER_CTRL is not 0x00 | SPI Controller $[SPI_EN] = 1$ | Fault Input $[FAULT_IN_EN] = 1$ |
| GPIO1 | √ | √ | √ | √ | √ | √ | √ | √ | | | |
| GPIO2 | √ | √ | √ | √ | √ | √ | √ | √ | | | |
| GPIO3 | √ | √ | √ | √ | √ | √ | √ | √ | √ (output, HIGH) | | |
| GPIO4 | √ | √ | √ | √ | √ | √ | √ | √ | | √ (SS) | |
| GPIO5 | √ | √ | √ | √ | √ | √ | √ | √ | | √ (MISO) | |
| GPIO6 | √ | √ | √ | √ | √ | √ | √ | √ | | √ (MOSI) | |
| GPIO7 | √ | √ | √ | √ | √ | √ | √ | √ | | √ (SCLK) | |
| GPIO8 | √ | √ | √ | √ | √ | √ | √ | √ | | | √ (input, active low) |

| GPIO Configuration | | Description |
|--------------------|--------------|--|
| DISABLE | High-Z | This is the default GPIO configuration at reset if OTP is not programmed |
| INPUT | Digital | When GPIO is configured as Digital Input, the device detects the input voltage level to determine a 1 or 0 with respect to its V_{IL} and V_{IH} levels. The result is shown in the $GPIO_STAT$ register. |
| | ADC and OTUT | The GPIO is configured to be measurable by the ADC (both main and AUX ADCs) and also as the input to the OTUT protectors. Example: use this selection for GPIO used for thermistor connection. |
| | ADC only | The GPIO is configured to be measurable by the ADC (both main and AUX ADCs) only. Example: use this selection to measurement voltage on GPIO. |
| OUTPUT | High | The GPIO is configured as digital output high (internally pull up to CVDD). The logic state is also shown in the $GPIO_STAT$ register. |
| | Low | The GPIO is configured as digital output low. The logic state is also shown in the $GPIO_STAT$ register. |

| GPIO Configuration | | Description |
|--------------------|------------------------|---|
| WEAK PULL-UP/DOWN | ADC and Weak Pull-up | The GPIO is pull up internally and is configured to be measured by the ADC (both main and AUX ADCs) |
| | ADC and Weak Pull-down | The GPIO is pull down internally and is configured to be measured by the ADC (both main and AUX ADCs) |
| SPECIAL | Module Balancing | When the <i>MB_TIMER_CTRL</i> register is non-zero, GPIO3 will be taken over to be used in the module balancing control. This configuration has higher priority over any of the INPUT/OUTPUT configurations on GPIO3. |
| | SPI Controller | When <i>GPIO_CONF1[SPI_EN]</i> = 1, GPIO4 to GPIO7 are taken over as the SPI controller communication lines. This configuration has higher priority over any of the INPUT/OUTPUT configurations on GPIO4 to GPIO7. |
| | Fault Input | When <i>GPIO_CONF1[FAULT_IN_EN]</i> = 1, GPIO8 is taken over as an input that if the GPIO was asserted (active low), will set <i>FAULT_SYS[GPIO]</i> = 1 and assert NFAULT (if enabled). |

6.3.6 Communication, OTP, Diagnostic Control

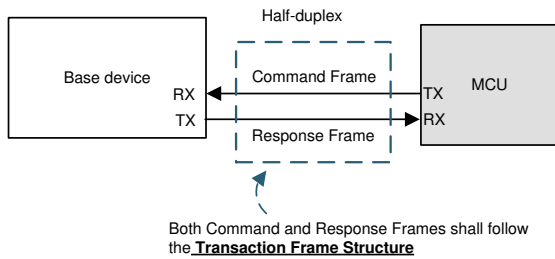
6.3.6.1 Communication

The device can operate as a standalone device in a multidrop configuration (*DEV_CONF[MULTIDROP_EN]* = 1) or as a base/stack device in a daisy chain configuration (*DEV_CONF[MULTIDROP_EN]* = 0). In multidrop configuration, the daisy chain communication is disabled and the host communicates only with a single device through UART interface. This document will focus on the daisy chain communication.

In daisy chain configuration, each device is identified by a 6-bit device address; hence, up to 64 devices can be connected in the daisy chain. In this configuration, a device is either defined to a base (interface with host through UART) or a stack (interface through the daisy chain ports COMH/COML to the base device). The base description in this document assumes the use of BQ79616 as base device. If a communication extender (also known as bridge device) is used as a base, user must refer to the bridge device's datasheet for details.

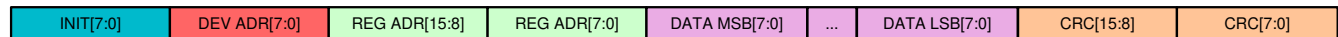
6.3.6.1.1 Serial Interface

The device has a serial interface which uses UART protocol as the physical layer to communicate between base device and host. The communication is specified in a proprietary frame structure.



Transaction Frame Structure (to/from system MCU to the base device):

A transaction frame consists of 5 types of information as shown above. Data are all sent in byte, and each byte is sent through UART protocol.



Each frame is sent through the UART Protocol

UART Protocol:

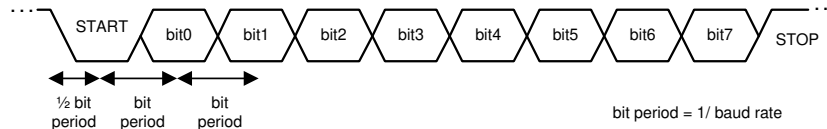


Figure 6-23. UART Communication to Host

6.3.6.1.1.1 UART Physical Layer

The UART interface follows the standard serial protocol of 8-N-1, where it sends information as a START bit, followed by eight data bits, and then one STOP bit. The STOP bit indicates the end of the byte. If a byte is received that does not have the STOP bit set, the *FAULT_COMM1[STOP_DET]* bit is set, indicating there may be a baud rate issue between the host and the device. The device supports 1-Mbps baud rate. Additionally, during development, a slower baud rate is needed to debug the communication, an optional 250-kbps baud rate can be enabled under communication debug mode.

The UART sends data on the TX pin and receives data on the RX pin. When idle, the TX and RX pins are high. The UART interface requires that RX is pulled up to CVDD through a resistor on the base device. The RX is pulled up on the device side. Do not leave RX unconnected. Ensure RX is connected directly to CVDD for stack devices.

The TX pin is disabled in stack devices, but must be pulled high through a resistor on the host side on base device to prevent triggering an invalid communications frame when the communication cable is not attached, or during power-off or SHUTDOWN state when TX is high impedance. TX is always pulled to CVDD internally while in ACTIVE or SLEEP mode, whether enabled or disabled. Leave TX unconnected if not used in stack devices.

The UART interface is strictly a half-duplex interface. While transmitting, any attempted communication on RX is ignored. The only exception is COMM CLEAR signal on RX pin, which immediately terminates the communication. See [Section 6.3.6.1.1.3](#) for details.

Using two STOP bits in UART:

The device can be set up with two stop bits (*DEV_CONF[TWO_STOP_EN] = 1*), the UART response frame transmits from device to host will always return with two STOP bits as shown below. Host is not required to send the command frame to the device with two STOP bits. The device is able to receive one or more stop bits with or without this function enabled.

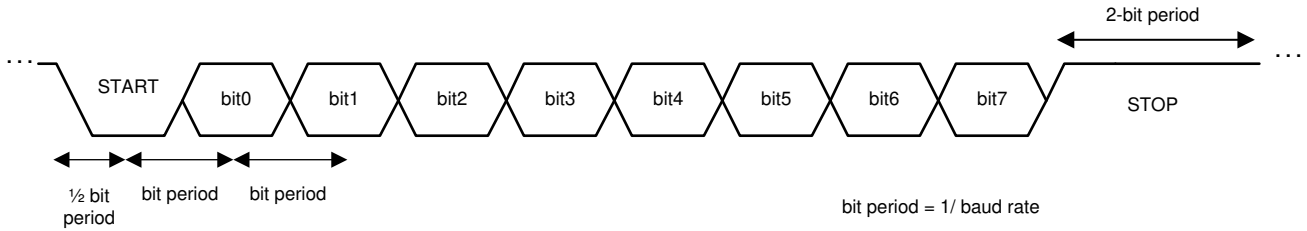


Figure 6-24. UART Response Frame with Two STOP Bits

Potential use of the two stop bits may be to:

- The host to gain extra time to process the data before receiving next data frame.
- The clock tolerance between device and host might cause the data detection out of sync. Having two STOP bits allows re-synchronization of the communication; hence, improving communication robustness.

Although UART is only used by the base device, if the $[TWO_STOP_EN] = 1$, the stack devices also set the $[TWO_STOP_EN] = 1$ even though UART is not used in stacks. It is because the stack devices will use the bit setting to determine the proper gap applying between two communication frames.

6.3.6.1.1.1 UART Transmitter

The transmitter is configured to wait a specified number of bit periods after the last bit reception before starting transmissions using the TX_HOLD_OFF register. This provides time for the host to switch the bus direction at the end of its transmission. The UART transmitter is disabled by default in the stack devices.

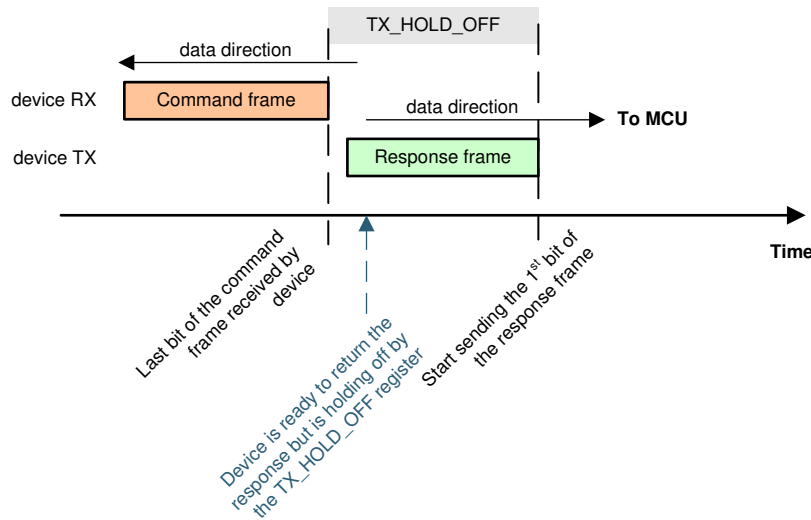


Figure 6-25. UART TX_HOLD_OFF

6.3.6.1.1.2 UART Receiver

While the device is transmitting data on TX, RX is ignored except when receiving a COMM CLEAR. To avoid collisions during data transmission up the daisy-chain interface, the host must wait until all bytes of a communication transmission are received from the device before attempting additional communication to the device. If the host starts transmitting without waiting to receive the preceding transaction's response, the communication is not considered reliable and the host must send a COMM CLEAR to restore normal communications to the base device.

6.3.6.1.1.3 COMM CLEAR

A COMM CLEAR is sent on the RX pin of the base device. It does not send to the stack devices. RX cannot be disabled and a COMM CLEAR can be sent at any time regardless of the TX status. Ensure that the COMM CLEAR does not exceed the maximum value of $t_{UART(CLR)}$ bit periods, as this may result in recognition of other communication pings.

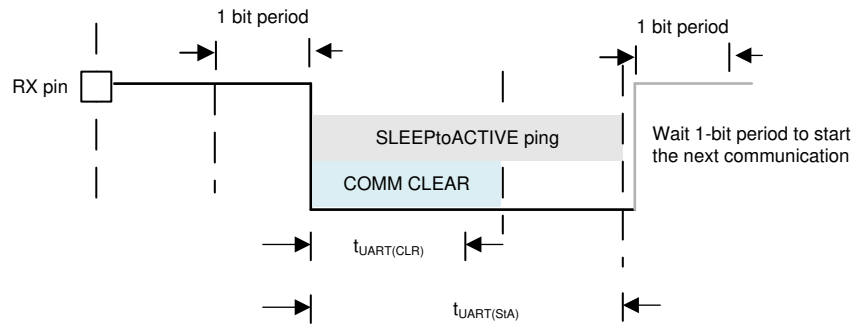


Figure 6-26. UART COMM CLEAR

Use the COMM CLEAR command to clear the receiver and instruct the UART engine to look for a new start of frame. The next byte following the COMM CLEAR is always considered a start-of-frame byte. When detected, a COMM CLEAR sets the *FAULT_COMM1[COMMCLR_DET]* flag. The host must wait at least $t_{UART(RXMIN)}$ after the COMM CLEAR to start sending a new frame. It must be noted that in addition to the *[COMMCLR_DET]* flag, the *FAULT_COMM1[STOP_DET]* flag is also set because the COMM CLEAR timing violates the typical byte timing and the STOP bit is seen as 0.

A SLEEPtoACTIVE ping/tone also clears the UART receiver. This ping/tone sets the *[COMMCLR_DET]* flag when transiting from SLEEP to ACTIVE mode. If this ping/tone is sent during ACTIVE mode, the *[COMMCLR_DET]* and *[STOP_DET]* flags are set.

COMM CLEAR sent during daisy chain communication:

When a read command is sent, but the response has not yet completely returned to the host, if a COMM CLEAR is received in the base device at this condition, the device response is discarded. In addition, the stack devices do not see the COMM CLEAR and continue to send their responses which are forwarded to the host, resulting in host receiving unexpected response frames. Hence, host should avoid this condition by waiting until all responses are received from the stack before sending a COMM CLEAR.

If the above condition occurs, the base device low-level communication debug register *DEBUG_UART_RR_TR[TR_WAIT]* (indicating device is waiting to transmit response) or *DEBUG_UART_RR_TR[TR_SOF]* (indicating a COMM CLEAR is received while device is transmitting data) bits can be set depending on the timing in receiving the COMM CLEAR signal.

When using the multidrop configuration, a COMM CLEAR signal must be used before every frame to ensure consistent communication.

6.3.6.1.1.2 Command and Response Protocol

The host initiates every transaction between the host and device. The device never transmits data without first receiving a command frame from the host. A command frame is a communication frame sent from host to the device; a response frame is a response (to a read command) from device to host. After a command frame is transmitted, the host must wait for all expected responses to be returned (or a timeout in case of error) before initiating a new command frame. The commands supported by the device are listed in [Table 6-9](#).

Table 6-9. Commands

| Command | Description |
|---------------------|---|
| Single Device Read | To read a register(s) from a single device (base or stack) |
| Single Device Write | To write a register(s) to a single device (base or stack) |
| Stack Read | To read a register(s) from the stack devices only. The device must be configured as a stack device with <i>COMM_CTRL[STACK_DEV] = 1</i> to respond to Stack Read commands.. |
| Stack Write | To write a register(s) for only the stack devices. The device must be configured as a stack device with <i>COMM_CTRL[STACK_DEV] = 1</i> to respond to Stack Write commands. |
| Broadcast Read | To read a register(s) for all of the devices in the daisy chain, including base device. |
| Broadcast Write | To write a register(s) for all of the devices in the daisy chain, including base device. |

Table 6-9. Commands (continued)

| Command | Description |
|-----------------------------------|--|
| Broadcast Write Reverse Direction | To send a broadcast write in the reverse direction set by <i>CONTROL1[DIR_SEL]</i> bit. This command is intended to be used for switching the communication direction with the RING interface. |

6.3.6.1.1.2.1 Transaction Frame Structure

The protocol layer is made up of transaction frames. There are two basic types of transaction frames: command frames (transactions from host) and response frames (transactions from device). The transaction frames are made up of the following five field types:

- Frame initialization (INIT, 1-byte)
- Device address (DEV ADR, 1-byte)
- Register address (REG ADR, 2-byte)
- Data (DATA, various byte length)
- Cyclic redundancy check (CRC, 2-byte)

6.3.6.1.1.2.1.1 Frame Initialization Byte

The frame initialization byte is used in both command and response frames. It is always the first byte of the frame. The frame initialization byte performs two functions. First, it defines the frame as either a command frame (host) or a response frame (device). Second, it defines the length of the frame that follows after the frame initialization byte. This provides the receiver an exact number of bytes to expect for a complete command or response.

Table 6-10. Command Frame Initialization Byte Definition

| | | Command Frame | | Response Frame | |
|-------------|-------------------------------|---------------|--|----------------|--|
| | Bit | Bit Name | Description | Bit Name | Description |
| INIT | 7 | FRAME_TYPE | 1 = Define Command Frame | FRAME_TYPE | 0 = Defines Response Frame |
| | 6 | REQ_TYPE | 000 = Single Device Read | RESPONSE_BYTE | Number of the data bytes 0x00 = 1 byte 0x01 = 2 bytes : 0x7F = 128 bytes |
| | 5 | | 001 = Single Device Write | | |
| | 4 | | 010 = Stack Read | | |
| | 3 | | 011 = Stack Write | | |
| | 2 | RSVD | 100 = Broadcast Read | | |
| | 1 | | 101 = Broadcast Write | | |
| 0 | 110 = Broadcast Write Reverse | | | | |
| | | | 111 = RSVD ⁽¹⁾ | | |
| | | | Reserved. This bit is ignored | | |
| | | | Number of data bytes of the command frame, excluding device address, register address or CRC | | |
| | | | 000 = 1 byte | | |
| | | | 001 = 2 bytes | | |
| | | | : | | |
| | | | 111 = 8 bytes | | |

(1) No function to this selection, however, selecting this setting will set the *DEBUG_COMMH[RC_IERR]* or *DEBUG_COMMH[RC_IERR]* flag depends on which daisy chain interface receives the command frame.

6.3.6.1.1.2.1.2 Device Address Byte

The device address byte identifies the device targeted by the single device read/write command. This byte is omitted for broadcast, stack, and broadcast reverse direction command frames. All response frames contain the device address byte. In single device read/write commands, the device that contains a matching value in the *DIR0_ADDR* (used for communication direction with *CONTROL1[DIR_SEL] = 0*) or in *DIR1_ADDR* (used for communication direction with *CONTROL1[DIR_SEL] = 1*) responds to the command. If multiple devices have matching values, all of those devices will respond and cause collision.

Table 6-11. Device Address Byte Definition

| | Command Frame | | | Response Frame | |
|---------|---------------|----------------|--|----------------|--|
| | Bit | Bit Name | Description | Bit Name | Description |
| DEV_ADR | 7 | RSVD | Should always write 0 | RSVD | Should always write 0 |
| | 6 | RSVD | Should always write 0 | RSVD | Should always write 0 |
| | 5 to 0 | Device Address | Set the device address range from 0x00 to 0x3F | Device Address | Set the device address range from 0x00 to 0x3F |

6.3.6.1.1.2.1.3 Register Address Bytes

Register addresses are two bytes in length. Any write command to an invalid register address is ignored. Any read from an invalid register returns a 0x00 response. This is true for command frames sent to an individual register with invalid address, or as part of command sent to multiple registers with invalid addresses. When read/write addresses a block of registers with only some invalid addresses, the valid addresses respond as normal, while the invalid addresses respond as previously described.

Table 6-12. Register Address Byte Definition

| | Command Frame | | | Response Frame | |
|---------|---------------|------------------------|---|------------------------|---|
| | Bit | Bit Name | Description | Bit Name | Description |
| REG_ADR | 7 to 0 | Register Address (MSB) | Target or beginning of the register address | Register Address (MSB) | Target or beginning of the register address |
| | 7 to 0 | Register Address (LSB) | Target or beginning of the register address | Register Address (LSB) | Target or beginning of the register address |

6.3.6.1.1.2.1.4 Data Bytes

The number of data bytes and the relevant information they convey is determined by the type of command frame sent and the target register specified in that command frame. When part of a command frame, the data bytes contain the values to be written to the registers. When part of a response frame, the data bytes contain the values returned from the registers.

Table 6-13. Data Bytes Definition

| | Command Frame | | | Response Frame | |
|------|---------------|---------------|--|----------------|--|
| | Bit | Bit Name | Description | Bit Name | Description |
| DATA | 7 | Data Byte[0] | For Write command: Data value to be written to the register(s) is specified in the REG_ADR frame For Read command: Specify the number of bytes need to be returned by the read command. 0x00 = 1 byte 0x01 = 2 bytes : 0x7F = 128 bytes | Data Byte[0] | Data value return from the register(s) is specified in the REG_ADR frame |
| | 6 | | | | |
| | 5 | | | | |
| | 4 | | | | |
| | 3 | | | | |
| | 2 | | | | |
| | 1 | | | | |
| | 0 | | | | |
| | ... | | | | |
| | 7 | Data Byte [n] | For Write command: Data value to be written to the register(s) is specified in the REG_ADR frame | Data Byte [n] | Data value return from the register(s) is specified in the REG_ADR frame |
| 6 | | | | | |
| 5 | | | | | |
| 4 | | | | | |
| 3 | | | | | |
| 2 | | | | | |
| 1 | | | | | |
| 0 | | | | | |

6.3.6.1.1.2.1.5 CRC Bytes

The device uses a CRC (cyclic redundancy check) to protect data integrity during transmission. The CRC represents the remainder of a process analogous to polynomial long division, where the frame being checked is divided by the generator. The CRC appended to the frame is the remainder. Because of this process, when the device receives a frame, the CRC calculated by the receiver across the entire frame including the transmitted CRC will be zero, indicating a correct transmission and reception. A non-zero result indicates a communication error. Specifically, the device uses the CRC-16-IBM polynomial ($x^{16} + x^{15} + x^2 + 1$) with 0xFFFF initialization.

The CRC value is checked as the first step after receiving the communication frame. If the CRC is incorrect, the entire frame is discarded and not processed. Any additional frame errors are not checked and any errors are not indicated other than CRC error. The bytes are still transferred up or down the stack, thus every device that processed the frame will indicate a CRC error. This results in multiple devices indicating CRC faults on the same communication frame.

6.3.6.1.1.2.1.6 Calculating Frame CRC Value

The CRC calculation by the transmitter is in bit-stream order across the entire transmission frame (except for the CRC). When determining bit-stream order for implementing the CRC algorithm, it is important to note that protocol bytes transmit serially, least-significant bit first. Figure 6-27 illustrates the bit-stream order concept.

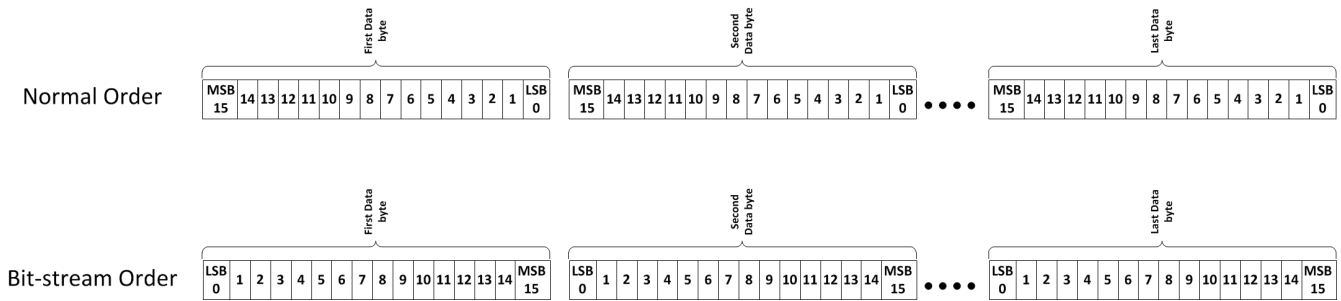


Figure 6-27. Bit-Stream Order Explanation

The CRC (0x0000) is appended to the end of the bit-stream. This bit-stream is then initialized by XOR'ing with 0xFFFF to catch any leading 0 errors. This new bit-stream is then divided by the polynomial (0xC002) until only the 2-byte CRC remains. During this process, the most significant 17 bits of the bit stream are XOR'd with the polynomial. The leading zeroes of the result are removed and that result is XOR'd with the polynomial once again. The process is repeated until only the 2-byte CRC remains. For example:

Example 1: CRC Calculation Using Polynomial Division

```

Command Frame = 0x80 00 02 0F 0B (0b1000 0000 0000 0000 0010 0000 1111 0000 1011)
Command Frame in bit stream order = 0x01 00 40 F0 D0 (0b0000 0001 0000 0000 0100 0000 1111 0000 1101 0000)
After Initialization (XOR with 0xFFFF) = 0b1111 1110 1111 1111 0100 0000 1111 0000 1101 0000
1111 1110 1111 1111 0100 0000 1111 0000 1101 0000 0000 0000 0000 0000 #append 0x0000 for CRC
1100 0000 0000 0010 1 #XOR with polynomial
0011 1110 1111 1101 1100 0000 1111 0000 1101 0000 0000 0000 0000 0000
11 1110 1111 1101 1100 0000 1111 0000 1101 0000 0000 0000 0000 0000 #delete leading zeros from
previous result
11 0000 0000 0000 101 #XOR with polynomial
00 1110 1111 1101 0110 0000 1111 0000 1101 0000
.....
.....
.....
1100 0110 0000 0001 0000 0000
1100 0000 0000 0010 1 #XOR with polynomial
0000 0110 0000 0011 1000 0000
110 0000 0011 1000 0000
110 0000 0000 0001 01 #XOR with polynomial
000 0000 0011 1001 0100
0000 0011 1001 0100 #CRC result in bit stream order
1100 0000 0010 1001 #final CRC result in normal order
CRC final 0xC029
    
```

6.3.6.1.1.2.1.7 Verifying Frame CRC

There are several methods for checking the CRC of a frame. One method is to simply calculate the CRC for the transmitted command except the last two bytes (CRC bytes) using the method described in the previous section, and then compare that result with the transmitted CRC bytes. A more simple option is to run the entire transmission through the CRC algorithm. If the CRC is correct, the result is 0000. In this case, the initial zero padding of the bit-stream with 16 zeroes is not necessary. Using the previous result and running through the algorithm produces the following results:

Example 1: CRC Verification Using Polynomial Division:

```

Command Frame = 0x80 00 02 0F 0B (0b1000 0000 0000 0000 0000 0010 0000 1111 0000 1011)
CRC to Check = 0xC029
Command Frame w/ CRC in bit stream order = 0x80 00 02 0F 0B C0 29 (0b1000 0000 0000 0000 0000 0010
0000 1111 0000 1011 0000 0011 1001 0100)
After Initialization (XOR with 0xFFFF) = 0b0 1111 1110 1111 1111 0100 0000 1111 0000 1101 0000 0000
0011 1001 0100
1111 1110 1111 1111 0100 0000 1111 0000 1101 0000 0000 0011 1001 010 #delete leading zeros from
previous result
1100 0000 0000 0010 1 #XOR with polynomial
0011 1110 1111 1101 1100 0000 1111 0000 1101 0000 0000 0011 1001 0100
11 1110 1111 1101 1100 0000 1111 0000 1101 0000 0000 0011 1001 0100 #delete leading zeros from
previous result
11 0000 0000 0000 101 #XOR with polynomial
00 1110 1111 1101 0110 0000 1111 0000 1101 0000 0000 0011 1001 0100
.....
.....
.....
1100 0110 0000 0010 1001 0100
1100 0000 0000 0010 1 #XOR with polynomial
0000 0110 0000 0000 0001 0100
1 1000 0000 0000 0101 00
1 1000 0000 0000 0101 #XOR with polynomial
0 0000 0000 0000 0000 00
0x0000 #verfiy that CRC checks out valid

```

Note

The result of '0b0000 0000 0000 0000' for the CRC indicates a successful check.

6.3.6.1.1.2.2 Transaction Frame Examples

Transaction frames are created using the frame structure discussed in the previous sections. This section outlines how the command and response frames are passing through the daisy chain. The CRC values in the examples are correct and can be used to verify the customer CRC algorithm. The CRC is verified by the device with every received command frame and the command is not executed unless the CRC is valid.

6.3.6.1.1.2.2.1 Single Device Read/Write

Single Device Read:

Device address must be set up before using this command. A single device read generates a response frame whose length depends on the requested number of register bytes read. The command frame send by host must contain the register address to start at (address field) and the number of bytes to return (number of registers to read). The DATA_SIZE field in the initialization byte for the single device read command is always 0b000.

The command frame travels to all devices in the daisy chain, but only the device that matches the command frame's device address field will respond to the single device read command. The corresponding device will respond with returned data request by the single device read, following the response frame format.

Single Device Write:

Device address must be set up before using this command. A write command for a single device enables the customer to update up to eight consecutive registers with one command. The single device write command frame must contain the register address to start at (address field) and the data bytes to write to the registers.

The DATA_SIZE field in the initialization byte for the single device write command is the number of registers to update.

The command frame travels to all devices in the daisy chain, but only the device that matches the command frame's device address field will execute the single device write command.

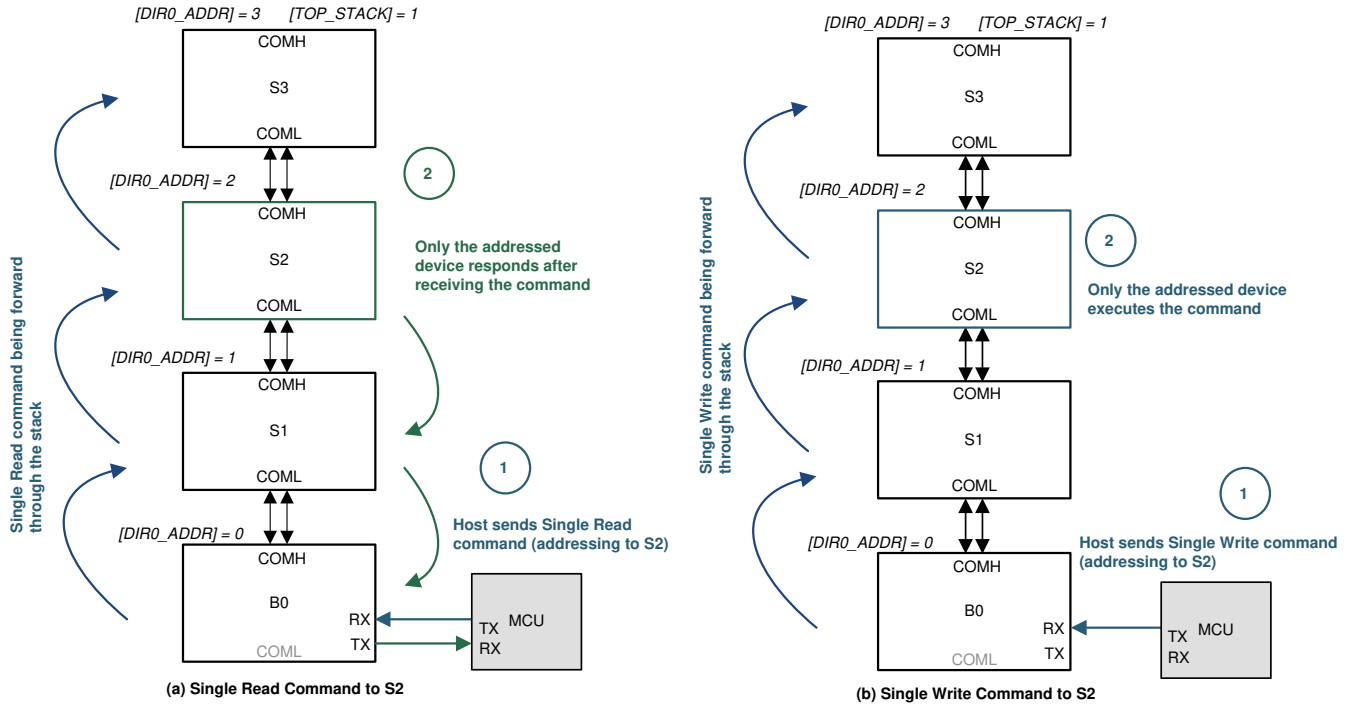


Figure 6-28. Single Device Read/Write

Table 6-14. Single Device Read/Write

| | Single Read Command Sent by Host | | Single Write Command Sent by Host | |
|---------------------|----------------------------------|--|--|---|
| Example | Read 16 Cell Voltages from S2 | | Write OTP Unlock Code to OTP_PROG_UNLOCK1A to 1D Registers | |
| Frame Field | Data | Comments | Data | Comments |
| Initialization Byte | 0x80 | Always 0x80 FRAME_TYPE = 1 REQ_TYPE = 0b000 = Single Read DATA_SIZE = 0b000 | 0x93 | 0x90 for 1 byte data read, 0x91 for 2 bytes data read, 0x92 for 3 bytes data read, and so on. For this example: FRAME_TYPE = 1 REQ_TYPE = 0b001 = Single Write DATA_SIZE = 0b11 = 4 bytes |
| Device Address | 0x02 | Device address 0x02 (S2) in this example | 0x02 | Device address 0x02 (B0) in this example |
| Register Address | 0x0568 | Start address of the register block to read (address of VCELL16_HI in this example) | 0x0300 | Start address of the register block to write (address of OTP_PROG_UNLOCK1A in this example) |
| Data | 0x1F | Instruct the target device to return 32 bytes of data (that is, from address 0x0568 to 0x0587), assuming each VCELLn_HI = 0x80, VCELLn_LO = 0x00, where n = 1 to 16. | 0x02B7 78BC | The unlock value to OTP_PROG_UNLOCK1A to OTP_PROG_UNLOCK1D |
| CRC | 0x5A6F | | 0xB8AE | |

6.3.6.1.1.2.2.2 Stack Read/Write

Stack Read:

The device address, *COMM_CTRL[STACK_DEV]* bit and *[TOP_STACK]* bit must be configured before using this command. A stack read command generates a number of response frames depending on the number of devices in the stack (that is, device with *COMM_CTRL[STACK_DEV]* = 1), whose length depends on the requested number of register bytes read. The stack read command frame must contain the register address to start at (address field) and the number of bytes to return (number of registers to read). The *DATA_SIZE* field in the initialization byte for the read command is always 0b000.

The command frame travels to all devices in the daisy chain, but only the device with *COMM_CTRL[STACK_DEV]* = 1 will respond. During the response, the device with *COMM_CTRL[TOP_STACK]* = 1 will return the response frame first. Each device (address N) in the stack waits until the device above (address N+1) responds before appending its response frame. The CRC is validated while receiving the responses. If a CRC error occurs in the response frame from address N+1, device N does not append its message and an invalid CRC fault is generated.

Use Figure 6-29 with the example of using reading 16 cell voltages from S1 to S3. The response to this command is 3 separate response frames (one response frame per device), each frame with a total length of 38 bytes (32 data bytes + 6 protocol bytes). Although the stack read command does not contain the device address field, each response frame will contain the corresponding device address field associating the data to a particular device. The host will receive a response frame from S3 first (ToS), following with a response frame from S2, and finally the response frame from S1.

Stack Write:

The *COMM_CTRL[STACK_DEV]* must be configured before using this command. A stack write command enables the host to update up to eight consecutive registers for the stack devices (that is, device with *COMM_CTRL[STACK_DEV]* = 1) with one command. The command frame must contain the register address to start at (address field) and the data bytes to write to the registers. The *DATA_SIZE* field in the initialization frame is the number of registers to update.

The command frame travels to all devices in the daisy chain, but only the device with *COMM_CTRL[STACK_DEV]* = 1 will execute the command.

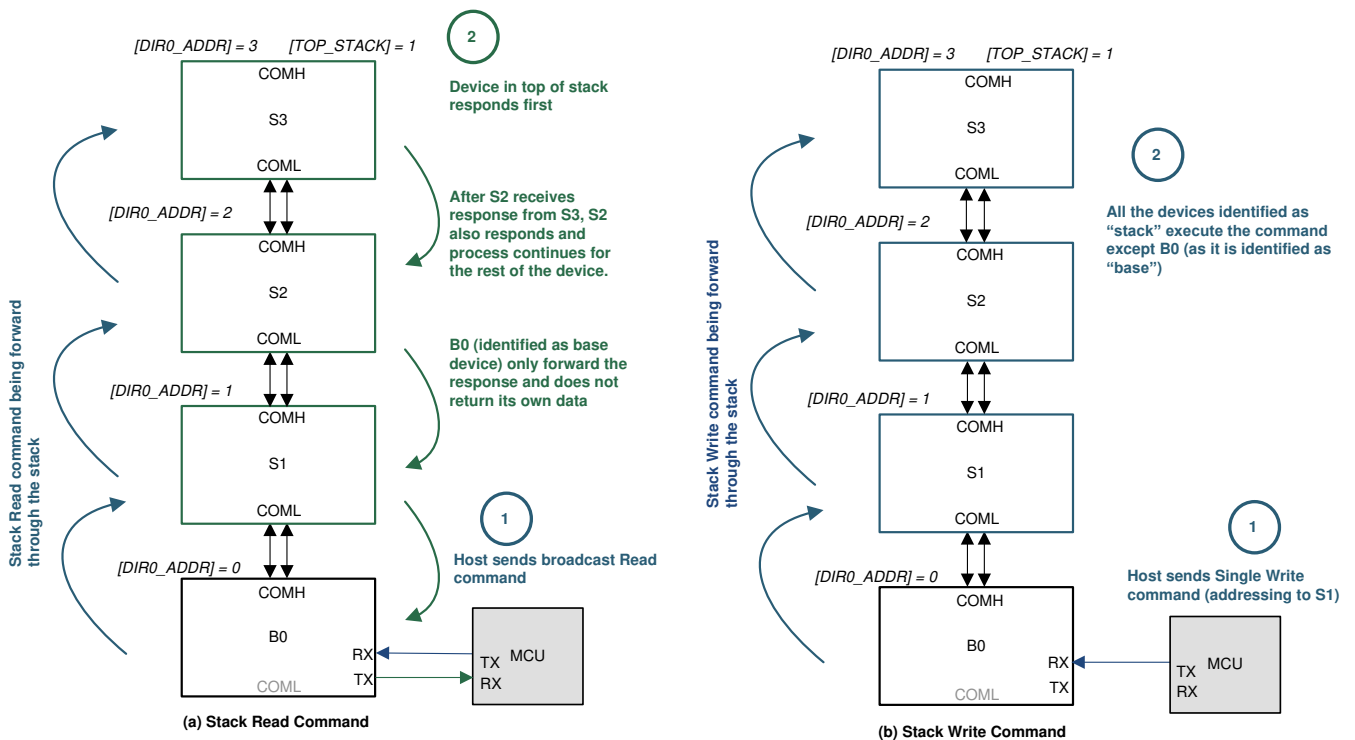


Figure 6-29. Stack Read/Write

Table 6-15. Stack Read/Write

| Stack Read Command Sent by Host | | | Stack Write Command Sent by Host | |
|---------------------------------|-------------------------------------|--|--|---|
| Example | Read 16 Cell Voltages from S1 to S3 | | Write OTP Unlock Code to OTP_PROG_UNLOCK1A to 1D Registers to S1, S2, and S3 | |
| Frame Field | Data | Comments | Data | Comments |
| Initialization Byte | 0xA0 | Always 0xA0 FRAME_TYPE = 1 REQ_TYPE = 0b010 = Stack Read DATA_SIZE = 0b000 | 0xB3 | 0xB0 for 1 byte data read, 0xB1 for 2 bytes data read, 0xB2 for 3 bytes data read, and so on. For this example: FRAME_TYPE = 1 REQ_TYPE = 0b011 = Stack Write DATA_SIZE = 0b011 = 4 bytes |
| Device Address | N/A | No need to include the device address byte in command frame | N/A | No need to include the device address byte in command frame |
| Register Address | 0x0568 | Start address of the register block to read (address of VCELL16_HI in this example) | 0x0300 | Start address of the register block to write (address of OTP_PROG_UNLOCK1A in this example) |
| Data | 0x1F | Instruct each device to return 32 bytes of data (that is, from address 0x0568 to 0x0587), assuming each VCELLn_HI = 0x80, VCELLn_LO = 0x00, where n = 1 to 16. | 0x02B7 78BC | The unlock value to OTP_PROG_UNLOCK1A to OTP_PROG_UNLOCK1D |
| CRC | 0x5C2D | | 0x0BD7 | |

6.3.6.1.1.2.2.3 Broadcast Read/Write

Broadcast Read:

The device address and $[TOP_STACK]$ bit must be configured before using this command. A broadcast read command generates a number of response frames depending on the number of devices in the daisy chain (both stack and base devices), whose length depends on the requested number of register bytes read. The broadcast read command frame must contain the register address to start at (address field) and the number of bytes to return (number of registers to read). The DATA_SIZE field in the initialization byte for the read command is always 0b000.

The command frame travels to all devices in the daisy chain, every device will respond. During the response, the device with $COMM_CTRL[TOP_STACK] = 1$ will return the response frame first, each device (address N) in the stack waits until the device above (address N+1) responds before appending its response frame. The CRC is validated while receiving the responses. If a CRC error occurs in the response frame from address N+1, device N does not append its message and an invalid CRC fault is generated.

Use [Table 6-16](#) with the example of reading 16 cell voltages from B0 to S3. The response to this command is 4 separate response frames (one response frame per device), each frame with a total length of 38 bytes (32 data bytes + 6 protocol bytes). Although the broadcast read command does not contain the device address field, each response frame will contain the corresponding device address field, associated the data to a particular device. The host will receive the response frame from S3 first (ToS), following with the response frame from S2, then S1, and finally the response frame from B0.

Broadcast Write:

This command can be used without auto-addressing. A broadcast write command enables the host to update up to eight consecutive registers for all devices in the daisy chain with one command. The command frame must contain the register address to start at (address field) and the data bytes to write to the registers. The DATA_SIZE field in the initialization frame is the number of registers to update.

The command frame travels to all the devices in the daisy chain, and every device in the daisy chain will execute the command.

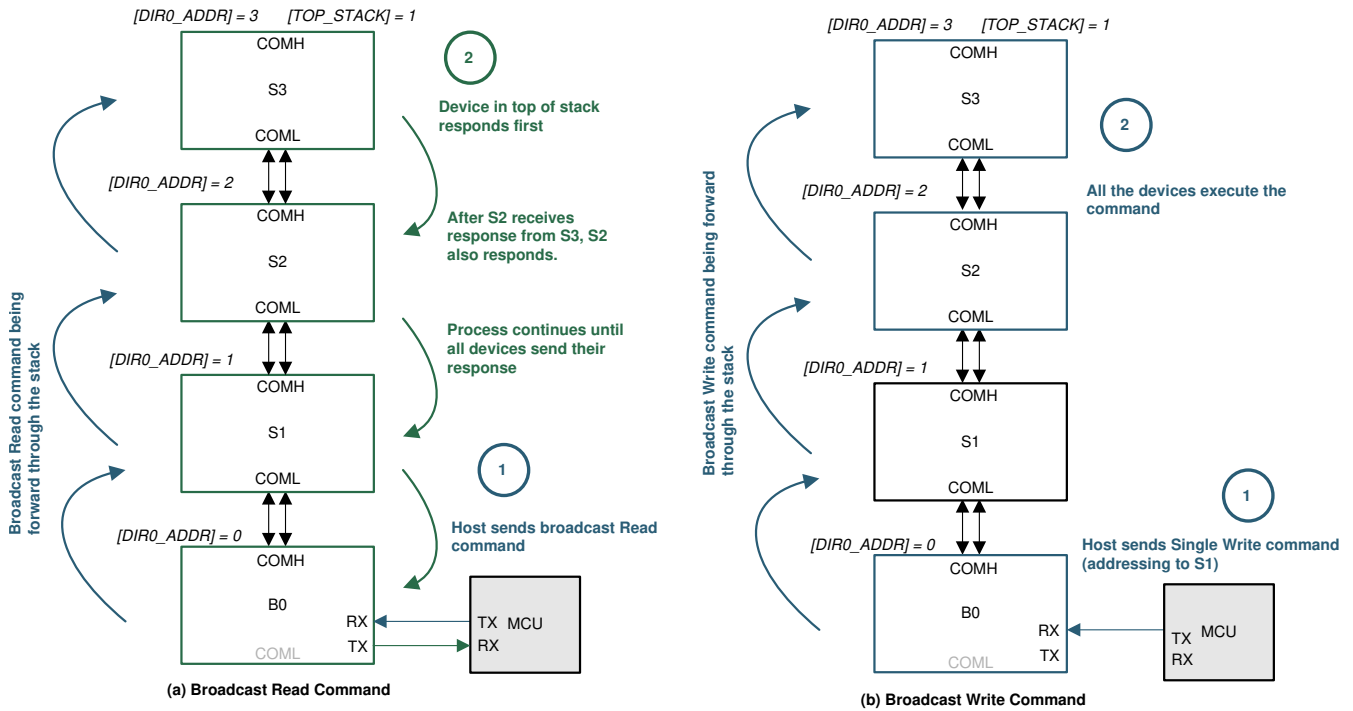


Figure 6-30. Broadcast Read/Write

Table 6-16. Broadcast Read/Write

| | Broadcast Read Command Sent by Host | | Broadcast Write Command Sent by Host | |
|---------------------|-------------------------------------|--|--|---|
| Example | Read 16 Cell Voltages from B0 to S3 | | Write OTP Unlock Code to OTP_PROG_UNLOCK1A to 1D Registers to B0, S1, S2, and S3 | |
| Frame Field | Data | Comments | Data | Comments |
| Initialization Byte | 0xC0 | Always 0xC0 FRAME_TYPE = 1 REQ_TYPE = 0b100 = Broadcast Read DATA_SIZE = 0b000 | 0xD3 | 0xD0 for 1 byte data read, 0xD1 for 2 bytes data read, 0xD2 for 3 bytes data read, and so on. For this example: FRAME_TYPE = 1 REQ_TYPE = 0b101 = Broadcast Write DATA_SIZE = 0b011 = 4 bytes |
| Device Address | N/A | No need to include the device address byte in command frame | N/A | No need to include the device address byte in command frame |
| Register Address | 0x0568 | Start address of the register block to read (address of VCELL16_HI in this example) | 0x0300 | Start address of the register block to write (address of OTP_PROG_UNLOCK1A in this example) |
| Data | 0x1F | Instruct each device to return 32-bytes of data (that is, from address 0x0568 to 0x0587), assuming each VCELLn_HI = 0x80, VCELLn_LO = 0x00, where n = 1 to 16. | 0x02B7 78BC | The unlock value to OTP_PROG_UNLOCK1A to OTP_PROG_UNLOCK1D |
| CRC | 0x422D | | 0x6BD1 | |

6.3.6.1.1.2.4 Broadcast Write Reverse Direction

Usually, device is expecting to receive communication based on the *[DIR_SEL]* setting. If a device receives communication frame opposite to the *[DIR_SEL]* setting, such as receiving command frame from COMH while *[DIR_SEL]* = 0, it will flag the communication as error. The broadcast write reverse direction is a command used to change flip the *[DIR_SEL]* setting when host needs to switch the daisy chain communication direction. This command is expected to receive from an opposite direction than the *[DIR_SEL]* setting during reverse communication direction procedure. See [Section 6.3.6.1.3.4](#) for details.

Although the broadcast write reverse direction is allowed to write any register value to the device, it is not recommended to write any other register setting other than the *CONTROL1[DIR_SEL]* to avoid communication collisions. Communication collisions are not detected and result in corrupted communication on the stack interface.

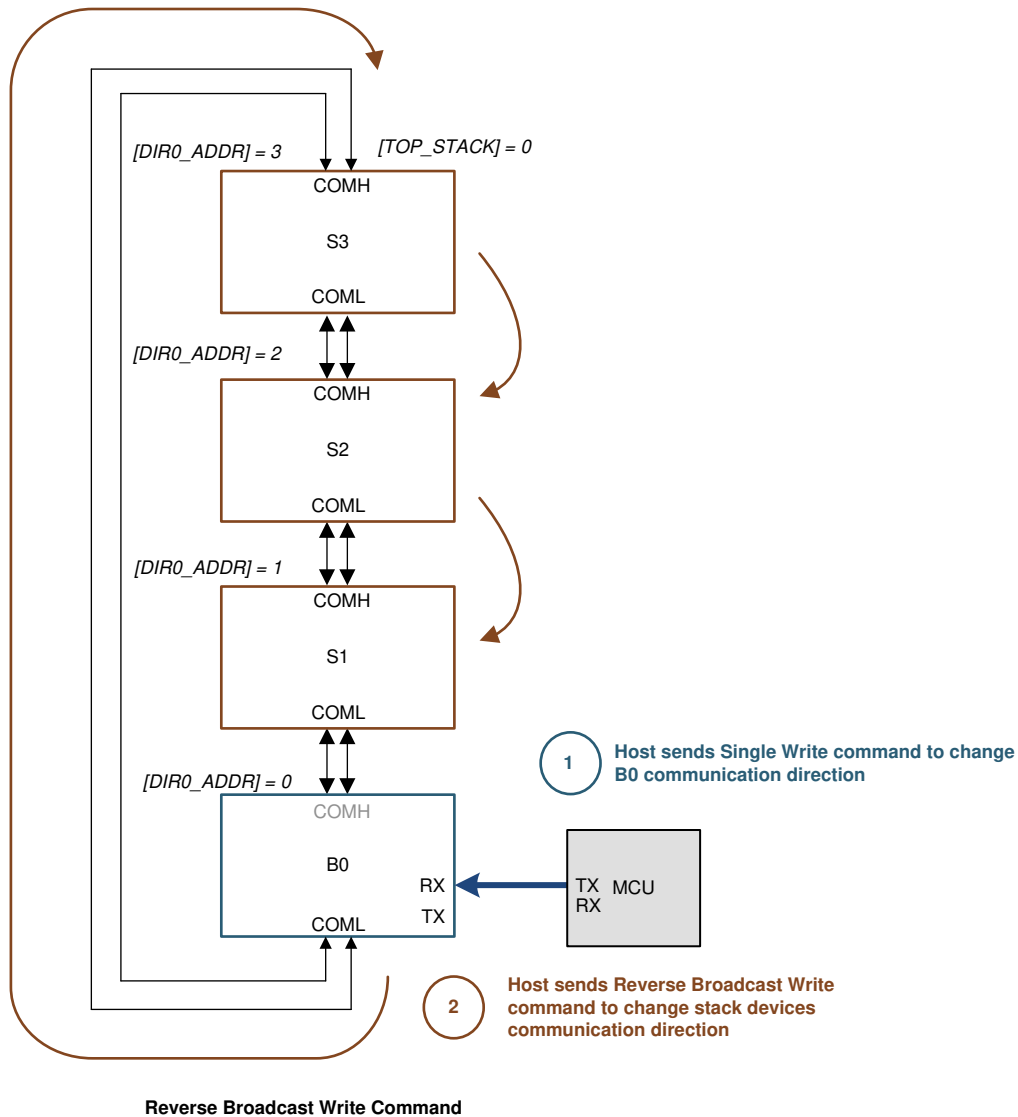


Figure 6-31. Broadcast Write Reverse Direction

Table 6-17. Broadcast Write Reverse Direction

| Broadcast Write Reverse Direction Command Sent by Host | | |
|--|--------|--|
| Set the <i>[DIR_SEL]</i> = 1 on All Devices in the Daisy Chain | | |
| Frame Field | Data | Comments |
| Initialization Byte | 0xE0 | Always 0xE0 FRAME_TYPE = 1 REQ_TYPE = 0b110 = Broadcast Write Reverse Direction DATA_SIZE = 0b000 |
| Device Address | N/A | No need to include the device address byte in command frame |
| Register Address | 0x0309 | Address of <i>CONTROL1</i> register |
| Data | 0x80 | Set <i>CONTROL1[DIR_SEL]</i> = 1 |

Table 6-17. Broadcast Write Reverse Direction (continued)

| Broadcast Write Reverse Direction Command Sent by Host | | |
|--|--------|----------|
| Set the $[DIR_SEL] = 1$ on All Devices in the Daisy Chain | | |
| Frame Field | Data | Comments |
| CRC | 0xC014 | |

6.3.6.1.2 Daisy Chain Interface

The daisy chain communication is created using differential signaling to minimize Electro-Magnetic Susceptibility (EMS) and Bulk Current Injection (BCI) immunity. The differential communication transmits true and complement data on the COM*P and COM*N pins, respectively. In a multiple device stack, there are configurations where the devices are physically located on the same board or located in entirely separate packs connected with twisted-pair wiring.

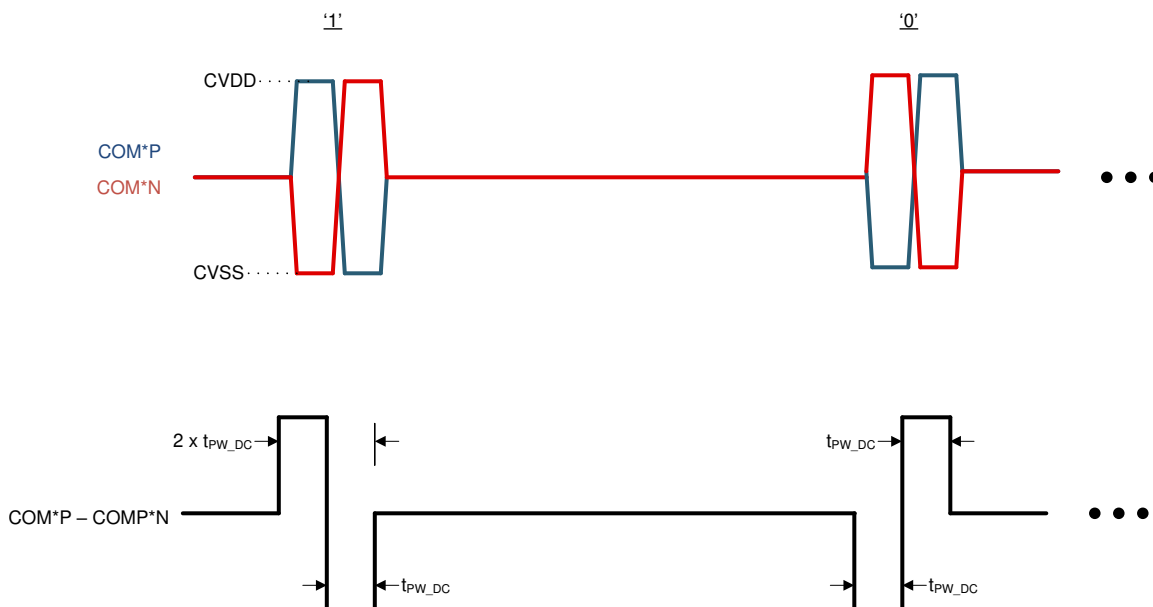
The device supports the use of transformers or capacitors to electrically isolate the signals between devices in the stack. For applications that have multiple devices on the same PCB, a single level-shifting capacitor is connected between the COMH/L pins of the devices. For extremely noisy environments, additional filtering may be necessary. For devices that are separated by cabling, additional isolation components are used. See [Section 7](#) for specific details on selecting components.

6.3.6.1.2.1 Daisy Chain Transmitter and Receiver Functionality

The daisy chain is bidirectional and half duplex, and, therefore, has a transmitter (TX) and receiver (RX) on the COMH and COML interfaces. The TX and RX functions are controlled automatically by the hardware based on the device's base/stack detection. When a WAKE ping/tone is received, the communication direction is set by $CONTROL1[DIR_SEL]$ and the $COMM_CTRL[TOP_STACK]$ configurations. See [Section 6.3.6.1.3](#) for details. Additionally, a user overwrites to take over the complete control of the COMH and COML is available under communication debug mode using the $DEBUG_CTRL_UNLOCK$, $DEBUG_COMM_CTRL1$, and $DEBUG_COMM_CTRL2$ registers. See [Section 6.5.4.14](#) for details.

6.3.6.1.2.2 Daisy Chain Protocol

The differential daisy chain (vertical) interface uses an asynchronous 13-bit byte-transfer protocol. Data is transferred LSB first and every bit is duplicated (with a complement) to ensure the transmission has no DC content.

**Figure 6-32. Daisy Chain Bit Definition**

A byte starts with a Preamble, followed by two SYNC bits, a start-of-frame bit, eight data bits starting from the LSB D0 to MSB D7 (D0 is transmitted just after State-Of-Frame and D7 comes last before the Byte Error and Postamble).

The device extracts timing information using the Preamble and SYNC bits to decode the rest of the bit value in the byte. If any of the following errors is detected, the byte is not processed and register error bit is set.

- The Preamble and SYNC bits are known values, if the decoded value has error, the *DEBUG_COMH/L_BIT[SYNC1]* = 1 depends on which COM port receives this data.
- If timing extracted from the Preamble and SYNC bits is outside of the expected range, the *DEBUG_COMH/L_BIT[SYNC2]* = 1.

Once the two valid SYNC bits are received, the additional bits are decoded and sent to the command processor. The device continues to detect any error on this byte, and if error is detected, the Byte Error (BERR) bit will be set in this byte. The *DEBUG_COMH/L_BIT[PERR]* = 1 depends on which COM port detects the error. The following condition will set the BERR bit in the byte.

- Not sufficient samples to indicate the logic level of a bit. That is, a bit is decoded as not a strong 1 or strong 0. The *DEBUG_COMH/L_BIT[BIT]* = 1 depends on which COM port detects the error.

In the meantime, each bit is still being retransmitted to the next device. If the device is unable to decode a 1 or a 0 for the bit, it will retransmit with 0 with the BERR bit set in the byte. When the new device detects the BERR bit is set to 1 in the receiving byte, it will ignore the questionable byte and set the *DEBUG_COMH/L_BIT[BERR_TAG]* = 1, indicating a byte is received with BERR. The questionable byte being ignored is likely to cause other communication errors and is likely to trigger the *DEBUG_COMH/L_BIT[PERR]* = 1 being set in the new device as well. The questionable byte continues to be retransmitted up the daisy chain with BERR set and the process continues.

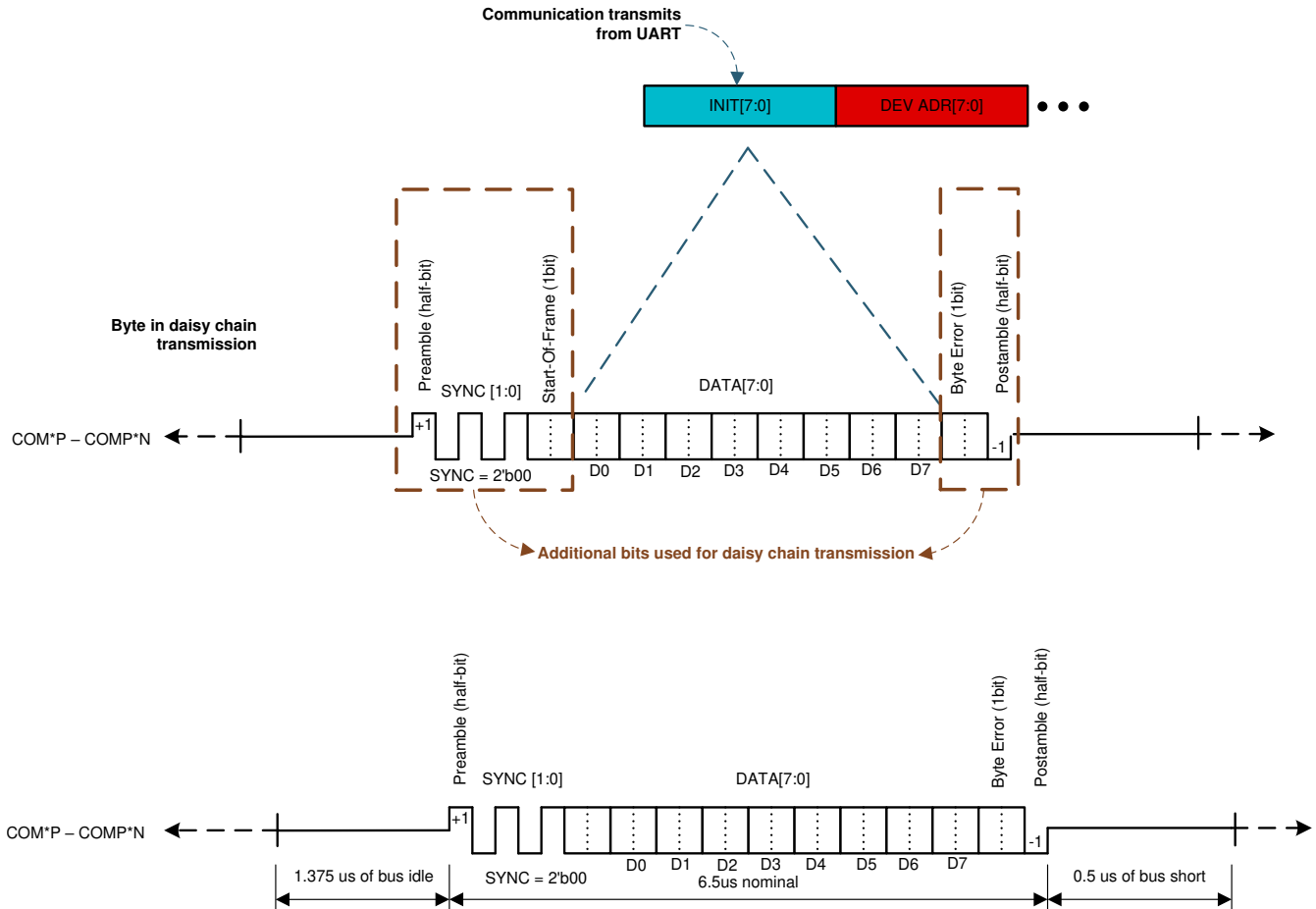


Figure 6-33. Daisy Chain Byte Definition

Table 6-18. Daisy Chain Byte Definition

| Bit Field | Description |
|-------------------------|--|
| Preamble (half-bit) | Indicates a start of transaction, signaling the receiver to start sampling. This half-bit and the following two SYNC bits are used to extra timing information. |
| SYNC[1:0] | Always 0b00. The SYNC bits are used for the digital to assess the timing and noise level on the byte, improving the detection of a 1 and 0 in a noisy environment. |
| Start-Of-Frame (1-bit) | The Start-Of-Frame (SOF) bit indicates the follow-on data byte is the initialization byte, a start of a communication transaction frame. Stack device needs this information to process the communication. For command frame transaction, the base device is responsible to set the SOF bit as it translates the UART communication to the daisy chain communication. The initialization byte contains data size information. Based on the data size information, the base device would count the number of bytes received and set the next SOF bit accordingly. The UART COMM CLEAR signal resets the UART receiver which includes the frame handling of the logic. Hence, the next byte after COMM CLEAR must have SOF set to 1 because the COMM CLEAR indicates the system clears UART and re-starts the communication. |
| Data[7:0] | The actual byte of the communication transaction frame |
| Byte Error BERR (1-bit) | Indicates an error detected in this byte. When a device receives a byte with BERR set by the lower device, it will retransmit the byte also with <i>BERR</i> = 1. Because each data bit is re-clocked from one device to the next, the next device may not detect a communication error. However, the tag of the [<i>BERR</i>] bit would indicate this communication frame has an error during its previous transaction. |
| Postamble (half-bit) | Indicates the end of transaction |

Each byte is transmitted at 2 MHz (250 ns per pulse or 500 ns per couplet). The time between each byte depends on the UART baud rate (1 Mbps in normal operation), but the byte time is always the same. The communication frame is defined with idle time between byte. In some rare cases, communication signal may not terminate cleanly, leaving ringing at the end of a byte. In such case, increasing the byte to byte gap can improve the communication robustness. The device allows additional byte gap insert between bytes in the response frame through *STACK_RESPONSE* register setting.

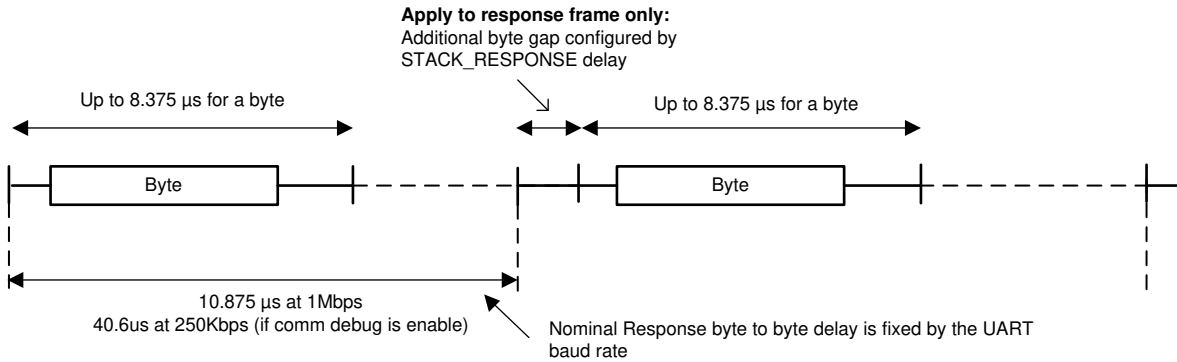


Figure 6-34. Daisy Chain Byte Transfer

6.3.6.1.3 Start Communication

From SHUTDOWN or after device reset, host follows the following steps to bring up the devices for communication.

- Host sends a WAKE ping to reset or bring the devices to ACTIVE mode. In this process, the devices in the daisy chain will configure their own COMH and COML ports based on their position in the daisy chain (base device or stack device)
 - After this step, the broadcast write is supported.
- Host performs auto-addressing to assign a device address to each device
 - After this step, the broadcast read/write and single device read/write are supported.
- Host configures the *COMM_CTRL[STACK_DEV]* and *[TOP_STACK]* bits. The Top of Stack (ToS) device will disable its transmitter of the COMH (or COML based on communication direction)
 - After this step, all commands, broadcast read/write, single device read/write, and stack read/write are supported.

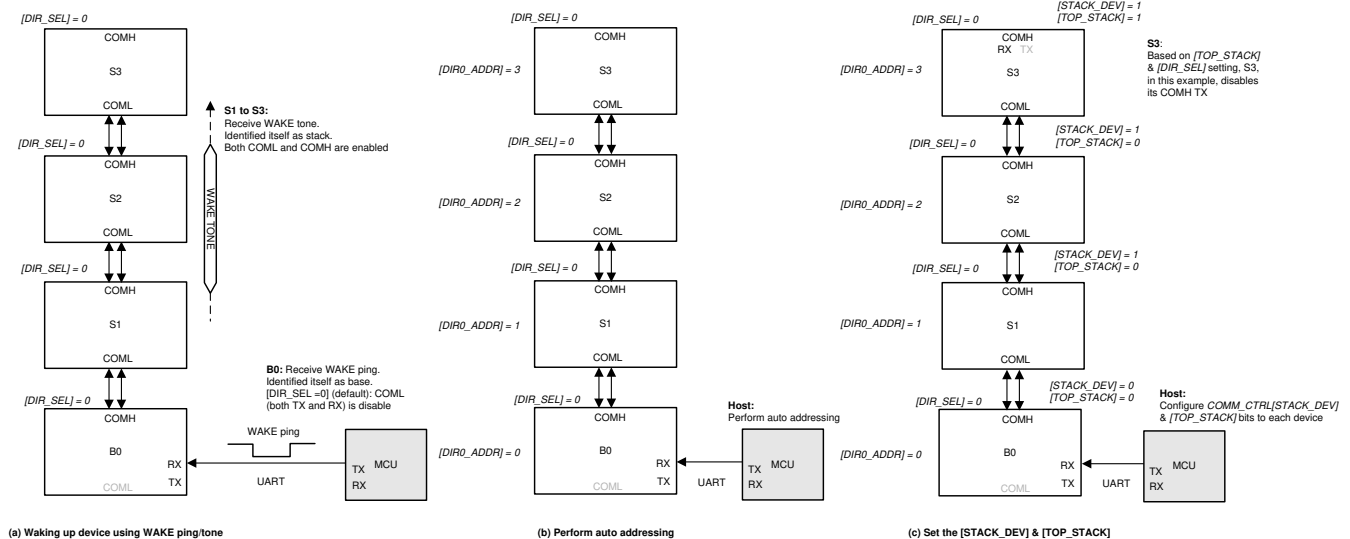


Figure 6-35. Configure Device for Communication

6.3.6.1.3.1 Identify Base and Stack

A WAKE ping/tone is used for the device to identify its position in the daisy chain.

- Base device: a device interfaces with host through UART
- Stack device: a device interfaces with the base device through COMH and COML

A base device will be woke up by a WAKE ping through RX pin, while a stack device will be woke up by WAKE tone via the COMH/COML port. Hence, a device is using a WAKE ping or WAKE tone to identify itself as base or stack. This information is stored in the AVAO_REF block which is available in all power modes and is refreshed whenever a WAKE ping/tone is received.

Using the *CONTROL1[DIR_SEL]* setting, a base device will disable the unused daisy chain ports (transmitter and receiver). If host changes the *CONTROL1[DIR_SEL]* setting, the base device will reconfigure its COMH/COML.

Note

The host starts communication at least 100 μ s after changing the *[DIR_SEL]* setting to ensure the device finishes the COMH/COML reconfiguration.

6.3.6.1.3.2 Auto-Addressing

Every device must have a unique device address for the read protocol to work. If, for any reason, two devices are assigned with the same device address, it is likely that broadcast and stack reads do not work. Additionally, single device read to the doubled address results in destroyed communication.

The default device address, assuming the device address in OTP is not programmed, is 0x00. For a host to talk to a standalone device (that is, a stack consisting with only one device), host can simply use the default 0x00 device address. Otherwise, device address follows the rules below:

- Base device address can start with any value, it is not necessary for it to be 0x00
- All device addresses must be sequential. That is, if base is 0x00, the next device must be 0x01, and next must be 0x02, and so on.

Before starting the auto-addressing procedure, all devices must be in ACTIVE mode. In this state, the device will only be able to process broadcast write command, which will be the command used for the auto-addressing procedure. Based on the *CONTROL1[DIR_SEL]* setting, the auto-addressing procedure sets up the device address to either *DIR0_ADDR* register (when *[DIR_SEL] = 0*) or *DIR1_ADDR* register (when *[DIR_SEL] = 1*).

6.3.6.1.3.2.1 Setting Up the Device Addresses

The *CONTROL1[ADDR_WR]* bit enables the auto-addressing mode. In this mode, the device turns off its COMH/COML (depends on the *[DIR_SEL]* setting) transmitter for one communication frame (following the auto-addressing procedure, that will be its own device's address), clear the *CONTROL1[ADDR_WR] = 0*. When the next communication is received (following the auto-addressing procedure, it will be the next device's address), the device will forward the communication to the next device.

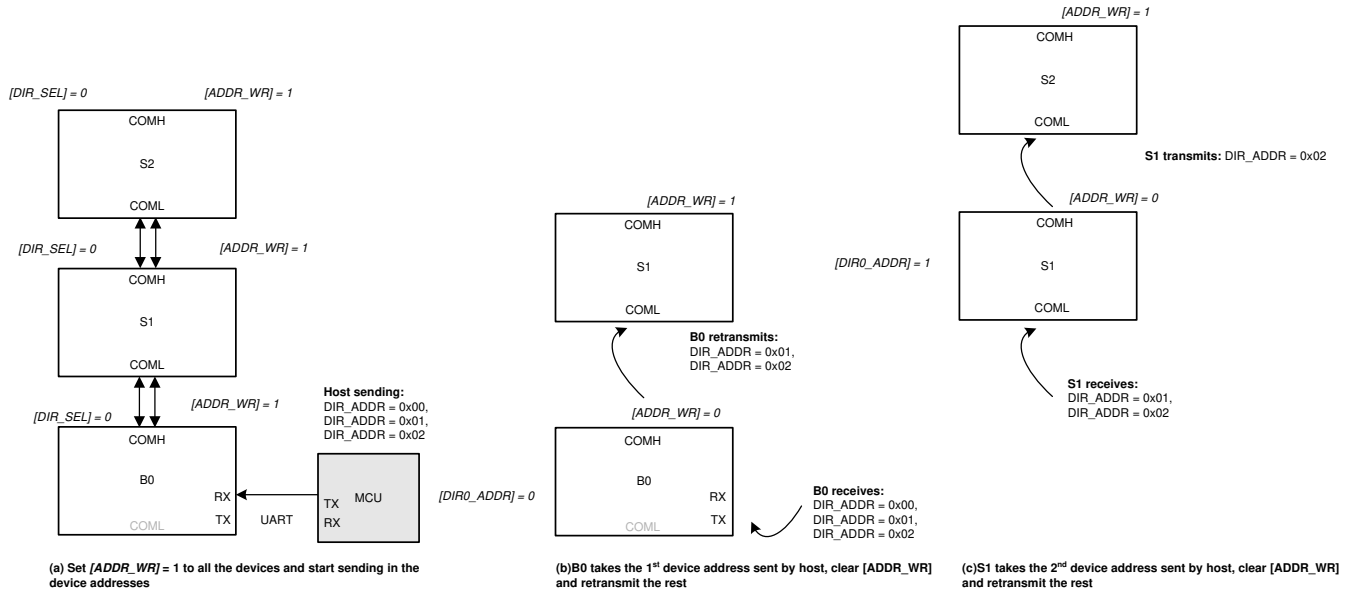


Figure 6-36. Auto-Addressing

6.3.6.1.3.2.2 Setting Up COMM_CTRL[STACK_DEV] and [TOP_STACK]

The last procedure in the auto-addressing is to configure the *COMM_CTRL[STACK_DEV]* and *[TOP_STACK]* settings. These bits need to be configured for the broadcast read and stack read/write to work properly.

- Base device: [STACK_DEV] = 0 and [TOP_STACK] = 0
- Stack devices (except ToS device): [STACK_DEV] = 1 and [TOP_STACK] = 0
- ToS device: [STACK_DEV] = 1 and [TOP_STACK] = 1

Table 6-19 shows the auto-addressing steps, assuming *CONTROL1[DIR_SEL] = 0* (that is, each device will be set up to transmit command frame sent by host from its COML to COMH).

Table 6-19. Auto-Addressing

| Step | Procedure |
|------|--|
| 1 | This step is required if a device reset has occurred before performing the auto-addressing procedure. Dummy Write to synchronize all daisy chain devices DLL (delay-locked loop) ramp in write direction. Host sends broadcast write to write 0x00 to <i>ECC_DATA1</i> to <i>ECC_DATA8</i> registers. |
| 2 | Enable auto-addressing procedure. Host sends broadcast write to set <i>CONTROL1[ADDR_WR] = 1</i> . |
| 3 | Sending in the device addresses. Host sends broadcast write to set the consecutive addresses to <i>DIR0_ADDR[ADDRESS5:0]</i> . With an example of a total of three devices in a daisy chain: <ol style="list-style-type: none"> 1. Send broadcast write to <i>DIR0_ADDR</i> register with data 0x00. 2. Send broadcast write to <i>DIR0_ADDR</i> register with data 0x01. 3. Send broadcast write to <i>DIR0_ADDR</i> register with data 0x02. |
| 4 | Set up the <i>COMM_CTRL[STACK_DEV]</i> and <i>[TOP_STACK]</i> bits for each device. <p>Option 1: Host sends single device write to each device to set the proper [STACK_DEV] and [TOP_STACK] values.</p> <p>Option 2 (less communication steps):</p> <ol style="list-style-type: none"> 1. Host sends broadcast write to set [STACK_DEV] = 1 and [TOP_STACK] = 0. 2. Host sends single device write to base device (device address 0x00 in this example) with [STACK_DEV] = 0. 3. Host send single device write to the ToS device address 0x02 in this example) with [TOP_STACK] = 1. |
| 5 | This step is required if a device reset has occurred before performing the auto-addressing procedure. Dummy read to synchronize all daisy chain devices DLL ramp in read direction. Host sends broadcast read to read <i>ECC_DATA1</i> to <i>ECC_DATA8</i> registers. Host may not receive all of the data as this step synchronizes the DLL. |

Table 6-19. Auto-Addressing (continued)

| Step | Procedure |
|------|--|
| 7 | Recommended as good practice. Use broadcast read to read <i>DIR0_ADDR</i> registers to read back all device addresses to ensure all devices are addressed properly. |
| 8 | If the dummy write and dummy read steps are performed to synchronize the DLL, it is normal if communication fault is triggered. Clear the fault registers if that is the case. |

6.3.6.1.3.2.3 Storing Device Address to OTP

The device uses *DIR0_ADDR* (used with $[DIR_SEL] = 0$) and *DIR1_ADDR* (used with $[DIR_SEL] = 1$) registers for its device address. In the auto-addressing procedure, device address is written to one of these registers and the new device address takes effect immediately.

The host has an option to program the device addresses for the $[DIR_SEL] = 0$ and 1 directions to the OTP, allowing the programmed addresses to be loaded whenever the device is reset. To program the device address to OTP, host writes the desired address to the OTP shadow registers, *DIR0_ADDR_OTP* (used when $[DIR_SEL] = 0$) and *DIR1_ADDR_OTP* (used when $[DIR_SEL] = 1$) and performs OTP programming. These two shadow registers only reflect the value programmed in OTP or use for the host to program the desired value to OTP. These two shadow registers are not the device address setting during communication. See [Section 6.3.6.3.2](#) for programming details.

6.3.6.1.3.3 Synchronize Daisy Chain DLL

When the device is reset or enters ACTIVE from SLEEP, the MCU performs a dummy write and read to synchronize the DLL on the daisy chain devices.

In the device reset case, if device address is not programmed in OTP. MCU must perform an auto-address. The DLL synchronization is part of the step. If device address is programmed in OTP, auto-address is not required after device reset. However, MCU should perform a dummy write and dummy read steps shown in [Table 6-19](#), step1 and step5 to synchronize the DLL.

When the device goes from SLEEP to ACTIVE using SLEEPtoACTIVE signal, the device is not reset. However, it is recommend to do a 1-data-byte dummy write and read to ensure robustness. Follow the similar dummy write and read steps in [Table 21](#), but only write and read to *OTP_ECC_DATAIN1*.

6.3.6.1.3.4 Ring Communication

The daisy chain communication for the device uses a Ring architecture. In this architecture, a cable break between two devices does not prevent communication to all upstream devices as in a normal non-Ring scheme. When the host detects a broken communication, the device allows the host to switch the communication direction to communicate with devices on both sides of the break. This allows for safe operation until the break in the lines is repaired.

The *CONTROL1[DIR_SEL]* controls the communication direction. The devices will reconfigure the COMH and COML ports depending on the $[DIR_SEL]$ and the $[TOP_STACK]$ settings. Auto-addressing procedure is needed to re-address the device address for the reverse communication direction.

Example to change the communication direction to $[DIR_SEL] = 1$ to the entire daisy chain:

1. Host sends Single Device Write to change the base device $[DIR_SEL] = 1$. The base device will disable its COMH and enable its COML.
2. Host sends Broadcast Write Reverse Direction to clear the *COMM_CTRL* register settings on all devices.
3. Host sends Broadcast Write Reverse Direction to change the rest of the devices' $[DIR_SEL] = 1$. In this step, the entire daisy chain set up to transmitting communication in the $[DIR_SEL] = 1$ direction (that is, each device set up to transmit command frames sent by host from its COMH to its COML).
4. Host performs auto-addressing procedure to set up device address in the *DIR1_ADDR* register. Unless the devices have been reset, host can skip the dummy read/write steps to synchronize the DLL in the auto-addressing procedure.
5. Host sets up the new Top of Stack device and the new ToS device will disable its COML transmitter.

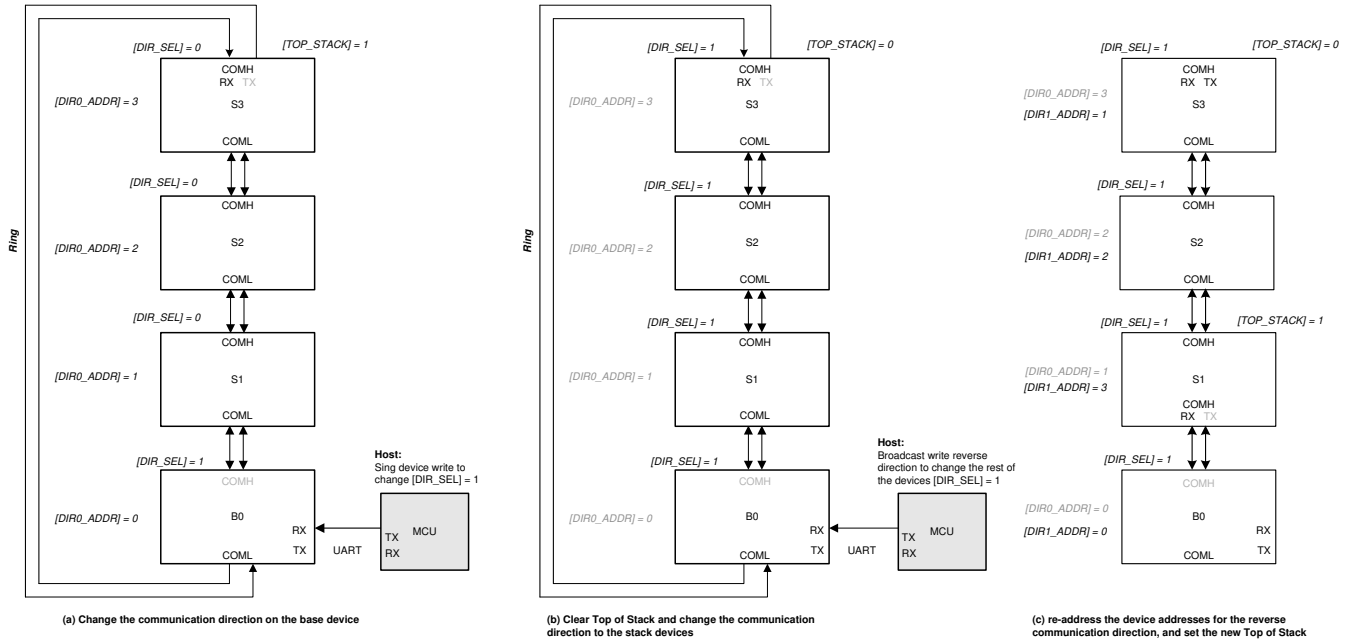


Figure 6-37. Example to Change Communication Direction in Daisy Chain

Once the device address in both communication directions is set up, host can skip auto-address step when switching communication direction.

In a broken cable case, host follows the same procedure to change the communication direction. To access all devices in the daisy chain, host will have to communicate with $[DIR_SEL] = 0$ on some devices and communicate with $[DIR_SEL] = 1$ on other devices in the daisy chain. The chain will also have two ToS devices, one for each communication direction.

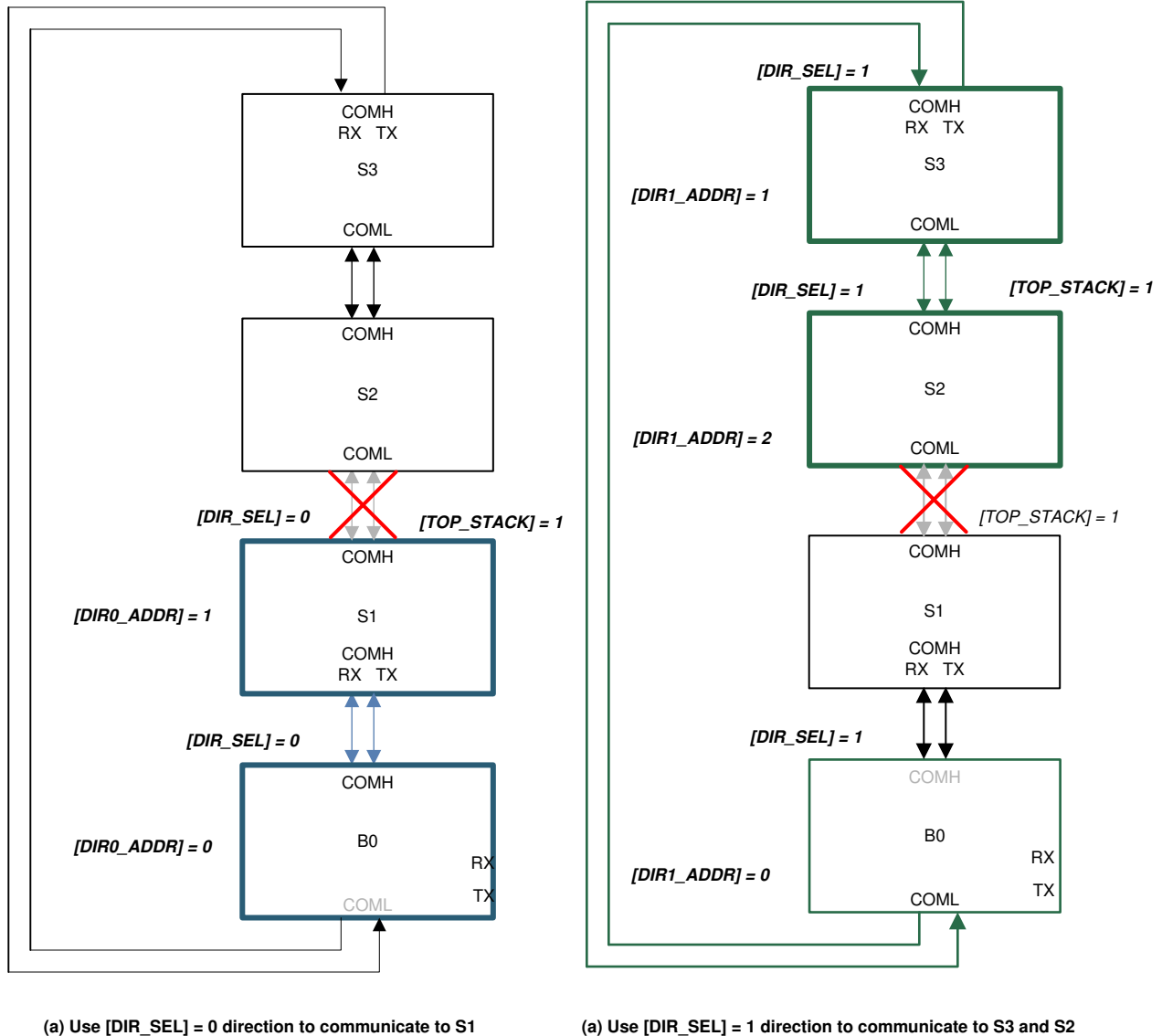


Figure 6-38. Using Ring Architecture to Access All Devices in a Broken Cable Case

6.3.6.1.4 Communication Timeout

There are two programmable communication timeout thresholds, CTS timer and CTL timer, that monitor the absence of a valid frame from either UART or daisy chain communication. A valid frame is defined as any frame (response or command) that does NOT contain any errors that prevent the frame from being processed. The communication timeouts are only actively counting while in ACTIVE mode. The counters are disabled and reset during SHUTDOWN mode. In SLEEP mode, the last counter values are held frozen.

6.3.6.1.4.1 Short Communication Timeout

The short communication timeout acts like an alert to the host when triggered. The timeout period is programmable through the *COMM_TIMEOUT_CONF[CTS_TIME2:0]* bits. If enabled, the timer is reset every time a valid response or command frame is received. If the timer expires, the *FAULT_SYS[CTS]* bit is set.

6.3.6.1.4.2 Long Communication Timeout

The long communication timeout allows the host to put the device in SLEEP or SHUTDOWN mode for power saving. The timeout period is programmable through *COMM_TIMEOUT_CONF[CTL_TIME2:0]* bits. If enabled,

the timer is reset every time a valid response or command frame is received. If the timer expires, host can choose one of the following actions through *COMM_TIMEOUT_CONF[CTL_ACT]* bit.

- Set *FAULT_SYS[CTL]* = 1 and enter SLEEP mode.
- Enter SHUTDOWN mode.

6.3.6.1.5 Communication Debug Mode

The device provides a communication debug mode to ease the initial development phase. To enter this debug mode, host writes an unlock code 0xA5 to register *DEBUG_CTRL_UNLOCK*. Once the debug mode is unlocked, the settings in *DEBUG_COMM_CTRL1* and *DEBUG_COMM_CTRL2* become effective.

To exit the debug mode simply write any value but 0xA5 (for example, writing 0x00) to the *DEBUG_CTRL_UNLOCK*. The COMH, COML, and UART will return to their normal operation status regardless of the settings in the *DEBUG_COMM_CTRL1* and *DEBUG_COMM_CTRL2* registers.

Once the communication debug mode is entered, the host gains control of the following:

Table 6-20. Communication Debug Mode Functions

| Control Function | Enable Bit | Description |
|--|---------------------------|--|
| Full COMH/L transmitter and receiver control | [<i>USER_DAISSY_EN</i>] | If [<i>USER_DAISSY_EN</i>] = 1, device will enable or disable its COMH/L transmitter and receiver based on the <i>DEBUG_COMM_CTRL2</i> register setting. If [<i>USER_DAISSY_EN</i>] = 0, COMH/L will be in its normal operation status even under communication debug mode. |
| Mirror out the data in daisy chain onto UART | [<i>USER_UART_EN</i>] | If [<i>USER_UART_EN</i>] = 1, host can set [<i>UART_MIRROR_EN</i>] = 1 to instruct the device to translate the daisy chain onto the UART, allowing host to read the data being received or forwarded in the daisy chain from the UART interface. Data will be presented in UART communication frame format. For stack devices, the UART TX is disabled by default. To use this feature, host also sets [<i>UART_TX_EN</i>] = 1. If [<i>USER_UART_EN</i>] = 0, any UART related debug functions are disabled. The UART will be in its normal operation status regardless of the [<i>UART_MIRROR_EN</i>] and [<i>UART_TX_EN</i>] settings. |
| Slow down UART baud rate to 250 kbps | [<i>USER_UART_EN</i>] | If [<i>USER_UART_EN</i>] = 1, host can set [<i>UART_BAUD</i>] = 1 to change the UART baud rate to 250 kbps. This will result in slow throughput rate on the daisy chain. If [<i>USER_UART_EN</i>] = 0, UART baud rate will stay on 1 Mbps regardless of the [<i>UART_BAUD</i>] setting. |

The *DEBUG_COMM_STAT* register has status bits indicating if UART and COMH/L are under user or hardware (device) control. The register also indicates the status of the COMH/L transmitter and receiver. This debug status register is updated per device status and is readable with or without the communication debug mode enabled.

In fact, the read-only debug registers are all readable in ACTIVE mode without communication debug mode enabled. Most of them are lower level communication fault status registers to provide extra information in a communication failure event like the *DEBUG_UART**, *DEBUG_COMH**, and *DEBUG_COML** registers. See [Section 6.3.6.2](#) and [Section 6.5.4](#) for more details.

6.3.6.1.6 Multidrop Configuration

A multidrop configuration is a configuration of multiple devices in a system communicating through UART to the host system. There is no daisy chain communication between devices. When [*MULTIDROP*] = 1, the device COMH and COML ports are disabled. All the communication protocols, single device read/write, broadcast read/write, stack read/write, and a reverse broadcast write are still supported as in daisy chain configuration (that is, [*MULTIDROP*] = 0). However, in a multidrop configuration, it is unlikely to have a use of the stack and reverse broadcast commands. If broadcast command is used, it is still required to set up the devices with sequential device address and set the [*TOP_STACK*] bit on the device with highest device address. The device with [*TOP_STACK*] = 1 initiates the data return when a broadcast read command is received, and the device with one lower device address will respond next, as in a daisy chain communication. Additionally, a *COMM_CLR* must be used before every frame to ensure consistent communication in multidrop configuration.

6.3.6.1.7 SPI Controller

The GPIO4 thru GPIO7 are configurable as a SPI controller interface when $GPIO_CONF1[SPI_EN] = 1$. The SPI controller includes four I/Os:

- SCLK: SPI clock, generated by the device and is used for synchronization
- MOSI: Controller data output, driven by the device to output data to target
- MISO: Controller data input, detecting data from target
- SS: target select, driven by the device during SPI communication.

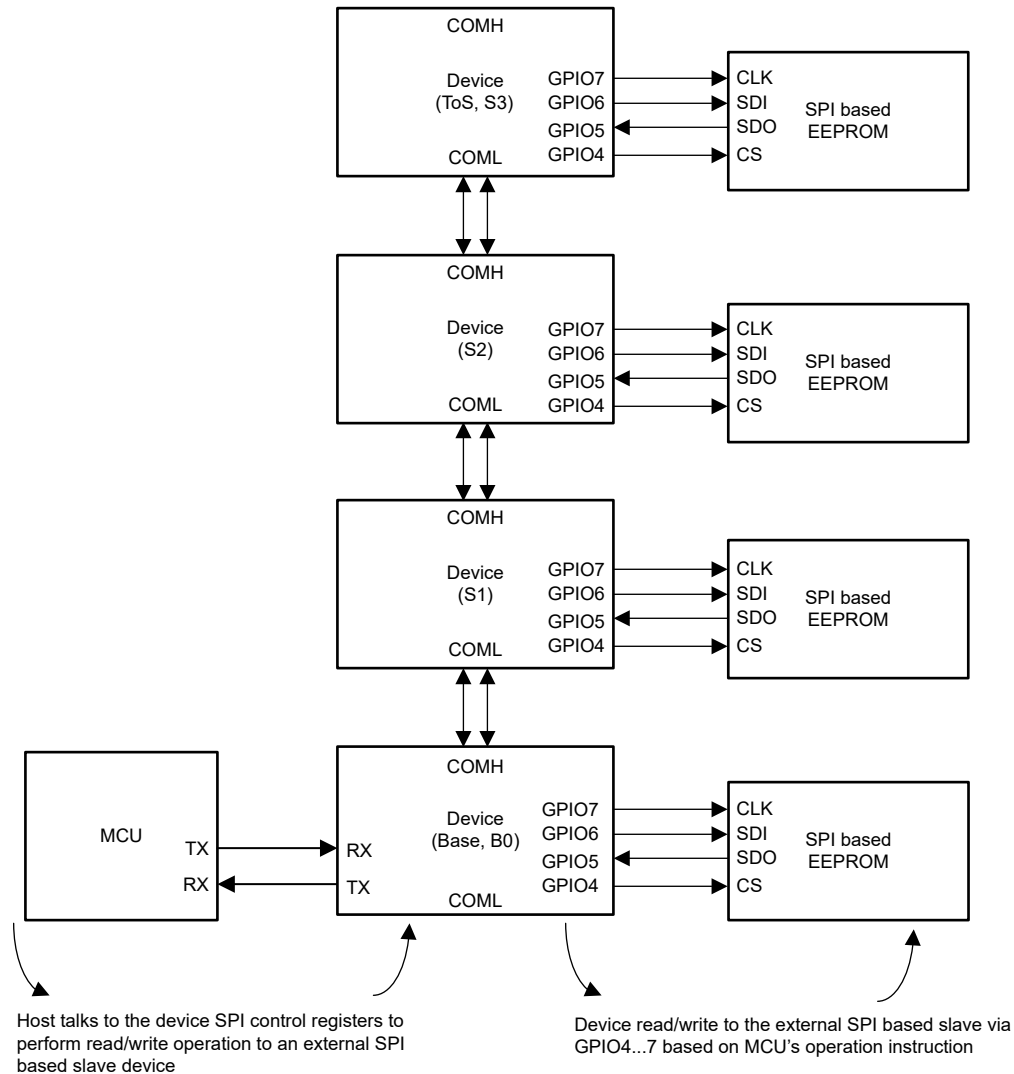


Figure 6-39. SPI Controller Stack Configuration

The $SPI_CONF[CPOL]$ (clock polarity) and $[CPHA]$ (clock phase) define the SPI clock format. The $[CPOL]$ is defined if the SPI clock is inverted or non-inverted. The $[CPHA]$ is defined if the MISO and MOSI are sampled on the leading (first) clock edge or on the trailing (second) clock edge, regardless of whether that clock edge is rising or falling. The $SPI_CONF[NUMBER4:0]$ defines how many bits the transaction is (1-bit to 24-bit transaction).

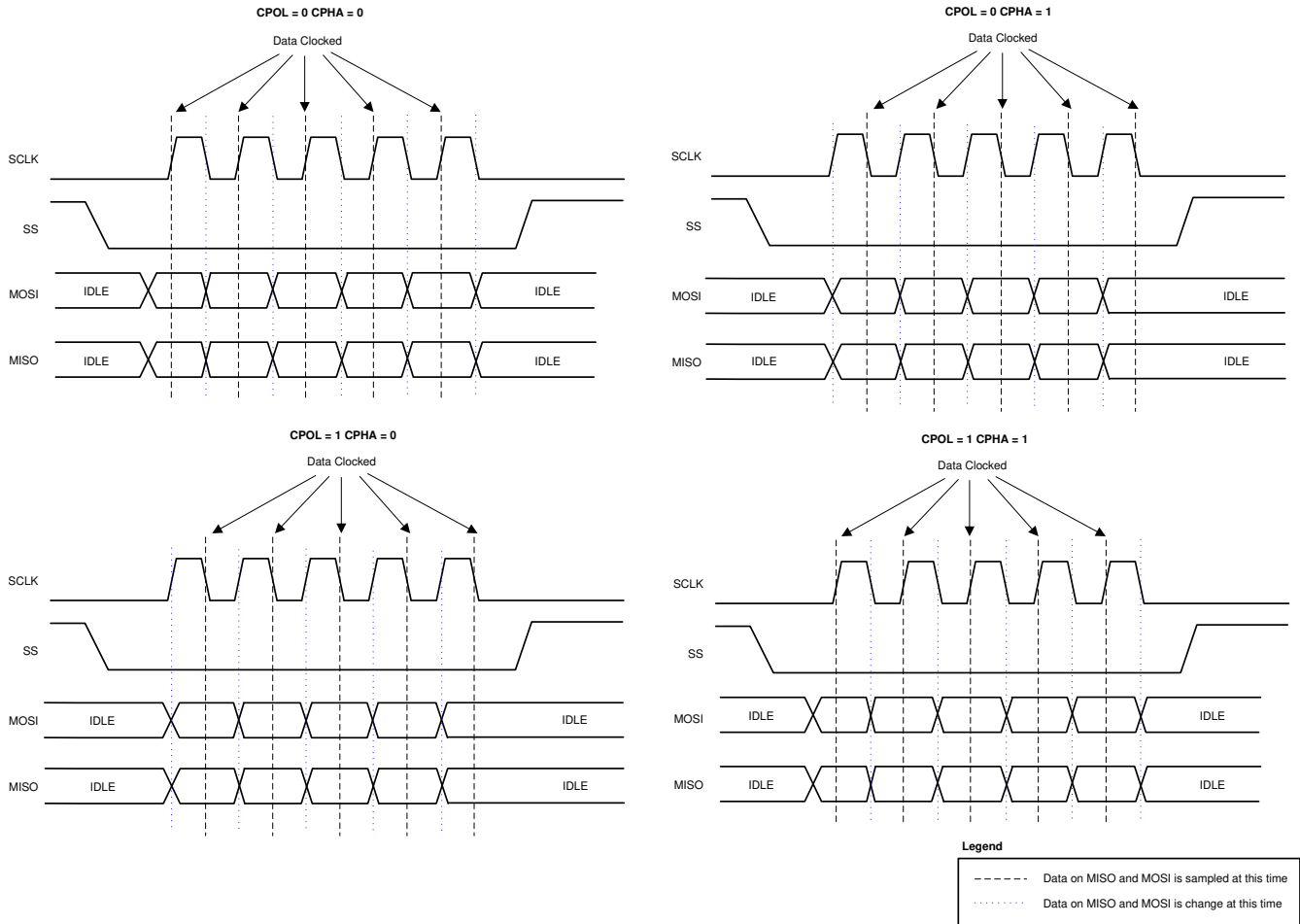


Figure 6-40. SPI Controller CPOL and CPHA

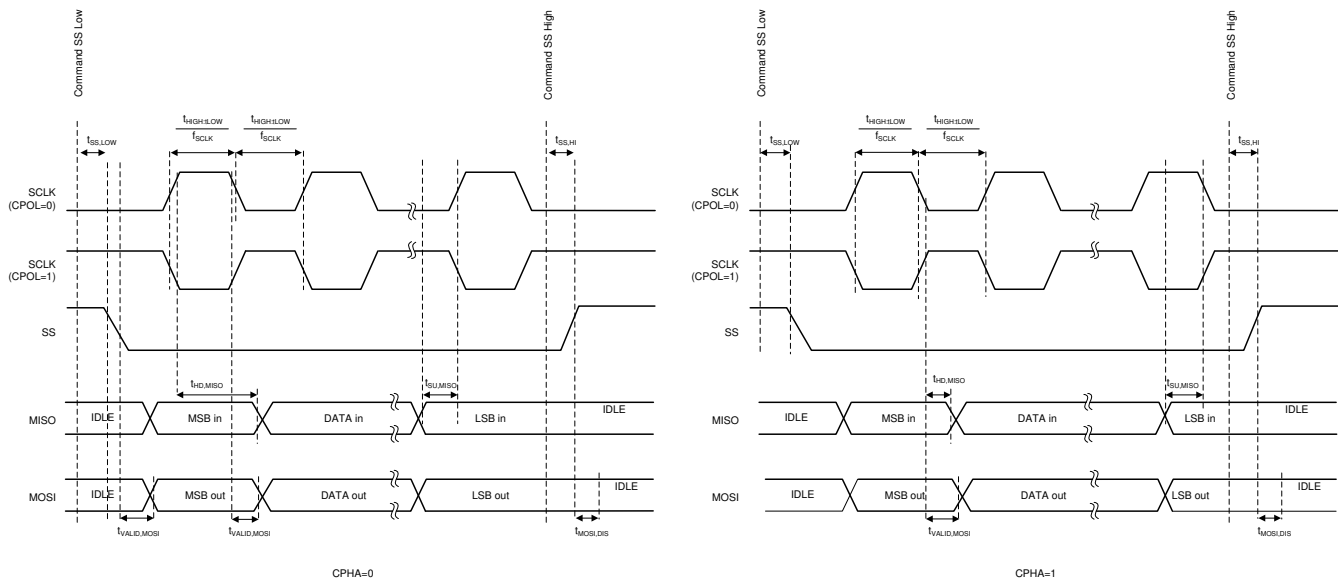


Figure 6-41. SPI Controller Timing Diagram

Table 6-21. Write to External SPI Target

| Step | Description |
|------|--|
| 1 | Configure the SPI clock polarity, clock phase, number of bit transactions: a. Write to <i>SPI_CONF</i> register to configure SPI communication |
| 2 | Write the data (from 1 to 24 bits, specified in the <i>SPI_CONF[NUMBIT4:0]</i> setting): a. Set up the data to send to SPI target to the <i>SPI_TX1</i> to <i>SPI_TX3</i> registers b. <i>SPI_TX1</i> is the LSByte and <i>SPI_TX3</i> is MSByte |
| 3 | Select the target (assuming active low) and execute the SPI write action: a. Send <i>SPI_EXE</i> register = 0x01 (that is, [<i>SS_CTRL</i>] = 0 and [<i>SPI_GO</i>] = 1) |
| 4 | Wait for the SPI communication to complete |
| 5 | Deselect the SS port (assuming active low, so deselecting means pull the SS pin high): a. Send <i>SPI_EXE</i> register = 0x02 (that is, [<i>SS_CTRL</i>] = 1 and [<i>SPI_GO</i>] = 0) |

Table 6-22. Read from External SPI Target

| Step | Description |
|------|---|
| 1 | Configure the SPI clock polarity, clock phase, number of bit transactions: a. Write to <i>SPI_CONF</i> register to configure SPI communication |
| 2 | Select the target and execute the SPI communication: a. Send <i>SPI_EXE</i> register = 0x01 (that is, [<i>SS_CTRL</i>] = 0 and [<i>SPI_GO</i>] = 1) |
| 3 | Wait for the data transaction to complete |
| 4 | Read the data (from 1 to 24 bits, specified in the <i>SPI_CONF[NUMBIT4:0]</i> setting): a. Read data from SPI target from the <i>SPI_RX1</i> to <i>SPI_RX3</i> registers b. <i>SPI_TX1</i> is the LSByte and <i>SPI_TX3</i> is MSByte |
| 5 | Deselect the SS port (assuming active low, so deselecting means pull the SS pin high): a. Send <i>SPI_EXE</i> register = 0x02 (that is, [<i>SS_CTRL</i>] = 1 and [<i>SPI_GO</i>] = 0) |

6.3.6.1.8 SPI Loopback

The SPI controller has a loopback function that is enabled using the *DIAG_COMM_CTRL[SPI_LOOPBACK]* bit. When enabled, the byte in the *SPI_TX** registers are clocked directly to the MISO pin of the SPI controller to verify the SPI controller functionality. This is performed internally, so no external connection is needed to run this test. This verifies that the SPI function is working correctly. The *SPI_CFG*, *SPI_TX**, and *SPI_EXE* registers are written as a normal SPI transaction, but the external pins do not toggle during this mode. That is, the external pins stay static in their last state and do not change state during the loopback operation.

The expected result of the test is that the byte in the *SPI_TX** register is read into the *SPI_RX** register. The SS pin is latched to the setting in *SPI_EXE[SS_CTRL]* that existed when the LOOPBACK mode was enabled. The CPHA and CPOL parameters must be set before entering LOOPBACK mode to ensure proper operation. Changing the CPOL or CPHA parameters while in LOOPBACK mode may result in errant pulses on the SPI outputs and is not recommended.

6.3.6.2 Fault Handling

6.3.6.2.1 Fault Status Hierarchy

The device monitors multiple types of faults such as:

- Battery cell monitoring through the hardware protector, like cell OV/UV, cell OT/UT, and so on
- System operation driven like device reset, communication timeout, thermal warning, and so on
- Command-based diagnostic check related like the various comparison through the main and AUX ADCs, BIST run, and so on
- Automatic diagnostic check running in the background like the internal power supplies, OTP CRC, and so on
- Communication fault.

Each bit in the *FAULT_SUMMARY* register represents a group of faults which are stored in one or more lower level fault registers. The *FAULT_SUMMARY* register represents the highest hierarchy level of fault status detected by the device. Host system can periodically poll the *FAULT_SUMMARY* register to check the fault status and only read the lower level fault registers if needed (for example, if *FAULT_SUMMARY[FAULT_OVUV]*

= 1, host can read *FAULT_OV1/2* and *FAULT_UV1/2* registers to determine which cell channel triggered the fault).

Table 6-23 shows which lower level register corresponds to the *FAULT_SUMMARY* register bit. The description of the register is covered in the Section 6.5.

Table 6-23. Low-Level Fault Registers

| FAULT_SUMMARY Bit Name | FAULT_PROT | FAULT_COMP_ADC | FAULT_OTP | FAULT_COMM | FAULT_OTUT | FAULT_OVUV | FAULT_SYS | FAULT_PWR |
|---------------------------|--------------------------------|--------------------------|---------------------------------|-----------------------------------|------------------|------------------|------------------|-------------------|
| Lower level register name | <i>FAULT_PROT</i> ₁ | <i>FAULT_COMP_GPIO</i> | <i>FAULT_OTP</i> ⁽¹⁾ | <i>FAULT_COMM1</i> ⁽¹⁾ | <i>FAULT_OT</i> | <i>FAULT_OV1</i> | <i>FAULT_SYS</i> | <i>FAULT_PWR1</i> |
| | <i>FAULT_PROT</i> ₂ | <i>FAULT_COMP_VCCB1</i> | | <i>FAULT_COMM2</i> ⁽¹⁾ | <i>FAULT_UT</i> | <i>FAULT_OV2</i> | | <i>FAULT_PWR2</i> |
| | | <i>FAULT_COMP_VCCB2</i> | | <i>FAULT_COMM3</i> | | <i>FAULT_UV1</i> | | <i>FAULT_PWR3</i> |
| | | <i>FAULT_COMP_VCOW1</i> | | | <i>FAULT_UV2</i> | | | |
| | | <i>FAULT_COMP_VCOW2</i> | | | | | | |
| | | <i>FAULT_COMP_CBOW1</i> | | | | | | |
| | | <i>FAULT_COMP_CBOW2</i> | | | | | | |
| | | <i>FAULT_COMP_CBFET1</i> | | | | | | |
| | | <i>FAULT_COMP_CBFET2</i> | | | | | | |
| | | <i>FAULT_COMP_MISC</i> | | | | | | |

(1) Some of the bits in the *FAULT_COMM1/2* and *FAULT_OTP* registers have a lower level of fault information than shown in the *DEBUG_COMM** and *DEBUG_OTP* registers.

6.3.6.2.1.1 Debug Registers

The *DEBUG_COMM** and *DEBUG_OTP* registers are a form of fault status showing lower hierarchy level of fault information for some of the bits in *FAULT_COMM1*, *FAULT_COMM2*, and *FAULT_OTP*.

Table 6-24 shows the hierarchy relationship. See Section 6.5 for the register description details.

Table 6-24. Debug Registers

| Low-level Fault Register | Low-level Register Bit | | Associated DEBUG Registers |
|--------------------------|--|---|--------------------------------|
| <i>FAULT_COMM1</i> | [<i>UART_RC</i>] | Fault related to received command frame from UART | <i>DEBUG_UART_RC</i> |
| | [<i>UART_RR</i>] [<i>UART_TR</i>] | Fault related to received or transmitted response frame from UART | <i>DEBUG_UART_RR_TR</i> |
| | [<i>COMH_BIT</i>] | Fault related to error in a byte from COMH | <i>DEBUG_COMH_BIT</i> |
| <i>FAULT_COMM2</i> | [<i>COMH_RC</i>] | Fault related to received command frame from COMH | <i>DEBUG_COMH_RC</i> |
| | [<i>COMH_RR</i>] [<i>COMH_TR</i>] | Fault related to received or transmitted response frame from COMH | <i>DEBUG_COMH_RR_TR</i> |
| | [<i>COML_BIT</i>] | Fault related to error in a byte from COML | <i>DEBUG_COML_BIT</i> |
| | [<i>COML_RC</i>] | Fault related to received command frame from COML | <i>DEBUG_COML_RC</i> |
| | [<i>COML_RR</i>] [<i>COML_TR</i>] | Fault related to received or transmitted response frame from COML | <i>DEBUG_COML_RR_TR</i> |
| | <i>FAULT_OTP</i> | [<i>SEC_DET</i>] | Single error correction in OTP |
| [<i>DED_DET</i>] | | Double error correction in OTP | <i>DEBUG_OTP_DED_BLK</i> |

6.3.6.2.2 Fault Masking and Reset

6.3.6.2.2.1 Fault Masking

When a device detects a fault, the corresponding low-level register bit, including the one in the related bit in the *DEBUG_** registers is set. Based on the fault hierarchy relationship, the fault will be reflected in the *FAULT_SUMMARY* register.

A group of faults can be masked, which the related low-level register flag will still be set, but the fault will not be reflected to the corresponding *FAULT_SUMMARY* register. The faults can be masked through the *FAULT_MSK1* and *FAULT_MSK2* registers.

For example, to mask the *FAULT_SUMMARY*[*FAULT_OTUT*] being set, host sets *FAULT_MSK1*[*MSK_OT*] = 1 and [*MSK_UT*] = 1.

When fault is masked, it will also prevent the device from asserting the NFAULT pin when the masked fault occurs. See [Section 6.3.6.2.3](#) for details on NFAULT signal.

Table 6-25. Fault Masking

| | Masking Bit Name | Related Low-level Register(s) Affected | <i>FAULT_SUMMARY</i> Register Bit That Will Be Masked |
|-------------------|----------------------------|---|---|
| <i>FAULT_MSK1</i> | [<i>MSK_PROT</i>] | <i>FAULT_PROT</i> * | [<i>FAULT_PROT</i>] |
| | [<i>MSK_UT</i>] | <i>FAULT_UT</i> | [<i>FAULT_OTUT</i>] |
| | [<i>MSK_OT</i>] | <i>FAULT_OT</i> | |
| | [<i>MSK_UV</i>] | <i>FAULT_UV</i> * | [<i>FAULT_OVUV</i>] |
| | [<i>MSK_OV</i>] | <i>FAULT_OV</i> * | |
| | [<i>MSK_COMP</i>] | <i>FAULT_COMP_*</i> | [<i>FAULT_COMP</i>] |
| | [<i>MSK_SYS</i>] | <i>FAULT_SYS</i> | [<i>FAULT_SYS</i>] |
| | [<i>MSK_PWR</i>] | <i>FAULT_PWR</i> * | [<i>FAULT_PWR</i>] |
| <i>FAULT_MSK2</i> | [<i>MSK_OTP_CRC</i>] | <i>FAULT_OTP</i> [<i>CUST_CRC</i>][<i>FACT_CRC</i>] | [<i>FAULT_OTP</i>] |
| | [<i>MSK_OTP_DATA</i>] | All non-CRC bits in <i>FAULT_OTP</i> , <i>DEBUG_OTP_*</i> | |
| | [<i>MSK_COMM3_FCOMM</i>] | <i>FAULT_COMM3</i> [<i>FCOMM_DET</i>] | [<i>FAULT_COMM3</i>] |
| | [<i>MSK_COMM3_FTONE</i>] | <i>FAULT_COMM3</i> [<i>FTONE_DET</i>] | |
| | [<i>MSK_COMM3_HB</i>] | <i>FAULT_COMM3</i> [<i>HB_FAIL</i>][<i>HB_FAST</i>] | |
| | [<i>MSK_COMM2</i>] | <i>FAULT_COMM2</i> , <i>DEBUG_COMH_*</i> , <i>DEBUG_COML_*</i> | [<i>FAULT_COMM2</i>] |
| | [<i>MSK_COMM1</i>] | <i>FAULT_COMM1</i> , <i>DEBUG_UART_*</i> | [<i>FAULT_COMM1</i>] |

6.3.6.2.2.2 Fault Reset

Once fault is detected, the fault status bit is latched until cleared using the reset bit. Similar to fault masking, when the specific fault reset bit is set, the associated low-level fault registers, including the *DEBUG_** registers are cleared. The corresponding bit in the *FAULT_SUMMARY* register will clear if all its associated low-level registers are cleared. If the fault condition persists and the reset bit is written, the fault status bit is not reset. The fault indicator cannot be reset until the underlying fault condition is eliminated.

The fault is reset through the *FAULT_RST1* and *FAULT_RST2* registers; the fault reset bits are structured in the same corresponding fault status registers as the fault masking bits.

6.3.6.2.3 Fault Signaling

Host can acquire the fault status with the following methods:

- Constantly polling the *FAULT_SUMMARY* status on each device in the daisy chain. If *FAULT_SUMMARY* is non-zero, read the low-level fault status registers to obtain more information.
- Enable fault status to pass down the daisy chain to the base device. Enable base device's NFAULT pin to be asserted when the *FAULT_SUMMARY* is non-zero in any of the devices in the daisy chain. Host monitors NFAULT. When NFAULT is triggered, host does a broadcast read on the *FAULT_SUMMARY* to determine which device(s) is at fault.

When using the NFAULT pin in the base device to signal the host under a fault detection, the stack devices have to transfer their fault status information to the base device. The information is transmitted through COMH/L through the same communication cables. In ACTIVE mode, each device embeds the fault status to the communication when a response frame is forwarded. In SLEEP mode, or using Heartbeat and Fault Tone in SLEEP mode.

The NFAULT pin can be masked by configuring *DEV_CONF*[*NFAULT_EN*] = 0. When NFAULT is disabled, the device will set the corresponding flag in *FAULT_SUMMARY* register but will not assert NFAULT.

6.3.6.2.3.1 Fault Status Transmitting in ACTIVE Mode

In ACTIVE mode, stack devices can embed their fault status before retransmitting a response frame if $DEV_CONF[FCOMM_EN] = 1$. When the $[FCOMM_EN] = 1$, the stack devices repurpose the SOF bit in the response frame's device address byte, register address bytes (both high and low address bytes) to a fault status bit instead. See Figure 6-42. This will be referred to as fault status bits in the rest of this section.

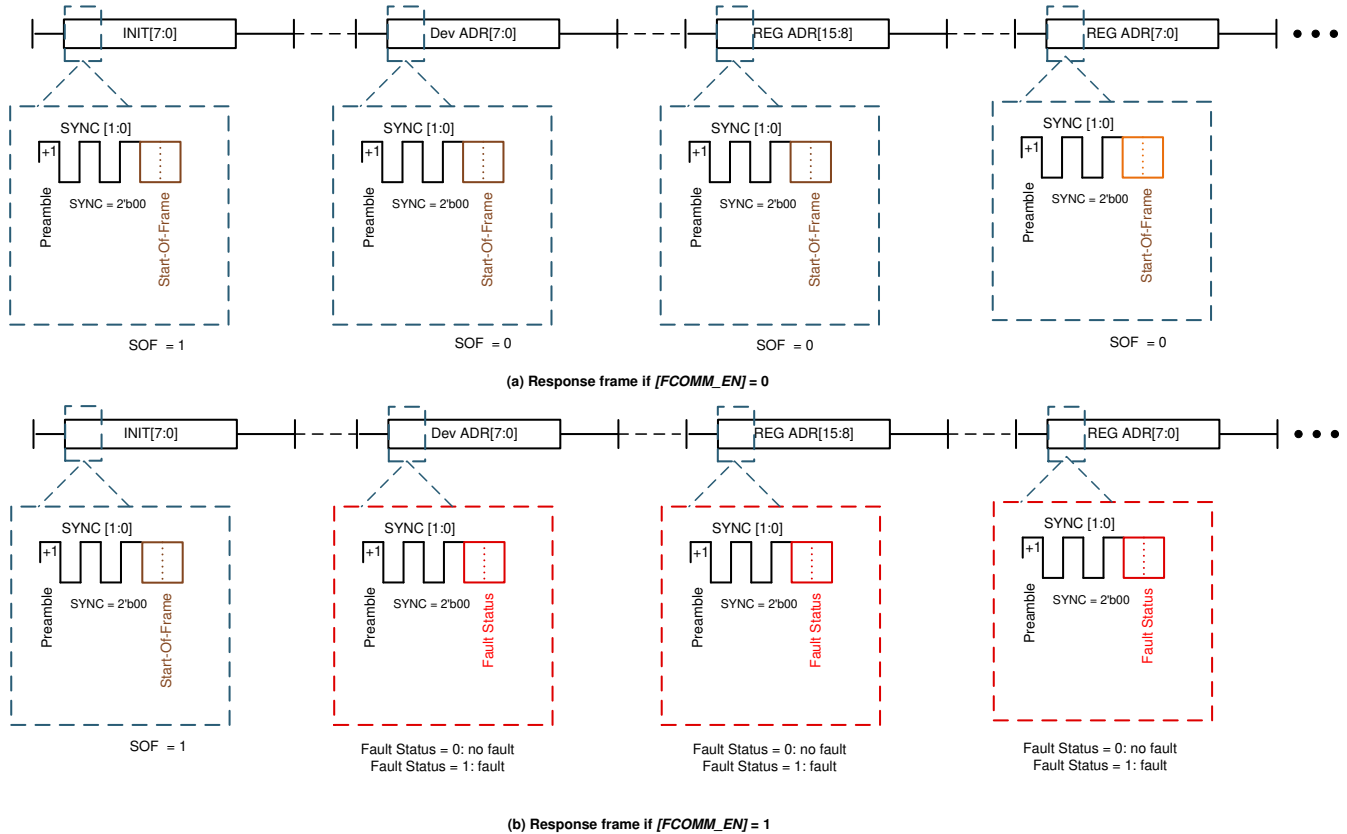


Figure 6-42. Embed Fault Status in Communication Response Frame

To pass on the fault status of the stack devices, the host sends a broadcast read or sends a single device read to the ToS device. Both types of reads will result in response frames passing through every device in the daisy chain, giving each device an opportunity to OR their fault status to the fault status bits in the response frame.

An example of a response frame going through a daisy chain from a single device read command to the top device is shown in Figure 6-43.

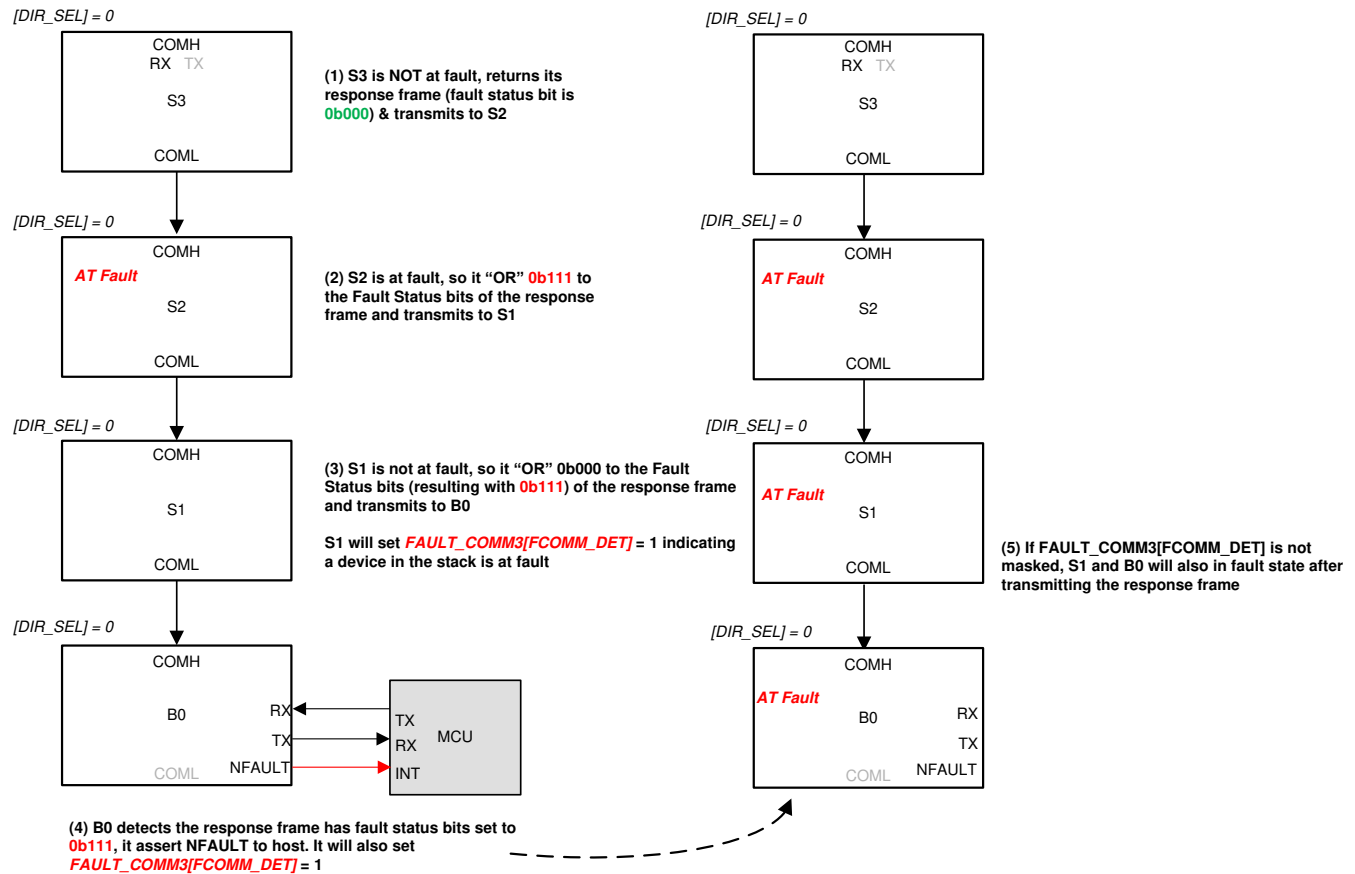


Figure 6-43. Transfer Fault Status in ACTIVE Mode (Respond to a Single Device Read)

When a device has no fault, it will OR the fault status bits with 0b000; otherwise, it will OR the fault status bits with 0b111. Hence, if a fault exists in any device in the daisy chain, the fault status bits will be 0b111. For the base device to assert the NFAULT pin, it requires at least two bits of the fault status bits to be 1.

Additionally, when a device detects a response frame with at least two of the fault status bits being 1, the device will also set the $FAULT_COMM3[FCOMM_DET] = 1$. If this fault is not masked, the device will be in fault state as well. Next time a response frame is transmitted, this device will OR the fault status bits with 0b111.

Host performs a broadcast read to detect which device in the daisy chain is at fault and what type of fault.

6.3.6.2.3.2 Fault Status Transmitting in SLEEP Mode

In SLEEP mode, the following fault detections are still active:

- Customer and Factory OTP shadow registers CRC check
- Device thermal warning
- Power supplies OV, UV, and oscillation detection
- If OVUV protectors are enabled, cell OV and UV detection.
- If OTUT protectors are enabled, thermistors OT and UT detection.

Because communication is not available in SLEEP mode, the device provides an option to transmit the fault status through Heartbeat (device in no fault state) and Fault (device in fault state) Tones. These tones are transmitted in the same direction as a communication command frame, which is based on the $CONTROL1[DIR_SEL]$ setting. For the tone signal to return back to the base device (so NFAULT can be triggered if needed), a Ring architecture must be used to support transmitting fault status in SLEEP mode.

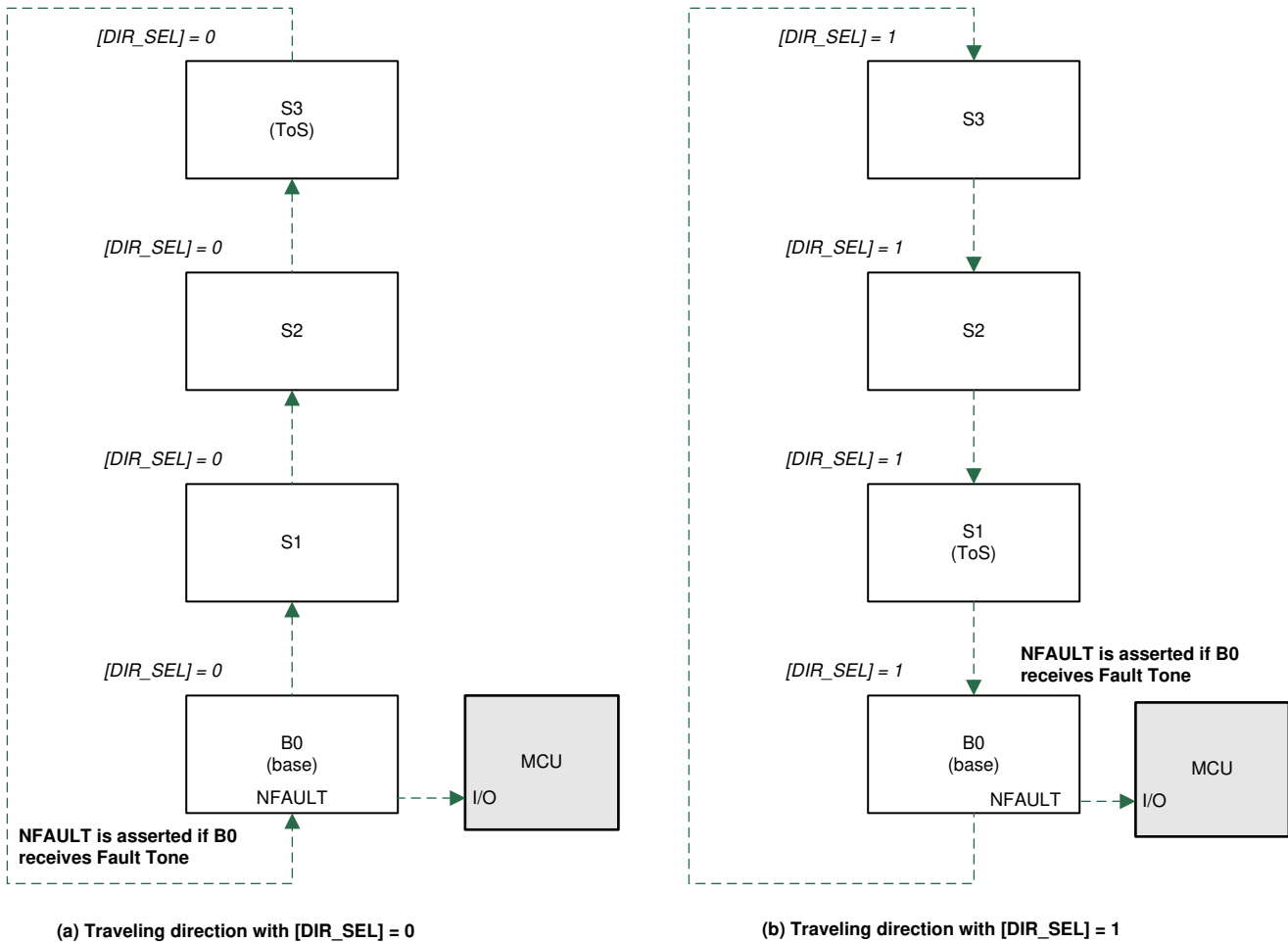


Figure 6-44. Heartbeat or Fault Tone Traveling Direction

Both the Heartbeat and Fault Tones are a type of tone similar to the communication tone. One main difference is a communication tone only transmits with a single burst of couplets, but Heartbeat and Fault Tones are sent with a burst of couplets periodically. See [Section 6.3.6.2.3.3](#) for details.

6.3.6.2.3.3 Heartbeat and Fault Tone

The tones are enabled by setting $DEV_CONF[HB_EN] = 1$ and $DEV_CONF[FTONE_EN] = 1$ to enable the Heartbeat and Fault Tone transmitters, respectively. The Heartbeat and Fault Tone receivers are always on in SLEEP mode regardless of the $[HB_EN]$ and $[FTONE_EN]$ settings. To avoid fault detection (asserting NFAULT or FAULT_SUMMARY register) by a Heartbeat or Fault Tone fault, mask the fault by $[MSK_COMM3_HB] = 1$ or $[MSK_COMM3_FTONE] = 1$.

The Heartbeat and Fault Tone are formed with couplets with "-" polarity. They are differentiated by the number of couplets. Unlike communication tones, Heartbeat and Fault Tone are transmitted periodically. The period between tones is a burst period. The number of couplets transmitted is always greater than the number of couplets needed for detection.

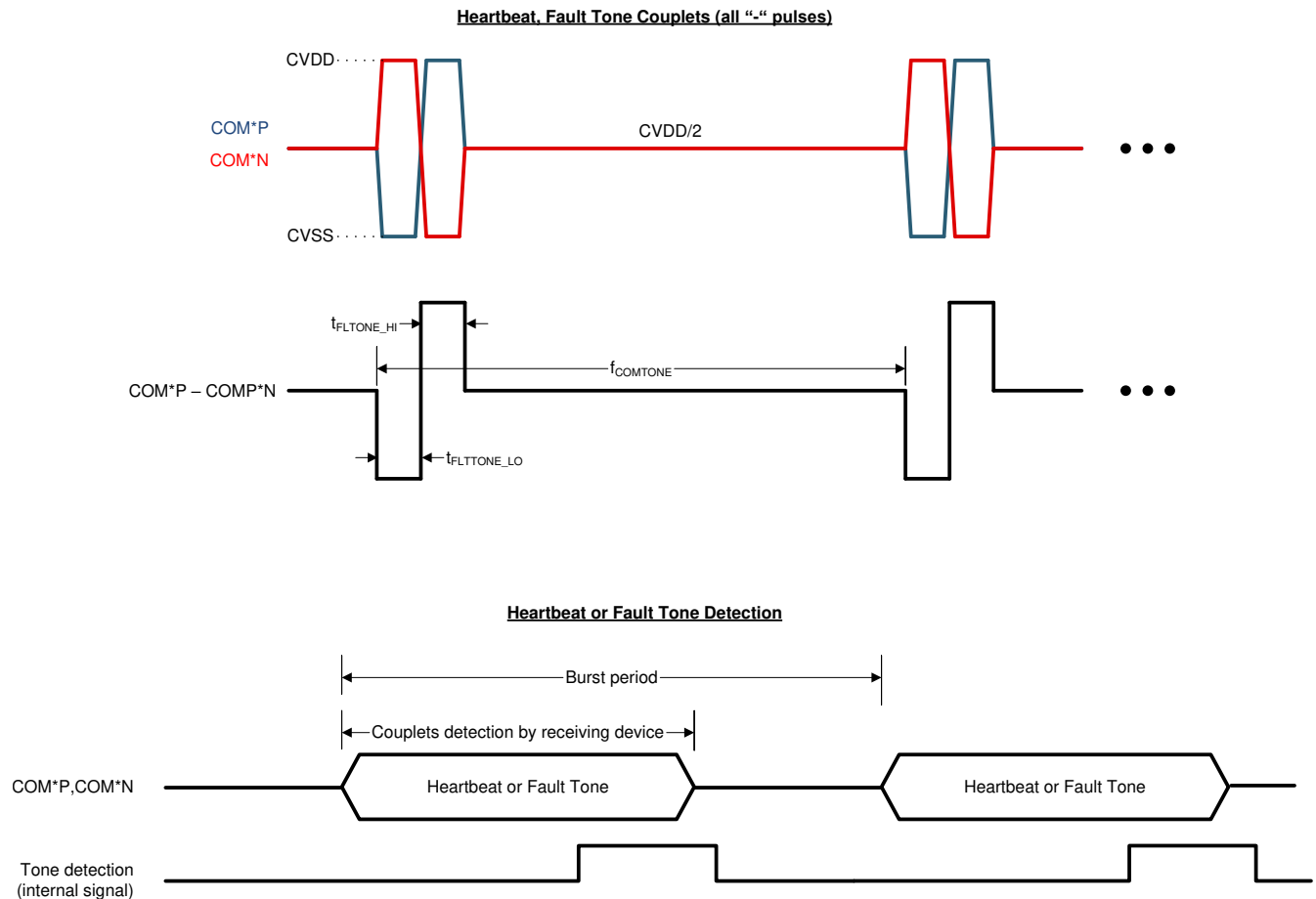


Figure 6-45. Heartbeat and Fault Tone

6.3.6.3 Nonvolatile Memory

There are memory locations that are programmable in nonvolatile memory (NVM) using OTP (One Time Programmable). The memory space is divided in two groups, factory space and customer space. The factory space stores the device configurations that are essential for normal operation. This space is not accessible by the host. The customer space contains the device default setting that host system can customize for their application configuration. This space is readable and programmable by the host.

When a device reset occurs, factory and customer OTP values are reloaded to their shadow registers. Error check and correction (ECC), single error correction (SEC) and double error detection (DED), are performed during the factory and customer space OTP load. The corresponding *FAULT_OTP[SEC_DET]* or *FAULT_OTP[DED_DET]* will be set if an error is detected.

Any load errors of the factory OTP space signal a fault using the *FAULT_OTP[FACTLDERR]*. Any load errors of the customer OTP space signal a fault using the *FAULT_OTP[CUSTLDERR]*. Additionally, the OTP space (factory and customer) are protected from data integrity problems using CRC. The corresponding *FAULT_OTP[FACT_CRC]* and *[CUST_CRC]* bits will be set if a CRC error is detected.

If any overvoltage error conditions exist in the OTP pages space (factory and customer) during programming, the *OTP_FAULT[GBLOVERR]* bit is set. Information received from the device with this error must not be considered reliable.

6.3.6.3.1 OTP Page Status

There are two unused pages of OTP memory available for the customer to program. Each page status is held in the *OTP_CUST1_STAT* and *OTP_CUST2_STAT* registers. The registers provide information on the current status of the page such as:

- Load status (if loaded, loaded with error, loaded but failed)
- Programmed successfully or available to be programmed
- Programmed status

When a reset occurs, the device evaluates the OTP page status and chooses the latest and valid OTP page to load. Page 2 has priority over Page 1. If both pages have not been written, the factory OTP default are loaded. [Section 6.5.1](#) shows all customer programmable OTP parameters. The register summary also shows the default values when Customer OTP Page 1 and Page 2 are not programmed.

- A valid page is one where the *OTP_CUST*_STAT[PROGOK]* = 1.
- When the page is selected for loading, the *OTP_CUST*_STAT1[LOADED]* = 1.
- If a single error occurs in the loading of the page, the page is loaded after the single error is corrected and the *OTP_CUST*_STAT1[LOADWRN]* = 1.
 - Additionally, the *DEBUG_OTP_SEC_BLK* register is updated with the location of the error corrected block.
- If a double error occurs, the loading of that block is terminated and the hardware defaults of that block are loaded (as indicated in [Section 6.5.1](#)).
 - The overall page loading process is not terminated for a DED, only the affected block is terminated.
 - When a DED occurs, the *OTP_CUST*_STAT1[LOADERR]* = 1. Additionally, the *DEBUG_OTP_DED_BLK* register is updated with the block where the double error occurred.

6.3.6.3.2 OTP Programming

[Section 6.5.1](#) shows all parameters that can be programmed to the customer OTP page. There are two pages of OTP memory available for customer to use.

Before programming the OTP, host ensures:

- All OTP shadow registers have the correct settings
- A customer OTP page is valid to be programmed. A valid page is one with *OTP_CUST*_STAT1[TRY]* = 0 and *OTP_CUST*_STAT1[FMTErr]* = 0.

Table 6-26. Program the OTP

| Step | Procedure |
|------|--|
| 1 | Unlock the OTP programming: <ol style="list-style-type: none"> a. Write the following data to <i>OTP_PROG_UNLOCK1A</i> to <i>OTP_PROG_UNLOCK1D</i> registers. <ul style="list-style-type: none"> • <i>OTP_PROG_UNLOCK1A</i> ≤ data 0x02 • <i>OTP_PROG_UNLOCK1B</i> ≤ data 0xB7 • <i>OTP_PROG_UNLOCK1C</i> ≤ data 0x78 • <i>OTP_PROG_UNLOCK1D</i> ≤ data 0xBC b. Do another write with the following data to <i>OTP_PROG_UNLOCK2A</i> to <i>OTP_PROG_UNLOCK2D</i> registers. <ul style="list-style-type: none"> • <i>OTP_PROG_UNLOCK2A</i> ≤ data 0x7E • <i>OTP_PROG_UNLOCK2B</i> ≤ data 0x12 • <i>OTP_PROG_UNLOCK2C</i> ≤ data 0x08 • <i>OTP_PROG_UNLOCK2D</i> ≤ data 0x6F Each block of registers must be written in order (that is, A, B, C, then D) with no other writes or reads between. The best practice is to use the same Write command to update. Any attempt to update the registers out of sequence, or if another register is written or read between writes, the entire sequence must be redone. |
| 2 | Check to confirm the OTP unlock procedure is successful: <ol style="list-style-type: none"> a. Read to confirm <i>OTP_PROG_STAT[UNLOCK]</i> = 1 Issuing a Read command after step 1 is ok, but issuing the <i>[PROG_GO]</i> must be the next write command after the unlock procedures. |
| 3 | Select the proper OTP page and start the OTP programming: <ol style="list-style-type: none"> a. To program page1, set <i>OTP_PROG_CTRL[PAGESEL][PROG_GO]</i> = 0x01, or b. To program page2, set <i>OTP_PROG_CTRL[PAGESEL][PROG_GO]</i> = 0x03 |

Table 6-26. Program the OTP (continued)

| Step | Procedure |
|------|---|
| 4 | Wait t_{PROG} for the OTP programming to complete |
| 5 | Check to ensure there is no error during OTP programming. The following bits are expected to be 1 after a successful OTP programming: a. $\text{OTP_PROG_STAT}[\text{DONE}] = 1$, OTP programming is done. No other bit will be set in this register. b. If page 1 is programmed, $\text{OTP_CUST1_STAT}[\text{PROGOK}]$, $[\text{TRY}]$, $[\text{OVOK}]$, and $[\text{UVOK}]$ bits are 1. Other bits are 0. c. If page 2 is programmed, $\text{OTP_CUST2_STAT}[\text{LOADED}]$, $[\text{PROGOK}]$, $[\text{TRY}]$, $[\text{OVOK}]$, and $[\text{UVOK}]$ bits are 1. Other bits are 0. |
| 6 | Issue a digital reset to reload the registers with the updated OTP values: a. $\text{CONTROL1}[\text{SOFT_RESET}] = 1$ |

During programming, if a programming voltage OV or UV event occurs, the $\text{OTP_CUST*_STAT}[\text{UVOK}]$ or $\text{OTP_CUST_STAT2}[\text{OVOK}]$ bit is 0 to indicate the programming voltage under- or overvoltage condition is detected during the programming attempts. In addition, the $[\text{UVERR}]$, $[\text{OVERR}]$, $[\text{SUVERR}]$, and $[\text{SOVERR}]$ bits in the OTP_PROG_STAT register indicate if there is programming voltage error during programming and stability test.

Note

- During the programming procedure, device performs a programming voltage stability test before actually programming the OTP. If a programming voltage fails the stability test, the device will not set the $\text{OTP_CUST*_STAT}[\text{TRY}]$ bit, giving the customer another attempt to program the page again.
- If the host incorrectly selects a page for programming, the $\text{OTP_PROG_STAT}[\text{PROGERR}]$ bit is set. This indicates that the selected page was not available to be programmed. Select the correct page and retry the programming.
- Device will not start OTP programming above 55°C temperature.
- OTP programming time (from $[\text{PROG_GO}] = 1$ to $[\text{DONE}] = 1$) for LDOIN capacitor of 0.1 μF is 100 ms.

6.3.6.4 Diagnostic Control/Status

The following subsections describe the diagnostic control and fault status that can be used as part of the safety mechanisms.

The Safety Manual for BQ79616 and the BQ79606 FMEDA documents are available separately from Texas Instruments. Contact TI Sales Associate or Applications Engineer for further information.

6.3.6.4.1 Power Supplies Check**6.3.6.4.1.1 Power Supply Diagnostic Check**

The internal power supply circuits have overvoltage, undervoltage, oscillation detection, and/or current limit checks. All these detections are continuously running in the background when the device is in ACTIVE or SLEEP mode. If a failure is detected, the corresponding flags in the FAULT_PWR* registers will be set or in certain failure modes, the device will reset. [Table 6-27](#) summarizes the diagnostics that apply for each power supply and the corresponding action when failure is detected.

Table 6-27. Power Supply Diagnostic Checks

| Supply/ Ground Pin | OV Check | UV Check | OSC Check | Current Limit | Pin Open |
|-----------------------|--|--|--|---|----------|
| LDOIN | | | | | |
| AVDD | If this fails, set $\text{FAULT_PWR1}[\text{AVDD_OV}]$ | If this fails, disable DVDD and trigger a digital reset. After soft reset, device sets $[\text{AVDDUV_DRST}]$ to indicate a reset is caused by AVDD UV. | If fails, set $\text{FAULT_PWR1}[\text{AVDD_OSC}]$ | Limit current to EC table current limit specification | |

Table 6-27. Power Supply Diagnostic Checks (continued)

| Supply/ Ground Pin | OV Check | UV Check | OSC Check | Current Limit | Pin Open |
|-----------------------|--|--|--|---|--|
| DVDD | If this fails, set <i>FAULT_PWR1</i> [DVDD_OV] | If this fails, trigger a digital reset | | Limit current to EC table current limit specification | |
| CVDD | If this fails, set <i>FAULT_PWR1</i> [CVDD_OV] | If this fails, set <i>FAULT_PWR1</i> [CVDD_UV] | | Limit current to EC table current limit specification | |
| TSREF | If this fails, set <i>FAULT_PWR2</i> [TSREF_OV] and <i>FAULT_OT</i> and <i>FAULT_UT</i> registers to all 1s. | If this fails, set <i>FAULT_PWR2</i> [TSREF_UV] and <i>FAULT_OT</i> and <i>FAULT_UT</i> registers to all 1s. | If fails, set <i>FAULT_PWR2</i> [TSREF_OSC] and <i>FAULT_OT</i> and <i>FAULT_UT</i> registers to all 1s. | Limit current to EC table current limit specification | |
| NEG5V | | If this fails, set <i>FAULT_PWR2</i> [NEG5V_UV] | | | |
| REFHP/REFHM | | | If REFHP fails, set <i>FAULT_PWR2</i> [REFH_OSC] | | If REFHM opens, set the <i>FAULT_PWR1</i> [REFHM_OPEN] |
| DVSS | | | | | If this opens, set the <i>FAULT_PWR1</i> [DVSS_OPEN] |
| CVSS | | | | | If this opens, set the <i>FAULT_PWR1</i> [CVSS_OPEN] |

Note

Due to the detection logic implemented, when AVDD OV or UV is detected, the AVDD OSC fault can also be triggered. Similarly, when TSREF OV or UV, the TSREF OSC fault can also be triggered.

6.3.6.4.1.2 Power Supply BIST

The device implements a power supply BIST (Built-In Self-Test) function to test the primary power supply failure diagnostic paths that cover the following detections:

- *FAULT_PWR1*[AVDD_OV], [AVDD_OSC], [DVDD_OV], [CVDD_OV], [CVDD_UV], [REFHM_OPEN], [DVSS_OPEN], and [CVSS_OPEN]
- *FAULT_PWR2*[TSREF_OV], [TSREF_UV], [TSREF_OSC], [NEG5V_UV], [REFHM_OSC], and [PWRBIST_FAIL]

The power supply BIST is essentially a check on the checker and it is a command base function initiated by host.

The power supply BIST, once started, will force a fault on failure detection path on each supply. Take AVDD OV diagnostic path as an example, when the BIST engine tests the AVDD OV path, the following occur:

1. The BIST engine forces a fail to the AVDD OV comparator
2. The BIST engine then checks to ensure the signal to trigger *FAULT* register is asserted, and the signal to trigger NFAULT is also asserted
3. The BIST engine resets the *FAULT* register and NFAULT signal (that is, clears the *FAULT_PWR1/2/3* registers and deasserts NFAULT)
4. The BIST engine repeats step 1 to step 3 on the next power supply diagnostic path check (for example, AVDD OSC) until all intended diagnostic paths covered by BIST are tested.

Note

- During the BIST run, the NFAULT pin will be toggled on and off. Host ignores the NFAULT pin status or can disable the NFAULT pin output by setting $DEV_CONF[NFAULT_EN] = 0$.
- Among all internal power supplies, TSREF is one that can be enabled or disabled by host. To ensure TSREF diagnostic paths are tested during BIST run, host enables TSREF before starting the power supply BIST. Otherwise, the BIST engine will ignore the TSREF diagnostic paths test result during the BIST run.
- Because other nonpower supply-related faults can also trigger NFAULT, it is recommended to mask all nonpower supply-related faults through $FAULT_MSK1/2$ registers before the power supply BIST run.
- Host also ensures there are no power supply faults before starting the power supply BIST run.

Start power supply BIST by sending $DIAG_PWR_CTRL[PWR_BIST_GO] = 1$. The BIST run will not abort even if a failure is detected during the run. At the end of the BIST run, the result is indicated by the $FAULT_PWR2[PWRBIST_FAIL]$ flag.

The power supply BIST forces a failure and ensures the diagnostic path triggers the fault accordingly. A failure on the BIST run indicates a diagnostic path is unable to trigger in a fault condition. To further examine which path is unable to indicate a failure, host can set the $DIAG_PWR_CTRL[BIST_NO_RST] = 1$. This bit disables the reset step during the BIST run. Re-start power supply BIST with this option enabled. At the end of the BIST run, examine the $FAULT_PWR1$ and $FAULT_PWR2$ registers. Any register flag that remains 0 indicates it is unable to flag a failure.

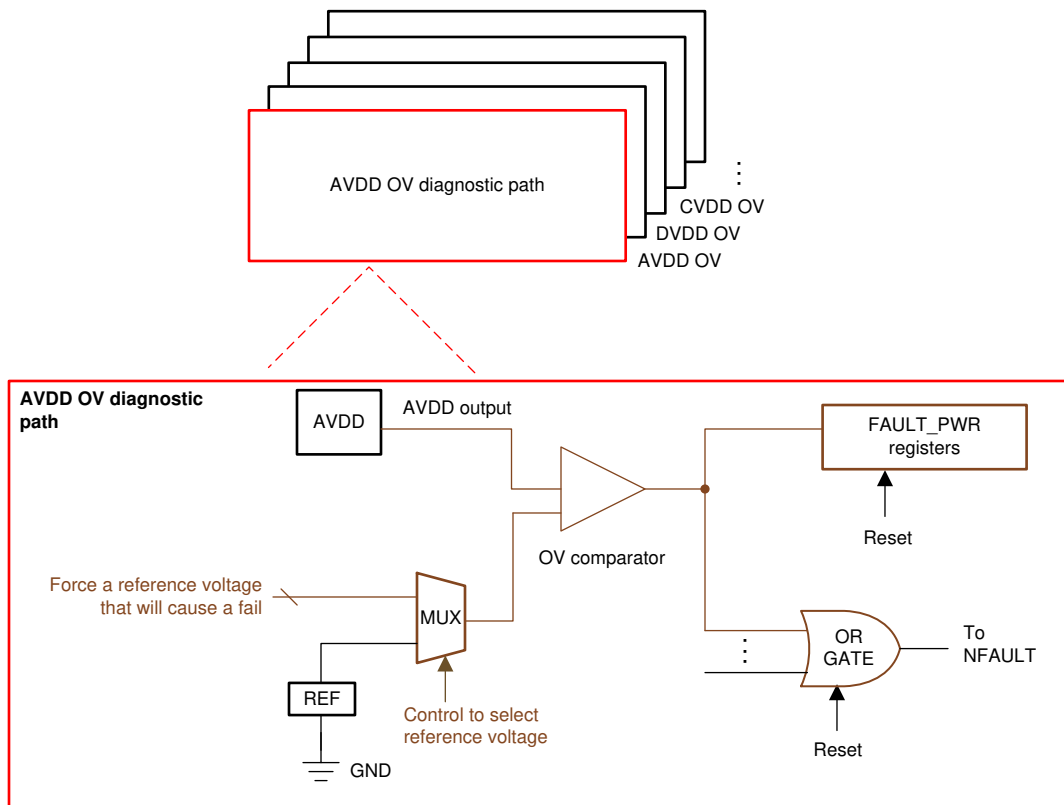


Figure 6-46. Power Supply BIST

6.3.6.4.2 Thermal Shutdown and Warning Check

6.3.6.4.2.1 Thermal Shutdown

Thermal shutdown occurs when the thermal shutdown sensor senses an overtemperature condition of the device. The sensor operates without interaction and is separated from the ADC measured die sensor. The thermal shutdown function has a register-status indicator flag (*FAULT_SYS[TSHUT]*) that is saved during the shutdown event and can be read after the device is awoken back up. When a TSHUT fault occurs, the part immediately enters the SHUTDOWN mode. Any pending transactions on UART or daisy chain are discarded. There is no fault signaling performed when a thermal shutdown event occurs as the device immediately shuts down.

To awaken the device, host ensures the ambient temperature is below T_{SHUT_FALL} and sends a WAKE ping to the base device. Host will not attempt to wake the device if the ambient temperature is still above T_{SHUT_FALL} .

Upon waking up, the *FAULT_SYS[TSHUT]* bit is set. See [Section 6.4.1.1](#) for more details. If the system faults are unmasked, *FAULT_MSK1[MSK_SYS]* = 0, the thermal shutdown will be reflected as a fault and will be indicated in the *FAULT_SUMMARY* register and the assertion of the NFAULT pin.

6.3.6.4.2.2 Thermal Warning

To warn the host of an impending thermal overload the device includes an overtemperature warning that signals a fault when the die temperature approaches thermal shutdown. The device detects the die temperature through the TWARN sensor against the thermal warning threshold. There are four threshold options configured by the *PWR_TRANSIT_CONF[TWARN_THR1:0]* setting.

When the system fault is unmasked, and the temperature warning fault occurs, the *FAULT_SYS[TWARN]* = 1. The host can take action to avoid a thermal shutdown.

6.3.6.4.3 Oscillators Watchdog

The oscillators are monitored by watchdog circuits. There are two oscillators in the device, the HFO and the LFO. If these oscillators are not functioning, the device does not operate. If the HFO or LFO does not transition within the expected time, the watchdog circuits cause a digital reset.

When this unexpected reset occurs, it is recommended that the host sends a SHUTDOWN ping/tone to the problem device and then send a WAKE ping to reset the daisy chain. If the oscillators are truly damaged, the device will not restart and must be replaced.

In addition to the watchdog, the LFO frequency is monitored to ensure it stays within acceptable limits. If the LFO frequency falls outside of the expected range, the *FAULT_SYS_FAULT[LFO]* bit is set.

6.3.6.4.4 OTP Error Check

6.3.6.4.4.1 OTP CRC Test and Faults

CRC Test:

The factory registers and customer OTP shadow registers are covered by a CRC check that constantly runs in the background. The *CUST_CRC_RSLT_HI* and *CUST_CRC_RSLT_LO* registers hold the current device's computed CRC value. This value is compared against the customer programmed value in the CRC registers, *CUST_CRC_HI* and *CUST_CRC_LO*. When updating any customer OTP shadow register covered in the CRC, the host must update a new CRC value to *CUST_CRC_HI* and *CUST_CRC_LO* registers. The CRC calculation is performed in the same manner (including the bit stream ordering) and with the same polynomial as described in [Section 6.3.6.1.2.1.6](#). The CRC check and comparison for factory and customer spaces is performed periodically and the *DEV_STAT[CUST_CRC_DONE]* and *[FACT_CRC_DONE]* bits are set after the check is complete. If the bit is already set, it remains set until cleared with a read.

CRC Faults:

When *CUST_CRC_HI/LO* and *CUST_CRC_RSLT_HI/LO* do not match, the *FAULT_OTP[CUST_CRC]* flag is set until the condition is corrected. Continuous monitoring of the factory NVM space occurs in a similar fashion, concurrently with the monitoring of the customer space. When a factory register change is detected, the *FAULT_OTP[FACT_CRC]* flag is set. When this fault occurs, the host should reset the fault flag to see if the

fault persists. If the fault persists, the host must perform a reset of the part. If reset does not correct the issue, the device is corrupted and must not be used.

6.3.6.4.4.2 OTP Margin Read

The device provides OTP margin read test modes, with which host can set up to reload the OTP with margin 1 or margin 0. To start the margin read test, host selects the desired test mode through `DIAG_OTP_CTRL[MARGIN_MODE2:0]` and sets `DIAG_OTP_CTRL[MARGIN_GO] = 1`. The device will reload the OTP per the `[MARGIN_MODE2:0]` setting. Any OTP related error will be flagged to the `FAULT_OTP` register.

6.3.6.4.4.3 Error Check and Correct (ECC) OTP

ECC:

Register values for selected registers (0x0000 to 0x002F) are permanently stored in OTP. All registers also exist as volatile storage locations at the same addresses, referred to as shadow registers. The volatile registers are for reading, writing, and device control. For a list of registers included in the OTP, see [Section 6.5.1](#).

During wakeup, the device first loads all shadow registers with hardware default values listed in [Section 6.5.1](#). Then the device loads the registers conditionally with OTP contents from the results of the Error Check and Correct (ECC) evaluation of the OTP. The OTP is loaded to shadow registers in 64-bit blocks; each block has its own Error Check and Correct (ECC) value stored. The ECC detects a single-bit (Single-Error-Correction) or double-bit (Double-Error-Detection) changes in OTP stored data. The ECC is calculated for each block, individually.

Single-bit errors are corrected, double-bit errors are only detected, not corrected. A block with good ECC is loaded. A block with a single-bit error is corrected, and the `FAULT_OTP[SEC_DET]` bit is set to flag the corrected error event. Additionally, the `DEBUG_OTP_SEC_BLK` register is updated with the location of the error corrected block. This enables the host to keep track of potentially damaged memory. The block is loaded to shadow registers after the single-bit error correction. Because the evaluation is on a block-by-block basis, it is possible for multiple blocks to have a single-correctable error and still be loaded correctly. Multiple-bit errors can exist with full correction, as long as they are limited to a single error per block.

A block with a bad ECC comparison (two-bit errors in one block) is not loaded and the `FAULT_OTP[DED_DET]` bit is set to flag the failed bit-error event. Additionally, the `DEBUG_OTP_DED_BLK` register is updated with the block where the double error occurred. The hardware default value remains in the register. This allows some blocks to be loaded correctly (no fail or single-bit corrected value) and some blocks not to load. When the `FAULT_OTP[SEC_DET]` or `FAULT_OTP[DED_DET]` bit is set and the condition is not cleared by a device reset, the device is corrupted and must not be used.

The ECC engine uses the industry standard 72,64 SEC DEC ECC implementation. The OTP is protected by a (72, 64) Hamming code, providing single error correction, double error detection (SECDED). For each 64 bits of data stored in OTP, an additional 8 bits of parity information are stored. The parity bits are designated p0, p1, p2, p4, p8, p16, p32, and p64. Bit p0 covers the entire encoded 72-bit ECC block. The remaining seven parity bits are assigned according to the following rule:

- Parity bit p1 covers odd bit positions, that is, bit positions which have the least significant bit of the bit position equal to 1 (1, 3, 5, and so on), including the p1 bit itself (bit 1).
- Parity bit p2 covers bit positions which have the second least significant bit of the bit position equal to 1 (2, 3, 6, 7, 10, 11, and so on), including the p2 bit itself (bit 2).
- The pattern continues for p4, p8, p16, p32, and p64. [Table 6-28](#) specifies the complete encoding.

Table 6-28. (72, 64) Parity Encoding

| Bit Position | 71 | 70 | 69 | 68 | 67 | 66 | 65 | 64 | 63 | 62 | 61 | 60 | 59 | 58 | 57 | 56 | 55 | 54 | |
|---------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---|
| Encoded Bits | d63 | d62 | d61 | d60 | d59 | d58 | d57 | p64 | d56 | d55 | d54 | d53 | d52 | d51 | d50 | d49 | d48 | d47 | |
| Parity Bit Coverage | p0 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | |
| | p1 | x | | x | | x | | x | | x | | x | | x | | x | | | |
| | p2 | x | x | | | x | x | | | x | x | | | x | x | | | x | x |
| | p4 | x | x | x | x | | | | | x | x | x | x | | | | | x | x |
| | p8 | | | | | | | | | x | x | x | x | x | x | x | x | | |
| | p16 | | | | | | | | | x | x | x | x | x | x | x | x | x | x |
| | p32 | | | | | | | | | x | x | x | x | x | x | x | x | x | x |
| | p64 | x | x | x | x | x | x | x | x | | | | | | | | | | |
| Bit Position | 53 | 52 | 51 | 50 | 49 | 48 | 47 | 46 | 45 | 44 | 43 | 42 | 41 | 40 | 39 | 38 | 37 | 36 | |
| Encoded Bits | d46 | d45 | d44 | d43 | d42 | d41 | d40 | d39 | d38 | d37 | d36 | d35 | d34 | d33 | d32 | d31 | d30 | d29 | |
| Parity Bit Coverage | p0 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | |
| | p1 | x | | x | | x | | x | | x | | x | | x | | x | | | |
| | p2 | | | x | x | | | x | x | | | x | x | | | x | x | | |
| | p4 | x | x | | | | | x | x | x | x | | | | | x | x | x | x |
| | p8 | | | | | | | x | x | x | x | x | x | x | | | | | |
| | p16 | x | x | x | x | x | x | | | | | | | | | | | | |
| | p32 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |
| | p64 | | | | | | | | | | | | | | | | | | |
| Bit Position | 35 | 34 | 33 | 32 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | |
| Encoded Bits | d28 | d27 | d26 | p32 | d25 | d24 | d23 | d22 | d21 | d20 | d19 | d18 | d17 | d16 | d15 | d14 | d13 | d12 | |
| Parity Bit Coverage | p0 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | |
| | p1 | x | | x | | x | | x | | x | | x | | x | | x | | | |
| | p2 | x | x | | | x | x | | | x | x | | | x | x | | | x | x |
| | p4 | | | | | x | x | x | x | | | | | x | x | x | x | | |
| | p8 | | | | | x | x | x | x | x | x | x | x | | | | | | |
| | p16 | | | | | x | x | x | x | x | x | x | x | x | x | x | x | x | x |
| | p32 | x | x | x | x | | | | | | | | | | | | | | |
| | p64 | | | | | | | | | | | | | | | | | | |
| Bit Position | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| Encoded Bits | d11 | p16 | d10 | d9 | d8 | d7 | d6 | d5 | d4 | p8 | d36 | d2 | d1 | p4 | d0 | p2 | p1 | p0 | |
| Parity Bit Coverage | p0 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | |
| | p1 | x | | x | | x | | x | | x | | x | | x | | x | | | |
| | p2 | | | x | x | | | x | x | | | x | x | | | x | x | | |
| | p4 | | | x | x | x | x | | | | | x | x | x | x | | | | |
| | p8 | | | x | x | x | x | x | x | x | x | | | | | | | | |
| | p16 | x | x | | | | | | | | | | | | | | | | |
| | p32 | | | | | | | | | | | | | | | | | | |
| | p64 | | | | | | | | | | | | | | | | | | |

Table 6-29. Encoder and Decoder Data IN and OUT Positioning

| ENCODER | | | |
|------------------|---------------|-------------------|---------------|
| DATA IN | Encoded Bits | DATA OUT | Bit Positions |
| OTP_ECC_DATAIN 1 | d0 to d7 | OTP_ECC_DATAOUT 1 | 0 to 7 |
| OTP_ECC_DATAIN 2 | d8 to d15 | OTP_ECC_DATAOUT 2 | 8 to 15 |
| OTP_ECC_DATAIN 3 | d16 to d23 | OTP_ECC_DATAOUT 3 | 16 to 23 |
| OTP_ECC_DATAIN 4 | d24 to d31 | OTP_ECC_DATAOUT 4 | 24 to 31 |
| OTP_ECC_DATAIN 5 | d32 to d39 | OTP_ECC_DATAOUT 5 | 32 to 39 |
| OTP_ECC_DATAIN 6 | d40 to d47 | OTP_ECC_DATAOUT 6 | 40 to 47 |
| OTP_ECC_DATAIN 7 | d48 to d55 | OTP_ECC_DATAOUT 7 | 48 to 55 |
| OTP_ECC_DATAIN 8 | d56 to d63 | OTP_ECC_DATAOUT 8 | 56 to 63 |
| | | OTP_ECC_DATAOUT 9 | 64 to 71 |
| DECODER | | | |
| DATA IN | Bit Positions | DATA IN | Encoded Bits |
| OTP_ECC_DATAIN 1 | 0 to 7 | OTP_ECC_DATAOUT 1 | d0 to d7 |
| OTP_ECC_DATAIN 2 | 8 to 15 | OTP_ECC_DATAOUT 2 | d8 to d15 |
| OTP_ECC_DATAIN 3 | 16 to 23 | OTP_ECC_DATAOUT 3 | d16 to d23 |
| OTP_ECC_DATAIN 4 | 24 to 31 | OTP_ECC_DATAOUT 4 | d24 to d31 |
| OTP_ECC_DATAIN 5 | 32 to 39 | OTP_ECC_DATAOUT 5 | d32 to d39 |
| OTP_ECC_DATAIN 6 | 40 to 47 | OTP_ECC_DATAOUT 6 | d40 to d47 |
| OTP_ECC_DATAIN 7 | 48 to 55 | OTP_ECC_DATAOUT 7 | d48 to d55 |
| OTP_ECC_DATAIN 8 | 56 to 63 | OTP_ECC_DATAOUT 8 | d56 to d63 |
| OTP_ECC_DATAIN 9 | 64 to 71 | | |

ECC Diagnostic Test: The device provides a diagnostic tool to test the ECC function. There are two modes that are available to run the diagnostic. The first, auto mode ($OTP_ECC_TEST[MANUAL_AUTO] = 0$), uses internal data to run the tests. In auto mode, the $OTP_ECC_TEST[DED_SEC]$ bit selects the type of test that is to be performed and the $OTP_ECC_TEST[ENC_DEC]$ bit determines if the encoder or decoder function is to be tested. The result of the ECC test is provided in the $OTP_ECC_DATAOUT^*$ registers within 1 μ s delay. The test steps and expected results from each test are shown below.

Automatic Decoding steps:

1. Set ECC Test to automatic $OTP_ECC_TEST[MANUAL_AUTO] = 0$
2. Set decoder setting $OTP_ECC_TEST[ENC_DEC] = 0$
3. Set decoder to single or double encoding setting with $OTP_ECC_TEST[DED_SEC]$ (1 for DED or 0 for SEC)
4. Clear all SEC/DED faults by $FAULT_RST2[RST_OTP_DATA] = 1$
5. Enable ECC test $OTP_ECC_TEST[ENABLE] = 1$
6. Read $FAULT_OTP[SEC_DET]$ flag for SEC or $FAULT_OTP[DED_DET]$ flag for DED
7. Block read $OTP_ECC_DATAOUT1$ to $OTP_ECC_DATAOUT8$ to verify the decoder test results as in [Table 6-30](#)
8. Disable ECC test $OTP_ECC_TEST[ENABLE] = 0$

Automatic Encoding steps:

1. Set ECC TEST to automatic $OTP_ECC_TEST[MANUAL_AUTO] = 0$
2. Set the encoder setting using $OTP_ECC_TEST[ENC_DEC] = 1$
3. Enable the ECC test with $OTP_ECC_TEST[ENABLE] = 1$
4. Block read $OTP_ECC_DATAOUT1$ to $OTP_ECC_DATAOUT9$ to verify the encoder test results as in [Table 6-30](#)
5. Disable ECC test $OTP_ECC_TEST[ENABLE] = 0$

Table 6-30. Decoder and Encoder Test Verification

| [DED_SEC] | [ENC_DEC] | [SEC_DET] | [DED_DET] | OTP_DATAOUT* |
|--------------|------------------|-----------|-----------|--------------------------|
| 0 (SEC test) | 0 (Decoder test) | 1 | 0 | 0x18C3 FF8A 68A9 8069 |
| 0 (SEC test) | 1 (Encoder test) | N/A | N/A | 0xCD 3968 C140 2EA5 ED6D |
| 1 (DED test) | 0 (Decoder test) | 0 | 1 | 0x0000 0000 0000 0000 |
| 1 (DED test) | 1 (Encoder test) | N/A | N/A | 0xCD 3968 C140 2EA5 ED6D |

6.3.6.4.5 Integrated Hardware Protector Check

6.3.6.4.5.1 Parity Check

When the OVUV and OTUT protectors are enabled, the register settings related to the OVUV and OTUT configurations are latched to protector blocks. The device will check periodically in the background to ensure the latched configurations remain the same throughout the protector operation.

The parity check covers the following latched setting. If a parity fault in the OVUV protector is detected, the device will set the *FAULT_PROT1[VPARITY_FAIL]* = 1. If a parity fault in the OTUT protector is detected, the device will set the *FAULT_PROT1[TPARITY_FAIL]* = 1.

Table 6-31. Protector Parity Check Settings

| OVUV Protector | OTUT Protector | Note |
|----------------------------|---|--|
| OV threshold, UV threshold | OT threshold, UT threshold | Ensure threshold settings remains the same during the operation |
| OVUV_MODE setting | OTUT_MODE setting | Ensure the protector doesn't switch to a different operation mode |
| NUM_CELL setting | <i>GPIO_CONF1</i> to <i>GPIO_CONF4</i> settings | Ensure the active channel (either cell channels for OVUV or GPIO channel for OTUT) remains the same during operation |

6.3.6.4.5.2 OVUV and OTUT DAC Check

The OV, UV, OT, and UT DAC values are multiplexed to the AUX ADC from which the host can read out the values as part of the diagnostic check on the protector threshold settings.

To measure the protector's DAC value, it is recommended to lock the OVUV or OTUT protectors to a single channel through *OVUV_CTRL[OVUV_LOCK3:0]* for OV and UV DAC measurement; and through *OTUT_CTRL[OTUT_LOCK2:0]* for OT and UT DAC measurement, and restart the OVUV protectors or OTUT protector to run in the single channel run mode. Host ensures the locked cell channel is not under OV or UV fault or the locked GPIO channel is not under OT or UT fault. Otherwise, the DAC measurement will not be reflecting the triggering threshold value. Note that the OV and UV DAC value is (0.8 x the threshold setting).

6.3.6.4.5.3 OVUV Protector BIST

The device implemented an OVUV BIST (Built-In-Self-Test) function to test the primary OVUV protector path. Host can start the BIST run by setting *[OVUV_MODE1:0]* = 0b10 and *[OVUV_GO]* = 1. The BIST run covers:

1. OV and UV comparators thresholds:
 - a. A higher and lower than the set threshold are checked to ensure the comparator is triggered correctly.
 - b. If failure is detected, the corresponding *FAULT_PROT2[OVCOMP_FAIL]* or *[UVCOMP_FAIL]* bit will be set.
2. The path from the OVUV MUX to UV fault status bit and NFAULT pin:
 - a. For each VC channel, a switch is open so that input to the OVUV MUX is open and will lead to a UV detection to the channel under test
 - b. The BIST engine then checks the logic to assert corresponding *FAULT_UV* register bit and the NFAULT is set properly.
 - c. The BIST engine resets the corresponding *FAULT_UV* bit and deasserts the NFAULT, then switches to test the next channel and repeats the process until all active channels are tested.
 - d. If failure is detected, the corresponding *[VPATH_FAIL]* bit is set.
3. OV fault bit and NFAULT path

- a. The BIST engine forces 1 to the *FAULT_OV** register, one bit at time, to ensure each *FAULT_OV** register bit can be set and the NFAULT can be asserted, accordingly.
- b. If failure is detected, the corresponding [*VPATH_FAIL*] bit will be set.

If NFAULT is enabled, host observes NFAULT toggling during the BIST run. Upon completion of the BIST run, the OVUV comparators will be turned off. Host starts the regular OVUV round robin mode by sending [*OVUV_GO*] = 1 with [*OVUV_MODE1:0*] = 0b01 (round robin mode).

Note

- If a [*OVUV_GO*] = 1 is sent during the OVUV BIST run, device will execute the new GO command based on the [*OVUV_MODE1:0*] setting.
 - Before starting the OVUV Protector BIST, host masks out all the non-OVUV related faults, and ensures there are no OV and UV faults on any cell channels (recommended all cell voltages to be at least 100 mV apart from the OV or UV threshold during the BIST run). Otherwise, the BIST result is not invalid.
 - After BIST starts, if pre-existing fault is detected before starting step 2, the BIST engine will be aborted and the *FAULT_PROT2[BIST_ABORT]* = 1.
 - A no reset option, *DIAG_PROT_CTRL[PROT_BIST_NO_RST]* = 1, is available to command the BIST engine not to reset the fault status and NFAULT pin after testing each channel. If a BIST run fails, host can select this option and re-run BIST to detect which cell channel path is unable reflect a fault condition in the fault registers.
-

6.3.6.4.5.4 OTUT Protector BIST

The device implemented an OTUT BIST function to test the primary OTUT protector path. Host can start the BIST run by setting [*OTUT_MODE1:0*] = 0b10 and [*OTUT_GO*] = 1. The BIST run covers:

1. OT and UT comparator thresholds
 - a. A higher and lower than the set threshold are checked to ensure the comparator is triggering correctly.
 - b. If failure is detected, the corresponding *FAULT_PROT2[OTCOMP_FAIL]* or [*UTCOMP_FAIL*] bit will be set.
2. The path from GPIO MUX to UT fault bit and NFAULT path
 - a. For each GPIO channel, the GPIO is internally pulled up so the input to the OTUT MUX is high and will lead to a UT detection to the channel under test.
 - b. The BIST cycle then checks the logic to assert the corresponding *FAULT_UT* register bit and the NFAULT is set properly.
 - c. The BIST engine resets the corresponding *FAULT_UT* bit and deasserts the NFAULT, then switches to test the next channel.
 - d. If failure is detected, the corresponding [*TPATH_FAIL*] bit will be set.
3. OV fault bit and NFAULT path
 - a. The BIST engine forces 1 to the *FAULT_OT* register, one bit at time, to ensure each *FAULT_OT* register bit can be set and the NFAULT can be asserted, accordingly.
 - b. If failure is detected, the corresponding [*TPATH_FAIL*] bit will be set.

If NFAULT is enabled, host observes NFAULT toggling during the BIST run. Upon completion of the BIST run, the OTUT comparators will be turned off. Host starts the regular OTUT round robin mode by sending [*OTUT_GO*] = 1 with [*OTUT_MODE1:0*] = 0b01 (round robin mode).

Note

- If a $[OTUT_GO] = 1$ is sent during the OTUT BIST run, device will execute the new GO command based on the $[OVUV_MODE1:0]$ setting.
- Before starting the OTUT Protector BIST, host masks out all non-OTUT related faults, and ensures there are no OT and UT faults on any GPIO during the BIST run). Otherwise, the BIST result is not invalid.
- After BIST starts, if pre-existing fault is detected before starting step 2, the BIST engine will be aborted and the $FAULT_PROT2[BIST_ABORT] = 1$.
- A no reset option, $DIAG_PROT_CTRL[PROT_BIST_NO_RST] = 1$, is available to command the BIST engine not to reset the fault status and NFAULT pin after testing each channel. If a BIST run fails, host can select this option and re-run BIST to detect which GPIO channel path is unable reflect a fault condition in the fault registers.

6.3.6.4.6 Diagnostic Through ADC Comparison

6.3.6.4.6.1 Cell Voltage Measurement Check

Cell voltage measurement path comparison:

The cell voltage measurement check is performed by comparing the prefiltered measurement result from Main ADC versus measurement result from AUX ADC. To read the compared value measured by Main ADC and AUX ADC, MCU has to set up this diagnostic check to lock on a single channel using $[AUX_CELL_SEL]$ setting and the start this diagnostic check. In this configuration, the compared values from Main ADC and AUX ADC are reported to $DIAG_MAIN_HI/LO$ registers and $DIAG_AUX_HI/LO$ registers respectively.

Both Main and AUX ADC has the same front end filters. This diagnostic time is mostly spent on waiting for the AAF on the AUX ADC path to settle. The $[AUX_SETTLE]$ setting allows the MCU to make trade-off between diagnostic time and noise filter level. Additionally, when AUX ADC starts, by default, AUXCELL slot always align to the Main ADC Cell1 slot. The $[AUX_CELL_ALIGN]$ setting allows MCU to change this alignment to Main ADC Cell8 slot, resulting with less sampling time delta between Main and AUX ADC on the higher channels.

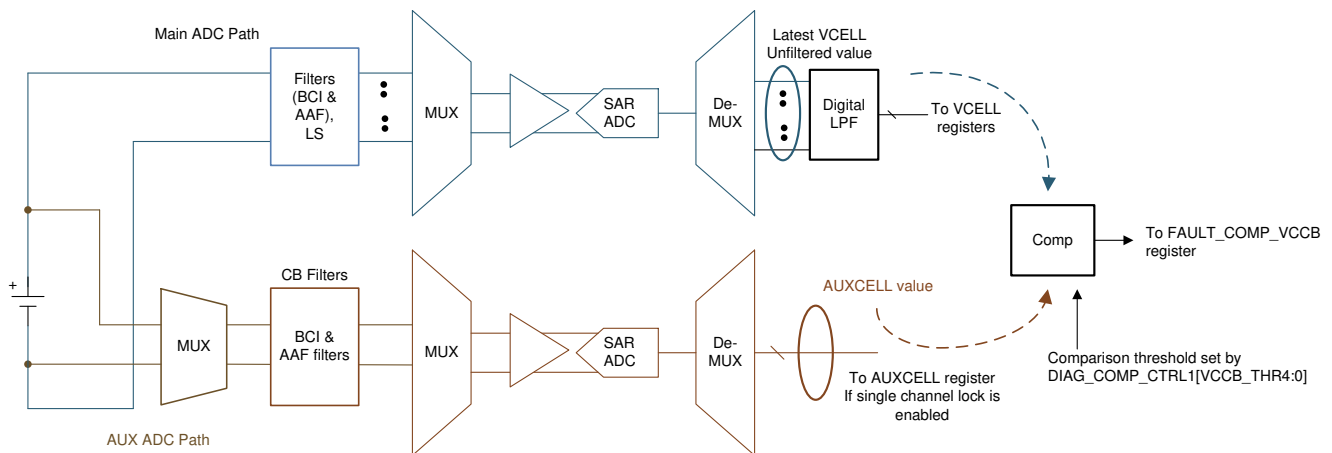


Figure 6-47. Cell Voltage Measurement Diagnostic

Before starting the cell voltage measurement comparison, host ensures:

- The desired AUXCELL channels to be tested are configured in the $ADC_CTRL2[AUX_CELL_SEL4:0]$ setting and AUX ADC is enabled and in continuous mode.
- Allow AUX ADC to run through all AUXCELL channels for the device to compensate for common mode error before starting this diagnostic check.
- Main ADC must be enabled and is in continuous mode.
- Select the (VCELL – AUXCELL) comparison threshold through $DIAG_COMP_CTRL1[VCCB_THR4:0]$ setting.

- Select the desired settling time for the AUX CELL channel through `ADC_CONF1[AUX_SETTLE1:0]`.

To start the cell voltage measurement comparison:

1. Set `DIAG_COMP_CTRL3[COMP_ADC_SEL2:0]` = cell voltage measurement check (that is, 0b001) and set `[COMP_ADC_GO] = 1`.
2. For each channel enabled by `[AUX_CELL_SEL4:0]`, the device will compare $\text{abs}[(V_{\text{CELL}} - \text{AUXCELL})] < [V_{\text{CCB_THR4:0}}]$.
3. Wait for the comparison to be accomplished, roughly $[(\text{number of channel}) * (\text{AUXCELL settling time} + \text{one round robin cycle time})]$.
4. The cell voltage measurement comparison is completed when `ADC_STAT2[DRDY_VCCB] = 1`.

Host checks the `FAULT_COMP_VCCB1` and `FAULT_COMP_VCCB2` registers for the comparison result.

ADC comparison abort conditions:

The device will not start the cell voltage measurement comparison under the invalid conditions listed below. When the comparison is aborted, the `FAULT_COMP_MISC[COMP_ADC_ABORT] = 1`, `[DRDY_AUX_CEL] = 1`, `[DRDY_VCCB] = 1`, and `FAULT_COMP_VCCB1/2` registers = 0xFF. If `[AUX_CELL_SEL4:0]` is set to locked at a single channel, the `AUX_CELL_HI/LO` registers will be reset to default value 0x8000 if the comparison run is aborted.

Invalid conditions or settings which will prevent the start of the cell voltage measurement comparison:

- Invalid `[AUX_CELL_SEL]` setting: results in no AUX ADC measurement on the selected channel. The `AUX_CELL_HI/LO` registers are kept in default value.
- Channel higher than the `NUM_CELL` configuration is selected.
- Invalid `BBVC_POSN` setting:
 - Adjacent channels are enabled in the `BBVC_POSN1/2` registers.
 - `BBVC_POSN2[CELL1]` is enabled.
 - More than two channels are selected in `BBVC_POSN1/2`.
 - `[AUX_CELL_SEL]` is locked to any of the selected channels in `BBVC_POSN1/2`.
- Main or AUX ADCs are off or not set in continuous mode.

Post-ADC digital LPF check:

The digital LPF is checked continuous whenever the Main ADC is running. A duplicate diagnostic LPF is implemented to check against each LPF for each VC channel and the BBP/N channel. The check is performed with one LPF at a time.

Example, to test LPF1 for cell channel 1, the input (that is, ADC measurement result from cell 1) is fed to the LPF1 and the diagnostic LPF for a period of time. The output of the LPF1 and the diagnostic LPF are compared against each other. Several outputs from LPF1 and diagnostic LPF will be compared to ensure the operation of the LFP1 before moving to check the next LFP. If any of the LPFs fail the diagnostic check, `FAULT_COMP_MISC[LPF_FAIL] = 1`.

When the LPF for each active cell channels is tested once, `ADC_STAT2[DRDY_LPF] = 1`. This diagnostic check of the LPFs will continuously run in the background as long as the Main ADC is running.

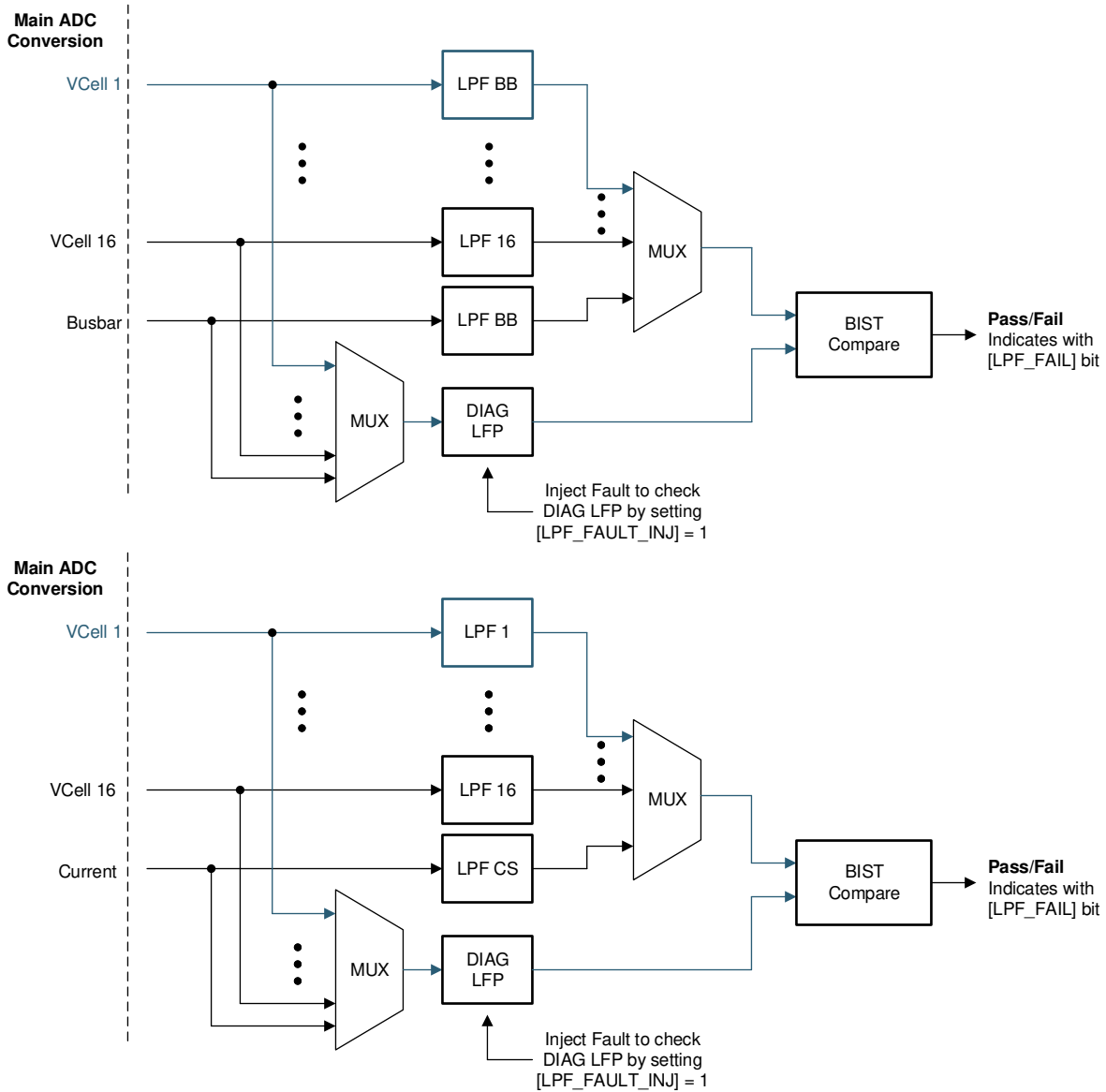


Figure 6-48. Post-ADC LPF Diagnostic (Blue Path as Example of Checking LPF1)

Furthermore, the device also implements a check to verify the functionality of the diagnostic LPF itself. By setting $DIAG_COMP_CTRL4[LPF_FAULT_INJ] = 1$ and restarting the Main ADC, the device will inject a fault into the diagnostic LPF, forcing a failure during the LPF diagnostic check which then sets the $[LPF_FAIL] = 1$. When the test is completed, simply set the $[LPF_FAULT_INJ] = 0$.

6.3.6.4.6.2 Temperature Measurement Check

Similar to the cell voltage measurement check, the device checks the thermistor temperature measurement by comparing the Main ADC measurement to the AUX ADC measurement. To read the compared value measured by Main ADC and AUX ADC, MCU has lock on a single channel using $[AUX_GPIO_SEL]$ setting and the start this diagnostic check. In this configuration, the compared values from Main ADC and AUX ADC are reported to $DIAG_MAIN_HI/LO$ registers and $DIAG_AUX_HI/LO$ registers respectively.

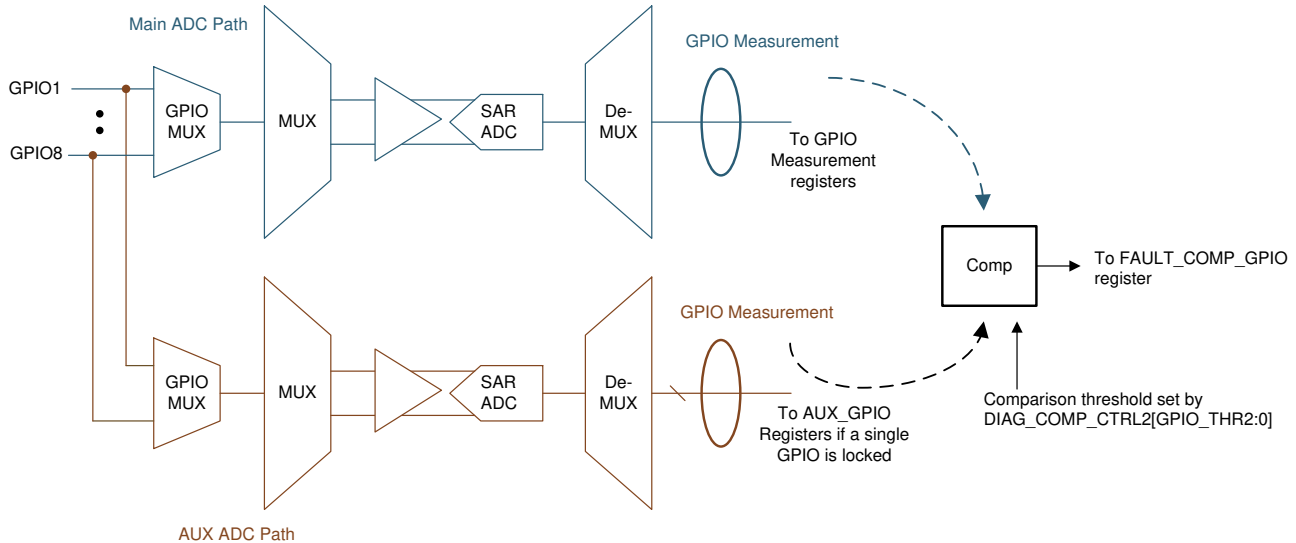


Figure 6-49. Thermistor Temperature (GPIO) Measurement Diagnostic

Before starting the temperature measurement comparison, host ensures:

- Main ADC must be enabled and is in continuous mode.
- The desired GPIO channels to be tested are configured in the `ADC_CTRL3[AUX_GPIO_SEL3:0]` setting and AUX ADC is enabled and in continuous mode.
- Select the comparison threshold through `DIAG_COMP_CTRL2[GPIO_THR2:0]` setting.

To start the cell voltage measurement comparison:

1. Set `DIAG_COMP_CTRL3[COMP_ADC_SEL2:0] = GPIO` measurement check (that is, 0b101) and set `[COMP_ADC_GO] = 1`.
2. For each channel enabled by `[AUX_GPIO_SEL4:0]`, the device will compare $\text{abs}[(\text{GPIO from Main} - \text{GPIO from AUX})] < [\text{GPIO_THR2:0}]$.
3. Wait for the comparison to be accomplished which can take up to 64 ADC round robin times.
4. The GPIO measurement comparison is completed when `ADC_STAT2[DRDY_GPIO] = 1`.

Host checks the `FAULT_COMP_GPIO` register for the comparison result.

ADC comparison abort conditions:

The device will not start the temperature measurement comparison under the invalid conditions listed below. When the comparison is aborted, the `FAULT_COMP_MISC[COMP_ADC_ABORT] = 1`, `[DRDY_GPIO] = 1`, and `FAULT_COMP_GPIO = 0xFF`. If `[AUX_GPIO_SEL3:0]` is set to locked at a single channel, the `AUX_GPIO_HI/LO` registers will be reset to default value 0x8000 if the comparison run is aborted.

Invalid conditions or settings which will prevent the start of the temperature measurement comparison:

- Invalid `[AUX_GPIO_SEL]` setting which the selected GPIO isn't configured for ADC measurement. The `AUX_GPIO_HI/LO` registers are kept in default value. This also applies to the case if `[AUX_GPIO_SEL]` is selected for all GPIOs but none of the GPIOs are configured for ADC measurement.
- Main or AUX ADCs are off or not set in continuous mode.

6.3.6.4.6.3 Cell Balancing FETs Check

The cell balancing FET check is performed by turning on the balancing FET and comparing the voltage across the FET (through the AUX ADC path) versus the cell voltage (through the Main ADC path). To read the AUXCELL measurement used for the check, MCU has to set up this diagnostic check to lock on a single channel using `[AUX_CELL_SEL]` setting and the start this diagnostic check. The AUXCELL compared value will be reported to `DIAG_AUX_HI/LO` registers.

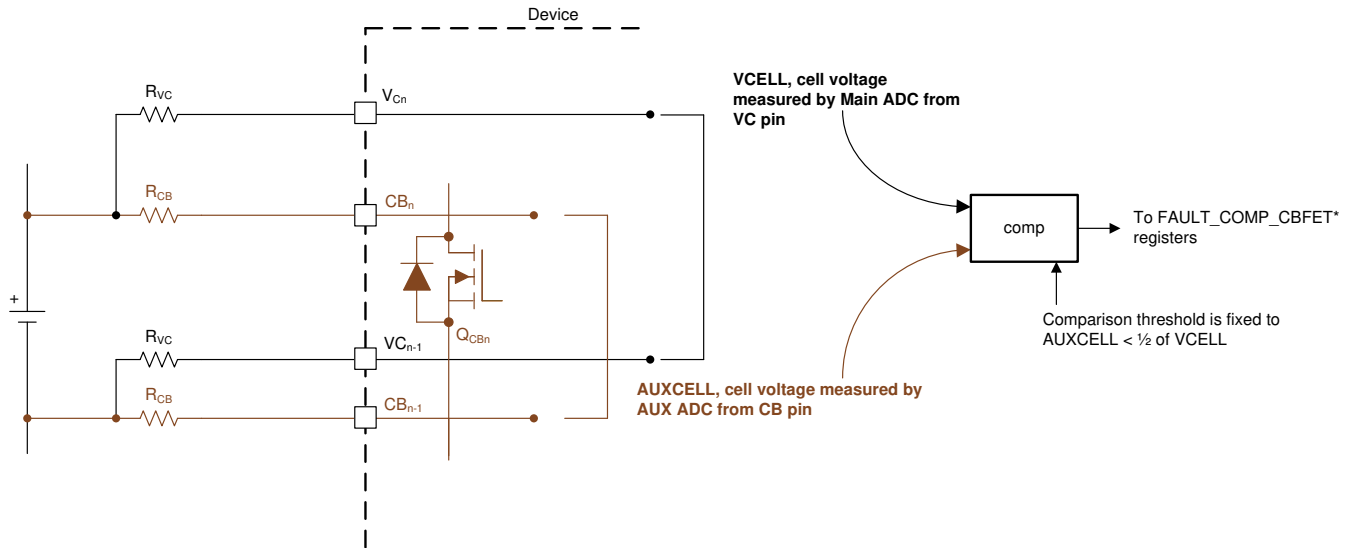


Figure 6-50. Cell Balancing FET Diagnostic

Before starting the cell balancing FET comparison, host ensures:

- Main ADC is running in continuous mode.
- Configured in the `ADC_CTRL2[AUX_CELL_SEL4:0]` to select the AUXCELL channels which the CB FETs are tested.
- Select the desired settling time for the AUX CELL channel through `ADC_CONF1[AUX_SETTLE1:0]`.
- Pause CB if balancing is running.
- Configured which CBFET to be tested through `DIAG_CBFET_CTRL1` and `DIAG_CBFET_CTRL2` registers.
 - The rules of maximum of eight CBFETs to be on and turn on no more than two consecutive CBFETs still apply.
 - Recommended to test in odd and even manner.

To start the CBFET comparison:

1. Start AUX ADC in continuous mode.
2. Turn on the selected CBFET by setting `DIAG_COMP_CTRL3[CBFET_CTRL_GO] = 1` and wait for appropriate dv/dt time.
3. Set `DIAG_COMP_CTRL3[COMP_ADC_SEL2:0] = CBFET check` (that is, 0b100) and set `[COMP_ADC_GO] = 1`.
4. The device turns on the CBFET configured in the above step and compares the AUXCELL measurement (through CB channel) < half of the VCELL measurement (through VC channel). Only the CBFETs that are enabled are checked.
5. The CBFET comparison is completed when `ADC_STAT2[DRDY_CBFET] = 1`.
6. Repeat this procedure for other set of CBFET test. To turn off the CBFET enabled for this test, MCU clear the `DIAG_CBFET1` and `DIAG_CBFET2` registers then set the `[CBFET_CTRL_GO] = 1`. Otherwise, exiting from the CB pause state by sending `[CB_PAUSE] = 0` will resume the regular balancing which turns off the CBFETs enabled for this test and resume on the CBFETs that are set for balancing.

Host checks the `FAULT_COMP_CBFET1` and `FAULT_COMP_CBFET2` registers for the comparison result. Repeat the steps to compare the remaining CBFETs.

ADC comparison abort conditions:

The device will not start the CBFET comparison under the invalid conditions listed below. When the comparison is aborted, the `FAULT_COMP_MISC[COMP_ADC_ABORT] = 1`, `[DRDY_AUX_CEL] = 1`, `[DRDY_CBFET] = 1`, and `FAULT_COMP_CBFET1/2 = 0xFF`. If `[AUX_CELL_SEL4:0]` is set to locked at a single channel, the `AUX_CELL_HI/LO` registers will be reset to default value 0x8000 if the comparison run is aborted.

Invalid conditions or settings which will prevent the start of the cell voltage measurement comparison:

- Invalid `[AUX_CELL_SEL]` setting which results in no AUX ADC measurement on the selected channel. The `AUX_CELL_HI/LO` registers are kept in default value.
- Channel higher than the `NUM_CELL` configuration is selected.
- Invalid `BBVC_POSN` setting:
 - Adjacent channels are enabled in the `BBVC_POSN1/2` registers.
 - `BBVC_POSN2[CELL1]` is enabled.
 - More than two channels are selected in `BBVC_POSN1/2`.
 - `[AUX_CELL_SEL]` is locked to any of the selected channels in `BBVC_POSN1/2`.
- Main or AUX ADCs are off or not set in continuous mode.
- CB is running and it is not in pause mode.
- More than eight CBFETs are enabled, or more than two consecutive CBFETs are enabled in `DIAG_CBFET_CTRL1/2` registers.

6.3.6.4.6.4 VC and CB Open Wire Check

The device can detect an open wire connection on the VC and CB pins. A current sink is connected to each VC and CB pin, except VC0 and CB0 pins which are connected with a current source.

When the current sink (or current source) is enabled and if there is an open wire connection, the external differential capacitor will be depleted and the cell voltage measurement will drop to an abnormal level over time. Similar detection concept applies to the VC0 and CB0 pins with a current source. If there is an open wire connection, the VC0 or CB0 will be pulled up by the current source, resulting in a reduced cell voltage measurement over time.

When the diagnostic comparison is enabled, the device will compare the cell voltage measurement from Main ADC (for VC pins open wire detection) against a host-programmed threshold; or comparing the AUX CELL measurement from the AUX ADC (for CB pins open wire detection) against a host-programmed threshold.

If MCU lock to a single CB channel though `[AUX_CELL_SEL]` before starting the CB open wire check. The device will report the AUXCELL measurement used for the check comparison. The value is reported in `DIAG_AUX_HI/LO` registers. Since there is no single channel lock mechanism in Main ADC, VC channel measurement used for VC open wire will not be reported in `DIAG_MAIN_HI/LO` registers.

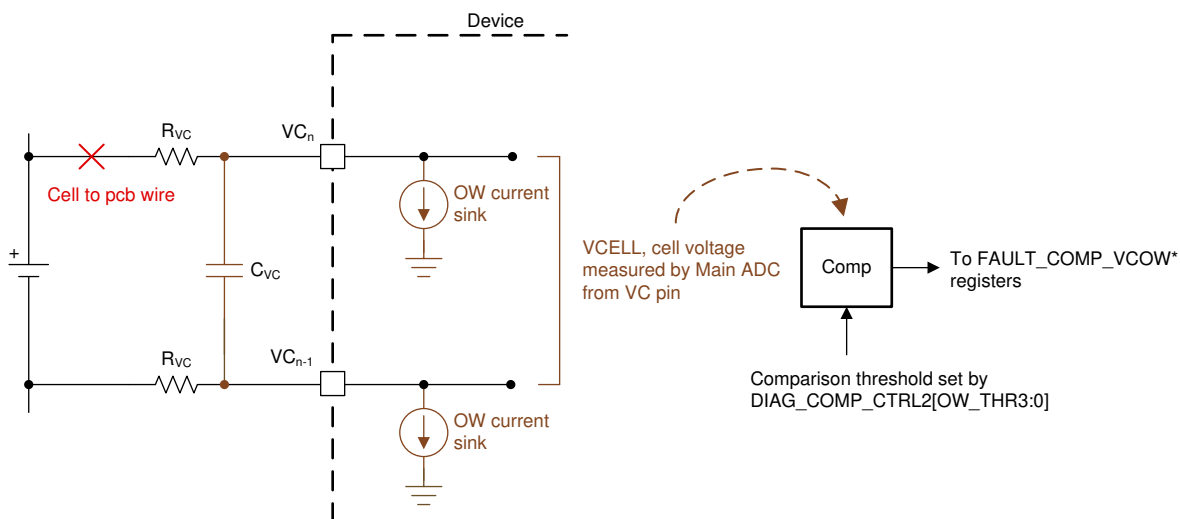


Figure 6-51. Open Wire Detection

Before starting the open wire comparison, host ensures:

- For VC open wire detection, Main ADC is running in continuous mode.
- For CB open wire detection, AUX ADC is running in continuous mode

- Configured in the *ADC_CTRL2[AUX_CELL_SEL4:0]* to select the AUXCELL channels
- Select the desired settling time for the AUX CELL channel through *ADC_CONF1[AUX_SETTLE1:0]*.
- Configure the open wire detection threshold through *DIAG_COMP_CTRL2[OW_THR3:0]*.

To start the open wire comparison:

1. Turn on the VC pins (or CB pins) current sink or source through *DIAG_COMP_CTRL3[OW_SNK1:0]*.
2. Wait for dV/dt time of the external capacitor to deplete to the detection threshold if there is an open wire fault.
3. For VC open wire detection, select *DIAG_COMP_CTRL3[COMP_ADC_SEL2:0] = OW VC* check (that is, 0b010) and set *[COMP_ADC_GO] = 1*. Or for CB open wire detection, *[COMP_ADC_SEL2:0] = OW CB* check (that is, 0b011).
4. The device compares all active VCELL measurement (for VC open wire) or AUX CELL measurement (for CB open wire) against the *[OW_THR3:0]* threshold setting.
5. When the comparison is completed, *ADC_STAT2[DRDY_VCOW] = 1* for VC open wire (or *[DRDY_CBOW] = 1* for CB open wire).
6. Host then turns off all current sinks and sources through *DIAG_COMP_CTRL3[OW_SNK1:0]*.

Host checks the *FAULT_COMP_VCOW1/2* or *FAULT_COMP_CBOW1/2* registers for the comparison result.

6.3.7 Bus Bar Support

The device supports bus bar measurement in two types of connections:

- Bus bar connected to a dedicated bus bar channel through BBP and BBN pins
- Bus bar connected to a VC channel

A total of three bus bars can be connected to a single device, one through BBP/N pins and two through VC channels. [Table 6-32](#) shows the difference between the two connection methods. Details are described in the later subsections.

Table 6-32. Bus Bar Connection Methods

| Supporting Feature/ Limitation | Bus Bar Connected Across BBP/BBN Pins | Bus Bar Connected Individually Across VC Pins |
|---|---|---|
| Number of bus bar can be connected per device | 1 | 2 |
| Connection channel | Can be connected to any channel but the bottom one | Can be connected to any channel but the bottom one |
| Bus bar measurement | Yes, result is output to <i>BUSBAR_HI/LO</i> registers | Yes, result is output to <i>VCELLx_HI/LO</i> registers, where x is the VC channel the bus bar is connected to |
| Integrated filters to bus bar measurement | Yes, same front end filters as the regular cell channels and post ADC digital LPF. BBP/N channel is x5 gain with ±1-V input range. Dedicated digital LPF setting separated from the LPF setting used for cell voltage measurements. | Yes, same front end filters as the regular cell channels and post ADC digital LPF. Use the same cell voltage LPF setting for bus bar measurement. |
| Host requires to adjust cell measurement adjustment | Yes, a cell + bus bar are sharing a single VC channel using this method. Host needs to separate out the bus bar measurement to obtain the actual cell measurement on the shared channel. | No, cell and bus bar are connected to their own VC channels and measurements are reported separately |
| Cell balancing limitation | No CB limitation, but host turns on adjacent CBFETs when balancing a cell above the bus bar connected channel | No CB limitation but require to float the upper CB pin on the bus bar connected channel |

The device supports bus bar measurement in a connection when a bus bar is connected to a VC channel. A total of two bus bars can be connected to a single device through the VC channels. [Table 6-33](#) shows the details as described in the later subsections.

Table 6-33. Bus Bar Connection Methods

| Supporting Feature/ Limitation | Bus Bar Connected Individually Across VC Pins |
|---|---|
| Number of bus bar can be connected per device | 2 |
| Connection channel | Can be connected to any channel but the bottom one |
| Bus bar measurement | Yes, result is output to <i>VCELLx_HI/LO</i> registers, where x is the VC channel the bus bar is connected to |
| Integrated filters to bus bar measurement | Yes, same front end filters as the regular cell channels and post ADC digital LPF. Use the same cell voltage LPF setting for bus bar measurement. |
| Host requires to adjust cell measurement adjustment | No, cell and bus bar are connected to their own VC channels and measurements are reported separately |
| Cell balancing limitation | No CB limitation but require to float the upper CB pin on the bus bar connected channel |

6.3.7.1 Bus Bar on BBP/BBN Pins

The device provides a dedicated bus bar channel through BBP/BBN pins for bus bar connection and measurement. It is a floating channel, allowing the bus bar to be connected to any cell except the bottom cell of a module. Using the bus bar channel maximizes the use of cell channels in the device across different module sizes.

6.3.7.1.1 Typical Connection

With the bus bar connected to BBP/BBN pins, it is intended to allow a single cell channel (VC channel) to be shared with a cell + a bus bar (see [Figure 6-52](#) (a) connection). Usually, such connection introduced additional IR error to the cell measurement to the system. The dedicated bus bar channel through BBP/BBN pins supported in the device allows the host to measure the bus bar voltage to obtain the actual cell measurement.

[Figure 6-52](#) (a) connection applies to bus bar connecting to any middle VC channel. That is, in a single device, there is a cell connected above and below the BBP/BBN channel. To support the hotplug on the bus bar channel, the device only requires a 400-Ω filter resistor each on the BBP/N pins and a 0.47-μF/16-V differential capacitor across the BBP/N pins.

If the bus bar connected to BBP/N is placed at the top of a module (see [Figure 6-52](#) (b) connection), such connection is the exception in the BBP/N case that a cell channel is not being shared. In this connection, actual cell measurements are made through the VC channels and host does not require additional calculations.

[Figure 6-52](#) (b) connection applies to bus bar connected to top of the module, where in a single device, no cell is connected above the bus bar. To support the hotplug on the bus bar channel, besides the 400-Ω filter resistor each on BBP/N pins and a 0.47-μF/16-V differential capacitor across BBP/N pins, and additional 0.47-μF/16-V differential capacitor is needed to connect from BBN to the top CB pins. This additional capacitor forms a complete capacitor ladder from all cells in the module to the bus bar, allowing high spike voltage during the hotplug to distributor across the capacitor ladder.

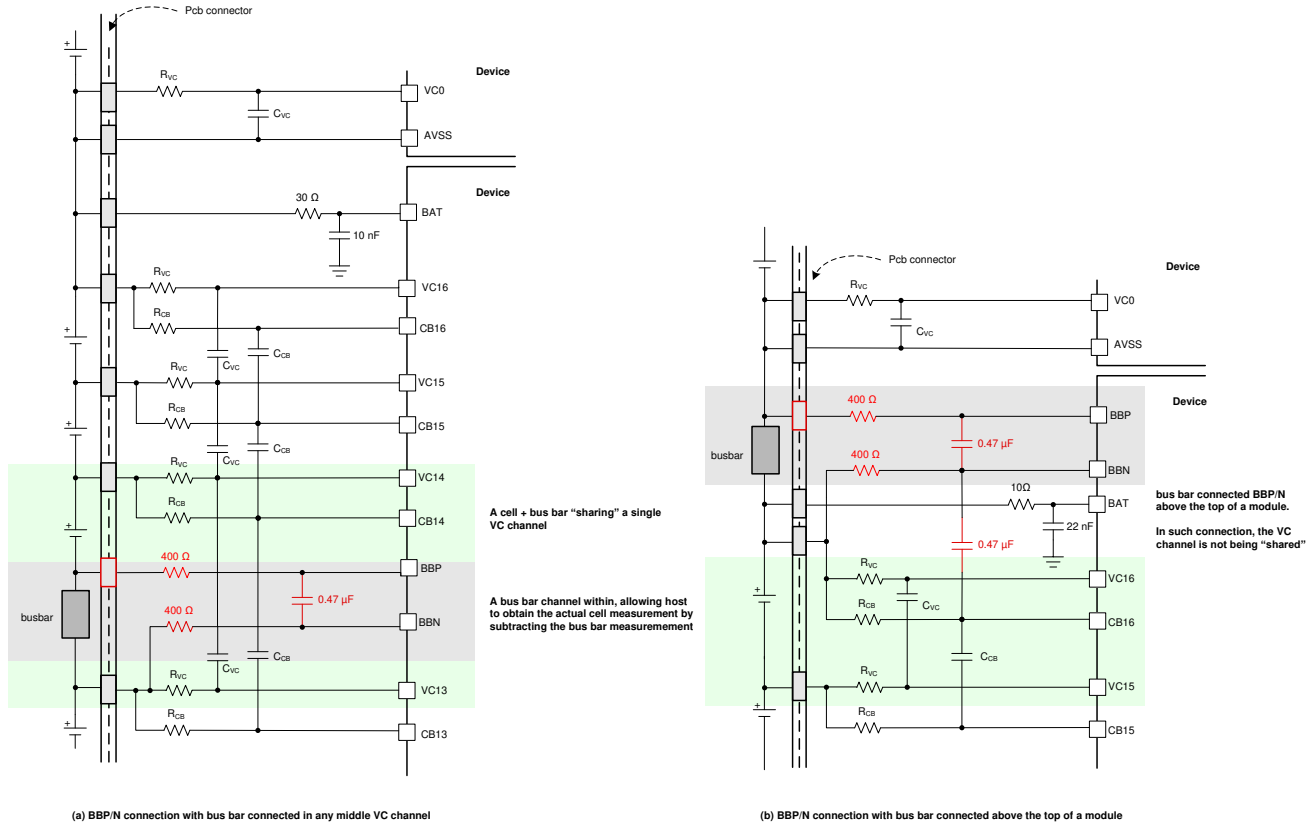


Figure 6-52. Bus Bar Connected Across BBP and BBN Pins

6.3.7.1.2 Bus Bar Measurement

The differential measurement across (BBP–BBN) is measured by the Main ADC and AUX ADC. See [Section 6.3.2.1.1](#) and [Section 6.3.2.2.1](#) for details. Use the `BBP_LOC` register to indicate which VC channel is shared with the BBP/N connection. This information enables the device to better have better common mode correction for the final ADC measurement. Host will be aware that additional IR error is introduced to the shared VC channel. If OVUV protector is enabled, this shared channel may trigger earlier OV or UV detection due to the additional IR increase (during charge) or decrease (during discharge) to the shared channel measurement.

6.3.7.1.3 Cell Balancing Handling

Because the bus bar is shared with a cell to a cell channel, there is no special handling on the cell balancing control. Host will be aware that additional IR error is introduced to the VCB_DONE detection (through VC channel) on the shared channel.

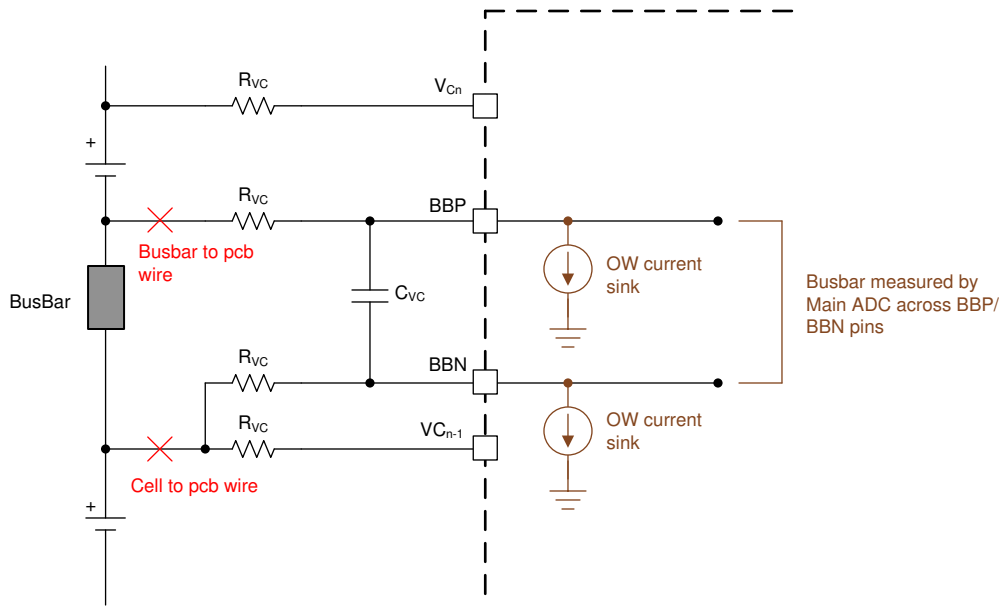


Figure 6-55. Current Sink for BBP/N Open Wire Detection

6.3.7.2 Bus Bar on Individual VC Channel

Besides connecting bus bar through BBP/N, the device also supports bus bar connection to an individual VC channel. All VC channels, except the bottom channel (VC1-VC0), support $-2V$ to $5V$ measurement.

The device supports bus bar connection to an individual VC channel. All VC channels, except the bottom channel (VC1-VC0), support $-2V$ to $5V$ measurement.

When bus bar is connected to an individual VC channel, host indicates the bus bar position in the *BBVC_POSN1* and *BBVC_POSN2* registers. The following configuration is not supported for bus bar connection through individual VC channel. Configuring *BBVC_POSN1* register with such configuration can cause error in balancing, OVUV detection and cell voltage measurement comparison check.

- Bottom channel does not support bus bar connection. That is, *BBVC_POSN1[CELL1]* must be 0.
- Maximum of two bus bars can be connected through this connection. That is, only two bits are set to 1 in the *BBVC_POSN1* registers.
- Bus bar cannot be connected to the adjacent channels.

6.3.7.2.1 Typical Connection

With bus bar connected to a VC channel individually, the upper CB pin on that channel is left floating to avoid forward biasing the internal CBFET (see [Figure 6-56 \(a\)](#) connection). This connection applies to bus bar connecting to any middle VC channel individually. That is, in a single device, there is a cell connected above and below the VC channel with bus bar connected. To ensure hotplug performance, the CB channel where the bus bar is connected will still have the differential capacitor even if the upper CB pin is floating. This capacitor forms a complete capacitor ladder from all cells and the bus bar connected to the device, allowing high-voltage spike during hotplug to distribute across the capacitor ladder.

If bus bar is connected to above the top of a module to an individual VC channel (see [Figure 6-56 \(b\)](#) connection), the upper CB pin on that channel is left floating but the CB differential capacitor will still be connected. Additionally, an additional RC filter is connected from the top CB pin to the BAT pin. This additional RC filter (using the same RC values as the other RC filter on the CB pins) is to ensure a complete capacitor ladder is formed for the device to distribute the high voltage spike with the same RC constant as the reset of the CB pins during hotplug event.

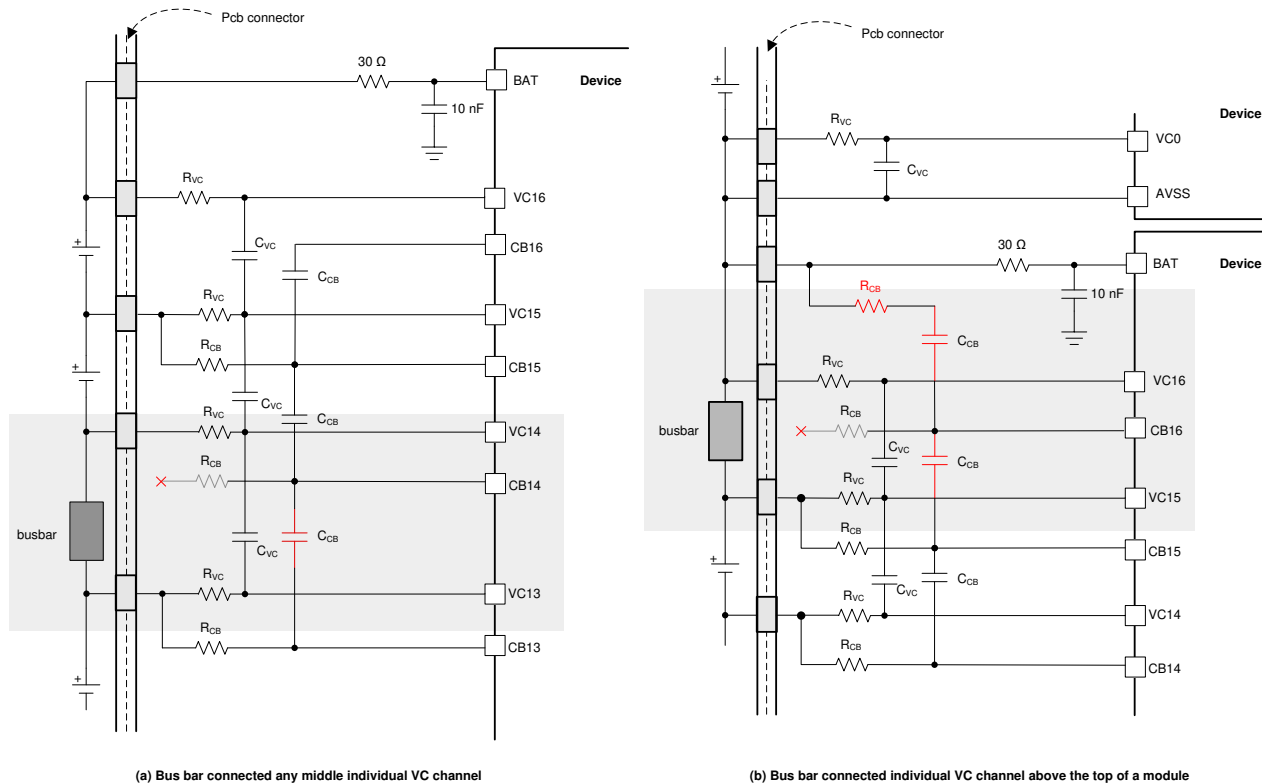


Figure 6-56. Bus Bar Connected to Individual VC Channel

6.3.7.2.2 Bus Bar Measurement

Bus bar measurement is performed through Main ADC measurement as one of the VC channels. The result is reported to $VCELLx_HI/LO$ register, where x is the channel connected with the bus bar. Digital LPF is enabled and applied as the rest of the VC channel measurement configuration.

The VC channel indicated for bus bar connection (through $BBVC_POSN1/2$ registers) will be skipped for VCB_DONE check during cell balancing, OV and UV detection when OVUV protectors are enabled, and will have special handling during cell voltage measurement comparison check.

6.3.7.2.3 Cell Balancing Handling

Because the upper CB pin is open on the channel where bus bar is connected, to balance the cell connected above bus bar, host turns on the adjacent CBFET and configures with the same timer setting.

Host configures $BBVC_POSN1/2$ register to indicate the bus bar connection. This information is used to avoid the channel connected with bus bar to trigger a VCB_DONE detection and turn off its CBFET, which disconnects the balancing path for the cell above the bus bar.

The balancing of the cell above the bus bar is still terminated based on the timer and cell voltage threshold, which its CBFET will be turned off when one of the stop conditions is met. The balancing path is disconnected even if the CBFET on the bus bar connected channel remains on.

Note

The CBFET on the bus bar connected channel will be on until the timer expired. This may lead to a delayed flagging of the $[CB_DONE] = 1$ even if the actual cell balancing is completed.

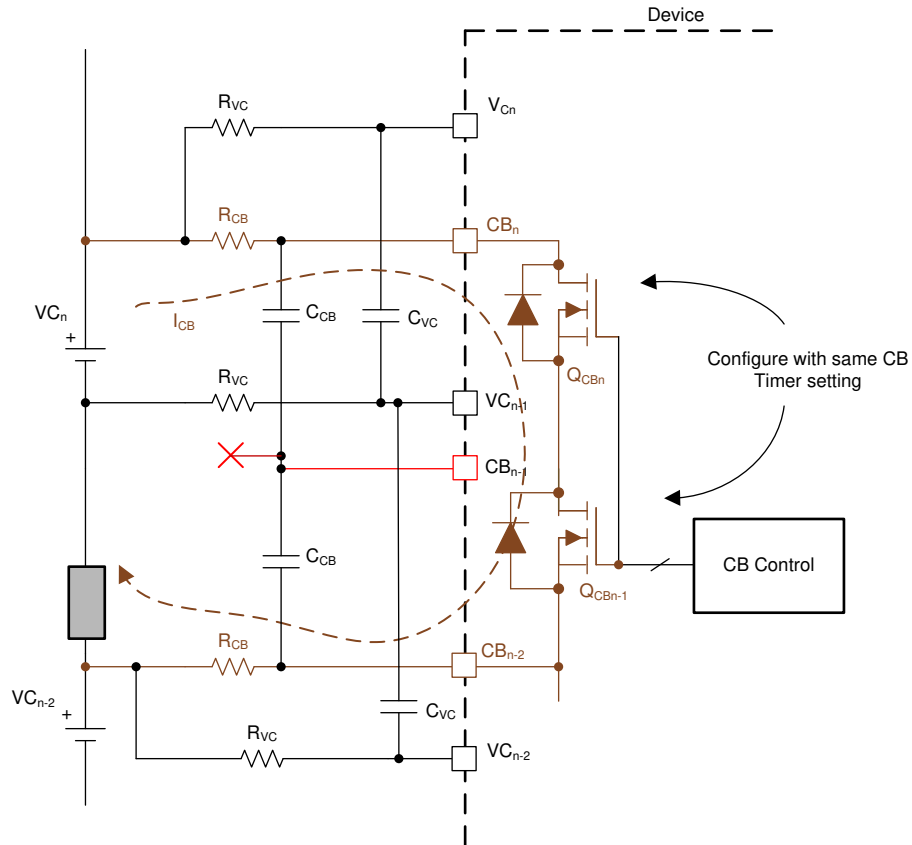


Figure 6-57. Cell Balancing with Bus Bar Connected to Individual VC Channel

6.3.7.2.4 Cell Voltage Diagnostic Control

The cell voltage comparison check is still performed by checking the Main ADC measurement versus the AUX ADC measurement. Because the upper CB pin of the CB channel, where a bus bar is connected, is open, the device handles the comparison check by comparing a sum of (cell + bus bar) from Main ADC versus sum of (cell + bus bar) from AUX ADC instead.

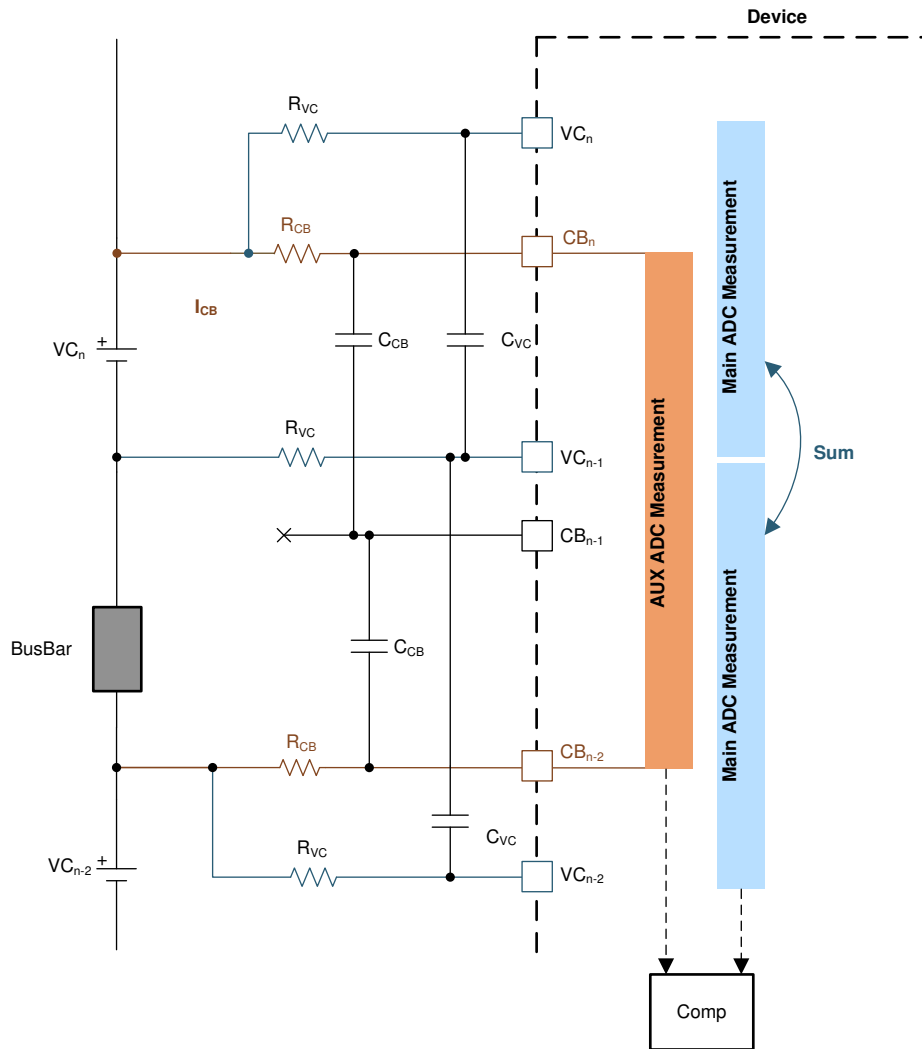


Figure 6-58. Cell Measurement Check with Bus Bar Connected to Individual VC Channel

6.4 Device Functional Modes

The device has three power modes plus an POR state.

- **POR:** This is not a power mode. This is a condition in which the voltage at the BAT pin is less than $V_{BAT\ min}$, and all circuits including the AVAO_REF block in the device are powered off.
- **SHUTDOWN:** This is the lowest power mode. AVDD, DVDD and CVDD supplies are off. Only a gross regulation at LDOIN pin is maintained. CVDD pin is will have a similar voltage as the LDOIN pin through internal circuit in order to support WAKE detection.
- **SLEEP:** This is the low power operation mode. Only limited functions are available.
- **ACTIVE:** This is the full power operation mode. All functions are supported under this state.

The various functions supported under different power modes are summarized in [Table 6-34](#) and the power state diagram is shown in [Figure 6-59](#).

Table 6-34. Active Functions Summary

| Functional Block | SHUTDOWN | SLEEP | ACTIVE | POR |
|--|----------|-------|--------|---|
| Main ADC | | | √ | This is not a power state. All circuits are off. A sufficient voltage on VBAT will POR the device and put it to SHUTDOWN mode |
| AUX ADC | | | √ | |
| OV/UV protector | | √(1) | √ | |
| OT/UT protector | | √(1) | √ | |
| Cell Balancing | | √(1) | √ | |
| OTCB Detection | | √(1) | √ | |
| Module Balancing (via control through MB_TIMER_CTRL) | | | √ | |
| UART | | | √ | |
| Comm Vertical Communication | | | √ | |
| Fault Status and NFAULT Communication | | √ | √ | |
| Comm timeout | | | √ | |
| SLEEP timeout | | √ | | |
| Thermal Shutdown Detection | | √ | √ | |
| SPI Controller | | | √ | |
| OTP programming | | | √ | |
| Always-on block to detect POR of the device | √ | √ | √ | |

(1) To enable cell balancing, OV/UV or OT/UT protector(s) in SLEEP mode, host must enable the function(s) in ACTIVE mode first, then put the device to SLEEP.

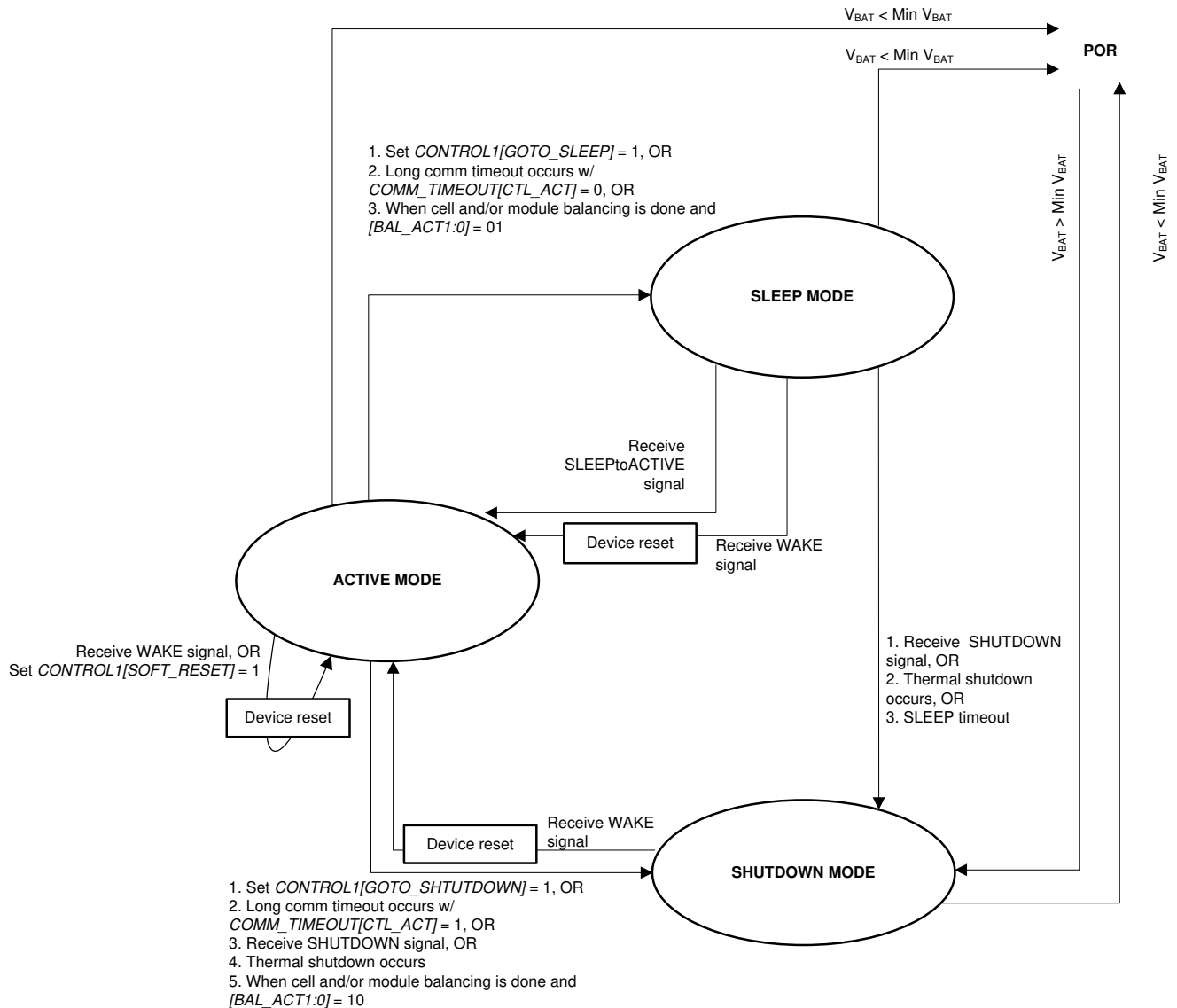


Figure 6-59. Power State Diagram

6.4.1 Power Modes

6.4.1.1 SHUTDOWN Mode

This is the lowest power mode. In SHUTDOWN mode, most of the functions are off. The device remains idle to simply monitor the WAKE ping/tone (see [Section 6.4.3](#) for details) to wake up from this state. Only a gross regulation on LDOIN and CVDD pins are maintain for WAKE ping/tone detection.

6.4.1.1.1 Exit SHUTDOWN Mode

Communication is not supported in SHUTDOWN mode, host must send a WAKE ping or WAKE tone to enter ACTIVE mode. Once device transitions from SHUTDOWN mode to ACTIVE mode, the following table indicates the expected fault bits being set under such transition has occurred.

Table 6-35. Expected Fault Bit After Device Wake From SHUTDOWN

| Device Position In The Daisy Chain | Expected Fault Bits After Waking Up From SHUTDOWN | |
|------------------------------------|---|------------------------------------|
| Base device | $FAULT_SYS[DRST] = 1$ | Digital reset by the wake ping |
| | $FAULT_COMM3[FCOMM_DET] = 1$ | $[DRST] = 1$ from the upper device |
| | $FAULT_COMM1[COMMCLR_DET] = 1$ | UART engine is reset |
| Stack device (except top of stack) | $FAULT_SYS[DRST] = 1$ | Digital reset by the wake tone |
| | $FAULT_COMM3[FCOMM_DET] = 1$ | $[DRST] = 1$ from the upper device |
| Top of stack device | $FAULT_SYS[DRST] = 1$ | Digital reset by the wake tone |

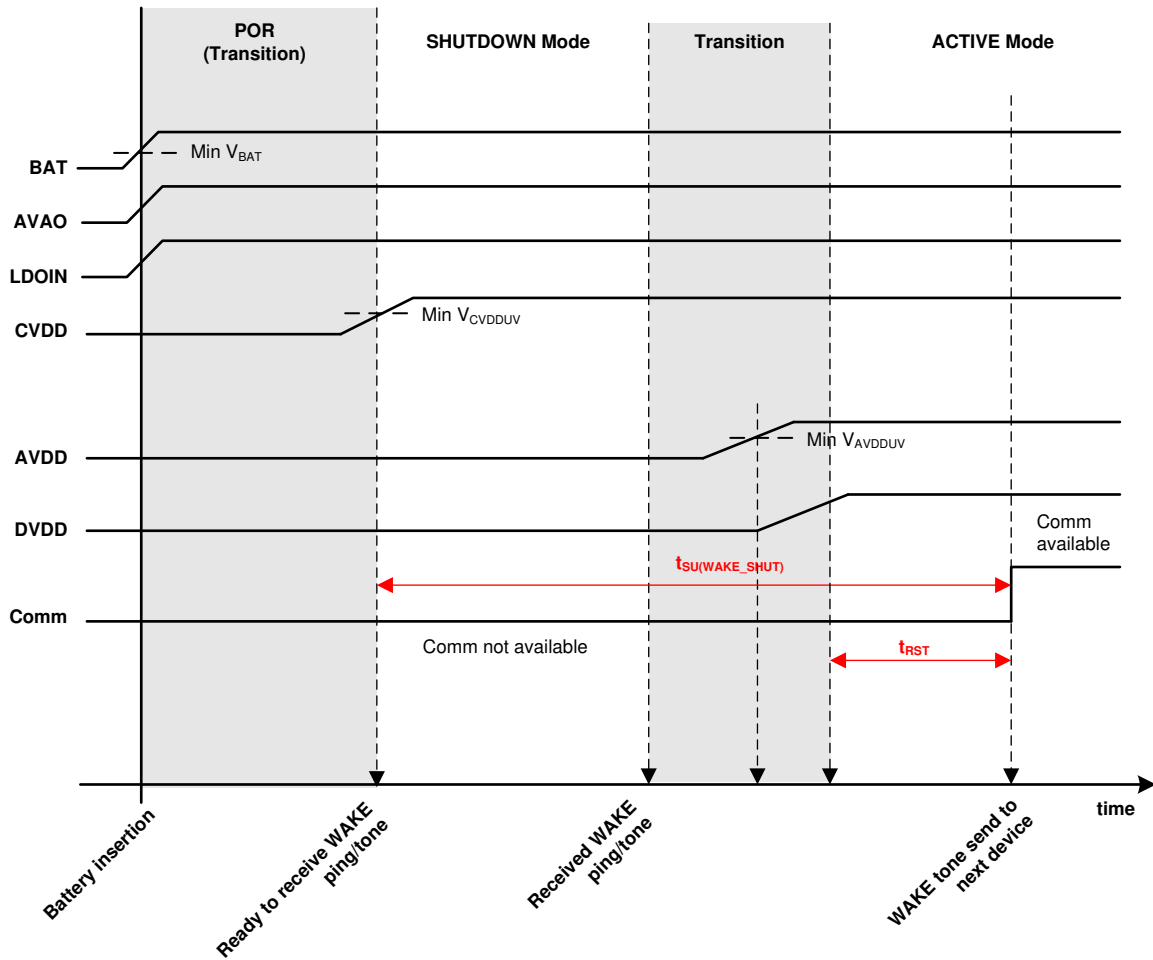


Figure 6-60. SHUTDOWN to ACTIVE Mode Transition

6.4.1.1.2 Enter SHUTDOWN Mode

During normal operation, host puts the device in SHUTDOWN mode through communication by sending $CONTROL1[GOTO_SHUTDOWN] = 1$. In a daisy chain configuration, using broadcast write to send this command will put all devices in the daisy chain in SHUTDOWN mode.

The device can also enter SHUTDOWN mode by one of the following conditions:

- Communication timeout: automatically transitions from ACTIVE mode to SHUTDOWN mode if there is no valid communication for the configured time. Host can enable this option through the `COMM_TIMEOUT_CONF` register.
- SLEEP mode timeout: automatically transitions from SLEEP mode to SHUTDOWN mode if device is in SLEEP mode for the configured time. Host can enable this option through `PWR_TRANSIT_CONF[SLP_TIME2:0]`.
- Upon balancing completion: automatically enter SHUTDOWN mode when all balancing of the devices is completed. See [Section 6.3.3](#) for details. This option can result with devices in different power modes for a period of time in a daisy chain configuration.
- Thermal shutdown: shuts down the device when the internal die temperature is greater than T_{SHUT}
- SHUTDOWN or HW_RESET ping/tone: These pings/tones are used as a recovery attempt on a loss communication situation. A SHUTDOWN ping/tone puts the device into SHUTDOWN mode without using communication, forcing most of the circuits to be off. A more aggressive recovery attempt uses HW_RESET ping/tone which turns off all circuits except a bandgap and restarts the device in SHUTDOWN mode.

6.4.1.2 SLEEP Mode

This is the low power operation mode. In SLEEP mode, all internal power supplies are still on, but functions are limited to cell balancing, OVUV and OTUT protectors, Heartbeat/Fault Tone/NFAULT transmission and detection.

6.4.1.2.1 Exit SLEEP Mode

Because host cannot communicate to the device, to exit SLEEP mode, host must send either a WAKE ping/tone or SLEEPtoACTIVE ping/tone to transition to ACTIVE mode. A WAKE wakes up and resets the device, which host will need to reconfigure the device setting; a SLEEPtoACTIVE only wakes up the device.

6.4.1.2.2 Enter SLEEP Mode

The device can enter SLEEP mode from ACTIVE mode only. During normal operation, host puts the device to SLEEP mode through communication by sending `CONTROL1[GOTO_SLEEP] = 1`. In a daisy chain configuration, using broadcast write to send this command will put all devices into SLEEP mode.

The device can also enter SLEEP mode in the following condition:

- Communication timeout: automatically transitions from ACTIVE mode to SLEEP mode if there is no valid communication for the configured time. Host can enable this option through the `COMM_TIMEOUT_CONF` register.

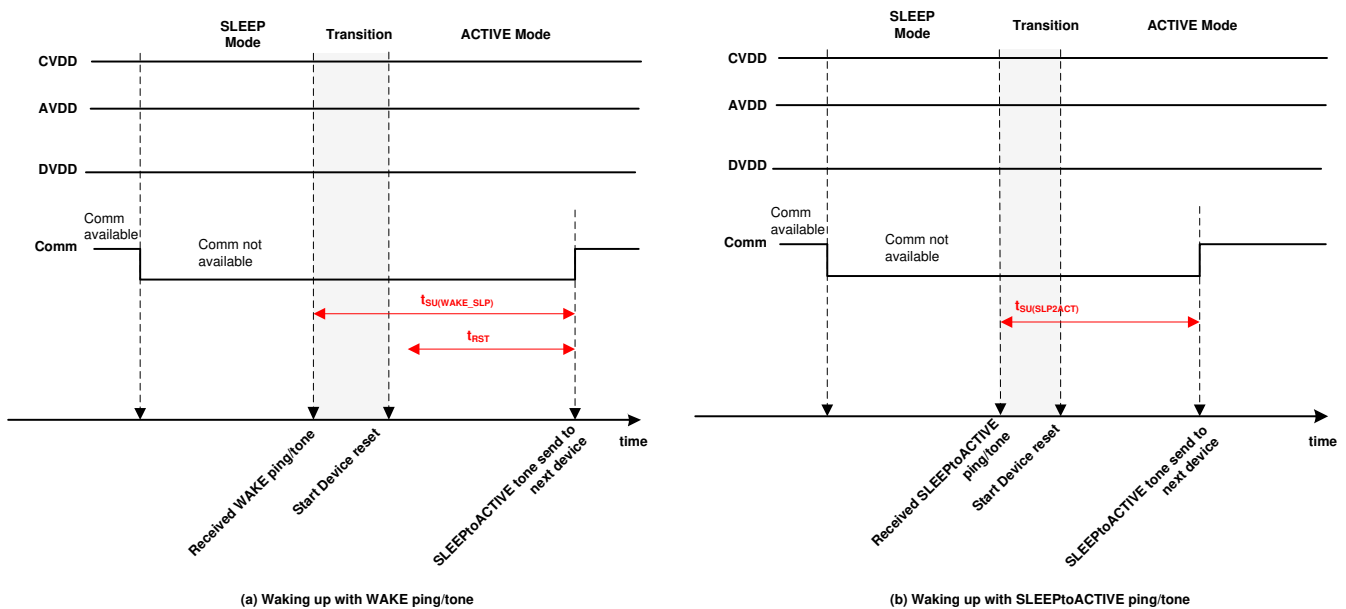


Figure 6-61. SLEEP to ACTIVE Mode Transition

6.4.1.3 ACTIVE Mode

This is the operation mode with full functionality support. Host can communicate to the device with full control on various features as well as performance diagnostic in this mode.

6.4.1.3.1 Exit ACTIVE Mode

From ACTIVE mode, device can enter SLEEP mode or SHUTDOWN mode through command, ping/tone, timer, or specific event. See [Section 6.4.1.1](#) and [Section 6.4.1.2](#) for details.

6.4.1.3.2 Enter ACTIVE Mode from SHUTDOWN Mode

The device can transition to ACTIVE mode from SHUTDOWN mode only through a WAKE ping/tone. Once in ACTIVE mode, the host clears some of the reset-related faults that are expected faults (see [Section 6.4.1.1](#) for details), indicating a POR on certain blocks due to the transition from SHUTDOWN mode to ACTIVE mode. Registers are reset to default; the OTP shadow registers are reloaded with the OTP programmed values.

6.4.1.3.3 Enter ACTIVE Mode from SLEEP Mode

From SLEEP mode, either a WAKE or SLEEPtoACTIVE ping/tone can put the device in ACTIVE mode. A WAKE ping/tone generates a digital reset to the device. Because the LDO supplies remain on during SLEEP mode, only the $FAULT_SYS[DRST] = 1$ is set, indicating a digital reset has occurred. Certain expected faults related to being reset are set. See SHUTDOWN mode for details. Registers are reset to default, the OTP shadow registers are reloaded with the OTP programmed values.

If a SLEEPtoACTIVE ping/tone is used to wake up the device from SLEEP mode to ACTIVE mode, the device enters ACTIVE mode without a digital reset, but the UART engine is reset; therefore, in the base device, the $FAULT_COMM1[COMMCLR_DET] = 1$ and the host clears it after entering ACTIVE mode.

6.4.2 Device Reset

There are several conditions which the device will go through: a digital reset, putting the registers to their default settings and reloading the OTP.

- A WAKE ping/tone is sent to transition from SHUTDOWN mode or SLEEP mode to ACTIVE mode.
- A WAKE ping/tone is received in ACTIVE mode.
- The $CONTROL1[SOFT_RESET] = 1$ command is sent in ACTIVE mode.
- A HW_RESET ping/tone is sent under any power mode. This generates a POR-like event to the device. Upon the detection of a HW_RESET ping/tone, the device will turn off all internal blocks except a bandgap for t_{HWRST} duration. Afterward, the device will restart in SHUTDOWN mode.
- Internal power supply faults. See [Section 6.3.6.4](#) for details.
 - AVDD UV, DVDD UV is detected.
- A HFO or LFP watchdog fault will reset the digital.

Apart from the full reset cases, the following conditions will only reset the UART engine. These conditions mainly affect the base device because UART is used to talk to the host MCU. In the base device, the $FAULT_COMM1[COMMCLR_DET] = 1$ will be set. These conditions do not affect the stack devices because UART is inactive in those devices.

- A SLEEPtoACTIVE ping is sent to transition from SLEEP mode to ACTIVE mode.
- The following conditions not only clear the UART engine and set the $[COMMCLR_DET] = 1$, they also set $FAULT_COMM1[STOP_DET] = 1$ as an indication that an unexpected UART STOP is detected.
 - A SLEEPtoACTIVE ping is sent in ACTIVE mode.
 - A COMM CLEAR signal is sent. This is a dedicated signal to clear the UART engine and instruct the engine to look for a new start of communication frame. See [Section 6.3.6.1.1.1](#) for more details.

6.4.3 Ping and Tone

In the noncommunicable conditions, such as in SHUTDOWN or SLEEP mode, or in the loss of communication situations when host would send an instruction for a reset or power down as a communication recovery attempt, a Ping or Tone is a form of communication to the device for a specific action.

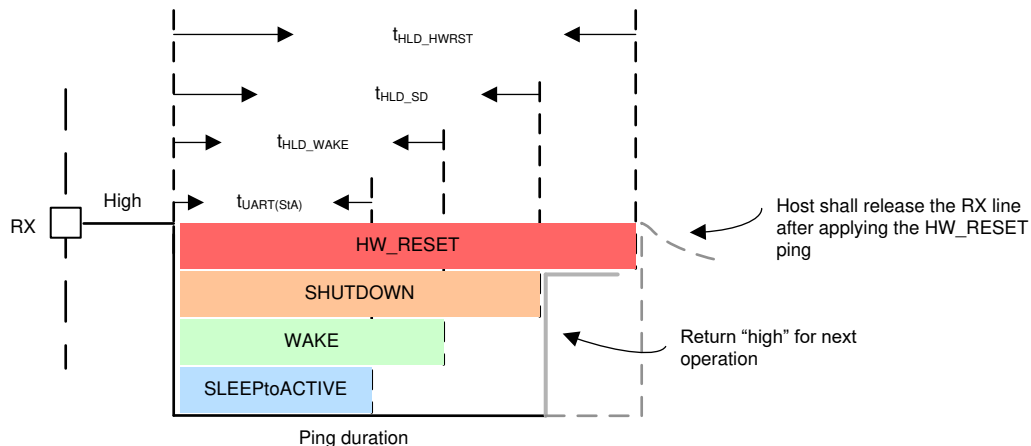
Table 6-36. Supported Ping and Tone in Different Power Modes

| Ping/Tone Detection | Detected Pin(s) | SHUTDOWN | SLEEP | ACTIVE |
|---------------------|-----------------|----------|-------|---|
| SHUTDOWN ping | RX | | √ | √ |
| SLEEPtoACTIVE ping | RX | | √ | √ |
| WAKE ping | RX | √ | √ | √ |
| HW_RESET ping | RX | | √ | √ |
| SHUTDOWN tone | COMH/L | | √ | √ |
| SLEEPtoACTIVE tone | COMH/L | | √ | √ |
| WAKE tone | COMH/L | √ | √ | √ |
| HW_RESET tone | COMH/L | | √ | √ |
| Fault tone | COMH/L | | √ | Only fault tone detection is available. |
| HEARTBEAT | COMH/L | | √ | |

6.4.3.1 Ping

A ping is a specific high-low-high signal to the RX pin of the device. Ping is used on the base device as only the base device is connected to the host which the UART RX is accessible. The device detects different low times of the ping signal to differentiate the different ping signals.

The communication pings are referring to the WAKE ping, SLEEPtoACTIVE ping, SHUTDOWN ping, and HW_RESET ping. These pings instruct the device to a specific power mode when normal communication is not available. By definition, a COMM CLEAR signal on the RX pin is a form of a ping. Because a COMM_CLR is to clear the UART engine, this signal is covered in [Section 6.3.6.1.1.1](#).

**Figure 6-62. Communication Pings**

6.4.3.2 Tone

A tone is a fixed number of couplets (pulses) with a specified polarity (all “+” or all “-”) sent through the differential vertical interface COMH and COML ports. Tone is used on stack devices as only the COMH/L ports are accessible. The number of couplets for transmission is always greater than the number of couplets needed for detection.

There are four communication tones corresponding to the four communication pings. They are WAKE tone, SLEEPtoACTIVE tone, SHUTDOWN tone, and HW_RESET tone. In addition to the communication tones, there are two extra tones related to device fault status: Heartbeat tone and Fault tone. These two fault status tones are only available in SLEEP mode. See [Section 6.3.6.2.3.3](#) for details.

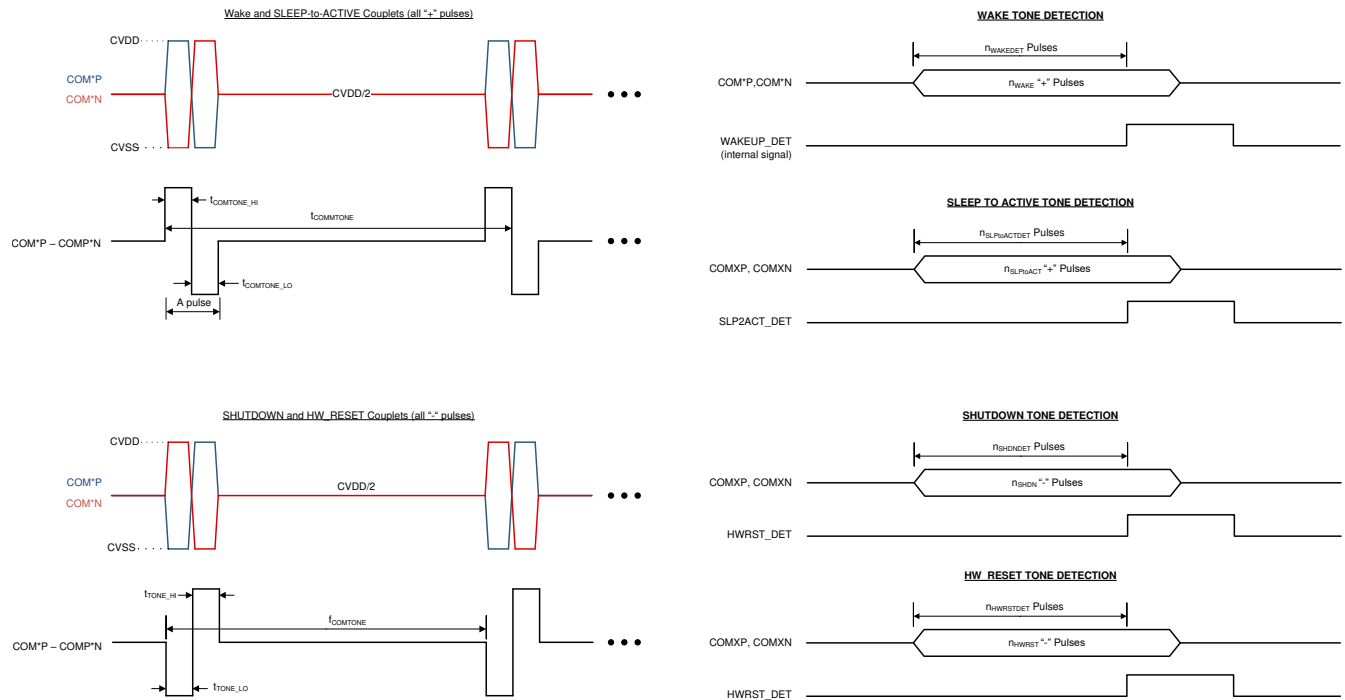


Figure 6-63. Communication Tones

6.4.3.3 Ping and Tone Propagation

Propagates:

The WAKE and SLEEPtoACTIVE pings/tones are part of the normal operation to wake up the device; hence, these two pings/tones can propagate to the next device in a daisy chain configuration. That is, when a device receives a WAKE ping/tone, it generates a WAKE tone and forwards it to the next device. Similar action applies to SLEEPtoACTIVE ping/tone.

The direction of the tone forwarding follows the communication direction, which is set by the *CONTROL1[DIR_SEL]* bit. See [Section 6.3.6.1](#) for more details. The detection of the tone is supported from the COMH and COML ports on stack devices regardless of the *[DIR_SEL]* setting. This does not apply to base device because base device detects pings instead.

During normal operation, host can simply send a WAKE or SLEEPtoACTIVE ping to the base device and the corresponding tone will be generated to the rest of the stack devices. During system development, if there is a need to send WAKE or SLEEPtoACTIVE to only some of the devices in the daisy chain, host can use the *CONTROL1[SEND_WAKE]* or *CONTROL1[SEND_SLPTOACT]* bit. Device that receives this command will send the corresponding tone to the next device in the daisy chain. Because the WAKE and SLEEPtoACTIVE tones propagate, the rest of the daisy chain connected above also receives the corresponding tone.

Does Not Propagate:

The SHUTDOWN and HW_RESET pings/tones are mostly used as a communication recovery attempt. Hence these pings/tones do not propagate. That is, when a device receives a SHUTDOWN ping/tone, it starts the shutdown process but the device does not generate another SHUTDOWN tone to the next device. Similar action applies to HW_RESET ping/tone.

For a base device, as RX pin is connected to the host, SHUTDOWN or HW_RESET ping can be used on the base device. For stack devices, it is required at least one stack device is connected to the problem device is communicable. Host has to talk to the neighboring device and sets the *CONTROL1[SEND_SHUTDOWN] = 1* or *CONTROL2[SEND_HW_RESET] = 1* to instruct the neighboring device to issue the corresponding tone to the problem device. The detection of the tone is supported from the COMH and COML ports on stack devices

regardless of the *[DIR_SEL]* setting. This does not apply to a base device because a base device detects pings instead.

Table 6-37. Ping and Tone Propagation Summary

| Ping/Tone | Propagable | Non-Propagable |
|---------------|--|---|
| WAKE | Receiving device will generate a WAKE tone to the next device | |
| SLEEPTOACTIVE | Receiving device will generate a SLEEPTOACTIVE tone to the next device | |
| SHUTDOWN | | Receiving device will initialize the shutdown process |
| HW_RESET | | Receiving device will initialize the HW reset process |

6.5 Register Maps

This section has three register map summary tables with registers listed per the order of the register address:

- The NVM (OTP) shadow registers. These read/write-able shadow registers are reset with OTP values programmed in the customer OTP space. To program the custom OTP space, host writes the desired values to these OTP shadow registers and follows the programming procedure. These registers are included in the OTP CRC check. If customer OTP space is not programmed. The shadow registers are loaded with factory configuration default value. If the OTP (either factory configuration default or value programmed in customer OTP space) is failing to load after a device reset, the shadow registers will be loaded with the hardware reset default value instead. The hardware reset default value and the factory configuration default values are the same for the majority of the OTP shadow registers. Only the *DIR0_ADDR_OTP*, *DIR1_ADDR_OTP*, *PWR_TRANSIT_CONF*, *CUST_CRC_HI/LO* registers have a reset value versus factory default, and are specified in [Section 6.5.1](#) and their register field descriptions.
- The Read/Write registers. These are registers that the host can read/write to during runtime. A device reset will put these registers back to their reset value.
- The Read registers. These are registers that the host only has read access. A device reset will put these registers back to their reset value.

The register summary tables use the following key:

- Addr = Register address
- Hex = Hexidecimal value
- NVM = Non-volatile memory (OTP) shadow registers
- RSVD = Reserved. Reserved register addresses or bits are not implemented in the device. Any write to these bits is ignored. Reads to these bits always return 0.
- OTP_SPARE: These are spare OTP and shadow register bits that are implemented in the device. These spare bits are included as part of the CRC calculation. These bits are read/write as normal, but do not perform any function or influence any device behaviors.
- OTP_RSVDn = OTP and shadowed registers that are implemented but are reserved for device internal usage, where n refers to the register address. MCU must keep these registers in their default value
- HW Reset default is the value loaded when digital resets (POR like event) whereas Factory Configuration Default is the default value loaded into the OTP cell if customer doesn't program it themselves. Customer cannot read the HW Reset value.

[Section 6.5.4](#) describes the definition of each bit in the registers. The registers in this section are grouped per functional blocks.

6.5.1 OTP Shadow Register Summary

| Register Name | Addr Hex | RW Type | Reset Value | Data | | | | | | | |
|-------------------------------|----------|---------|---|------------|-----------|---------------|----------|-------------|-----------|----------|-------|
| | | | | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
| DIR0_ADDR_OTP | 0 | NVM | HW Reset Default = 0x00 Factory Configuration Default = 0x01 | SPARE[1:0] | | ADDRESS[5:0] | | | | | |
| DIR1_ADDR_OTP | 1 | NVM | HW Reset Default = 0x00 Factory Configuration Default = 0x01 | SPARE[1:0] | | ADDRESS[5:0] | | | | | |
| DEV_CONF | 2 | NVM | 0x54 | RSVD | NO_ADJ_CB | MULTI_DROP_EN | FCOMM_EN | TWO_STOP_EN | NFAULT_EN | FTONE_EN | HB_EN |

| Register Name | Addr Hex | RW Type | Reset Value | Data | | | | | | | |
|------------------|----------|---------|---|-----------------|-------------|--------------|-----------------|-----------------|----------------|-----------|-----------|
| | | | | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
| ACTIVE_CELL | 3 | NVM | HW Reset Default = 0x00 Factory Configuration Default = 0x0A | SPARE[3:0] | | | | NUM_CELL[3:0] | | | |
| OTP_SPARE15 | 4 | NVM | 0x00 | SPARE[7:0] | | | | | | | |
| BBVC_POSN1 | 5 | NVM | 0x00 | CELL16 | CELL15 | CELL14 | CELL13 | CELL12 | CELL11 | CELL10 | CELL9 |
| BBVC_POSN2 | 6 | NVM | 0x00 | CELL8 | CELL7 | CELL6 | CELL5 | CELL4 | CELL3 | CELL2 | CELL1 |
| ADC_CONF1 | 7 | NVM | 0x00 | AUX_SETTLE[1:0] | | LPF_BB[2:0] | | | LPF_VCELL[2:0] | | |
| ADC_CONF2 | 8 | NVM | 0x00 | SPARE[1:0] | | | ADC_DLY[5:0] | | | | |
| OV_THRESH | 9 | NVM | 0x3F | SPARE | SPARE | OV_THR[5:0] | | | | | |
| UV_THRESH | A | NVM | 0x00 | SPARE | SPARE | UV_THR[5:0] | | | | | |
| OTUT_THRESH | B | NVM | 0xE0 | UT_THR[2:0] | | | OT_THR[4:0] | | | | |
| UV_DISABLE1 | C | NVM | 0x00 | CELL16 | CELL15 | CELL14 | CELL13 | CELL12 | CELL11 | CELL10 | CELL9 |
| UV_DISABLE2 | D | NVM | 0x00 | CELL8 | CELL7 | CELL6 | CELL5 | CELL4 | CELL3 | CELL2 | CELL1 |
| GPIO_CONF1 | E | NVM | 0x00 | FAULT_IN_EN | SPI_EN | GPIO2[2:0] | | | GPIO1[2:0] | | |
| GPIO_CONF2 | F | NVM | 0x00 | SPARE | SPARE | GPIO4[2:0] | | | GPIO3[2:0] | | |
| GPIO_CONF3 | 10 | NVM | 0x00 | SPARE[1:0] | | GPIO6[2:0] | | | GPIO5[2:0] | | |
| GPIO_CONF4 | 11 | NVM | 0x00 | SPARE[1:0] | | GPIO8[2:0] | | | GPIO7[2:0] | | |
| OTP_SPARE14 | 12 | NVM | 0x00 | SPARE[7:0] | | | | | | | |
| OTP_SPARE13 | 13 | NVM | 0x00 | SPARE[7:0] | | | | | | | |
| OTP_SPARE12 | 14 | NVM | 0x00 | SPARE[7:0] | | | | | | | |
| OTP_SPARE11 | 15 | NVM | 0x00 | SPARE[7:0] | | | | | | | |
| FAULT_MSK1 | 16 | NVM | 0x00 | MSK_PROT | MSK_UT | MSK_OT | MSK_UV | MSK_OV | MSK_COMP | MSK_SYS | MSK_PWR |
| FAULT_MSK2 | 17 | NVM | 0x00 | SPARE[1] | MSK_OTP_CRC | MSK_OTP_DATA | MSK_COMM3_FCOMM | MSK_COMM3_FTONE | MSK_COMM3_HB | MSK_COMM2 | MSK_COMM1 |
| PWR_TRANSIT_CONF | 18 | NVM | HW Reset Default = 0x18 Factory Configuration Default = 0x10 | SPARE[2:0] | | | TWARN_THR[1:0] | | SLP_TIME[2:0] | | |

| Register Name | Addr Hex | RW Type | Reset Value | Data | | | | | | | |
|--|----------|---------|-------------|--|---------------|------|------------|----------|---------------|------|------|
| | | | | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
| COMM_TIM EOUT_CON F | 19 | NVM | 0x00 | SPARE | CTS_TIME[2:0] | | | CTL_ACT | CTL_TIME[2:0] | | |
| TX_HOLD_ OFF | 1A | NVM | 0x00 | DLY[7:0] | | | | | | | |
| MAIN_ADC_ CAL1 | 1B | NVM | 0x00 | GAINL[7:0] | | | | | | | |
| MAIN_ADC_ CAL2 | 1C | NVM | 0x00 | GAINH | OFFSET[6:0] | | | | | | |
| AUX_ADC_ CAL1 | 1D | NVM | 0x00 | GAINL[7:0] | | | | | | | |
| AUX_ADC_ CAL2 | 1E | NVM | 0x00 | GAINH | OFFSET[6:0] | | | | | | |
| OTP_RSVD 1F | 1F | NVM | 0x00 | INTERNAL USE. DO NOT WRITE TO THIS ADDRESS | | | | | | | |
| OTP_RSVD 20 | 20 | NVM | 0x00 | INTERNAL USE. DO NOT WRITE TO THIS ADDRESS | | | | | | | |
| CUST_MISC 1 through CUST_MISC 8 | 21 | NVM | 0x00 | DATA[7:0] | | | | | | | |
| | 22 | NVM | 0x00 | DATA[7:0] | | | | | | | |
| | 23 | NVM | 0x00 | DATA[7:0] | | | | | | | |
| | 24 | NVM | 0x00 | DATA[7:0] | | | | | | | |
| | 25 | NVM | 0x00 | DATA[7:0] | | | | | | | |
| | 26 | NVM | 0x00 | DATA[7:0] | | | | | | | |
| | 27 | NVM | 0x00 | DATA[7:0] | | | | | | | |
| 28 | NVM | 0x00 | DATA[7:0] | | | | | | | | |
| STACK_RES PONSE | 29 | NVM | 0x00 | SPARE[1:0] | | | DELAY[5:0] | | | | |
| BBP_LOC | 2A | NVM | 0x00 | SPARE[2:0] | | | | LOC[4:0] | | | |
| OTP_RSVD 2B | 2B | NVM | 0x00 | INTERNAL USE. DO NOT WRITE TO THIS ADDRESS | | | | | | | |
| OTP_SPARE 10 | 2C | NVM | 0x00 | SPARE[7:0] | | | | | | | |
| OTP_SPARE 9 | 2D | NVM | 0x00 | SPARE[7:0] | | | | | | | |
| OTP_SPARE 8 | 2E | NVM | 0x00 | SPARE[7:0] | | | | | | | |
| OTP_SPARE 7 | 2F | NVM | 0x00 | SPARE[7:0] | | | | | | | |
| OTP_SPARE 6 | 30 | NVM | 0x00 | SPARE[7:0] | | | | | | | |
| OTP_SPARE 5 | 31 | NVM | 0x00 | SPARE[7:0] | | | | | | | |
| OTP_SPARE 4 | 32 | NVM | 0x00 | SPARE[7:0] | | | | | | | |
| OTP_SPARE 3 | 33 | NVM | 0x00 | SPARE[7:0] | | | | | | | |
| OTP_SPARE 2 | 34 | NVM | 0x00 | SPARE[7:0] | | | | | | | |
| OTP_SPARE 1 | 35 | NVM | 0x00 | SPARE[7:0] | | | | | | | |

BQ79616

SLUSF21A – JUNE 2023 – REVISED JUNE 2026

| Register Name | Addr Hex | RW Type | Reset Value | Data | | | | | | | |
|---------------|----------|---------|---|----------|------|------|------|------|------|------|------|
| | | | | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
| CUST_CRC_HI | 36 | NVM | HW Reset Default = 0x57 Factory Configuration Default = 0x31 | CRC[7:0] | | | | | | | |
| CUST_CRC_LO | 37 | NVM | HW Reset Default = 0x89 Factory Configuration Default = 0xF3 | CRC[7:0] | | | | | | | |

6.5.2 Read/Write Register Summary

| Register Name | Addr Hex | RW Type | Reset Value | Data | | | | | | | | |
|--|----------|---------|-------------|--|-------------------|----------------|-------------------|----------------|---------------|----------------|---------------|-----------|
| | | | | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | |
| OTP_PROG_UNL OOCK1A through OTP_PROG_UNL OOCK1D | 300 | RW | 0x00 | CODE[7:0] | | | | | | | | |
| | 301 | RW | 0x00 | CODE[7:0] | | | | | | | | |
| | 302 | RW | 0x00 | CODE[7:0] | | | | | | | | |
| | 303 | RW | 0x00 | CODE[7:0] | | | | | | | | |
| DIR0_ADDR | 306 | RW | 0x00 | RSVD | | | ADDRESS[5:0] | | | | | |
| DIR1_ADDR | 307 | RW | 0x00 | RSVD | | | ADDRESS[5:0] | | | | | |
| COMM_CTRL | 308 | RW | 0x00 | RSVD | | | | | | | STACK_DEV | TOP_STACK |
| CONTROL1 | 309 | RW | 0x00 | DIR_SEL | SEND_SHUT_DOWN | SEND_WAKE | SEND_SLPTO_ACT | GOTO_SHUT_DOWN | GOTO_SLEEP | SOFT_RESET | ADDR_WR | |
| CONTROL2 | 30A | RW | 0x00 | RSVD | | | | | | | SEND_HW_RESET | TSREF_EN |
| OTP_PROG_CTRL | 30B | RW | 0x00 | RSVD | | | | | | | PAGE_SEL | PROG_GO |
| ADC_CTRL1 | 30D | RW | 0x00 | RSVD | RSVD | | LPF_BB_EN | LPF_VCELL_EN | MAIN_GO | MAIN_MODE[1:0] | | |
| ADC_CTRL2 | 30E | RW | 0x00 | RSVD | | AUX_CELL_ALIGN | AUX_CELL_SEL[4:0] | | | | | |
| ADC_CTRL3 | 30F | RW | 0x00 | RSVD | AUX_GPIO_SEL[3:0] | | | AUX_GO | AUX_MODE[1:0] | | | |
| REG_INT_RSVD | 310 | RW | 0x00 | INTERNAL USE. DO NOT WRITE TO THIS ADDRESS | | | | | | | | |
| CB_CELL16_CTRL L through CB_CELL1_CTRL | 318 | RW | 0x00 | RSVD | | | TIME[4:0] | | | | | |
| | 319 | RW | 0x00 | RSVD | | | TIME[4:0] | | | | | |
| | 31A | RW | 0x00 | RSVD | | | TIME[4:0] | | | | | |
| | 31B | RW | 0x00 | RSVD | | | TIME[4:0] | | | | | |
| | 31C | RW | 0x00 | RSVD | | | TIME[4:0] | | | | | |
| | 31D | RW | 0x00 | RSVD | | | TIME[4:0] | | | | | |
| | 31E | RW | 0x00 | RSVD | | | TIME[4:0] | | | | | |
| | 31F | RW | 0x00 | RSVD | | | TIME[4:0] | | | | | |
| | 320 | RW | 0x00 | RSVD | | | TIME[4:0] | | | | | |
| | 321 | RW | 0x00 | RSVD | | | TIME[4:0] | | | | | |
| | 322 | RW | 0x00 | RSVD | | | TIME[4:0] | | | | | |
| | 323 | RW | 0x00 | RSVD | | | TIME[4:0] | | | | | |
| | 324 | RW | 0x00 | RSVD | | | TIME[4:0] | | | | | |
| | 325 | RW | 0x00 | RSVD | | | TIME[4:0] | | | | | |
| | 326 | RW | 0x00 | RSVD | | | TIME[4:0] | | | | | |
| | 327 | RW | 0x00 | RSVD | | | TIME[4:0] | | | | | |
| VMB_DONE_THRESH | 328 | RW | 0x3F | RSVD | | | MB_THR[5:0] | | | | | |
| MB_TIMER_CTRL | 329 | RW | 0x00 | RSVD | | | TIME[4:0] | | | | | |
| VCB_DONE_THRESH | 32A | RW | 0x00 | RSVD | | | CB_THR[5:0] | | | | | |
| OTCB_THRESH | 32B | RW | 0x0F | RSVD | COOLOFF[2:0] | | | OTCB_THR[3:0] | | | | |

BQ79616

SLUSF21A – JUNE 2023 – REVISED JUNE 2026

| Register Name | Addr Hex | RW Type | Reset Value | Data | | | | | | | | |
|---|----------|---------|-------------|-------------------|----------------|----------------|-------------------|-------------------|----------------|----------------|------------------|--|
| | | | | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | |
| OVUV_CTRL | 32C | RW | 0x00 | VCB DONE_THR_LOCK | OVUV_LOCK[3:0] | | | OVUV_GO | OVUV_MODE[1:0] | | | |
| OTUT_CTRL | 32D | RW | 0x00 | RSVD | OTCB_THR_LOCK | OTUT_LOCK[2:0] | | OTUT_GO | OTUT_MODE[1:0] | | | |
| BAL_CTRL1 | 32E | RW | 0x00 | RSVD | | | | DUTY[2:0] | | | | |
| BAL_CTRL2 | 32F | RW | 0x00 | RSVD | CB_PAUSE | FLTSTOP_EN | OTCB_EN | BAL_ACT[1:0] | | BAL_GO | AUTO_BAL | |
| BAL_CTRL3 | 330 | RW | 0x00 | RSVD | | | BAL_TIME_SEL[3:0] | | | BAL_TIME_GO | | |
| FAULT_RST1 | 331 | RW | 0x00 | RST_PROT | RST_UT | RST_OT | RST_UV | RST_OV | RST_COMP | RST_SYS | RST_PWR | |
| FAULT_RST2 | 332 | RW | 0x00 | RSVD | RST_OTP_CRC | RST_OTP_DATA | RST_COMM3_FCOMM | RST_COMM3_FTONE | RST_COMM3_HB | RST_COMM2 | RST_COMM1 | |
| DIAG_OTP_CTRL | 335 | RW | 0x00 | RSVD | | | FLIP_FACT_CRC | MARGIN_MODE[2:0] | | | MARGIN_GO | |
| DIAG_COMM_CTRL | 336 | RW | 0x00 | RSVD | | | | | | SPI_LOOP_BACK | FLIP_TR_CRC | |
| DIAG_PWR_CTRL | 337 | RW | 0x00 | RSVD | | | | | | BIST_NO_RST | PWR_BIST_GO | |
| DIAG_CBFET_CTRL1 | 338 | RW | 0x00 | CBFET16 | CBFET15 | CBFET14 | CBFET13 | CBFET12 | CBFET11 | CBFET10 | CBFET9 | |
| DIAG_CBFET_CTRL2 | 339 | RW | 0x00 | CBFET8 | CBFET7 | CBFET6 | CBFET5 | CBFET4 | CBFET3 | CBFET2 | CBFET1 | |
| DIAG_COMP_CTRL1 | 33A | RW | 0x00 | VCCB_THR[4:0] | | | | BB_THR[2:0] | | | | |
| DIAG_COMP_CTRL2 | 33B | RW | 0x00 | RSVD | GPIO_THR[2:0] | | | OW_THR[3:0] | | | | |
| DIAG_COMP_CTRL3 | 33C | RW | 0x00 | RSVD | CBFET_CTRL_GO | OW_SNK[1:0] | | COMP_ADC_SEL[2:0] | | | COMP_ADC_GO | |
| DIAG_COMP_CTRL4 | 33D | RW | 0x00 | RSVD | | | | | | COMP_FAULT_INJ | LPF_FAULT_INJ | |
| DIAG_PROT_CTRL | 33E | RW | 0x00 | RSVD | | | | | | | PROT_BIST_NO_RST | |
| OTP_ECC_DATA1 N1 through OTP_ECC_DATA1 N9 | 343 | RW | 0x00 | DATA[7:0] | | | | | | | | |
| | 344 | RW | 0x00 | DATA[7:0] | | | | | | | | |
| | 345 | RW | 0x00 | DATA[7:0] | | | | | | | | |
| | 346 | RW | 0x00 | DATA[7:0] | | | | | | | | |
| | 347 | RW | 0x00 | DATA[7:0] | | | | | | | | |
| | 348 | RW | 0x00 | DATA[7:0] | | | | | | | | |
| | 349 | RW | 0x00 | DATA[7:0] | | | | | | | | |
| | 34A | RW | 0x00 | DATA[7:0] | | | | | | | | |
| 34B | RW | 0x00 | DATA[7:0] | | | | | | | | | |
| OTP_ECC_TEST | 34C | RW | 0x00 | RSVD | | | | DED_SEC | MANUAL_AUTO | ENC_DEC | ENABLE | |
| SPI_CONF | 34D | RW | 0x00 | RSVD | CPOL | CPHA | NUMBIT[4:0] | | | | | |

| Register Name | Addr Hex | RW Type | Reset Value | Data | | | | | | | |
|--|----------|---------|-------------|-----------|------|------|----------------|-------------------------|----------------|-----------------------|------------------------|
| | | | | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
| SPI_TX3, SPI_TX2, and SPI_TX1 | 34E | RW | 0x00 | DATA[7:0] | | | | | | | |
| | 34F | RW | 0x00 | DATA[7:0] | | | | | | | |
| | 350 | RW | 0x00 | DATA[7:0] | | | | | | | |
| SPI_EXE | 351 | RW | 0x02 | RSVD | | | | | | SS_CTRL | SPI_GO |
| OTP_PROG_UNL OCK2A through OTP_PROG_UNL OCK2D | 352 | RW | 0x00 | CODE[7:0] | | | | | | | |
| | 353 | RW | 0x00 | CODE[7:0] | | | | | | | |
| | 354 | RW | 0x00 | CODE[7:0] | | | | | | | |
| | 355 | RW | 0x00 | CODE[7:0] | | | | | | | |
| DEBUG_CTRL_U NLOCK | 700 | RW | 0x00 | CODE[7:0] | | | | | | | |
| DEBUG_COMM_ CTRL1 | 701 | RW | 0x04 | RSVD | | | UART_ BAUD | UART_ MIRROR_ _EN | UART_ TX_EN | USER_ UART_ _EN | USER_ DAISY_ _EN |
| DEBUG_COMM_ CTRL2 | 702 | RW | 0x0F | RSVD | | | COML_ TX_EN | COML_ RX_EN | COMH_ TX_EN | COMH_ RX_EN | |

6.5.3 Read-Only Register Summary

| Register Name | Addr Hex | RW Type | Reset Value | Data | | | | | | | |
|--|----------|---------|-------------|----------------|-------------|-------------|--------|--------|--------|-------------|------|
| | | | | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
| PARTID | 500 | R | 0x00 | REV[7:0] | | | | | | | |
| DEV_REVID | E00 | R | 0x00 | DEV_REVID[7:0] | | | | | | | |
| DIE_ID1 through DIE_ID9 | 501 | R | 0x00 | ID[7:0] | | | | | | | |
| | 502 | R | 0x00 | ID[7:0] | | | | | | | |
| | 503 | R | 0x00 | ID[7:0] | | | | | | | |
| | 504 | R | 0x00 | ID[7:0] | | | | | | | |
| | 505 | R | 0x00 | ID[7:0] | | | | | | | |
| | 506 | R | 0x00 | ID[7:0] | | | | | | | |
| | 507 | R | 0x00 | ID[7:0] | | | | | | | |
| | 508 | R | 0x00 | ID[7:0] | | | | | | | |
| 509 | R | 0x00 | ID[7:0] | | | | | | | | |
| CUST_CRC_RSLT_ _HI | 50C | R | 0x31 | CRC[7:0] | | | | | | | |
| CUST_CRC_RSLT_ _LO | 50D | R | 0xF3 | CRC[7:0] | | | | | | | |
| OTP_ECC_DATA OUT1 through OTP_ECC_DATA OUT9 | 510 | R | 0x00 | DATA[7:0] | | | | | | | |
| | 511 | R | 0x00 | DATA[7:0] | | | | | | | |
| | 512 | R | 0x00 | DATA[7:0] | | | | | | | |
| | 513 | R | 0x00 | DATA[7:0] | | | | | | | |
| | 514 | R | 0x00 | DATA[7:0] | | | | | | | |
| | 515 | R | 0x00 | DATA[7:0] | | | | | | | |
| | 516 | R | 0x00 | DATA[7:0] | | | | | | | |
| | 517 | R | 0x00 | DATA[7:0] | | | | | | | |
| 518 | R | 0x00 | DATA[7:0] | | | | | | | | |
| OTP_PROG_STA T | 519 | R | 0x00 | UNLOCK | OTERR | UVERR | OVERR | SUVERR | SOVERR | PROG ERR | DONE |
| OTP_CUST1_STA T | 51A | R | 0x00 | LOADED | LOAD WRN | LOAD ERR | FMTERR | PROGOK | UVOK | OVOK | TRY |

BQ79616

SLUSF21A – JUNE 2023 – REVISED JUNE 2026

| Register Name | Addr Hex | RW Type | Reset Value | Data | | | | | | | | |
|-------------------------------------|----------|---------|-------------|--------------------|------------------------|-----------------------|-----------------|-----------------------|------------------------|------------------------|-----------------------|------------------|
| | | | | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | |
| OTP_CUST2_STA T | 51B | R | 0x00 | LOADED | LOAD WRN | LOAD ERR | FMTERR | PROGOK | UVOK | OVOK | TRY | |
| SPI_RX3, SPI_RX2, and SPI_RX1 | 520 | R | 0x00 | DATA[7:0] | | | | | | | | |
| | 521 | R | 0x00 | DATA[7:0] | | | | | | | | |
| | 522 | R | 0x00 | DATA[7:0] | | | | | | | | |
| DIAG_STAT | 526 | R | 0x00 | RSVD | | | DRDY_ OTUT | DRDY_ OVUV | DRDY_ BIST_ OTUT | DRDY_ BIST_ OVUV | DRDY_ BIST_ PWR | |
| ADC_STAT1 | 527 | R | 0x00 | RSVD | | | RSVD | DRDY_ AUX_ GPIO | DRDY_ AUX_ CELL | DRDY_ AUX_ MISC | DRDY_ MAIN_ ADC | |
| ADC_STAT2 | 528 | R | 0x00 | RSVD | | DRDY_ LPF | DRDY_ GPIO | DRDY_ VCOW | DRDY_ CBOW | DRDY_ CBFET | DRDY_ VCCB | |
| GPIO_STAT | 52A | R | 0x00 | GPIO8 | GPIO7 | GPIO6 | GPIO5 | GPIO4 | GPIO3 | GPIO2 | GPIO1 | |
| BAL_STAT | 52B | R | 0x00 | INVALID_ CBCONF | OT_ PAUSE_ DET | CB_ INPAUSE | MB_RUN | CB_RUN | ABORT FLT | MB_ DONE | CB_ DONE | |
| DEV_STAT | 52C | R | 0x00 | RSVD | FACT_ CRC_ DONE | CUST_ CRC_ DONE | OTUT_ RUN | OVUV_ RUN | RSVD | AUX_ RUN | MAIN_ RUN | |
| FAULT_SUMMAR Y | 52D | R | 0x00 | FAULT_ PROT | FAULT_ COMP_ ADC | FAULT_ OTP | FAULT_ COMM | FAULT_ OTUT | FAULT_ OVUV | FAULT_ SYS | FAULT_ PWR | |
| FAULT_COMM1 | 530 | R | 0x00 | RSVD | | | UART_TR | UART_ RR | UART_ RC | COMM CLR_ DET | STOP_ DET | |
| FAULT_COMM2 | 531 | R | 0x00 | COML_ TR | COML_ RR | COML_ RC | COML_ BIT | COMH_ TR | COMH_ RR | COMH_ RC | COMH_ BIT | |
| FAULT_COMM3 | 532 | R | 0x00 | RSVD | | | | FCOMM_ DET | FTONE_ DET | HB_FAIL | HB_FAST | |
| FAULT_OTP | 535 | R | 0x00 | RSVD | DED_ DET | SEC_DET | CUST_ CRC | FACT_ CRC | CUSTLD ERR | FACTLD ERR | GBLOV ERR | |
| FAULT_SYS | 536 | R | 0x00 | LFO | RSVD | GPIO | DRST | CTL | CTS | TSHUT | TWARN | |
| FAULT_PROT1 | 53A | R | 0x00 | RSVD | | | | | | | TPARITY_ FAIL | VPARITY_ FAIL |
| FAULT_PROT2 | 53B | R | 0x00 | RSVD | BIST_ ABORT | TPATH_ FAIL | VPATH_ FAIL | UTCOMP_ FAIL | OTCOMP_ FAIL | OVCOMP_ FAIL | UVCOMP_ FAIL | |
| FAULT_OV1 | 53C | R | 0x00 | OV16_ DET | OV15_ DET | OV14_ DET | OV13_ DET | OV12_ DET | OV11_ DET | OV10_ DET | OV9_DET | |
| FAULT_OV2 | 53D | R | 0x00 | OV8_DET | OV7_DET | OV6_DET | OV5_DET | OV4_DET | OV3_DET | OV2_DET | OV1_DET | |
| FAULT_UV1 | 53E | R | 0x00 | UV16_ DET | UV15_ DET | UV14_ DET | UV13_ DET | UV12_ DET | UV11_ DET | UV10_ DET | UV9_DET | |
| FAULT_UV2 | 53F | R | 0x00 | UV8_DET | UV7_DET | UV6_DET | UV5_DET | UV4_DET | UV3_DET | UV2_DET | UV1_DET | |
| FAULT_OT | 540 | R | 0x00 | OT8_DET | OT7_DET | OT6_DET | OT5_DET | OT4_DET | OT3_DET | OT2_DET | OT1_DET | |
| FAULT_UT | 541 | R | 0x00 | UT8_DET | UT7_DET | UT6_DET | UT5_DET | UT4_DET | UT3_DET | UT2_DET | UT1_DET | |
| FAULT_COMP_G PIO | 543 | R | 0x00 | GPIO8_ FAIL | GPIO7_ FAIL | GPIO6_ FAIL | GPIO5_ FAIL | GPIO4_ FAIL | GPIO3_ FAIL | GPIO2_ FAIL | GPIO1_ FAIL | |
| FAULT_COMP_V CCB1 | 545 | R | 0x00 | CELL16_ FAIL | CELL15_ FAIL | CELL14_ FAIL | CELL13_ FAIL | CELL12_ FAIL | CELL11_ FAIL | CELL10_ FAIL | CELL9_ FAIL | |
| FAULT_COMP_V CCB2 | 546 | R | 0x00 | CELL8_ FAIL | CELL7_ FAIL | CELL6_ FAIL | CELL5_ FAIL | CELL4_ FAIL | CELL3_ FAIL | CELL2_ FAIL | CELL1_ FAIL | |
| FAULT_COMP_V COW1 | 548 | R | 0x00 | VCOW16_ FAIL | VCOW15_ FAIL | VCOW14_ FAIL | VCOW13_ FAIL | VCOW12_ FAIL | VCOW11_ FAIL | VCOW10_ FAIL | VCOW9_ FAIL | |

| Register Name | Addr Hex | RW Type | Reset Value | Data | | | | | | | | |
|--------------------|----------|---------|-------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|------------------|----------|
| | | | | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | |
| FAULT_COMP_V COW2 | 549 | R | 0x00 | VCOW8 _FAIL | VCOW7 _FAIL | VCOW6 _FAIL | VCOW5 _FAIL | VCOW4 _FAIL | VCOW3 _FAIL | VCOW2 _FAIL | VCOW1 _FAIL | |
| FAULT_COMP_VB OW1 | 54B | R | 0x00 | CBOW16 _FAIL | CBOW15 _FAIL | CBOW14 _FAIL | CBOW13 _FAIL | CBOW12 _FAIL | CBOW11 _FAIL | CBOW10 _FAIL | CBOW9 _FAIL | |
| FAULT_COMP_VB OW2 | 54C | R | 0x00 | CBOW8 _FAIL | CBOW7 _FAIL | CBOW6 _FAIL | CBOW5 _FAIL | CBOW4 _FAIL | CBOW3 _FAIL | CBOW2 _FAIL | CBOW1 _FAIL | |
| FAULT_COMP_C BFET1 | 54E | R | 0x00 | CBFET16 _FAIL | CBFET15 _FAIL | CBFET14 _FAIL | CBFET13 _FAIL | CBFET12 _FAIL | CBFET11 _FAIL | CBFET10 _FAIL | CBFET9 _FAIL | |
| FAULT_COMP_C BFET2 | 54F | R | 0x00 | CBFET8 _FAIL | CBFET7 _FAIL | CBFET6 _FAIL | CBFET5 _FAIL | CBFET4 _FAIL | CBFET3 _FAIL | CBFET2 _FAIL | CBFET1 _FAIL | |
| FAULT_COMP_MIS C | 550 | R | 0x00 | RSVD | | | | | | | COMP_ ADC_ ABORT | LPF_FAIL |
| FAULT_PWR1 | 552 | R | 0x00 | CVSS_ OPEN | DVSS_ OPEN | REFHM_ OPEN | CVDD_ UV | CVDD_ OV | DVDD_ OV | AVDD_ OSC | AVDD_ OV | |
| FAULT_PWR2 | 553 | R | 0x00 | RSVD | PWRBIST_ FAIL | RSVD | REFH_ OSC | NEG5V_ UV | TSREF_ OSC | TSREF_ UV | TSREF_ OV | |
| FAULT_PWR3 | 554 | R | RSVD | RSVD | | | | | RSVD | RSVD | AVDDUV_ DRST | |
| CB_COMPLETE1 | 556 | R | 0x00 | CELL16_ DONE | CELL15_ DONE | CELL14_ DONE | CELL13_ DONE | CELL12_ DONE | CELL11_ DONE | CELL10_ DONE | CELL9_ DONE | |
| CB_COMPLETE2 | 557 | R | 0x00 | CELL8_ DONE | CELL7_ DONE | CELL6_ DONE | CELL5_ DONE | CELL4_ DONE | CELL3_ DONE | CELL2_ DONE | CELL1_ DONE | |
| BAL_TIME | 558 | R | 0x00 | TIME_ UNIT | TIME[6:0] | | | | | | | |
| VCELL16_HI/LO | 568 | R | 0x80 | RESULT[7:0] | | | | | | | | |
| | 569 | R | 0x00 | RESULT[7:0] | | | | | | | | |
| VCELL15_HI/LO | 56A | R | 0x80 | RESULT[7:0] | | | | | | | | |
| | 56B | R | 0x00 | RESULT[7:0] | | | | | | | | |
| VCELL14_HI/LO | 56C | R | 0x80 | RESULT[7:0] | | | | | | | | |
| | 56D | R | 0x00 | RESULT[7:0] | | | | | | | | |
| VCELL13_HI/LO | 56E | R | 0x80 | RESULT[7:0] | | | | | | | | |
| | 56F | R | 0x00 | RESULT[7:0] | | | | | | | | |
| VCELL12_HI/LO | 570 | R | 0x80 | RESULT[7:0] | | | | | | | | |
| | 571 | R | 0x00 | RESULT[7:0] | | | | | | | | |
| VCELL11_HI/LO | 572 | R | 0x80 | RESULT[7:0] | | | | | | | | |
| | 573 | R | 0x00 | RESULT[7:0] | | | | | | | | |
| VCELL10_HI/LO | 574 | R | 0x80 | RESULT[7:0] | | | | | | | | |
| | 575 | R | 0x00 | RESULT[7:0] | | | | | | | | |
| VCELL9_HI/LO | 576 | R | 0x80 | RESULT[7:0] | | | | | | | | |
| | 577 | R | 0x00 | RESULT[7:0] | | | | | | | | |
| VCELL8_HI/LO | 578 | R | 0x80 | RESULT[7:0] | | | | | | | | |
| | 579 | R | 0x00 | RESULT[7:0] | | | | | | | | |
| VCELL7_HI/LO | 57A | R | 0x80 | RESULT[7:0] | | | | | | | | |
| | 57B | R | 0x00 | RESULT[7:0] | | | | | | | | |
| VCELL6_HI/LO | 57C | R | 0x80 | RESULT[7:0] | | | | | | | | |
| | 57D | R | 0x00 | RESULT[7:0] | | | | | | | | |
| VCELL5_HI/LO | 57E | R | 0x80 | RESULT[7:0] | | | | | | | | |
| | 57F | R | 0x00 | RESULT[7:0] | | | | | | | | |

BQ79616

SLUSF21A – JUNE 2023 – REVISED JUNE 2026

| Register Name | Addr Hex | RW Type | Reset Value | Data | | | | | | | |
|--------------------|----------|---------|-------------|-------------|------|------|------|------|------|------|------|
| | | | | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
| VCELL4_HI/LO | 580 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 581 | R | 0x00 | RESULT[7:0] | | | | | | | |
| VCELL3_HI/LO | 582 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 583 | R | 0x00 | RESULT[7:0] | | | | | | | |
| VCELL2_HI/LO | 584 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 585 | R | 0x00 | RESULT[7:0] | | | | | | | |
| VCELL1_HI/LO | 586 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 587 | R | 0x00 | RESULT[7:0] | | | | | | | |
| BUSBAR_HI/LO | 588 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 589 | R | 0x00 | RESULT[7:0] | | | | | | | |
| TSREF_HI/LO | 58C | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 58D | R | 0x00 | RESULT[7:0] | | | | | | | |
| GPIO1_HI/LO | 58E | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 58F | R | 0x00 | RESULT[7:0] | | | | | | | |
| GPIO2_HI/LO | 590 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 591 | R | 0x00 | RESULT[7:0] | | | | | | | |
| GPIO3_HI/LO | 592 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 593 | R | 0x00 | RESULT[7:0] | | | | | | | |
| GPIO4_HI/LO | 594 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 595 | R | 0x00 | RESULT[7:0] | | | | | | | |
| GPIO5_HI/LO | 596 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 597 | R | 0x00 | RESULT[7:0] | | | | | | | |
| GPIO6_HI/LO | 598 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 599 | R | 0x00 | RESULT[7:0] | | | | | | | |
| GPIO7_HI/LO | 59A | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 59B | R | 0x00 | RESULT[7:0] | | | | | | | |
| GPIO8_HI/LO | 59C | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 59D | R | 0x00 | RESULT[7:0] | | | | | | | |
| DIETEMP1_HI/LO | 5AE | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 5AF | R | 0x00 | RESULT[7:0] | | | | | | | |
| DIETEMP2_HI/LO | 5B0 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 5B1 | R | 0x00 | RESULT[7:0] | | | | | | | |
| AUX_CELL_HI/LO | 5B2 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 5B3 | R | 0x00 | RESULT[7:0] | | | | | | | |
| AUX_GPIO_HI/LO | 5B4 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 5B5 | R | 0x00 | RESULT[7:0] | | | | | | | |
| AUX_BAT_HI/LO | 5B6 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 5B7 | R | 0x00 | RESULT[7:0] | | | | | | | |
| AUX_REFL_HI/LO | 5B8 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 5B9 | R | 0x00 | RESULT[7:0] | | | | | | | |
| AUX_VBG2_HI/LO | 5BA | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 5BB | R | 0x00 | RESULT[7:0] | | | | | | | |
| AUX_AVAO_REF_HI/LO | 5BE | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 5BF | R | 0x00 | RESULT[7:0] | | | | | | | |

| Register Name | Addr Hex | RW Type | Reset Value | Data | | | | | | | |
|-----------------------|----------|---------|------------------------------|-------------|-------------|--------------|------------|-------------|-------------|------------|------|
| | | | | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
| AUX_AVDD_REF_HI/LO | 5C0 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 5C1 | R | 0x00 | RESULT[7:0] | | | | | | | |
| AUX_OV_DAC_HI/LO | 5C2 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 5C3 | R | 0x00 | RESULT[7:0] | | | | | | | |
| AUX_UV_DAC_HI/LO | 5C4 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 5C5 | R | 0x00 | RESULT[7:0] | | | | | | | |
| AUX_OT_OTCB_DAC_HI/LO | 5C6 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 5C7 | R | 0x00 | RESULT[7:0] | | | | | | | |
| AUX_UT_DAC_HI/LO | 5C8 | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 5C9 | R | 0x00 | RESULT[7:0] | | | | | | | |
| AUX_VCBDONE_DAC_HI/LO | 5CA | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 5CB | R | 0x00 | RESULT[7:0] | | | | | | | |
| AUX_VCM_HI/LO | 5CC | R | 0x80 | RESULT[7:0] | | | | | | | |
| | 5CD | R | 0x00 | RESULT[7:0] | | | | | | | |
| REFOVDAC_HI/LO | 5D0 | R | 0x00 | RESULT[7:0] | | | | | | | |
| | 5D1 | R | 0x00 | RESULT[7:0] | | | | | | | |
| DIAG_MAIN_HI/LO | 5D2 | R | 0x00 | RESULT[7:0] | | | | | | | |
| | 5D3 | R | 0x00 | RESULT[7:0] | | | | | | | |
| DIAG_AUX_HI/LO | 5D4 | R | 0x00 | RESULT[7:0] | | | | | | | |
| | 5D5 | R | 0x00 | RESULT[7:0] | | | | | | | |
| DEBUG_COMM_STAT | 780 | R | 0x33 for base 0x3F for stack | RSVD | HW_UART_DRV | HW_DAI5Y_DRV | COML_TX_ON | COML_RX_ON | COMH_TX_ON | COMH_RX_ON | |
| DEBUG_UART_RC | 781 | R | 0x00 | RSVD | RC_IERR | RC_TXDIS | RC_SOF | RC_BYTE_ERR | RC_UNEXP | RC_CRC | |
| DEBUG_UART_RR_TR | 782 | R | 0x00 | RSVD | | TR_SOF | TR_WAIT | RR_SOF | RR_BYTE_ERR | RR_CRC | |
| T | 783 | R | 0x00 | RSVD | | PERR | BERR_TAG | SYNC2 | SYNC1 | BIT | |
| DEBUG_COMH_RC | 784 | R | 0x00 | RSVD | RC_IERR | RC_TXDIS | RC_SOF | RC_BYTE_ERR | RC_UNEXP | RC_CRC | |
| DEBUG_COMH_RR_TR | 785 | R | 0x00 | RSVD | TR_WAIT | RR_TXDIS | RR_SOF | RR_BYTE_ERR | RR_UNEXP | RR_CRC | |
| DEBUG_COML_BIT | 786 | R | 0x00 | RSVD | | PERR | BERR_TAG | SYNC2 | SYNC1 | BIT | |
| DEBUG_COML_RC | 787 | R | 0x00 | RSVD | RC_IERR | RC_TXDIS | RC_SOF | RC_BYTE_ERR | RC_UNEXP | RC_CRC | |
| DEBUG_COML_RR_TR | 788 | R | 0x00 | RSVD | TR_WAIT | RR_TXDIS | RR_SOF | RR_BYTE_ERR | RR_UNEXP | RR_CRC | |
| DEBUG_UART_DISCARD | 789 | R | 0x00 | COUNT[7:0] | | | | | | | |
| DEBUG_COMH_DISCARD | 78A | R | 0x00 | COUNT[7:0] | | | | | | | |

BQ79616

SLUSF21A – JUNE 2023 – REVISED JUNE 2026

| Register Name | Addr Hex | RW Type | Reset Value | Data | | | | | | | |
|----------------------------|----------|---------|-------------|------------|------|------|------|------|------|------|------|
| | | | | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
| DEBUG_COML_D ISCARD | 78B | R | 0x00 | COUNT[7:0] | | | | | | | |
| DEBUG_UART_V ALID_HI/LO | 78C | R | 0x00 | COUNT[7:0] | | | | | | | |
| | 78D | R | 0x00 | COUNT[7:0] | | | | | | | |
| DEBUG_COMH_V ALID_HI/LO | 78E | R | 0x00 | COUNT[7:0] | | | | | | | |
| | 78F | R | 0x00 | COUNT[7:0] | | | | | | | |
| DEBUG_COML_V ALID_HI/LO | 790 | R | 0x00 | COUNT[7:0] | | | | | | | |
| | 791 | R | 0x00 | COUNT[7:0] | | | | | | | |
| DEBUG_OTP_SE C_BLK | 7A0 | R | 0x00 | BLOCK[7:0] | | | | | | | |
| DEBUG_OTP_DE D_BLK | 7A1 | R | 0x00 | BLOCK[7:0] | | | | | | | |

6.5.4 Register Field Descriptions

6.5.4.1 Device Addressing Setup

6.5.4.1.1 DIR0_ADDR_OTP

| Address | 0x0000 | | | | | | | |
|--|------------|-------|--------------|-------|-------|-------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | SPARE[1:0] | | ADDRESS[5:0] | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| SPARE[1:0] = Spare | | | | | | | | |
| ADDRESS[5:0] = This register shows the default device address used when $[DIR_SEL] = 0$ and programmed in the OTP. Writing to this register won't change the device address actively in use. This register is used for the system to program the device address to OTP, which will be loaded to the <i>DIR0_ADDR</i> register at POR. For programming, follow the OTP programming procedure. | | | | | | | | |

6.5.4.1.2 DIR1_ADDR_OTP

| Address | 0x0001 | | | | | | | |
|--|------------|-------|--------------|-------|-------|-------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | SPARE[1:0] | | ADDRESS[5:0] | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| SPARE[1:0] = Spare | | | | | | | | |
| ADDRESS[5:0] = This register shows the default device address used when $[DIR_SEL] = 1$ and programmed in the OTP. Writing to this register won't change the device address actively in use. This register is used for the system to program the device address to OTP, which will be loaded to the <i>DIR1_ADDR</i> register at POR. For programming, follow the OTP programming procedure. | | | | | | | | |

6.5.4.1.3 CUST_MISC1 through CUST_MISC8

| Address | 0x0021 to 0x0028 | | | | | | | |
|----------------------------------|------------------|-------|-------|-------|-------|-------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | DATA[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| DATA[7:0] = Customer scratch pad | | | | | | | | |

6.5.4.1.4 DIR0_ADDR

| Address | 0x0306 | | | | | | | |
|--|--------|-------|--------------|-------|-------|-------|-------|-------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | ADDRESS[5:0] | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| ADDRESS[5:0] = Always shows the current device address used by the device when $[DIR_SEL] = 0$. At POR, this register is loaded from the device address value in the OTP (same OTP device address loaded to <i>DIR0_ADDR_OTP</i> register). Host can re-address the device by writing a different device address to this register, and the device will take on the new address immediately. Note: $CONTROL1[ADDR_WR] = 1$ is required to write to this register. See Section 6.5.4.3.11 for details. | | | | | | | | |

6.5.4.1.5 DIR1_ADDR

| Address | 0x0307 | | | | | | | |
|-----------------|--------|-------|--------------|-------|-------|-------|-------|-------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | ADDRESS[5:0] | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |

ADDRESS[5:0] = Always shows the current device address used by the device when $[DIR_SEL] = 1$. At POR, this register is loaded from the device address value in the OTP (same OTP device address loaded to $DIR1_ADDR_OTP$ register). Host can re-address the device by writing a different device address to this register, and the device will take on the new address immediately.
Note: $CONTROL1[ADDR_WR] = 1$ is required to write to this register. See [Section 6.5.4.3.11](#) for details.

6.5.4.2 Device ID and Scratch Pad

6.5.4.2.1 PARTID

| Address | 0x0500 | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | PARTID[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| PARTID[7:0] Device Identification: = 0x21 = BQ79616 All other codes = Reserved | | | | | | | | |

6.5.4.2.2 DEV_REVID

| Address | 0xE00 | | | | | | | |
|--|-------|------|------|------|------|------|------|------|
| Read Only | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| A value of 0x00 indicates that the device is in normal operating mode. If a fault activates the Factory Testmode Detection, the value will be non-zero. Refer Safety Manual for details on SM426: Fact Testmode Detection. | | | | | | | | |

6.5.4.2.3 DIE_ID1 through DIE_ID9

| Address | 0x0501 | | | | | | | |
|---|---------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | ID[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| ID[7:0] = Device Revision 0x10 = Revision A0 0x11 = Revision A1 0x20 = Revision B0 0x21 = Revision B1 0x22 = Revision B2 All other codes = Reserved | | | | | | | | |

| Address | 0x0502 to 0x0509 | | | | | | | |
|-------------------------------------|------------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | ID[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| ID[7:0] = Die ID for TI factory use | | | | | | | | |

6.5.4.3 General Configuration and Control

6.5.4.3.1 DEV_CONF

| Address | 0x0002 | | | | | | | |
|---------|--------|-----------|--------------|----------|-------------|-----------|----------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | NO_ADJ_CB | MULTIDROP_EN | FCOMM_EN | TWO_STOP_EN | NFAULT_EN | FTONE_EN | HB_EN |

| | | | | | | | | |
|--|---|---|---|---|---|---|---|---|
| Reset | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| NO_ADJ_CB = Indicates the device will not allow an adjacent CB FET to be turned on in manual CB control. If MCU has enabled an adjacent CB FET, device will not start CB even if host sends [BAL_GO] = 1. 0 = Device will allow two adjacent CB FETs to be enabled. 1 = Device will not allow adjacent CB FET to be enabled. | | | | | | | | |
| MULTIDROP_EN = Defines if the device is used in a multidrop or daisy chain configuration. The TX and RX for COML and COMH will be enabled or disabled based on the configuration. 0 = Daisy chain of base device 1 = Multidrop | | | | | | | | |
| FCOMM_EN = Enables the fault state detection through communication in ACTIVE mode. 0 = Disable 1 = Enable | | | | | | | | |
| TWO_STOP_EN = Enables two stop bits for the UART in case of severe oscillator error in the host and device. 0 = One STOP bit 1 = Two STOP bits | | | | | | | | |
| NFAULT_EN = Enables the NFAULT function. 0 = NFAULT always pulled up 1 = NFAULT pulled low to indicate an unmasked fault is detected. Note: This bit setting does not affect the FAULT_SUMMARY register. | | | | | | | | |
| FTONE_EN = Enables FAULT TONE transmitter when device is in SLEEP mode. 0 = Disable 1 = Enable | | | | | | | | |
| HB_EN = Enables HEARTBEAT transmitter when device is in SLEEP mode. 0 = Disable 1 = Enable | | | | | | | | |

6.5.4.3.2 ACTIVE_CELL

| Address | 0x0003 | | | | | | | |
|---|------------|-------|-------|---------------|-------|-------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | SPARE[3:0] | | | NUM_CELL[3:0] | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Factory OTP Reset | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| SPARE[3:0] = Spare | | | | | | | | |
| NUM_CELL[3:0] = Configures the number of cells in series. 0x0 = 6S 0x1 = 7S 0x2 = 8S : 0xA = 16S Unused code defaults to CHIP_TYPE[MAX_CH1:0] setting (in factory trim). If the NUM_CELL setting has more channels than the device offers, it would be capped to highest number of channel the device offers. | | | | | | | | |

6.5.4.3.3 BBVC_POSN1

| Address | 0x0005 | | | | | | | |
|--|--------|--------|--------|--------|--------|--------|--------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CELL16 | CELL15 | CELL14 | CELL13 | CELL12 | CELL11 | CELL10 | CELL9 |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CELL9 to CELL16 = This register specifies which channel is connected with a bus bar. The cell voltage measurement diagnostic comparison will handle those channel differently 0 = No special handling 1 = Special handling of the enabled channel(s). See Section 6.3.7 for details. | | | | | | | | |

6.5.4.3.4 BBVC_POSN2

| Address | 0x0006 | | | | | | | |
|--|--------|-------|-------|-------|-------|-------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CELL8 | CELL7 | CELL6 | CELL5 | CELL4 | CELL3 | CELL2 | CELL1 |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CELL1 to CELL8 = Among the active cells specified by the <i>ACTIVE_CELL</i> register, this register indicates which active channel is excluded from the OV, UV and VCB_DONE monitoring. This register information is also used for cell voltage measurement diagnostic comparison. 0 = No special handling of the functions mentioned above. 1 = Special handling of the functions mentioned above. See Section 6.3.7 for details. | | | | | | | | |

6.5.4.3.5 PWR_TRANSIT_CONF

| Address | 0x0018 | | | | | | | |
|--|------------|-------|-------|-----------------|-------|---------------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | SPARE[2:0] | | | TWAR_N_THR[1:0] | | SLP_TIME[2:0] | | |
| Reset | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| Factory Configuration default | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| SPARE[2:0] = Spare | | | | | | | | |
| TWARN_THR[1:0] = Sets the TWARN threshold. 00 = 85°C 01 = 95°C 10 = 105°C (default) 11 = 115°C | | | | | | | | |
| SLP_TIME[2:0] = A timeout in SLEEP mode. This timer starts counting when device enters SLEEP mode. When the timer expires, the device enters SHUTDOWN mode. The timer resets if device wakes up to ACTIVE mode. 000 = No timeout. Device remains in SLEEP mode (default at reset) 001 = 5 s 010 = 10 s 011 = 1 min 100 = 10 min 101 = 30 min 110 = 1 hour 111 = 2 hour | | | | | | | | |

6.5.4.3.6 COMM_TIMEOUT_CONF

| Address | 0x0019 | | | | | | | |
|--|--------|---------------|-------|-------|---------|---------------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | SPARE | CTS_TIME[2:0] | | | CTL_ACT | CTL_TIME[2:0] | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| SPARE = Spare | | | | | | | | |
| CTS_TIME[2:0] = Sets the short communication timeout. When this timer expires, the device sets the <i>FAULT_SYS[CTL]</i> bit. This can be used as an alert to the system to prevent a long communication timeout. 000 = Disables short communication timeout (default at reset) 001 = 100 ms 010 = 2 s 011 = 10 s 100 = 1 min 101 = 10 min 110 = 30 min 111 = 1 hr | | | | | | | | |
| CTL_ACT = Configures the device action when long communication timeout timer expires. 0 = Sets <i>FAULT_SYS[CTL]</i> and sends device to SLEEP mode 1 = Sends the device to SHUTDOWN. <i>FAULT_SYS[CTL]</i> bit will not be set.(default at reset) | | | | | | | | |

CTL_TIME[2:0] = Sets the long communication timeout. When this timer expires, the device takes the action configured by the *[CTL_ACT]* bit.
 000 = Disables long communication timeout
 001 = 100 ms
 010 = 2 s
 011 = 10 s
 100 = 1 min (default at reset)
 101 = 10 min
 110 = 30 min
 111 = 1 hr

6.5.4.3.7 TX_HOLD_OFF

| Address | 0x001A | | | | | | | |
|---|----------|-------|-------|-------|-------|-------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | DLY[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| DLY[7:0] = Sets the number of bit periods from 0 to 255 to delay after receiving the STOP bit of a command frame and before transmitting the 1st bit of response frame. | | | | | | | | |

6.5.4.3.8 STACK_RESPONSE

| Address | 0x0029 | | | | | | | |
|---|------------|-------|------------|-------|-------|-------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | SPARE[1:0] | | DELAY[5:0] | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| DELAY[5:0] Add additional byte delay gap in daisy chain data response frame = 0x00 = 0- μ s 0x01 to 0x3F = 0.25- μ s to 15.75- μ s in 0.25us step | | | | | | | | |

6.5.4.3.9 BBP_LOC

| Address | 0x002A | | | | | | | |
|---|------------|-------|-------|----------|-------|-------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | SPARE[3:0] | | | LOC[4:0] | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| LOC[4:0] = Indicate the BBP pin location and show which VC channel the BB channel is sharing with. This information is used for correcting the common mode error during BB channel ADC measurement 0x00 = BBP/N not in used 0x01 = BBP to negative side of Cell2, BBN to VC1 0x02 = BBP to negative side of Cell3, BBN to VC2 0x03 = BBP to negative side of Cell4, BBN to VC3 : 0x0F = BBP to negative side of Cell16, BBN to VC15 0x10 = BBP to VC16, BBN to positive side of Cell16 | | | | | | | | |

6.5.4.3.10 COMM_CTRL

| Address | 0x0308 | | | | | | | |
|---|--------|-------|-------|-------|-------|-------|-----------|-----------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | | | | STACK_DEV | TOP_STACK |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| STACK_DEV = Defines device as a base or stack device in daisy chain configuration. 0 = Base device 1 = Stack device | | | | | | | | |

TOP_STACK = Defines device as highest addressed device in the stack.
 0 = Not the ToS device
 1 = Is the ToS device

6.5.4.3.11 CONTROL1

| Address | 0x0309 | | | | | | | |
|--|---------|---------------|-----------|---------------|---------------|------------|------------|---------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | DIR_SEL | SEND_SHUTDOWN | SEND_WAKE | SEND_SLPTOACT | GOTO_SHUTDOWN | GOTO_SLEEP | SOFT_RESET | ADDR_WR |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| DIR_SEL = Selects daisy chain communication direction. 0 = With two devices connected in daisy chain, command frame travels from COMH of the lower device to COML of the next device. 1 = With two devices connected in daisy chain, command frame travels from COML of the lower device to COMH of the next device. | | | | | | | | |
| SEND_SHUTDOWN = Sends SHUTDOWN tone to next device up the stack. The device receiving this bit set is unaffected. Bit is cleared on read. 0 = Ready 1 = Send SHUTDOWN tone up the stack | | | | | | | | |
| SEND_WAKE = Sends WAKE tone to next device up the stack. Bit is cleared on read. 0 = Ready 1 = Send WAKE tone to next device up the stack. | | | | | | | | |
| SEND_SLPTOACT = Sends SLEEPtoACTIVE tone up the stack. Bit is cleared on read. 0 = Ready 1 = Send SLEEPtoACTIVE tone up the stack | | | | | | | | |
| GOTO_SHUTDOWN = Transitions device to SHUTDOWN mode. Bit is cleared on read. 0 = Ready 1 = Enter SHUTDOWN mode | | | | | | | | |
| GOTO_SLEEP = Transitions device to SLEEP mode. Bit is cleared on read. 0 = Ready 1 = Enter SLEEP mode | | | | | | | | |
| SOFT_RESET = Resets the digital to OTP default. Bit is cleared on read. Setting this bit will cause the device to generate WAKE tone to the upper stack devices. 0 = Ready 1 = Reset device | | | | | | | | |
| ADDR_WR = Enables device to start auto-addressing. When this bit is set, device will not forward the first transition it receives, allowing the device address to be written to a single device. See Section 6.3.6.1.3.2 for details. 0 = Not performing auto-address. Device forwards communication transaction as normal. 1 = Device is being auto-addressed; the first communication transaction it receives will not be forwarded. | | | | | | | | |

6.5.4.3.12 CONTROL2

| Address | 0x030A | | | | | | | |
|--|--------|-------|-------|-------|-------|-------|---------------|----------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | | | | SEND_HW_RESET | TSREF_EN |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| SEND_HW_RESET = Sends HW_RESET tone up the stack. Bit is cleared on read. 0 = Ready 1 = Send HW_RESET tone to next stack device up | | | | | | | | |
| TSREF_EN = Enables TSREF LDO output. Used to bias NTC thermistor. 0 = Disabled 1 = Enabled | | | | | | | | |

6.5.4.3.13 CUST_CRC_HI

| Address | 0x0036 | | | | | | |
|---------|--------|--|--|--|--|--|--|
| | | | | | | | |

| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|----------|-------|-------|-------|-------|-------|-------|-------|
| Name | CRC[7:0] | | | | | | | |
| Reset | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 |
| Factory Configuration Reset | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 |
| CRC[7:0] = High-byte of the host-calculated CRC for customer OTP space. | | | | | | | | |

6.5.4.3.14 CUST_CRC_LO

| Address | 0x0037 | | | | | | | |
|--|----------|-------|-------|-------|-------|-------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CRC[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| Factory Configuration Reset | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
| CRC[7:0] = Low-byte of the host-calculated CRC for customer OTP space. | | | | | | | | |

6.5.4.3.15 CUST_CRC_RSLT_HI

| Address | 0x050C | | | | | | | |
|---|----------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CRC[7:0] | | | | | | | |
| Reset | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 |
| CRC[7:0] = High-byte of the device-calculated CRC for customer OTP space. | | | | | | | | |

6.5.4.3.16 CUST_CRC_RSLT_LO

| Address | 0x050D | | | | | | | |
|--|----------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CRC[7:0] | | | | | | | |
| Reset | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
| CRC[7:0] = Low-byte of the device-calculated CRC for customer OTP space. | | | | | | | | |

6.5.4.4 Operation Status

6.5.4.4.1 DIAG_STAT

| Address | 0x0526 | | | | | | | |
|---|--------|-------|-------|-----------|-----------|----------------|----------------|---------------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | DRDY_OTUT | DRDY_OVUV | DRDY_BIST_OTUT | DRDY_BIST_OVUV | DRDY_BIST_PWR |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| DRDY_OTUT = Indicates the OTUT round robin has run at least once. This bit is cleared when [OTUT_GO] = 1 with [OTUT_MODE1:0] = 01 (start the OTUT round robin run) and set when at least 1 cycle of round robin is completed. 0 = OTUT has not started or first round robin has not completed yet. 1 = At least 1 cycle of round robin has completed. | | | | | | | | |

| |
|--|
| <p>DRDY_OVUV = Indicates the OVUV round robin has at least run once. This bit is cleared when $[OVUV_GO] = 1$ with $[OVUV_MODE1:0] = 01$ (start the OVUV round robin run) and set when at least 1 cycle of round robin is completed. 0 = OVUV has not started or first round robin has not completed yet. 1 = At least 1 cycle of round robin has completed.</p> |
| <p>DRDY_BIST_OTUT = Indicates the status of the OTUT protector diagnostic. This bit is cleared when $[OTUT_GO] = 1$ with $[OTUT_MODE1:0] = 10$ (start the BIST run) and set when the BIST cycle is completed. 0 = Not started or still running. 1 = BIST cycle completed.</p> |
| <p>DRDY_BIST_OVUV = Indicates the status of the OVUV protector diagnostic. This bit is cleared when $[OVUV_GO] = 1$ with $[OVUV_MODE1:0] = 10$ (start the BIST run) and set when the BIST cycle is completed. 0 = Not started or still running. 1 = BIST cycle completed.</p> |
| <p>DRDY_BIST_PWR = Indicates the status of the power supplies diagnostic. This bit is cleared when $[PWR_BIST_GO] = 1$ (start the BIST run) and set when the BIST cycle is completed. 0 = Not started or still running. 1 = BIST cycle completed.</p> |

6.5.4.4.2 ADC_STAT1

| Address | 0x0527 | | | | | | | |
|--|--------|-------|-------|---------------|---------------|---------------|---------------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | RSVD | DRDY_AUX_GPIO | DRDY_AUX_CELL | DRDY_AUX_MISC | DRDY_MAIN_ADC | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| <p>DRDY_AUX_GPIO = AUX ADC has completed at least a single measurement on all active GPIO channels configured for ADC measurement. This bit is cleared when $[AUX_GO]$ is changed from 0 to 1. 0 = Not ready 1 = All GPIO inputs have completed at least a single measurement by the AUX ADC</p> | | | | | | | | |
| <p>DRDY_AUX_CELL = Device has completed at least a single measurement on all AUXCELL channel(s) set by $[AUX_CELL_SEL4:0]$. This bit is cleared when $[AUX_GO]$ is changed from 0 to 1. 0 = Not ready 1 = All $[AUX_CELL_SEL4:0]$ configured channels have completed at least a single measurement</p> | | | | | | | | |
| <p>DRDY_AUX_MISC = Device has completed at least a single measurement on all AUX ADC MISC input channels (that is, completed a single round robin run). This bit is cleared when $[AUX_GO]$ is changed from 0 to 1. 0 = Not ready 1 = All AUX ADC MISC inputs have completed at least a single measurement</p> | | | | | | | | |
| <p>DRDY_MAIN_ADC = Device has completed at least a single measurement on all Main ADC input channels, including all GPIOs (that is, completed a single round robin run). This bit is cleared when $[MAIN_GO]$ is changed from 0 to 1. 0 = Not ready 1 = All Main ADC inputs have completed at least a single measurement</p> | | | | | | | | |

6.5.4.4.3 ADC_STAT2

| Address | 0x0528 | | | | | | | |
|-----------------|--------|-------|----------|-----------|-----------|-----------|------------|-----------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | DRDY_LPF | DRDY_GPIO | DRDY_VCOW | DRDY_CBOU | DRDY_CBFET | DRDY_VCCB |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |

| |
|--|
| <p>DRDY_LPF = Device has finished at least 1 round of LPF checks on all active cell channels. The comparison continues in the background as long as the Main ADC is running. This bit is cleared when $[MAIN_GO] = 1$. This data ready bit is also used when a fault is injected to test the DIAG_LPF engine using the $[LPF_FLT_INJ]$ bit. When $[LPF_FLT_INJ] = 1$, this bit is cleared to 0 and the device will restart the VC and BB and BB channel LPF checks from the beginning using the fault inject $[DIAG_LPF]$. Once all channel LPFs are checked, the $[DRDY_LPF] = 1$. 0 = Not ready 1 = Diagnostic comparison finished</p> |
| <p>DRDY_GPIO = Device has finished the GPIO Main and AUX ADC diagnostic comparisons on all active channels and the comparisons are stopped. This bit is cleared when $[COMP_ADC_GO] = 1$. 0 = Not ready 1 = Diagnostic comparison finished</p> |
| <p>DRDY_VCOW = Device has finished VC OW diagnostic comparison on all active channels and the comparison is stopped. This bit is cleared when $[COMP_ADC_GO] = 1$. 0 = Not ready 1 = Diagnostic comparison finished</p> |
| <p>DRDY_CBOW = Device has finished CB OW diagnostic comparison on all active channels and the comparison is stopped. This bit is cleared when $[COMP_ADC_GO] = 1$. 0 = Not ready 1 = Diagnostic comparison finished</p> |
| <p>DRDY_CBFET = Device has finished CB FET diagnostic comparison on all active channels and the comparison is stopped. This bit is cleared when $[COMP_ADC_GO] = 1$. 0 = Not ready 1 = Diagnostic comparison finished</p> |
| <p>DRDY_VCCB = Device has finished VCELL vs. AUXCELL diagnostic comparison on all active channels. This bit is cleared when $[COMP_ADC_GO] = 1$. 0 = Not ready 1 = Diagnostic comparison finished</p> |

6.5.4.4.4 GPIO_STAT

| Address | 0x052A | | | | | | | |
|---|--------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | GPIO8 | GPIO7 | GPIO6 | GPIO5 | GPIO4 | GPIO3 | GPIO2 | GPIO1 |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| GPIO1 through GPIO8 = When GPIO is configured as digital input or output, this register shows the GPIO status. 0 = Low 1 = High | | | | | | | | |

6.5.4.4.5 BAL_STAT

| Address | 0x052B | | | | | | | |
|---|----------------|--------------|------------|--------|--------|----------|---------|---------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | INVALID_CBCONF | OT_PAUSE_DET | CB_INPAUSE | MB_RUN | CB_RUN | ABORTFLT | MB_DONE | CB_DONE |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | R | R | R | R | R | R | R | R |
| INVALID_CBCONF = Indicates CB is unable to start (after $[BAL_GO] = 1$) due to improper CB control settings. Incorrect settings include: <ul style="list-style-type: none"> • More than eight cells are enabled for CB. • More than two adjacent cells are enabled for CB if $DEVICE_CONF[NO_ADJ_CB] = 0$. • Any adjacent cells are enabled for CB if $DEVICE_CONF[NO_ADJ_CB] = 1$. This bit is updated every time $[BAL_GO] = 1$. 0 = Valid CB setting 1 = Invalid CB setting | | | | | | | | |

| |
|---|
| <p>OT_PAUSE_DET = Indicates the OTCB is detected if $[OTCB_EN] = 1$. The bit is also set if CB TWARN is detected, which will also pause CB. Valid only after $[BAL_GO] = 1$ 0 = No OTCB or CB TWARN is detected 1 = Any NTC thermistor measurement is greater than $OTCB_THR[3:0]$ setting, or die (CBFET) temperature is greater than CB TWARN</p> |
| <p>CB_INPAUSE = Indicates the cell balancing pause status. 0 = CB is running or not started 1 = Paused (can be caused by OTCB detection, or host sets $[CB_PAUSE] = 1$)</p> |
| <p>MB_RUN = Indicates module balancing, controlled by the device, is running. Only valid if MB_TIMER_CTRL is not 0x00 and after $[BAL_GO] = 1$. Does not indicate the cell balancing status. 0 = Completed or not started 1 = Module balancing, controlled by the device, is running</p> |
| <p>CB_RUN = Indicates cell balancing is running. Only valid after $[BAL_GO] = 1$. Does not indicate the module balancing status. This bit remains as 1 even if CB is in pause state. 0 = Completed or not started 1 = At least 1 cell is in active cell balancing</p> |
| <p>ABORTFLT = Indicates cell balancing is aborted due to detection of unmasked fault. Cleared when $BAL_CTRL1[BAL_GO] = 1$. CB abort does not trigger if CB is in pause ($[CB_INPAUSE] = 1$) even if an unmasked fault is detected. The abort at fault function will resume if CB is no longer in pause state. 0 = Not aborted or cell balancing not running 1 = Aborted</p> |
| <p>MB_DONE = Indicates module balancing is completed. Cleared when $BAL_CTRL1[BAL_GO] = 1$. 0 = Module balancing is still running or has not started 1 = Module balancing completed</p> |
| <p>CB_DONE = Indicates all cell balancing is completed. Cleared when $BAL_CTRL1[BAL_GO] = 1$. 0 = Cell balancing is still running or has not started 1 = All cell balancing is completed</p> |

6.5.4.4.6 DEV_STAT

| Address | 0x052C | | | | | | | |
|---|--------|---------------|---------------|----------|----------|-------|---------|----------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | FACT_CRC_DONE | CUST_CRC_DONE | OTUT_RUN | OVUV_RUN | RSVD | AUX_RUN | MAIN_RUN |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| FACT_CRC_DONE = Indicates the status of the factory CRC state machine. This bit is set when the factory CRC is calculated and verified internally at least once. A read from this register will clear this bit. 0 = Not complete 1 = Complete (cleared on read) | | | | | | | | |
| CUST_CRC_DONE = Indicates the status of the customer CRC state machine. This bit is set when the CRC is calculated and compared to the $CUST_CRC^*$ registers at least once. A read from this register will clear this bit. 0 = Not complete 1 = Complete (cleared on read) | | | | | | | | |
| OTUT_RUN = Shows the status of the OTUT protector comparators. This bit is set when OTUT BIST starts. When BIST is completed or aborted, the device will turn off the OT and UT comparators automatically, and then this bit will be cleared). 0 = off (that is, OTUT is not started or when $[OTUT_GO] = 1$ and $[OTUT_MODE1:0] = 0$) 1 = on (that is, when $[OTUT_GO] = 1$ and $[OTUT_MODE1:0]$ is non-zero) | | | | | | | | |
| OVUV_RUN = Shows the status of the OVUV protector comparators. This bit is set when OVUV BIST starts. When BIST is completed or aborted, the device will turn off the OV and UV comparators automatically, and then this bit will be cleared). 0 = off (that is, OVUV is not started or when $[OVUV_GO] = 1$ and $[OVUV_MODE1:0] = 0$) 1 = on (that is, when $[OVUV_GO] = 1$ and $[OVUV_MODE1:0]$ is non-zero) | | | | | | | | |
| AUX_RUN = Shows the status of the AUX ADC. 0 = off 1 = on | | | | | | | | |

MAIN_RUN = Shows the status of the Main ADC.
0 = off
1 = on

6.5.4.5 ADC Configuration and Control

6.5.4.5.1 ADC_CONF1

| Address | 0x0007 | | | | | | | |
|--|-----------------|-------|-------------|-------|-------|----------------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | AUX_SETTLE[1:0] | | LPF_BB[2:0] | | | LPF_VCELL[2:0] | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| <p>AUX_SETTLE[1:0] = The AUXCELL configures the AUX CELL settling time. Each AUXCELL has to wait for the anti-aliasing filter (AAF) settling time in order to consider as a valid measurement. These bits provide the option to use different AAF or bypass an AAF to trade for a fast measurement. 00 = 4.3 ms 01 = 2.3 ms 10 = 1.3 ms 11 = Reserved</p> <p>LPF_BB[2:0] = Configures the post main SAR ADC low-pass filter cut-off frequency for BBP/N measurement. Same options as the LPF_VCELL[2:0].</p> <p>LPF_VCELL[2:0] = Configures the post ADC low-pass filter cut-off frequency for VCELL measurement. 0x0 = 6.5 Hz (154 ms average) 0x1 = 13 Hz (77 ms average) 0x2 = 26 Hz (38 ms average) 0x3 = 53 Hz (19 ms average) 0x4 = 111 Hz (9 ms average) 0x5 = 240 Hz (4 ms average) 0x6 = 600 Hz (1.6 ms average) 0x7 = 240 Hz</p> | | | | | | | | |

6.5.4.5.2 ADC_CONF2

| Address | 0x0008 | | | | | | | |
|---|------------|-------|--------------|-------|-------|-------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | SPARE[1:0] | | ADC_DLY[5:0] | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| <p>SPARE[1:0] = Spare</p> <p>ADC_DLY[5:0] = If <i>[MAIN_GO]</i> bit is written to 1, bit Main ADC is delayed for this setting time before being enabled to start the conversion. This setting synchronizes the start of Main ADC throughout the daisy-chained stack. The option ranges from 0 μs (no delay) to 200 μs in 5-μs steps. Undefined code = 0 μs (no delay)</p> | | | | | | | | |

6.5.4.5.3 MAIN_ADC_CAL1

| Address | 0x001B | | | | | | | |
|--|------------|-------|-------|-------|-------|-------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | GAINL[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| <p>GAINL[7:0] = Main ADC 25°C gain calibration result (lower 8-bit). If customer performs gain calibration during production flow, the gain result can be programmed to OTP and will be sent to this gain register at device reset. The device automatically applies this data during ADC correction step. Range from -0.78125% to 0.7782% in 0.0031% steps.</p> | | | | | | | | |

6.5.4.5.4 MAIN_ADC_CAL2

| Address | 0x001C | | | | | | | |
|---------|--------|-------------|-------|-------|-------|-------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | GAINH | OFFSET[6:0] | | | | | | |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| GAINH Main ADC 25°C gain calibration result (MS bit). If customer performs gain calibration during production flow, the gain result can be programmed to OTP and will be sent to this gain register at device reset. The device automatically applies this data during ADC correction step. Range from -0.78125% to 0.7782% in 0.0031% steps. | | | | | | | | |
| OFFSET[6:0] = Main ADC 25°C offset calibration result. If customer performs offset calibration during production flow, the offset result can be programmed to OTP and will be sent to this offset register at device reset. The device automatically applies this data during ADC correction step. Range from -12.20703-mV to 12.01630-mV in 0.19073-mV steps | | | | | | | | |

6.5.4.5.5 AUX_ADC_CAL1

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x001D | | | | | | | |
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | GAINL[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| GAINL[7:0] = AUX ADC 25°C gain calibration result (lower 8-bit). If customer performs gain calibration during production flow, the gain result can be programmed to OTP and will be sent to this gain register at device reset. The device automatically applies this data during ADC correction step. Range from -0.78125% to 0.7782% in 0.0031% steps. | | | | | | | | |

6.5.4.5.6 AUX_ADC_CAL2

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x001E | | | | | | | |
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | GAINH | OFFSET[6:0] | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| GAINH AUX ADC 25°C gain calibration result (MS bit). If customer performs gain calibration during production flow, the gain result can be programmed to OTP and will be sent to this gain register at device reset. The device automatically applies this data during ADC correction step. Range from -0.78125% to 0.7782% in 0.0031% steps. | | | | | | | | |
| OFFSET[6:0] = AUX ADC 25°C offset calibration result. If customer performs offset calibration during production flow, the offset result can be programmed to OTP and will be sent to this offset register at device reset. The device automatically applies this data during ADC correction step. Range from -12.20703-mV to 12.01630-mV in 0.19073-mV steps | | | | | | | | |

6.5.4.5.7 ADC_CTRL1

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|----------------|--------------|
| Address | 0x030D | | | | | | | |
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | RSVD | RSVD | LPF_BB_EN | LPF_VCELL_EN | MAIN_GO | MAIN_MODE[1:0] | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| LPF_BB_EN = Enables digital low-pass filter post-ADC conversion. LPF applies to BBP/N measurements only. The cut-off frequency is configured by <i>ADC_CONFIG1[LPF_BB[2:0]]</i> . | | | | | | | | |
| LPF_VCELL_EN = Enables digital low-pass filter post-ADC conversion. LPF applies to VCELL measurements only. The cut-off frequency is configured by <i>ADC_CONFIG1[LPF_VCELL[2:0]]</i> . | | | | | | | | |
| MAIN_GO = Starts main ADC conversion. When this bit is written to 1, all Main ADC inputs are sampled. Once the Main ADC is started, any change to the Main ADC control setting has no effect until this bit is written to 1 again. This bit is cleared to 0 in read. 0 = Ready. Writing 0 has no effect 1 = Start Main ADC | | | | | | | | |

MAIN_MODE[1:0] = Sets the Main ADC run mode. In continuous run, if user would like to stop ADC, user must read all the ADC conversion results, then stop it. ADC results are not valid before ADC is reenabled next time.
 00 = Main ADC not running
 01 = Single run. Run the main ADC round robin 8 times and then stop
 10 = Continuous run. Continuous running the Main ADC round robin until host sends command to stop
 11 = Reserved

6.5.4.5.8 ADC_CTRL2

| Address | 0x030E | | | | | | | |
|---|--------|-------|----------------|-------------------|-------|-------|-------|-------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | RSVD | AUX_CELL_ALIGN | AUX_CELL_SEL[4:0] | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| AUX_CELL_ALIGN = Align the AUX ADC AUXCELL measurement to Main ADC CELL1 or CELL8 0 = Dynamic Alignment 1 = Align to Main ADC CELL8 | | | | | | | | |
| AUX_CELL_SEL[4:0] = Selects which AUXCELL channel(s) will be multiplexed through the AUX ADC. 0x00 = Run all active cell channels set by ACTIVE_CELL_CONF register 0x01 = Lock to AUX Busbar (BBP-BBN) 0x02 = Lock to AUXCELL1 0x03 = Lock to AUXCELL2 0x04 = Lock to AUXCELL3 : 0x11 = Lock to AUXCELL16 0x12 to 0x1F = RSVD NOTE: If inactive channel or RSVD code is selected, device will not perform AUX ADC conversion on the AUXCELL slot and the AUX_CELL_HI/LO registers will be kept in reset value. | | | | | | | | |

6.5.4.5.9 ADC_CTRL3

| Address | 0x030F | | | | | | | |
|--|--------|-------------------|-------|-------|-------|--------|---------------|-------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | AUX_GPIO_SEL[3:0] | | | | AUX_GO | AUX_MODE[1:0] | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| AUX_GPIO_SEL[3:0] = Selects which GPIO channel(s) will be multiplexed through the AUX ADC to use for temperature measurement diagnostic. If this selection is not set to 0x00, the AUX ADC will lock onto a single GPIO channel and the measurement result is output to the AUX_GPIO_HI/LO registers. 0x00 = AUX ADC cycles through all GPIO channel(s) that are configured as ADC only or ADC and OTUT. 0x01 = Lock to GPIO1 0x02 = Lock to GPIO2 : 0x08 = Lock to GPIO8 All other codes are RSVD. NOTE: If GPIO is not configured for ADC measurement or RSVD codes are selected, device will not perform AUX ADC conversion on the GPIO slot and the AUX_GPIO_HI/LO registers will be kept in reset value. | | | | | | | | |
| AUX_GO = Starts AUX ADC conversion. When this bit is written to 1, all AUX ADC inputs are sampled. Once the AUX ADC is started, any change to the AUX ADC control setting has no effect until this bit is written to 1 again. This bit is cleared to 0 in read. 0 = Ready. Writing 0 has no effect. 1 = Start AUX ADC | | | | | | | | |
| AUX_MODE[1:0] = Sets the Main ADC run mode. In continuous run, if user would like to stop ADC, user must read all the ADC conversion results, then stop it. ADC results are not valid before ADC is reenabled next time. 00 = AUX ADC not running 01 = Single run. Run the AUX ADC round robin once and then stop. 10 = Continuous run. Continually run the AUX ADC round robin until host sends command to stop. 11 = 8-round-robin run to measure all eight GPIOs once. | | | | | | | | |

6.5.4.6 ADC Measurement Results

6.5.4.6.1 VCELL16_HI/LO

VCELL16_HI

| Address | 0x0568 | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the high-byte of the Cell16 voltage in 2s complement. When host reads this register, the device locks the Cell16 voltage low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

VCELL16_LO

| Address | 0x0569 | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the low-byte of the Cell16 voltage in 2s complement. | | | | | | | | |

6.5.4.6.2 VCELL15_HI/LO

VCELL15_HI

| Address | 0x056A | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the high-byte of the Cell15 voltage in 2s complement. When host reads this register, the device locks the Cell15 voltage low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

VCELL15_LO

| Address | 0x056B | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the low-byte of the Cell15 voltage in 2s complement. | | | | | | | | |

6.5.4.6.3 VCELL14_HI/LO

VCELL14_HI

| Address | 0x056C | | | | | | | |
|-----------|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

RESULT[7:0] = The ADC measurement result of the high-byte of the Cell14 voltage in 2s complement. When host reads this register, the device locks the Cell14 voltage low-byte from updating until the high-byte and low-byte registers are read.

VCELL14_LO

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x056D | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the low-byte of the Cell14 voltage in 2s complement. | | | | | | | | |

6.5.4.6.4 VCELL13_HI/LO

VCELL13_HI

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x056E | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the high-byte of the Cell13 voltage in 2s complement. When host reads this register, the device locks the Cell13 voltage low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

VCELL13_LO

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x056F | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the low-byte of the Cell13 voltage in 2s complement. | | | | | | | | |

6.5.4.6.5 VCELL12_HI/LO

VCELL12_HI

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0570 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the high-byte of the Cell12 voltage in 2s complement. When host reads this register, the device locks the Cell12 voltage low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

VCELL12_LO

| | | | | | | | | |
|----------------|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0571 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

RESULT[7:0] = The ADC measurement result of the low-byte of the Cell12 voltage in 2s complement.

6.5.4.6.6 VCELL11_HI/LO

VCELL11_HI

| Address | 0x0572 | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the high-byte of the Cell11 voltage in 2s complement. When host reads this register, the device locks the Cell11 voltage low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

VCELL11_LO

| Address | 0x0573 | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the low-byte of the Cell11 voltage in 2s complement. | | | | | | | | |

6.5.4.6.7 VCELL10_HI/LO

VCELL10_HI

| Address | 0x0574 | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the high-byte of the Cell10 voltage in 2s complement. When host reads this register, the device locks the Cell10 voltage low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

VCELL10_LO

| Address | 0x0575 | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the low-byte of the Cell10 voltage in 2s complement. | | | | | | | | |

6.5.4.6.8 VCELL9_HI/LO

VCELL9_HI

| Address | 0x0576 | | | | | | | |
|-----------|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

RESULT[7:0] = The ADC measurement result of the high-byte of the Cell9 voltage in 2s complement. When host reads this register, the device locks the Cell9 voltage low-byte from updating until the high-byte and low-byte registers are read.

VCELL9_LO

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0577 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the low-byte of the Cell9 voltage in 2s complement. | | | | | | | | |

6.5.4.6.9 VCELL8_HI/LO

VCELL8_HI

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0578 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the high-byte of the Cell8 voltage in 2s complement. When host reads this register, the device locks the Cell8 voltage low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

VCELL8_LO

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0579 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the low-byte of the Cell8 voltage in 2s complement. | | | | | | | | |

6.5.4.6.10 VCELL7_HI/LO

VCELL7_HI

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x057A | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the high-byte of the Cell7 voltage in 2s complement. When host reads this register, the device locks the Cell7 voltage low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

VCELL7_LO

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x057B | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the low-byte of the Cell7 voltage in 2s complement. | | | | | | | | |

6.5.4.6.11 VCELL6_HI/LO

VCELL6_HI

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x057C | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the high-byte of the Cell6 voltage in 2s complement. When host reads this register, the device locks the Cell6 voltage low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

VCELL6_LO

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x057D | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the low-byte of the Cell6 voltage in 2s complement. | | | | | | | | |

6.5.4.6.12 VCELL5_HI/LO

VCELL5_HI

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x057E | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the high-byte of the Cell5 voltage in 2s complement. When host reads this register, the device locks the Cell5 voltage low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

VCELL5_LO

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x057F | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the low-byte of the Cell5 voltage in 2s complement. | | | | | | | | |

6.5.4.6.13 VCELL4_HI/LO

VCELL4_HI

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0580 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the high-byte of the Cell4 voltage in 2s complement. When host reads this register, the device locks the Cell4 voltage low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

VCELL4_LO

| | | | | | | | | |
|----------------|---------------|--|--|--|--|--|--|--|
| Address | 0x0581 | | | | | | | |
|----------------|---------------|--|--|--|--|--|--|--|

| | | | | | | | | |
|---|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the low-byte of the Cell4 voltage in 2s complement. | | | | | | | | |

6.5.4.6.14 VCELL3_HI/LO

VCELL3_HI

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0582 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the high-byte of the Cell3 voltage in 2s complement. When host reads this register, the device locks the Cell3 voltage low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

VCELL3_LO

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0583 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the low-byte of the Cell3 voltage in 2s complement. | | | | | | | | |

6.5.4.6.15 VCELL2_HI/LO

VCELL2_HI

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0584 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the high-byte of the Cell2 voltage in 2s complement. When host reads this register, the device locks the Cell2 voltage low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

VCELL2_LO

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0585 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the low-byte of the Cell2 voltage in 2s complement. | | | | | | | | |

6.5.4.6.16 VCELL1_HI/LO

VCELL1_HI

| | | | | | | | | |
|----------------|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0586 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |

| | | | | | | | | |
|--|---|---|---|---|---|---|---|---|
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the high-byte of the Cell1 voltage in 2s complement. When host reads this register, the device locks the Cell1 voltage low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

VCELL1_LO

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0587 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the low-byte of the Cell1 Voltage in 2s complement. | | | | | | | | |

6.5.4.6.17 BUSBAR_HI/LO**BUSBAR_HI**

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0588 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the high-byte of the differential bus bar pins (BBP – BBN) in 2s complement. When host reads this register, the device locks the low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

BUSBAR_LO

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0589 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the low-byte of the differential bus bar pins (BBP – BBN) in 2s complement. | | | | | | | | |

6.5.4.6.18 TSREF_HI/LO**TSREF_HI**

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x058C | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The TSREF high-byte result from Main ADC. When host reads this register, the device locks the TSREF low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

TSREF_LO

| | | | | | | | | |
|----------------|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x058D | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

RESULT[7:0] = The TSREF low-byte result from Main ADC

6.5.4.6.19 GPIO1_HI/LO

GPIO1_HI

| Address | 0x058E | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement high-byte result of the GPIO1. When host reads this register, the device locks the GPIO1 low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

GPIO1_LO

| Address | 0x058F | | | | | | | |
|---|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement low-byte result of the GPIO1. | | | | | | | | |

6.5.4.6.20 GPIO2_HI/LO

GPIO2_HI

| Address | 0x0590 | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement high-byte result of the GPIO2. When host reads this register, the device locks the GPIO2 low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

GPIO2_LO

| Address | 0x0591 | | | | | | | |
|---|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement low-byte result of the GPIO2. | | | | | | | | |

6.5.4.6.21 GPIO3_HI/LO

GPIO3_HI

| Address | 0x0592 | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement high-byte result of the GPIO3. When host reads this register, the device locks the GPIO3 low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

GPIO3_LO

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0593 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement low-byte result of the GPIO3. | | | | | | | | |

6.5.4.6.22 GPIO4_HI/LO**GPIO4_HI**

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0594 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement high-byte result of the GPIO4. When host reads this register, the device locks the GPIO4 low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

GPIO4_LO

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0595 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement low-byte result of the GPIO4. | | | | | | | | |

6.5.4.6.23 GPIO5_HI/LO**GPIO5_HI**

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0596 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement high-byte result of the GPIO5. When host reads this register, the device locks the GPIO5 low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

GPIO5_LO

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0597 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement low-byte result of the GPIO5. | | | | | | | | |

6.5.4.6.24 GPIO6_HI/LO**GPIO6_HI**

| | | | | | | | | |
|----------------|---------------|--|--|--|--|--|--|--|
| Address | 0x0598 | | | | | | | |
|----------------|---------------|--|--|--|--|--|--|--|

| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement high-byte result of the GPIO6. When host reads this register, the device locks the GPIO6 low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

GPIO6_LO

| Address | 0x0599 | | | | | | | |
|---|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement low-byte result of the GPIO6. | | | | | | | | |

6.5.4.6.25 GPIO7_HI/LO

GPIO7_HI

| Address | 0x059A | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement high-byte result of the GPIO7. When host reads this register, the device locks the GPIO7 low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

GPIO7_LO

| Address | 0x059B | | | | | | | |
|---|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement low-byte result of the GPIO7. | | | | | | | | |

6.5.4.6.26 GPIO8_HI/LO

GPIO8_HI

| Address | 0x059C | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement high-byte result of the GPIO8. When host reads this register, the device locks the GPIO8 low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

GPIO8_LO

| Address | 0x059D | | | | | | | |
|-----------|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement low-byte result of the GPIO8. | | | | | | | | |

6.5.4.6.27 DIETEMP1_HI/LO

DIETEMP1_HI

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05AE | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The DieTemp1 high-byte result from Main ADC. When host reads this register, the device locks the DIETEMP1 low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

DIETEMP1_LO

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05AF | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The DieTemp1 low-byte (temperature used for ADC correction) result from Main ADC. | | | | | | | | |

6.5.4.6.28 DIETEMP2_HI/LO

DIETEMP2_HI

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05B0 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The DieTemp2 high-byte result from AUX ADC. When host reads this register, the device locks the DIETEMP2 low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

DIETEMP2_LO

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05B1 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The DieTemp2 low-byte (temperature used for ADC correction) result from AUX ADC | | | | | | | | |

6.5.4.6.29 AUX_CELL_HI/LO

AUX_CELL_HI

| | | | | | | | | |
|----------------|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05B2 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

RESULT[7:0] = The ADC measurement result of the high-byte of the AUXCELL voltage in 2s complement.
 These *AUX_CELL_HI/LO* registers will only report AUXCELL voltage measurement if host configures [*AUX_CELL_SEL4:0*] to lock to a single AUXCELL channel.
 When host reads this register, the device locks the AUXCELL voltage low-byte from updating until the high-byte and low-byte registers are read.

AUX_CELL_LO

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05B3 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The ADC measurement result of the low-byte of the AUX cell voltage in 2s complement. These <i>AUX_CELL_HI/LO</i> registers will only report AUXCELL voltage measurement if host configures [<i>AUX_CELL_SEL4:0</i>] to lock to a single AUXCELL channel. | | | | | | | | |

6.5.4.6.30 AUX_GPIO_HI/LO

AUX_GPIO_HI

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05B4 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The AUX ADC measurement high-byte result of the GPIO that is locked by the [<i>AUXGPIO_SEL3:0</i>] bits. When host reads this register, the device locks the AUX_GPIO low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

AUX_GPIO_LO

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05B5 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The AUX ADC measurement low-byte result of the GPIO that is locked by the [<i>AUXGPIO_SEL3:0</i>] bits. | | | | | | | | |

6.5.4.6.31 AUX_BAT_HI/LO

AUX_BAT_HI

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05B6 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The high-byte result of the BAT pin measurement from AUX ADC. When host reads this register, the device locks the AUX_BAT low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

AUX_BAT_LO

| | | | | | | | | |
|----------------|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05B7 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |

| | | | | | | | | |
|--|---|---|---|---|---|---|---|---|
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The low-byte result of the BAT pin measurement from AUX ADC. | | | | | | | | |

6.5.4.6.32 AUX_REFL_HI/LO

AUX_REFL_HI

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05B8 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The high-byte result of the internal reference, REFL, measurement from AUX ADC. When host reads this register, the device locks the AUX_REL low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

AUX_REFL_LO

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05B9 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The low-byte result of the internal reference, REFL, measurement from AUX ADC. | | | | | | | | |

6.5.4.6.33 AUX_VBG2_HI/LO

AUX_VBG2_HI

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05BA | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The high-byte result of the internal reference, VBG2, measurement from AUX ADC. When host reads this register, the device locks the AUX_VBG2 low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

AUX_VBG2_LO

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05BB | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The low-byte result of the internal reference, VBG2, measurement from AUX ADC. | | | | | | | | |

6.5.4.6.34 AUX_AVAO_REF_HI/LO

AUX_AVAO_REF_HI

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05BE | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The high-byte result of the AVAO_REF measurement from AUX ADC. When host reads this register, the device locks the AUX_AVAO_REF low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

AUX_AVAO_REF_LO

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05BF | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The low-byte result of the AVAO_REF measurement from AUX ADC. | | | | | | | | |

6.5.4.6.35 AUX_AVDD_REF_HI/LO

AUX_AVDD_REF_HI

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05C0 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The high-byte result of the AVDD_REF measurement from AUX ADC. When host reads this register, the device locks the AUX_AVDD_REF low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

AUX_AVDD_REF_LO

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05C1 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The low-byte result of the AVDD_REF measurement from AUX ADC. | | | | | | | | |

6.5.4.6.36 AUX_OV_DAC_HI/LO

AUX_OV_DAC_HI

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05C2 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The high-byte result of the OV comparator DAC measurement, which is (0.8 x OV threshold), from AUX ADC. When host reads this register, the device locks the AUX_OV_DAC low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

AUX_OV_DAC_LO

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x05C3 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The low-byte result of the OV comparator DAC measurement, which is (0.8 x OV threshold), from AUX ADC. | | | | | | | | |

6.5.4.6.37 AUX_UV_DAC_HI/LO

AUX_UV_DAC_HI

| | | | | | | | | |
|----------------|---------------|--|--|--|--|--|--|--|
| Address | 0x05C4 | | | | | | | |
|----------------|---------------|--|--|--|--|--|--|--|

| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The high-byte result of the UV comparator DAC measurement, which is (0.8 x UV threshold), from AUX ADC. When host reads this register, the device locks the AUX_UV_DAC low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

AUX_UV_DAC_LO

| Address | 0x05C5 | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The low-byte result of the UV comparator DAC measurement, which is (0.8 x UV threshold), from AUX ADC. | | | | | | | | |

6.5.4.6.38 AUX_OT_OTCB_DAC_HI/LO

AUX_OT_OTCB_DAC_HI

| Address | 0x05C6 | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The high-byte result of the OT comparator (either OT or OTCB threshold based on [OTCB_THR_LOCK] setting) DAC measurement from AUX ADC. When host reads this register, the device locks the AUX_OT_OTCB_DAC low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

AUX_OT_OTCB_DAC_LO

| Address | 0x05C7 | | | | | | | |
|---|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The low-byte result of the OT comparator (either OT or OTCB threshold based on [OTCB_THR_LOCK] setting) DAC measurement from AUX ADC. | | | | | | | | |

6.5.4.6.39 AUX_UT_DAC_HI/LO

AUX_UT_DAC_HI

| Address | 0x05C8 | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The high-byte result of the UT comparator DAC measurement from AUX ADC. When host reads this register, the device locks the AUX_UT_DAC low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

AUX_UT_DAC_LO

| Address | 0x05C9 | | | | | | | |
|-----------|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

RESULT[7:0] = The low-byte result of the UT comparator DAC measurement from AUX ADC.

6.5.4.6.40 AUX_VCBDONE_DAC_HI/LO

AUX_VCBDONE_DAC_HI

| Address | 0x05CA | | | | | | | |
|---|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The high-byte result of the UV comparator (VCBDONE Threshold) DAC measurement from AUX ADC. When host reads this register, the device locks the AUX_VCBDONE_DAC low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

AUX_VCBDONE_DAC_LO

| Address | 0x05CB | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The low-byte result of the UV comparator (VCBDONE Threshold) DAC measurement from AUX ADC. | | | | | | | | |

6.5.4.6.41 AUX_VCM_HI/LO

AUX_VCM_HI

| Address | 0x05CC | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|-------------|-------|-------|-------|-------|-------|-------|-------|
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The high-byte result of the VCM (common mode voltage on Main ADC) measurement from AUX ADC. When host reads this register, the device locks the AUX_VCM low-byte from updating until the high-byte and low-byte registers are read. | | | | | | | | |

AUX_VCM_LO

| Address | 0x05CD | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The low-byte result of the VCM (common mode voltage on Main ADC) measurement from AUX ADC. | | | | | | | | |

6.5.4.6.42 REFOVDAC_HI/LO

REFOVDAC_HI

| Address | 0x05D0 | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The high-byte result of the recorded OVDAC reference voltage trimmed at factory. | | | | | | | | |

REFOVDAC_LO

| Address | 0x05D1 | | | | | | | |
|-----------|--------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |

| | | | | | | | | |
|---|-------------|---|---|---|---|---|---|---|
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The low-byte result of the recorded OVDAC reference voltage trimmed at factory. | | | | | | | | |

6.5.4.6.43 DIAG_MAIN_HI/LO

DIAG_MAIN_HI

| | | | | | | | | |
|---|-------------|-------|-------|-------|-------|-------|-------|-------|
| Address | 0x05D2 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The high-byte result of reported Main ADC comparison value used in the diagnostic ADC comparison. Valid if the diagnostic ADC comparison is run when a single channel is locked | | | | | | | | |

DIAG_MAIN_LO

| | | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Address | 0x05D3 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The low-byte result of reported Main ADC comparison value used in the diagnostic ADC comparison. Valid if the diagnostic ADC comparison is run when a single channel is locked | | | | | | | | |

6.5.4.6.44 DIAG_AUX_HI/LO

DIAG_AUX_HI

| | | | | | | | | |
|--|-------------|-------|-------|-------|-------|-------|-------|-------|
| Address | 0x05D4 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The high-byte result of reported AUX ADC comparison value used in the diagnostic ADC comparison. Valid if the diagnostic ADC comparison is run when a single channel is locked | | | | | | | | |

DIAG_AUX_LO

| | | | | | | | | |
|---|-------------|-------|-------|-------|-------|-------|-------|-------|
| Address | 0x05D5 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RESULT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RESULT[7:0] = The low-byte result of reported AUX ADC comparison value used in the diagnostic ADC comparison. Valid if the diagnostic ADC comparison is run when a single channel is locked | | | | | | | | |

6.5.4.7 Balancing Configuration, Control and Status

6.5.4.7.1 CB_CELL16_CTRL through CB_CELL1_CTRL

| | | | | | | | | |
|-----------------|---------------------|-------|-------|-----------|-------|-------|-------|-------|
| Address | 0x0318 to 0x0327 | | | | | | | |
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | TIME[4:0] | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |

TIME[4:0] = Sets the timer for cell* balancing. The selection is sampled whenever [BAL_GO] = 1 is set by the host MCU.
 0x00 = 0 s = stop balancing
 0x01 = 10 s
 0x02 = 30 s
 0x03 = 60 s
 0x04 = 300 s
 0x05 to 0x10 = range from 10 min to 120 min in 10-min steps
 0x11 to 0x1F = range from 150 min to 540 min in 30-min steps and 600 min

6.5.4.7.2 VMB_DONE_THRESH

| Address | 0x0328 | | | | | | | |
|---|--------|-------|-------------|-------|-------|-------|-------|-------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | MB_THR[5:0] | | | | | |
| Reset | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| RSVD = Reserved | | | | | | | | |
| MB_THR[5:0] = If MB_TIMER_CTRL is not 0x00 and BAT voltage is less than this threshold, the module balancing through GPIO3 stops. The selection is sampled whenever [AUX_GO] = 1 is set by the host MCU. Note: To use this option, MCU enables the AUX ADC first before sending [BAL_GO] = 1. A new threshold setting will take effect if MCU resends [AUX_GO] = 1. It is not necessary to resend [BAL_GO] = 1 as balancing is already running. Range from 18 V to 65 V with 1-V steps. Unused codes default to 65 V. | | | | | | | | |

6.5.4.7.3 MB_TIMER_CTRL

| Address | 0x0329 | | | | | | | |
|--|--------|-------|-------|-----------|-------|-------|-------|-------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | TIME[4:0] | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| TIME[4:0] = Sets the timer for module balancing. The selection is sampled whenever [BAL_GO] = 1 is set by the host MCU. 0x00 = 0 s = stop balancing 0x01 = 10 s 0x02 = 30 s 0x03 = 60 s 0x04 = 300 s 0x05 to 0x10 = range from 10 min to 120 min in 10-min steps 0x11 to 0x1F = range from 150 min to 540 min in 30-min steps and 600 min | | | | | | | | |

6.5.4.7.4 VCB_DONE_THRESH

| Address | 0x032A | | | | | | | |
|---|--------|-------|-------------|-------|-------|-------|-------|-------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | CB_THR[5:0] | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| CB_THR[5:0] = If a cell voltage is less than this threshold, the cell balancing on that cell stops. This threshold setting applies to all cells. The selection is sampled whenever [OVUV_GO] = 1 is set by the host MCU. Note: To use the VCB_DONE detection feature, host sets this threshold, then issues [OVUV_GO] = 1 before starting CB (that is, sending [BAL_GO] = 1). To change the VCB_DONE threshold detection, set a new threshold then re-issue [OVUV_GO] = 1 for the new threshold to take effect. It is not necessary to re-issue [BAL_GO] = 1 to restart balancing in this case. Range from 2.45-V to 4-V with 25-mV steps, where 0x00 = Disables voltage based on CB_DONE comparison 0x01 = threshold of 2.45-V 0x3F = threshold of 4-V | | | | | | | | |

6.5.4.7.5 OTCB_THRESH

| Address | 0x032B | | | | | | | |
|---------|--------|--|--|--|--|--|--|--|
| | | | | | | | | |

BQ79616

SLUSF21A – JUNE 2023 – REVISED JUNE 2026

| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--|-------|--------------|-------|-------|---------------|-------|-------|-------|
| Name | RSVD | COOLOFF[2:0] | | | OTCB_THR[3:0] | | | |
| Reset | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| RSVD = Reserved | | | | | | | | |
| COOLOFF[2:0] = Sets the COOLOFF hysteresis (resume temperature = OTCB_THR - COOLOFF hysteresis) to resume CB when <i>BAL_CTRL1[OTCB_EN]</i> = 1 and OTCB is detected. The MCU configures the corresponding GPIO(s) to the ADC and OTUT option. Range from 4% to 14% in steps of 2%. Unused code is set to 14%. | | | | | | | | |
| OTCB_THR[3:0] = Sets the OTCB threshold when <i>BAL_CTRL1[OTCB_EN]</i> = 1. The MCU configures the corresponding GPIO(s) to the ADC and OTUT option. Range from 10% to 24% in steps of 2%. Unused code is set to 24%. | | | | | | | | |

6.5.4.7.6 BAL_CTRL1

| Address | 0x032E | | | | | | | |
|---|--------|-------|-------|-------|-------|-----------|-------|-------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | | | DUTY[2:0] | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| DUTY[2:0] = Selection is sampled whenever <i>[BAL_GO]</i> = 1 is set by the host MCU. 0x0 = 5 s 0x1 = 10 s 0x2 = 30 s 0x3 = 60 s 0x4 = 5 min 0x5 = 10 min 0x6 = 20 min 0x7 = 30 min | | | | | | | | |

6.5.4.7.7 BAL_CTRL2

| Address | 0x032F | | | | | | | |
|--|--------|----------|------------|---------|--------------|-------|--------|----------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | CB_PAUSE | FLTSTOP_EN | OTCB_EN | BAL_ACT[1:0] | | BAL_GO | AUTO_BAL |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| CB_PAUSE = Pauses cell balancing on all cells to allow diagnostics to run. 0 = Normal cell balancing operation 1 = Pause all cell balancing | | | | | | | | |
| FLTSTOP_EN = Stops cell or module balancing if unmasked fault occurs. The selection is sampled whenever <i>[BAL_GO]</i> = 1 is set by the host MCU. 0 = Balancing is continuous regardless of fault condition (excluding thermal shutdown) 1 = All CB balancing stops when any unmasked fault occurs | | | | | | | | |
| OTCB_EN = Enables the OTCB detection during cell balancing. The selection is sampled whenever <i>[BAL_GO]</i> = 1 is set by the host MCU. 0 = Disable OTCB detection 1 = Enable OTCB detection | | | | | | | | |
| BAL_ACT[1:0] = Controls the device action when the MB and CB are completed. These bits are samples whenever <i>[BAL_GO]</i> = 1 is set by the host MCU. The action is valid. 00 = No action 01 = Enters SLEEP 10 = Enters SHUTDOWN 11 = Reserved | | | | | | | | |
| BAL_GO = Starts cell or module balancing. When written to 1, all balancing configuration registers are sampled. Any change to the configuration registers has no effect until this bit is written to 1 again. The bit is self-clearing. 0 = Ready 1 = Start balancing | | | | | | | | |

AUTO_BAL = Selects between auto or manual cell balance control. The selection is sampled whenever [BAL_GO] = 1 is set by the host MCU.
 0 = Manual cell balancing
 1 = Auto cell balancing

6.5.4.7.8 BAL_CTRL3

| Address | 0x0330 | | | | | | | |
|---|--------|-------|-------|-------------------|-------|-------|-------|-------------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | BAL_TIME_SEL[3:0] | | | | BAL_TIME_GO |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| BAL_TIME_GO Instruct the device to report the selected CB channel (set by [BAL_TIME_SEL3:0]) remaining balancing time to BAL_TIME register | | | | | | | | |
| BAL_TIME_SEL[3:0] = Select a single CB channel to report its remaining balancing time 0x0 = CB Channel 1 0x1 = CB Channel 2 : 0xF = CB Channel 16 | | | | | | | | |

6.5.4.7.9 CB_COMPLETE1

| Address | 0x0556 | | | | | | | |
|---|-------------|-------------|-------------|-------------|-------------|-------------|-------------|------------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CELL16_DONE | CELL15_DONE | CELL14_DONE | CELL13_DONE | CELL12_DONE | CELL11_DONE | CELL10_DONE | CELL9_DONE |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CELL9_DONE to Cell balance completion for cell9 to cell16. This register is cleared when MCU sets [BAL_GO] = 1. CELL16_DONE = 0 = Balancing on the particular cell is still running or has not started 1 = Balancing completed on the particular cell | | | | | | | | |

6.5.4.7.10 CB_COMPLETE2

| Address | 0x0557 | | | | | | | |
|---|------------|------------|------------|------------|------------|------------|------------|------------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CELL8_DONE | CELL7_DONE | CELL6_DONE | CELL5_DONE | CELL4_DONE | CELL3_DONE | CELL2_DONE | CELL1_DONE |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CELL1_DONE to Cell balance completion for cell1 to cell8. This register is cleared when MCU sets [BAL_GO] = 1. CELL8_DONE = 0 = Balancing on the particular cell is still running or has not started 1 = Balancing completed on the particular cell | | | | | | | | |

6.5.4.7.11 BAL_TIME

| Address | 0x0558 | | | | | | | |
|---|-----------|-----------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | TIME_UNIT | TIME[6:0] | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| TIME_UNIT = Indicates the unit reported by [TIME6:0] 0 = sec 1 = min | | | | | | | | |
| TIME[6:0] = Report the selected CB channel remaining balancing time If [TIME_UNIT] = 0. Time report in sec with 5sec step If [TIME_UNIT] = 1. Time report in min with 5min step | | | | | | | | |

6.5.4.8 Protector Configuration and Control

6.5.4.8.1 OV_THRESH

| Address | 0x0009 | | | | | | | |
|---|--------|-------|-------------|-------|-------|-------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | SPARE | SPARE | OV_THR[5:0] | | | | | |
| Reset | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| SPARE = Spare | | | | | | | | |
| OV_THR[5:0] = Sets the overvoltage threshold for the OV comparator. Changes on these bits require host to send another [OVUV_GO] = 1 command. All settings are at 25-mV steps. 0x02 to 0x0E: range from 2700 mV to 3000 mV 0x12 to 0x1E: range from 3500 mV to 3800 mV 0x22 to 0x2E: range from 4175 mV to 4475 mV All other settings will default to 2700 mV. | | | | | | | | |

6.5.4.8.2 UV_THRESH

| Address | 0x000A | | | | | | | |
|--|--------|-------|-------------|-------|-------|-------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | SPARE | SPARE | UV_THR[5:0] | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| SPARE = Spare | | | | | | | | |
| UV_THR[5:0] = Sets the undervoltage threshold for the UV comparator. Changes on these bits require host to send another [OVUV_GO] = 1 command. All settings are at 50-mV steps. 0x00 to 0x26: range from 1200 mV to 3100 mV All other settings will default to 3100 mV. | | | | | | | | |

6.5.4.8.3 UV_DISABLE1

| Address | 0x000C | | | | | | | |
|---|--------|--------|--------|--------|--------|--------|--------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CELL16 | CELL15 | CELL14 | CELL13 | CELL12 | CELL11 | CELL10 | CELL9 |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CELL9 to CELL10: Indicate which channels shall be excluded from UV and VCB_DONE detection CELL16 = 0 = UV and VCB_DONE monitoring apply to the channel 1 = UV and VCB_DOME monitoring are excluded from the channel | | | | | | | | |

6.5.4.8.4 UV_DISABLE2

| Address | 0x000D | | | | | | | |
|---|--------|-------|-------|-------|-------|-------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CELL8 | CELL7 | CELL6 | CELL5 | CELL4 | CELL3 | CELL2 | CELL1 |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CELL8 to CELL9: Indicate which channels shall be excluded from UV and VCB_DONE detection CELL1 = 0 = UV and VCB_DONE monitoring apply to the channel 1 = UV and VCB_DOME monitoring are excluded from the channel | | | | | | | | |

6.5.4.8.5 OTUT_THRESH

| Address | 0x000B | | | | | | | |
|---------|-------------|-------|-------|-------------|-------|-------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | UT_THR[2:0] | | | OT_THR[4:0] | | | | |
| Reset | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |

UT_THR[2:0] = Sets the UT threshold for the UT comparator. Changes on these bits require host to send another [OTUT_GO] = 1 command. The MCU configures the corresponding GPIO(s) to ADC and OTUT input.
Range from 66% to 80% in steps of 2%

OT_THR[4:0] = Sets the OT threshold for the OT comparator. Changes on these bits require host to send another [OTUT_GO] = 1 command. The MCU configures the corresponding GPIO(s) to ADC and OTUT input.
Range from 10% to 39% in steps of 1%
Unused code defaults to 39%.

6.5.4.8.6 OVUV_CTRL

| Address | 0x032C | | | | | | | | |
|---|------------------|----------------|-------|-------|-------|---------|----------------|-------|--|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| Name | VCBDONE_THR_LOCK | OVUV_LOCK[3:0] | | | | OVUV_GO | OVUV_MODE[1:0] | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| VCBDONE_THR_LOCK = As the UV comparator is switching between UV threshold and VCBDONE threshold to measure the UV DAC or the VCBDONE DAC result for diagnostics, the UV comparator has to lock onto only one threshold before starting the AUX ADC measurement. This bit selects which threshold is locked to the UV comparator. The bit is sampled when <i>OVUV_MODE[1:0]</i> is 0b11 which is locked to a single channel mode. 0 = UV threshold is selected 1 = VCBDONE threshold is selected | | | | | | | | | |
| OVUV_LOCK[3:0] = Configures a particular single channel as the OV and UV comparators input when [OVUV_MOD1:0] = 0b11. Changes on these bits require host to send another [OVUV_GO] = 1 command. 0x0 = Lock to Cell1 0x1 = Lock to Cell2 0x2 = Lock to Cell3 : 0xF = Lock to Cell16 | | | | | | | | | |
| OVUV_GO = Starts the OV and UV comparators. When written to 1, all OVUV configuration settings are sampled. This bit is self-clearing. 0 = Ready 1 = Start OV and UV comparators | | | | | | | | | |
| OVUV_MODE[1:0] = Sets the OV and UV comparators operation mode when [OVUV_GO] = 1. Changes on these bits require host to send another [OVUV_GO] = 1 command. 00 = Do not run OV and UV comparators 01 = Run the OV and UV round robin with all active cells 10 = Run the OV and UV BIST cycle. 11 = Lock OV and UV comparators to a single channel configured by [OVUV_LOCK3:0] Note: Active cells are defined by the <i>ACTIVE_CELL[NUM_CELL3:0]</i> register. | | | | | | | | | |

6.5.4.8.7 OTUT_CTRL

| Address | 0x032D | | | | | | | | |
|--|--------|---------------|----------------|-------|-------|---------|----------------|-------|--|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| Name | RSVD | OTCB_THR_LOCK | OTUT_LOCK[2:0] | | | OTUT_GO | OTUT_MODE[1:0] | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| RSVD = Reserved | | | | | | | | | |
| OTCB_THR_LOCK = As the OT comparator is switching between OT threshold and OTCB threshold to measure the OT or OTCB DAC threshold result for diagnostics, the OT comparator has to lock onto only one threshold before starting the AUX ADC measurement. This bit selects which threshold is locked to the OT comparator. The bit is sampled when <i>OTUT_MODE[1:0]</i> = 0b11 which is locked to a single channel mode. 0 = OT threshold is selected 1 = OTCB threshold is selected | | | | | | | | | |
| OTUT_LOCK[2:0] = Configures a particular single channel as the OT and UT comparators input when [OTUT_MOD1:0] = 0b11. Changes on these bits require host to send another [OTUT_GO] = 1 command. 0x0 = Lock to GPIO1A 0x1 = Lock to GPIO2A : 0x7 = Lock to GPIO8A | | | | | | | | | |

OTUT_GO = Starts the OT and UT comparators. When written to 1, all OTUT configuration settings are sampled. This bit is self-clearing.
 0 = Ready
 1 = Start OT and UT comparators

OTUT_MODE[1:0] = Sets the OT and UT comparators operation mode when [OTUT_GO] = 1. Changes on these bits require host to send another [OTUT_GO] = 1 command.
 00 = Do not run OT and UT comparators
 01 = Run the OT and UT round robin with all active cells
 10 = Run the OT and UT BIST cycle.
 11 = Lock OT and UT comparators to a single channel configured by [OTUT_LOCK3:0]

6.5.4.9 GPIO Configuration

6.5.4.9.1 GPIO_CONF1

| Address | 0x000E | | | | | | | |
|---|-------------|--------|------------|-------|-------|------------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | FAULT_IN_EN | SPI_EN | GPIO2[2:0] | | | GPIO1[2:0] | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| FAULT_IN_EN = Enables GPIO8 as an active-low input to trigger the NFAULT pin when the input signal is low. 0 = No fault input function. GPIO8 is configured based on [GPIO8_CONF2:0] setting. 1 = GPIO8 is set as active-low input to trigger NFAULT pin, [GPIO8_CONF2:0] setting is ignored. | | | | | | | | |
| SPI_EN = Enables SPI controller on GPIO4, GPIO5 and GPIO6, GPIO7. 0 = SPI controller disabled. 1 = SPI controller enabled. Overwrite the [GPIO4_CONF2:0], [GPIO5_CONF2:0], [GPIO6_CONF2:0], and [GPIO7_CONF2:0] settings. | | | | | | | | |
| GPIO2[2:0] = Configures GPIO2. 000 = As disabled, high-Z 001 = As ADC and OTUT inputs 010 = As ADC only input 011 = As digital input 100 = As output high 101 = As output low 110 = As ADC input and weak pull-up enabled 111 = As ADC input and weak pull-down enabled | | | | | | | | |
| GPIO1[2:0] = Configures GPIO1. 000 = As disabled, high-Z 001 = As ADC and OTUT inputs 010 = As ADC only input 011 = As digital input 100 = As output high 101 = As output low 110 = As ADC input and weak pull-up enabled 111 = As ADC input and weak pull-down enabled | | | | | | | | |

6.5.4.9.2 GPIO_CONF2

| Address | 0x000F | | | | | | | |
|---------------|--------|-------|------------|-------|-------|------------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | SPARE | RSVD | GPIO4[2:0] | | | GPIO3[2:0] | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| SPARE = Spare | | | | | | | | |

| |
|--|
| <p>GPIO4[2:0] = Configures GPIO4. If $[SPI_EN] = 1$, these configuration bits are ignored and the pin is used as SS for SPI controller. See Section 6.3.6.1.7 for details.</p> <ul style="list-style-type: none"> 000 = As disabled, high-Z 001 = As ADC and OTUT inputs 010 = As ADC only input 011 = As digital input 100 = As output high 101 = As output low 110 = As ADC input and weak pull-up enabled 111 = As ADC input and weak pull-down enabled |
| <p>GPIO3[2:0] = Configures GPIO3. If MB_TIMER_CTRL is not 0x00, this configuration is ignored and the pin is configured for module balancing.</p> <ul style="list-style-type: none"> 000 = As disabled, high-Z 001 = As ADC and OTUT inputs 010 = As ADC only input 011 = As digital input 100 = As output high 101 = As output low 110 = As ADC input and weak pull-up enabled 111 = As ADC input and weak pull-down enabled |

6.5.4.9.3 GPIO_CONF3

| Address | 0x0010 | | | | | | | |
|--|------------|-------|------------|-------|-------|------------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | SPARE[1:0] | | GPIO6[2:0] | | | GPIO5[2:0] | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| <p>SPARE[1:0] = Spare</p> | | | | | | | | |
| <p>GPIO6[2:0] = Configures GPIO6. If $[SPI_EN] = 1$, these configuration bits are ignored and the pin is used as MOSI for SPI controller. See Section 6.3.6.1.7 for details.</p> <ul style="list-style-type: none"> 000 = As disabled, high-Z 001 = As ADC and OTUT inputs 010 = As ADC only input 011 = As digital input 100 = As output high 101 = As output low 110 = As ADC input and weak pull-up enabled 111 = As ADC input and weak pull-down enabled | | | | | | | | |
| <p>GPIO5[2:0] = Configures GPIO5. If $[SPI_EN] = 1$, these configuration bits are ignored and the pin is used as MISO for SPI controller. See Section 6.3.6.1.7 for details.</p> <ul style="list-style-type: none"> 000 = As disabled, high-Z 001 = As ADC and OTUT inputs 010 = As ADC only input 011 = As digital input 100 = As output high 101 = As output low 110 = As ADC input and weak pull-up enabled 111 = As ADC input and weak pull-down enabled | | | | | | | | |

6.5.4.9.4 GPIO_CONF4

| Address | 0x0011 | | | | | | | |
|---------------------------|------------|-------|------------|-------|-------|------------|-------|-------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | SPARE[1:0] | | GPIO8[2:0] | | | GPIO7[2:0] | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| <p>SPARE[1:0] = Spare</p> | | | | | | | | |

GPIO8[2:0] = Configures GPIO8. If $[FAULT_IN_EN] = 1$, these configuration bits are ignored and the pin is used as an input such that an active low will trigger NFAULT.

- 000 = As disabled, high-Z
- 001 = As ADC and OTUT inputs
- 010 = As ADC only input
- 011 = As digital input
- 100 = As output high
- 101 = As output low
- 110 = As ADC input and weak pull-up enabled
- 111 = As ADC input and weak pull-down enabled

GPIO7[2:0] = Configures GPIO7. If $[SPI_EN] = 1$, these configuration bits are ignored and the pin is used as SCLK for SPI controller. See [Section 6.3.6.1.7](#) for details.

- 000 = As disabled, high-Z
- 001 = As ADC and OTUT inputs
- 010 = As ADC only input
- 011 = As digital input
- 100 = As output high
- 101 = As output low
- 110 = As ADC input and weak pull-up enabled
- 111 = As ADC input and weak pull-down enabled

6.5.4.10 SPI Controller

6.5.4.10.1 SPI_CONF

| Address | 0x034D | | | | | | | |
|---|--------|-------|-------|-------------|-------|-------|-------|-------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | CPOL | CPHA | NUMBIT[4:0] | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| CPOL = Sets the SCLK polarity. 0 = Idles low and clocks high 1 = Idles high and clocks low | | | | | | | | |
| CPHA = Sets the edge of SCLK where data is sampled on MISO. 0 = Leading clock transition 1 = Trailing clock transition | | | | | | | | |
| NUMBIT[4:0] = SPI transaction length. Set the number of SPI bits to read/write. 00000 = 24-bit 00001 = 1-bit 00010 = 2-bit : 10111 = 23-bit All others = 23-bit | | | | | | | | |

6.5.4.10.2 SPI_EXE

| Address | 0x0351 | | | | | | | |
|--|--------|-------|-------|-------|-------|-------|---------|--------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | | | | SS_CTRL | SPI_GO |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| RSVD = Reserved | | | | | | | | |
| SS_CTRL = Programs the state of SS. 0 = Output low 1 = Output high | | | | | | | | |
| SPI_GO = Executes the SPI transaction. This bit is self-clearing. 0 = Idle 1 = Execute the SPI | | | | | | | | |

6.5.4.10.3 SPI_TX3, SPI_TX2, and SPI_TX1

| Address | 0x034E to 0x0350 | | | | | | | |
|---|---------------------|-------|-------|-------|-------|-------|-------|-------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | DATA[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| DATA[7:0] = Data to be used to write to SPI target device. The bits are programmed by using <i>SPI_CONF[NUMBIT4:0]</i> and are clocked out of MOSI starting from the LSB <i>SPI_TX1</i> -> LSB <i>SPI_TX2</i> -> LSB <i>SPI_TX3</i> . | | | | | | | | |

6.5.4.10.4 SPI_RX3, SPI_RX2, and SPI_RX1

| Address | 0x0520 to 0x0522 | | | | | | | |
|---|---------------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | DATA[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | R | R | R | R | R | R | R | R |
| DATA[7:0] = Data returned from a read during SPI transaction. Updated, starting with LSB <i>SPI_RX1</i> -> LSB <i>SPI_RX2</i> -> LSB <i>SPI_RX3</i> , with the number of bits set by <i>SPI_CONF[NUMBIT4:0]</i> clocked in from MISO. | | | | | | | | |

6.5.4.11 Diagnostic Control

6.5.4.11.1 DIAG_OTP_CTRL

| Address | 0x0335 | | | | | | | |
|---|--------|-------|-------|---------------|------------------|-------|-------|-----------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | FLIP_FACT_CRC | MARGIN_MODE[2:0] | | | MARGIN_GO |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| FLIP_FACT_CRC = An enable bit to flip the factory CRC value. This is for factory CRC diagnostic. 0 = Normal operation. No modification of the factory CRC 1 = Flip the CRC value. This causes a factory CRC fault, <i>FAULT_OTP[FACT_CRC]</i> . | | | | | | | | |
| MARGIN_MODE[2:0] = Configures OTP Margin read mode: 0b000 = Normal Read 0b001 = Reserved 0b010 = Margin 1 Read 0b011 to 0b111 = Reserved | | | | | | | | |
| MARGIN_GO = Starts OTP Margin test set by the <i>[MARGIN_MOD]</i> bit. This bit self-clears and always reads 0. 0 = Ready 1 = Start the test | | | | | | | | |

6.5.4.11.2 DIAG_COMM_CTRL

| Address | 0x0336 | | | | | | | |
|--|--------|-------|-------|-------|-------|-------|--------------|-------------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | | | | SPI_LOOPBACK | FLIP_TR_CRC |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| SPI_LOOPBACK = Enables SPI loopback function to verify SPI functionality. See the Section 6.3.6.1.7 for more details. 0 = Disable 1 = Enable | | | | | | | | |

FLIP_TR_CRC = Sends a purposely incorrect communication (during transmitting response) CRC by inverting all of the calculated CRC bits.
 0 = Send CRC as calculated
 1 = Send inverted CRC

6.5.4.11.3 DIAG_PWR_CTRL

| Address | 0x0337 | | | | | | | |
|---|--------|-------|-------|-------|-------|-------|-------------|-------------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | | | | BIST_NO_RST | PWR_BIST_GO |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| BIST_NO_RST = Use for further diagnostic if the power supply BIST detects a failure. When this bit is set to 1, and then BIST cycle is run using [PWR_BIST_GO], the device will not clear the FAULT_PWR1 and FAULT_PWR2 register, and does not deassert the NFAULT signal at the end of BIST cycle. 0 = Cycle through BIST on the LDO comparators. The FAULT_PWR* registers are reset to 0 and NFAULT is deasserted at the end of each LDO BIST run. 1 = Cycle through BIST on the LDO comparators. The FAULT_PWR* registers are not reset to 0, and NFAULT remains asserted at the end of each LDO BIST run. | | | | | | | | |
| PWR_BIST_GO = When written to 1, the power supply BIST diagnostic will start. Any change in [BIST_NO_RST] has no effect until this bit is written to 1 again. The bit self-clears. 0 = Ready 1 = Start power supply BIST diagnostic. | | | | | | | | |

6.5.4.11.4 DIAG_CBFET_CTRL1

| Address | 0x0338 | | | | | | | |
|--|-------------------|-------|-------|-------|-------|-------|-------|-------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CBFET16 to CBFET9 | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CBFET16 to CBFET9 = Enables CBFET for CBFET diagnostic. This register is only sampled when [COMP_ADC_SEL2:0] = 0b100. 0 = CBFET off 1 = CBFET on | | | | | | | | |

6.5.4.11.5 DIAG_CBFET_CTRL2

| Address | 0x0339 | | | | | | | |
|---|------------------|-------|-------|-------|-------|-------|-------|-------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CBFET8 to CBFET1 | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CBFET8 to CBFET1 = Enables CBFET for CBFET diagnostic. This register is only sampled when [COMP_ADC_SEL2:0] = 0b100. 0 = CBFET off 1 = CBFET on | | | | | | | | |

6.5.4.11.6 DIAG_COMP_CTRL1

| Address | 0x033A | | | | | | | | |
|--|---------------|-------|-------|-------|-------|-------|-------|-------|--|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| Name | VCCB_THR[4:0] | | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| VCCB_THR[4:0] = Configures the VCELL vs. AUXCELL delta. The VCELL vs. AUXCELL check is considered pass if the measured delta is less than this threshold. This threshold applies to the bus bar comparison from Main to AUX ADC as well. Range from 6 to 99mV in 3mV step | | | | | | | | | |
| BB_THR[2:0] = RSVD = Additional delta value added to the VCCB_THR setting, used during VCELL vs. AUXCELL comparison when comparing a cell connected above a bus bar (with the bus bar connected to a VC channel individually). Range is from 5 mV to 40 mV in 5-mV steps. Reserved | | | | | | | | | |

6.5.4.11.7 DIAG_COMP_CTRL2

| Address | 0x033B | | | | | | | |
|---|--------|---------------|-------|-------|-------------|-------|-------|-------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | GPIO_THR[2:0] | | | OW_THR[3:0] | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| GPIO_THR[2:0] = Configures the GPIO comparison delta threshold between Main and AUX ADC measurements. Range is from 4-mV to 32-mV in 4-mV steps | | | | | | | | |
| OW_THR[3:0] = Configures the OW detection threshold for diagnostic comparison. This threshold applies to the CB OW and VC OW diagnostics. Range is from 500 mV to 5 V in 300-mV steps. | | | | | | | | |

6.5.4.11.8 DIAG_COMP_CTRL3

| Address | 0x033C | | | | | | | |
|--|--------|---------------|-------------|-------|-------------------|-------|-------|-------------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | CBFET_CTRL_GO | OW_SNK[1:0] | | COMP_ADC_SEL[2:0] | | | COMP_ADC_GO |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| CBFET_CTRL_GO = When this GO bit = 1, device turns on the CBFET configured and turns off whichever CBFET is clear in DIAG_CBFET_CTRL1/2 registers. This GO action is executed only if CB is not running or it's in pause, otherwise, CBFETs are controlled by regular CB control. If CBFET are turned on by this GO bit, once CB is started or resume, the CBFET controls returns to the regular CB control (and not by this GO bit action) | | | | | | | | |
| OW_SNK[1:0] = Turns on current sink on VC pins, CB pins, or BBP/N pins. Changes to these bits take effect immediately. Host MCU is responsible to turn on the correct sink current before performing open wire (OW) test and to turn off the sink current after OW test is completed. 00 = All VC, BBP/N, CB pins current sink is off. 01 = Turn on current sink on all VC pins 10 = Turn on current sink on all CB pins 11 = Turn on current sink on BBP/N pins | | | | | | | | |

| |
|--|
| <p>COMP_ADC_SEL[2:0] = Enables the device diagnostic comparison through the ADC measurements. Host enables the corresponding ADCs in continuous mode before enabling this diagnostic. These bits are sampled when [COMP_ADC_GO] = 1.</p> <p>000 = No ADC comparison is performed 001 = Cell voltage measurement check.</p> <p>Device compares the cell channels specified by [AUX_CELL_SEL4:0] against the following criteria: VCELL (from Main ADC) vs. AUXCELL (from AUX ADC) delta is less than [VCCB_THR4:0]. The [DRDY_VCCB] = 1 when this comparison is completed.</p> <p>010 = Open wire (OW) check on VC pins. MCU enables the current sink on all VC pins through the [OW_SNK1:0] before enabling this comparison. Device compares corresponding VC pins specified by ACTIVE_CELL register against the following criteria: VCELL (from Main ADC) is less than DIAG_COMP_CTRL2 [OW_THR3:0]. The [DRDY_VC_OW] = 1 when the comparison is completed.</p> <p>011 = Open wire (OW) check on CB pins MCU enables the current sink on all VC pins through the [OW_SNK1:0] before enabling this comparison. Device compares corresponding CB pins specified by [AUX_CELL_SEL4:0] against the following criteria: AUXCELL (from AUX ADC) is less than DIAG_COMP_CTRL2 [OW_THR3:0]. The [DRDY_CBOW] = 1 when the comparison is completed.</p> <p>100 = CBFET check. MCU preconfigures the following before starting this check:</p> <ul style="list-style-type: none"> • Pause cell balancing if balancing is enabled. • Enable the CBFET configured by DIAG_CBFET_CTRL1/2 registers. • Configure the [EXTD_CBFET] to decide if all CBFET returns to pause state (that is, turn off all CBFET) or remains their status as specified by DIAG_CBFET_CTRL1/2 registers. <p>When this test starts, device will turn on CBFET specified by DIAG_CBFET_CTRL1/2 registers and then compares the channel specified by [AUX_CELL_SEL4:0] with the following criteria: AUXCELL (from AUX ADC) < 1/3 of VCELL (from Main ADC). [DRDY_CBFET] = 1 when the comparison is completed.</p> <p>101 = GPIO measurement check (applies to GPIO configured as ADC and OTUT inputs or ADC only input). Device compares main GPIO measurement vs. AUX GPIO measurements delta is less than [GPIO_THR2:0]. The [DRDY_GPIO] = 1 when the comparison is completed. Other codes: No ADC comparison is performed</p> |
| <p>COMP_ADC_GO = Device starts diagnostic test specified by [COMP_ADC_SEL2:0] setting. When this bit is written to 1, the selected [COMP_ADC_SEL2:0] is sampled. Change of [COMP_ADC_SEL2:0] setting has no effect unless this GO bit is written to 1 again. This bit is cleared to 0 in read. 0 = Ready. Writing 0 has no effect 1 = Star diagnostic selected by [COMP_ADC_SEL2:0]</p> |

6.5.4.11.9 DIAG_COMP_CTRL4

| Address | 0x033D | | | | | | | |
|--|--------|-------|-------|-------|-------|-------|----------------|---------------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | | | | COMP_FAULT_INJ | LPF_FAULT_INJ |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| COMP_FAULT_INJ = Injects fault to the ADC comparison logic. If any ADC comparison diagnostic is run with this bit set, the comparison result is expected to fail. 0 = Disable 1 = Enable | | | | | | | | |
| LPF_FAULT_INJ = Injects fault condition to the diagnostic LPF during LPF diagnostic. The FAULT_COMP_MISC[LPF_FAIL] is expected to be set. 0 = Disable 1 = Enable | | | | | | | | |

6.5.4.11.10 DIAG_PROT_CTRL

| Address | 0x033E | | | | | | | |
|---------|--------|-------|-------|-------|-------|-------|-------|------------------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | | | | | PROT_BIST_NO_RST |

| | | | | | | | | |
|--|---|---|---|---|---|---|---|---|
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| <p>PROT_BIST_NO_RST = Use for further diagnostic if the protector BIST detects a failure. When this bit is set to 1, the device will not clear the <i>FAULT_OV1/2</i>, <i>FAULT_UV1/2</i>, <i>FAULT_OT</i>, and <i>FAULT_UT</i> registers. The NFAULT signal will be latched once it is asserted.</p> <p>Note: Host ensures there is no fault before starting the BIST run with this bit set to 0.</p> <p>0 = During BIST run, when the device asserts a fault to check the protector comparators and MUX and asserts the correct OV, UV, OT, and UT fault bits the NFAULT pin. When this bit is 0, the device clears the fault and deasserts NFAULT before switching to the next channel.</p> <p>1 = During BIST run, the fault created during the test will not be cleared before switching to next cell or GPIO channel. The NFAULT pin is latched once it is asserted.</p> | | | | | | | | |

6.5.4.12 Fault Configuration and Reset

6.5.4.12.1 FAULT_MSK1

| Address | 0x0016 | | | | | | | |
|---|----------|--------|--------|--------|--------|----------|---------|---------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | MSK_PROT | MSK_UT | MSK_OT | MSK_UV | MSK_OV | MSK_COMP | MSK_SYS | MSK_PWR |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| <p>MSK_PROT = Masks the <i>FAULT_PROT*</i> registers to trigger NFAULT.</p> <p>0 = Assert NFAULT if any bit from <i>FAULT_PROT*</i> is set to 1.</p> <p>1 = No NFAULT action regardless of <i>FAULT_PROT*</i> bit status.</p> | | | | | | | | |
| <p>MSK_UT = Masks the <i>FAULT_UT*</i> registers to trigger NFAULT.</p> <p>0 = Assert NFAULT if any bit from <i>FAULT_UT*</i> is set to 1.</p> <p>1 = No NFAULT action regardless of <i>FAULT_UT*</i> bit status.</p> | | | | | | | | |
| <p>MSK_OT = Masks the <i>FAULT_OT*</i> registers to trigger NFAULT.</p> <p>0 = Assert NFAULT if any bit from <i>FAULT_OT*</i> is set to 1.</p> <p>1 = No NFAULT action regardless of <i>FAULT_OT*</i> bit status.</p> | | | | | | | | |
| <p>MSK_UV = Masks the <i>FAULT_UV*</i> registers to trigger NFAULT.</p> <p>0 = Assert NFAULT if any bit from <i>FAULT_UV*</i> is set to 1.</p> <p>1 = No NFAULT action regardless of <i>FAULT_UV*</i> bit status.</p> | | | | | | | | |
| <p>MSK_OV = Masks the <i>FAULT_OV*</i> registers to trigger NFAULT.</p> <p>0 = Assert NFAULT if any bit from <i>FAULT_OV*</i> is set to 1.</p> <p>1 = No NFAULT action regardless of <i>FAULT_OV*</i> bit status.</p> | | | | | | | | |
| <p>MSK_COMP = Masks the <i>FAULT_COMP_*</i> registers to trigger NFAULT.</p> <p>0 = Assert NFAULT if any bit from <i>FAULT_COMP_*</i> is set to 1.</p> <p>1 = No NFAULT action regardless of <i>FAULT_COM_*</i> bit status.</p> | | | | | | | | |
| <p>MSK_SYS = To mask the NFAULT assertion from any <i>FAULT_SYS</i> register bit.</p> <p>0 = Assert NFAULT if any bit from <i>FAULT_SYS</i> is set to 1.</p> <p>1 = No NFAULT action regardless of <i>FAULT_SYS</i> bit status.</p> | | | | | | | | |
| <p>MSK_PWR = To mask the NFAULT assertion from any <i>FAULT_PWR1</i> to <i>FAULT_PWR3</i> register bit.</p> <p>0 = Assert NFAULT if any bit from <i>FAULT_PWR1</i> to <i>FAULT_PWR3</i> is set to 1.</p> <p>1 = No NFAULT action regardless of <i>FAULT_PWR1</i> to <i>FAULT_PWR3</i> bit status.</p> | | | | | | | | |

6.5.4.12.2 FAULT_MSK2

| Address | 0x0017 | | | | | | | |
|--|----------|-------------|--------------|-----------------|-----------------|--------------|-----------|-----------|
| NVM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | SPARE[1] | MSK_OTP_CRC | MSK_OTP_DATA | MSK_COMM3_FCOMM | MSK_COMM3_FTONE | MSK_COMM3_HB | MSK_COMM2 | MSK_COMM1 |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| SPARE[1] = Spare | | | | | | | | |
| <p>MSK_OTP_CRC = Masks the <i>FAULT_OTP</i> register (<i>[CUST_CRC]</i> and <i>[FACT_CRC]</i> only) on NFAULT triggering.</p> <p>0 = Assert NFAULT if any bit described above is set to 1.</p> <p>1 = No NFAULT action regardless of the status of the bits described above.</p> | | | | | | | | |
| <p>MSK_OTP_DATA = Masks the <i>FAULT_OTP</i> register (all bits except <i>[CUST_CRC]</i> and <i>[FACT_CRC]</i>) on NFAULT triggering.</p> <p>0 = Assert NFAULT if any bit described above is set to 1.</p> <p>1 = No NFAULT action regardless of the status of the bits described above.</p> | | | | | | | | |

| |
|---|
| MSK_COMM3_FCOMM Masks <i>FAULT_COMM3[FCOMM_DET]</i> fault on NFAULT triggering. = 0 = Assert NFAULT if <i>FAULT_COMM3[FCOMM_DET]</i> is set to 1. 1 = No NFAULT action regardless of <i>FAULT_COMM3[FCOMM_DET]</i> status. |
| MSK_COMM3_FTONE Masks <i>FAULT_COMM3[FTONE_DET]</i> fault on NFAULT triggering. = 0 = Assert NFAULT if <i>FAULT_COMM3[FTONE_DET]</i> is set to 1. 1 = No NFAULT action regardless of <i>FAULT_COMM3[FTONE_DET]</i> status. |
| MSK_COMM3_HB = Masks <i>FAULT_COMM3[HB_FAST]</i> or <i>[HB_FAIL]</i> faults on NFAULT triggering. 0 = Assert NFAULT if <i>FAULT_COMM3[HB_FAST]</i> or <i>[HB_FAIL]</i> is set to 1. 1 = No NFAULT action regardless of <i>FAULT_COMM3[HB_FAST]</i> or <i>[HB_FAIL]</i> status. |
| MSK_COMM2 = Masks <i>FAULT_COMM2</i> register on NFAULT triggering. 0 = Assert NFAULT if any bit from <i>FAULT_COMM2</i> register is set to 1. 1 = No NFAULT action regardless of <i>FAULT_COMM2</i> register bit status. |
| MSK_COMM1 = Masks <i>FAULT_COMM1</i> register on NFAULT triggering. 0 = Assert NFAULT if any bit from <i>FAULT_COMM1</i> register is set to 1. 1 = No NFAULT action regardless of <i>FAULT_COMM1</i> register bit status. |

6.5.4.12.3 FAULT_RST1

| Address | 0x0331 | | | | | | | |
|--|----------|--------|--------|--------|--------|----------|---------|---------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RST_PROT | RST_UT | RST_OT | RST_UV | RST_OV | RST_COMP | RST_SYS | RST_PWR |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RST_PROT = Resets the <i>FAULT_PROT1</i> and <i>FAULT_PROT2</i> registers to 0x00. 0 = No reset 1 = Reset registers to 0x00 | | | | | | | | |
| RST_UT = Resets all <i>FAULT_UT</i> registers to 0x00. 0 = No reset 1 = Reset registers to 0x00 | | | | | | | | |
| RST_OT = Resets all <i>FAULT_OT</i> registers to 0x00. 0 = No reset 1 = Reset registers to 0x00 | | | | | | | | |
| RST_UV = Resets all <i>FAULT_UV*</i> registers to 0x00. 0 = No reset 1 = Reset registers to 0x00 | | | | | | | | |
| RST_OV = Resets all <i>FAULT_OV*</i> registers to 0x00. 0 = No reset 1 = Reset registers to 0x00 | | | | | | | | |
| RST_COMP = Resets all <i>FAULT_COMP_*</i> registers to 0x00. 0 = No reset 1 = Reset registers to 0x00 | | | | | | | | |
| RST_SYS = To reset the <i>FAULT_SYS</i> register to 0x00. This bit self-clears to 0 after writing to 1. 0 = Do not reset 1 = Reset to 0x00 | | | | | | | | |
| RST_PWR = To reset the <i>FAULT_PWR1</i> to <i>FAULT_PWR3</i> registers to 0x00. This bit self-clears to 0 after writing to 1. 0 = Do not reset 1 = Reset to 0x00 | | | | | | | | |

6.5.4.12.4 FAULT_RST2

| Address | 0x0332 | | | | | | | |
|--|--------|-------------|--------------|-----------------|-----------------|--------------|-----------|-----------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | RST_OTP_CRC | RST_OTP_DATA | RST_COMM3_FCOMM | RST_COMM3_FTONE | RST_COMM3_HB | RST_COMM2 | RST_COMM1 |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| RST_OTP_CRC = Resets the <i>FAULT_OTP</i> register (<i>[CUST_CRC]</i> and <i>[FACT_CRC]</i> only). 0 = No reset 1 = Reset the register to 0x00 | | | | | | | | |

| |
|--|
| RST_OTP_DATA = Resets the <i>FAULT_OTP</i> register (<i>[SEC_DETECT]</i> and <i>[DED_DETECT]</i> only). 0 = No reset 1 = Reset the register to 0x00 |
| RST_COMM3_FCOMM Resets <i>FAULT_COMM3[FCOMM_DET]</i> . = 0 = No reset 1 = Reset the related bit to 0 |
| RST_COMM3_FTONE Resets <i>FAULT_COMM3[FTONE_DET]</i> . = 0 = No reset 1 = Reset the related bit to 0 |
| RST_COMM3_HB = Resets <i>FAULT_COMM3[HB_FAST]</i> and <i>[HB_FAIL]</i> bits. 0 = No reset 1 = Reset the related bits to 0 |
| RST_COMM2 = Resets <i>FAULT_COMM2</i> , <i>DEBUG_COML*</i> , and <i>DEBUG_COMM_COMH*</i> registers. 0 = No reset 1 = Reset registers to 0x00 |
| RST_COMM1 = Resets <i>FAULT_COMM1</i> and <i>DEBUG_COMM_UART*</i> registers. 0 = No reset 1 = Reset registers to 0x00 |

6.5.4.13 Fault Status

6.5.4.13.1 FAULT_SUMMARY

This register is the soft version of the NFAULT.

| Address | 0x052D | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--|------------|----------------|-----------|------------|------------|------------|-----------|-----------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | FAULT_PROT | FAULT_COMP_ADC | FAULT_OTP | FAULT_COMM | FAULT_OTUT | FAULT_OVUV | FAULT_SYS | FAULT_PWR |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| FAULT_PROT = This bit is set if <i>[MSK_PROT]</i> = 0 and any of the <i>FAULT_PROT1</i> or <i>FAULT_PROT2</i> register bits is set. 0 = No protector (OVUV, OTUT comparators) fault. 1 = Protector fault is detected | | | | | | | | |
| FAULT_COMP_ADC = This bit is set if <i>[MSK_COMP]</i> = 0 and any of the following registers is set: <ul style="list-style-type: none"> • <i>FAULT_COMP_VCCB1/2</i> • <i>FAULT_COMP_VCOW1/2</i> • <i>FAULT_COMP_CBOW1/2</i> • <i>FAULT_COMP_CBFET1/2</i> • <i>FAULT_COMP_GPIO</i> • <i>FAULT_COMP_MISC</i> 0 = No ADC comparison fault (that is, none of the <i>FAULT_COMP_*</i> registers are set). 1 = ADC comparison fault is detected. | | | | | | | | |
| FAULT_OTP = This bit is set if <i>[MSK_OTP]</i> = 0 and any of the <i>FAULT_OTP</i> register bits is set. 0 = No OTP-related fault detected or OTP faults are masked. 1 = OTP-related fault is detected. | | | | | | | | |
| FAULT_COMM = This bit is set if any of the following is true: <ul style="list-style-type: none"> • <i>[MSK_COMM1]</i> = 0 and any of the <i>FAULT_COMM1</i> register bits is set. • <i>[MSK_COMM2]</i> = 0 and any of the <i>FAULT_COMM2</i> register bits is set. • <i>[MSK_COMM3_HB]</i> = 0 and the <i>FAULT_COMM3[HB_FAST]</i> bit or <i>[HB_FAIL]</i> bit is set. • <i>[MSK_COMM3_FTONE]</i> = 0 and the <i>FAULT_COMM3[FTONE_DET]</i> is set. • <i>[MSK_COMM3_FCOMM]</i> = 0 and if <i>FAULT_COMM3[FCOMM_DET]</i> is set. 0 = No UART, VIF, or FTONE fault is detected, or UART, VIF, and FTONE faults are masked. 1 = UART, VIF, or UT fault is detected. | | | | | | | | |

FAULT_OTUT = This bit is set if any of the following is true:

- $[MSK_OT] = 0$ and any of the *FAULT_OT1* or *FAULT_OT2* bits is set.
- $[MSK_UT] = 0$ and any of the *FAULT_UT1* or *FAULT_UT2* bits is set.

0 = No OT or UT fault is detected, or OT and UT faults are masked.
1 = OT or UT fault is detected

FAULT_OVUV = This bit is set if any of the following is true:

- $[MSK_OV] = 0$ and any of the *FAULT_OV1* or *FAULT_OV2* bits is set.
- $[MSK_UV] = 0$ and any of the *FAULT_UV1* or *FAULT_UV2* bits is set.

0 = No OV or UV fault is detected, or OV and UV faults are masked.
1 = OV or UV fault is detected.

FAULT_SYS = This bit is set if $[MSK_SYS] = 0$ and any of the *FAULT_SYS* register bits is set.

0 = No system related fault detected or system faults are masked.
1 = System related fault is detected.

FAULT_PWR = This bit is set if $[MSK_PWR] = 0$ and any of the *FAULT_PWR1* to *FAULT_PWR3* register bits is set.

0 = No power rail related fault is detected or power rail faults are masked.
1 = Power rail related fault is detected.

6.5.4.13.2 FAULT_COMM1

| Address | 0x0530 | | | | | | | |
|--|--------|-------|-------|---------|---------|---------|-------------|----------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | UART_TR | UART_RR | UART_RC | COMMCLR_DET | STOP_DET |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| UART_TR = Indicates a UART FAULT is detected when transmitting a response frame. Further details of the fault information are available in the <i>DEBUG_UART_RR_TR</i> register. 0 = No fault 1 = Fault | | | | | | | | |
| UART_RR = Indicates a UART FAULT is detected when receiving a response frame. Further details of the fault information are available in the <i>DEBUG_UART_RR_TR</i> register. 0 = No fault 1 = Fault | | | | | | | | |
| UART_RC = Indicates a UART FAULT is detected during receiving a command frame. Further details of the fault information are available in the <i>DEBUG_UART_RC</i> register. 0 = No fault 1 = Fault | | | | | | | | |
| COMMCLR_DET = A UART communication clear signal is detected. A detection of SLEEPtoACTIVE ping in ACTIVE or SLEEP mode or detection of WAKE pin in ACTIVE mode will also set this bit. 0 = No UART Clear 1 = UART Clear detected | | | | | | | | |
| STOP_DET = Indicates an unexpected STOP condition is received. A detection of SLEEPtoACTIVE signal in ACTIVE mode will also set this bit. 0 = No fault 1 = Fault | | | | | | | | |

6.5.4.13.3 FAULT_COMM2

| Address | 0x0531 | | | | | | | |
|--|---------|---------|---------|----------|---------|---------|---------|----------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | COML_TR | COML_RR | COML_RC | COML_BIT | COMH_TR | COMH_RR | COMH_RC | COMH_BIT |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| COML_TR = Indicates a COML byte level fault is detected when transmitting a response frame. Further details of the fault information are available in the <i>DEBUG_COML_RR_TR</i> register. 0 = No fault 1 = Fault | | | | | | | | |

| |
|--|
| <p>COML_RR = Indicates a COML byte level fault is detected when receiving a response frame. Further details of the fault information are available in the <i>DEBUG_COML_RR_TR</i> register. 0 = No fault 1 = Fault</p> |
| <p>COML_RC = Indicates a COML byte level fault is detected when receiving a command frame. Further details of the fault information are available in the <i>DEBUG_COML_RR_RC</i> register. 0 = No fault 1 = Fault</p> |
| <p>COML_BIT = Indicates a COML bit level fault is detected which would cause at least one byte level fault. Further details of the fault information are available in the <i>DEBUG_COML_BIT</i> register. 0 = No fault 1 = Fault</p> |
| <p>COMH_TR = Indicates a COMH byte level fault is detected when transmitting a response frame. Further details of the fault information are available in the <i>DEBUG_COMH_RR_TR</i> register. 0 = No fault 1 = Fault</p> |
| <p>COMH_RR = Indicates a COMH byte level fault is detected when receiving a response frame. Further details of the fault information are available in the <i>DEBUG_COMH_RR_TR</i> register. 0 = No fault 1 = Fault</p> |
| <p>COMH_RC = Indicates a COMH byte level fault is detected when receiving a command frame. Further details of the fault information are available in the <i>DEBUG_COMH_RR_RC</i> register. 0 = No fault 1 = Fault</p> |
| <p>COMH_BIT = Indicates a COMH bit level fault is detected which would cause at least one byte level fault. Further details of the fault information are available in the <i>DEBUG_COMH_BIT</i> register. 0 = No fault 1 = Fault</p> |

6.5.4.13.4 FAULT_COMM3

| Address | 0x0532 | | | | | | | |
|--|--------|-------|-------|-------|-----------|-----------|---------|---------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | | FCOMM_DET | FTONE_DET | HB_FAIL | HB_FAST |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| FCOMM_DET = Received communication transaction with the Fault Status bits set by any of the upper stack device(s). 0 = Fault Status are clear, indicating no fault is detected from any of the upper stack device(s). 1 = Fault Status are set from the receiving communication transaction. | | | | | | | | |
| FTONE_DET = Indicates a FAULT TONE is received. Detection is monitoring the COML side if <i>[DIR_SEL] = 0</i> and vice versa. 0 = No FAULT TONE detected 1 = FAULT TONE detected | | | | | | | | |
| HB_FAIL = Indicates HEARTBEAT is not received within an expected time. Detection is monitoring the COML side if <i>[DIR_SEL] = 0</i> and vice versa. 0 = No fault 1 = Fault | | | | | | | | |
| HB_FAST = Indicates HEARTBEAT is received too frequently. Detection is monitoring the COML side if <i>[DIR_SEL] = 0</i> and vice versa. This bit may also be set when <i>[FTONE_DET] = 1</i> depends on how soon the FAULT TONE is detected from the previous HEARTBEAT 0 = No fault 1 = Fault | | | | | | | | |

6.5.4.13.5 FAULT_OTP

| Address | 0x0535 | | | | | | | |
|-----------|--------|---------|---------|----------|----------|-----------|-----------|----------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | DED_DET | SEC_DET | CUST_CRC | FACT_CRC | CUSTLDERR | FACTLDERR | GBLOVERR |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| |
|--|
| RSVD = Reserved |
| DED_DET = Indicates a DED error has occurred during the OTP load. (Unknown during encoding) 0 = No fault 1 = Fault |
| SEC_DET = Indicates a SEC error has occurred during the OTP load. (Unknown during encoding) 0 = No fault 1 = Fault |
| CUST_CRC = Indicates a CRC error has occurred in the customer register space. 0 = No fault 1 = Fault |
| FACT_CRC = Indicates a CRC error has occurred in the factory register space. 0 = No fault 1 = Fault |
| CUSTLDERR = Indicates errors during the customer space OTP load process. Read <i>OTP_CUST1_STAT</i> and <i>OTP_CUST2_STAT</i> registers for the specific error condition. This error bit is set if one of the following is true: <ul style="list-style-type: none"> No Customer OTP page is programmed. The highest Customer OTP page has a <i>[FMterr]</i>. The highest Customer OTP page has <i>[TRY] = 1</i> and is not <i>[PROGOK]</i>. LOADERR happened on the selected Customer OTP page. <p>Information received from the device with this error must not be considered reliable. Writing <i>[RST_OTP_DATA] = 1</i> does not reset this bit. To recheck this error, a device reset or HW_RESET is needed.</p> 0 = No fault 1 = Fault |
| FACTLDERR = Indicates errors during the factory space OTP load process. This error bit is set if one of the following is true: <ul style="list-style-type: none"> No factory OTP page is programmed. The highest factory OTP page has a <i>[FMterr]</i>. The highest factory OTP page has <i>[TRY] = 1</i> and is not <i>[PROGOK]</i>. LOADERR happened on the selected factory OTP page. <p>Information received from the device with this error must not be considered reliable. Writing <i>[RST_OTP_DATA] = 1</i> does not reset this bit. To recheck this error, a device reset or HW_RESET is needed.</p> 0 = No fault 1 = Fault |
| GBLOVERR = Indicates that an overvoltage error is detected on one of the OTP pages. Read <i>OTP_CUST1_STAT</i> and <i>OTP_CUST2_STAT</i> registers to determine the specific page(s). Information received from the device with this error must not be considered reliable. Writing <i>[RST_OTP_DATA] = 1</i> does not reset this bit. To clear this bit, a device reset or HW_RESET is needed. Repeat the programming procedure on a different page (if available) will force the device to re-evaluate the condition. 0 = No fault 1 = Fault |

6.5.4.13.6 FAULT_SYS

| Address | 0x0536 | | | | | | | |
|---|--------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | LFO | RSVD | GPIO | DRST | CTL | CTS | TSHUT | TWARN |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| LFO = Indicated LFO frequency is outside an expected range 0 = No fault detected 1 = Fault detected | | | | | | | | |
| RSVD = Reserved | | | | | | | | |
| GPIO = Indicates GPIO8 detects a FAULT input when <i>GPIO_CONF1[FAULT_IN_EN] = 1</i> . 0 = No fault detected 1 = FAULT input detected | | | | | | | | |

| |
|--|
| <p>DRST = Indicates a digital reset has occurred. 0 = No digital reset 1 = Digital reset has occurred</p> |
| <p>CTL = Indicates a long communication timeout occurred. Device action is configured by <i>[CTL_ACT]</i>. This bit is not observable if the action is set to device shutdown. 0 = No fault 1 = Long communication timeout occurs. Observable if long timeout action is set to SLEEP.</p> |
| <p>CTS = Indicates a short communication timeout occurred. No action from the device. This can be served as an alert to system before reaching long communication timeout. 0 = No fault 1 = Short communication timeout occurs</p> |
| <p>TSHUT = Indicates the previous shutdown was a thermal shutdown, in which the die temperature (die temp 2) is higher than the thermal shutdown threshold. 0 = Die temperature is less than thermal shutdown threshold 1 = The previous shutdown was a thermal shutdown</p> |
| <p>TWARN = Indicates the die temperature (die temp 2) is higher than the <i>TWARN_THR[1:0]</i> setting. No action is taken by the device at the moment yet. This serves as a warning signal that the die temperature is approaching thermal shutdown. 0 = Die temperature is less than <i>TWARN_THR[1:0]</i> 1 = Die temperature is greater than <i>TWARN_THR[1:0]</i></p> |

6.5.4.13.7 FAULT_PROT1

| Address | 0x053A | | | | | | | |
|---|--------|-------|-------|-------|-------|-------|--------------|--------------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | | | | TPARITY_FAIL | VPARITY_FAIL |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| <p>TPARITY_FAIL = Indicates a parity fault is detected on any of the following OTUT related configurations:</p> <ul style="list-style-type: none"> • OT or UT threshold setting • <i>[OTUT_MODE1:0]</i> setting • <i>GPIO_CONF1...4</i> settings <p>0 = No fault 1 = Fault</p> | | | | | | | | |
| <p>VPARITY_FAIL = Indicates a parity fault is detected on any of the following OVUV related configurations:</p> <ul style="list-style-type: none"> • OV or UV threshold setting • <i>[OVUV_MODE1:0]</i> setting • <i>[NUM_CELL3:0]</i> setting <p>0 = No fault 1 = Fault</p> | | | | | | | | |

6.5.4.13.8 FAULT_PROT2

| Address | 0x053B | | | | | | | |
|---|--------|------------|------------|------------|-------------|-------------|-------------|-------------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | BIST_ABORT | TPATH_FAIL | VPATH_FAIL | UTCOMP_FAIL | OTCOMP_FAIL | OVCOMP_FAIL | UVCOMP_FAIL |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| <p>BIST_ABORT = Indicates either OVUV or OTUT BIST run is aborted. 0 = BIST runs to completion 1 = BIST abort</p> | | | | | | | | |

| |
|--|
| TPATH_FAIL = Indicates a fault is detected along the OTUT signal path during BIST test. 0 = No fault 1 = Fault |
| VPATH_FAIL = Indicates a fault is detected along the OVUV signal path during BIST test. 0 = No fault 1 = Fault |
| UTCOMP_FAIL = Indicates the UT comparator fails during BIST test. 0 = No fault 1 = Fault |
| OTCOMP_FAIL = Indicates the OT comparator fails during BIST test. 0 = No fault 1 = Fault |
| OVCOMP_FAIL = Indicates the OV comparator fails during BIST test. 0 = No fault 1 = Fault |
| UVCOMP_FAIL = Indicates the UV comparator fails during BIST test. 0 = No fault 1 = Fault |

6.5.4.13.9 FAULT_OV1

| Address | 0x053C | | | | | | | |
|--|----------|----------|----------|----------|----------|----------|----------|---------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | OV16_DET | OV15_DET | OV14_DET | OV13_DET | OV12_DET | OV11_DET | OV10_DET | OV9_DET |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| OV9_DET to OV16_DET = OV fault status for Cell9 to Cell16, results are from the OV comparator detection. | | | | | | | | |

6.5.4.13.10 FAULT_OV2

| Address | 0x053D | | | | | | | |
|--|---------|---------|---------|---------|---------|---------|---------|---------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | OV8_DET | OV7_DET | OV6_DET | OV5_DET | OV4_DET | OV3_DET | OV2_DET | OV1_DET |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| OV1_DET to OV8_DET = OV fault status for Cell1 to Cell8, results are from the OV comparator detection. | | | | | | | | |

6.5.4.13.11 FAULT_UV1

| Address | 0x053E | | | | | | | |
|--|----------|----------|----------|----------|----------|----------|----------|---------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | UV16_DET | UV15_DET | UV14_DET | UV13_DET | UV12_DET | UV11_DET | UV10_DET | UV9_DET |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| UV9_DET to UV16_DET = UV fault status for Cell9 to Cell16, results are from the UV comparator detection. | | | | | | | | |

6.5.4.13.12 FAULT_UV2

| Address | 0x053F | | | | | | | |
|--|---------|---------|---------|---------|---------|---------|---------|---------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | UV8_DET | UV7_DET | UV6_DET | UV5_DET | UV4_DET | UV3_DET | UV2_DET | UV1_DET |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| UV1_DET to UV8_DET = UV fault status for Cell1 to Cell8, results are from the UV comparator detection. | | | | | | | | |

6.5.4.13.13 FAULT_OT

| Address | 0x0540 | | | | | | | |
|-----------|---------|---------|---------|---------|---------|---------|---------|---------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | OT8_DET | OT7_DET | OT6_DET | OT5_DET | OT4_DET | OT3_DET | OT2_DET | OT1_DET |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

OT1_DET to OT8_DET = OT fault status for GPIO1 to GPIO8, results are from the OT comparator detection.

6.5.4.13.14 FAULT_UT

| Address | 0x0541 | | | | | | | |
|-----------|---------|---------|---------|---------|---------|---------|---------|---------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | UT8_DET | UT7_DET | UT6_DET | UT5_DET | UT4_DET | UT3_DET | UT2_DET | UT1_DET |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

UT1_DET to UT8_DET = UT fault status for GPIO1 to GPIO8, results are from the UT comparator detection.

6.5.4.13.15 FAULT_COMP_GPIO

| Address | 0x0543 | | | | | | | |
|-----------|------------|------------|------------|------------|------------|------------|------------|------------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | GPIO8_FAIL | GPIO7_FAIL | GPIO6_FAIL | GPIO5_FAIL | GPIO4_FAIL | GPIO3_FAIL | GPIO2_FAIL | GPIO1_FAIL |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

GPIO1_FAIL to GPIO8_FAIL = Indicates ADC vs. AUX ADC GPIO measurement diagnostic results for GPIO1 to GPIO8.
GPIO8_FAIL = 0 = Diagnostic pass
1 = Diagnostic fail. GPIO from Main ADC vs. AUX ADC measurement is greater than [GPIO_THR2:0]

6.5.4.13.16 FAULT_COMP_VCCB1

| Address | 0x0545 | | | | | | | |
|-----------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|------------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CELL16_FAIL | CELL15_FAIL | CELL14_FAIL | CELL13_FAIL | CELL12_FAIL | CELL11_FAIL | CELL10_FAIL | CELL9_FAIL |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

CELL9_FAIL to CELL16_FAIL = Indicates voltage diagnostic results for cell9 to cell16.
CELL16_FAIL = 0 = Diagnostic pass
1 = Diagnostic fail. VCELL vs. AUXCELL measurement is greater than [VCCB_THR4:0]

6.5.4.13.17 FAULT_COMP_VCCB2

| Address | 0x0546 | | | | | | | |
|-----------|------------|------------|------------|------------|------------|------------|------------|------------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CELL8_FAIL | CELL7_FAIL | CELL6_FAIL | CELL5_FAIL | CELL4_FAIL | CELL3_FAIL | CELL2_FAIL | CELL1_FAIL |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

CELL1_FAIL to CELL8_FAIL = Indicates voltage diagnostic results for cell1 to cell8.
CELL8_FAIL = 0 = Diagnostic pass
1 = Diagnostic fail. VCELL vs. AUXCELL measurement is greater than [VCCB_THR4:0]

6.5.4.13.18 FAULT_COMP_VCOW1

| Address | 0x0548 | | | | | | | |
|---------|--------|--|--|--|--|--|--|--|
|---------|--------|--|--|--|--|--|--|--|

BQ79616

SLUSF21A – JUNE 2023 – REVISED JUNE 2026

| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-----------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|------------|
| Name | VCOW16_FAIL | VCOW15_FAIL | VCOW14_FAIL | VCOW13_FAIL | VCOW12_FAIL | VCOW11_FAIL | VCOW10_FAIL | VCOW9_FAIL |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

VCOW9_FAIL to VCOW16_FAIL = 0 = Diagnostic pass
1 = Diagnostic fail. VCELL measurement is less than [OW_THR3:0]

6.5.4.13.19 FAULT_COMP_VCOW2

| Address | 0x0549 | | | | | | | |
|-----------|------------|------------|------------|------------|------------|------------|------------|------------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | VCOW8_FAIL | VCOW7_FAIL | VCOW6_FAIL | VCOW5_FAIL | VCOW4_FAIL | VCOW3_FAIL | VCOW2_FAIL | VCOW1_FAIL |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

VCOW1_FAIL to VCOW8_FAIL = 0 = Diagnostic pass
1 = Diagnostic fail. VCELL measurement is less than [OW_THR3:0]

6.5.4.13.20 FAULT_COMP_CBOW1

| Address | 0x054B | | | | | | | |
|-----------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|------------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CBOW16_FAIL | CBOW15_FAIL | CBOW14_FAIL | CBOW13_FAIL | CBOW12_FAIL | CBOW11_FAIL | CBOW10_FAIL | CBOW9_FAIL |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

CBOW9_FAIL to CBOW16_FAIL = 0 = Pass
1 = Fail

6.5.4.13.21 FAULT_COMP_CBOW2

| Address | 0x054C | | | | | | | |
|-----------|------------|------------|------------|------------|------------|------------|------------|------------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CBOW8_FAIL | CBOW7_FAIL | CBOW6_FAIL | CBOW5_FAIL | CBOW4_FAIL | CBOW3_FAIL | CBOW2_FAIL | CBOW1_FAIL |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

CBOW1_FAIL to CBOW8_FAIL = 0 = Pass
1 = Fail

6.5.4.13.22 FAULT_COMP_CBFET1

| Address | 0x054E | | | | | | | |
|-----------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|-------------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CBFET16_FAIL | CBFET15_FAIL | CBFET14_FAIL | CBFET13_FAIL | CBFET12_FAIL | CBFET11_FAIL | CBFET10_FAIL | CBFET9_FAIL |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

CBFET9_FAIL to CBFET16_FAIL = 0 = Pass
1 = Fail

6.5.4.13.23 FAULT_COMP_CBFET2

| Address | 0x054F | | | | | | | |
|---|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CBFET8_FAIL | CBFET7_FAIL | CBFET6_FAIL | CBFET5_FAIL | CBFET4_FAIL | CBFET3_FAIL | CBFET2_FAIL | CBFET1_FAIL |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CBFET1_FAIL to CBFET8_FAIL = Results of the CB FET diagnostic for CB FET1 to CB FET8. CBFET8_FAIL = 0 = Pass 1 = Fail | | | | | | | | |

6.5.4.13.24 FAULT_COMP_MISC

| Address | 0x0550 | | | | | | | |
|--|--------|-------|-------|-------|-------|-------|----------------|----------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | | | | COMP_ADC_ABORT | LPF_FAIL |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| COMP_ADC_ABORT = Indicates the most recent ADC comparison diagnostic is aborted due to improper setting. Valid only if one of the ADC comparison diagnostics has started. 0 = ADC comparison diagnostic run to completion 1 = ADC comparison diagnostic is aborted | | | | | | | | |
| LPF_FAIL = Indicates LPF diagnostic result. 0 = Pass 1 = Fail | | | | | | | | |

6.5.4.13.25 FAULT_PWR1

| Address | 0x0552 | | | | | | | |
|---|-----------|-----------|------------|---------|---------|---------|----------|---------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CVSS_OPEN | DVSS_OPEN | REFHM_OPEN | CVDD_UV | CVDD_OV | DVDD_OV | AVDD_OSC | AVDD_OV |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CVSS_OPEN = Indicates an open condition on CVSS pin. 0 = No fault 1 = Fault | | | | | | | | |
| DVSS_OPEN = Indicates an open condition on DVSS pin. 0 = No fault 1 = Fault | | | | | | | | |
| REFHM_OPEN = Indicates an open condition on REFHM pin. 0 = No fault 1 = Fault | | | | | | | | |
| CVDD_UV = Indicates an undervoltage fault on the CVDD LDO. 0 = No fault 1 = Fault | | | | | | | | |
| CVDD_OV = Indicates an overvoltage fault on the CVDD LDO. 0 = No fault 1 = Fault | | | | | | | | |
| DVDD_OV = Indicates an overvoltage fault on the DVDD LDO. 0 = No fault 1 = Fault | | | | | | | | |

AVDD_OSC = Indicates AVDD is oscillating outside of acceptable limits.

0 = No fault

1 = Fault

This fault could trigger when transitioning from SLEEP to ACTIVE mode. So, if this fault is set, please ignore it and reset the fault.

AVDD_OV = Indicates an overvoltage fault on the AVDD LDO.

0 = No fault

1 = Fault

6.5.4.13.26 FAULT_PWR2

| Address | 0x0553 | | | | | | | |
|---|--------|--------------|-------|----------|----------|-----------|----------|----------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | PWRBIST_FAIL | RSVD | REFH_OSC | NEG5V_UV | TSREF_OSC | TSREF_UV | TSREF_OV |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| PWRBIST_FAIL = Indicates a fail on the power supply BIST run. 0 = No fault 1 = Fault | | | | | | | | |
| REFH_OSC = Indicates REGH reference is oscillating outside of an acceptable limit. 0 = No fault 1 = Fault | | | | | | | | |
| NEG5V_UV = Indicates an undervoltage fault on the NEG5V charge pump. 0 = No fault 1 = Fault | | | | | | | | |
| TSREF_OSC = Indicates TSREF is oscillating outside of an acceptable limit. 0 = No fault 1 = Fault | | | | | | | | |
| TSREF_UV = Indicates an undervoltage fault on the TSREF LDO. 0 = No fault 1 = Fault | | | | | | | | |
| TSREF_OV = Indicates an overvoltage fault on the TSREF LDO. 0 = No fault 1 = Fault | | | | | | | | |

6.5.4.13.27 FAULT_PWR3

| Address | 0x0554 | | | | | | | |
|---|--------|-------|-------|-------|-------|-------|-------|-------------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | | | RSVD | RSVD | AVDDUV_DRST |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| AVDDUV_DRST = Indicates a digital reset occurred due to AVDD UV detected. This also applies when device wakes up after a SHUTDOWN or HW Reset event. 0 = No reset 1 = Digital reset occurred due to AVDD UV | | | | | | | | |

6.5.4.14 Debug Control and Status

6.5.4.14.1 DEBUG_CTRL_UNLOCK

| Address | 0x0700 | | | | | | | |
|---------|-----------|-------|-------|-------|-------|-------|-------|-------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CODE[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

CODE[7:0] = Write the unlock code (0xA5) to this register to activate the setting in the *DEBUG_COMM_CTRL** register. Any other value than the unlock code will deactivate any effect in the *DEBUG_COMM_CTRL** setting and return to the normal settings of the device.

6.5.4.14.2 *DEBUG_COMM_CTRL1*

| Address | 0x0701 | | | | | | | |
|---|--------|-------|-------|-----------|----------------|------------|--------------|----------------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | UART_BAUD | UART_MIRROR_EN | UART_TX_EN | USER_UART_EN | USER_DAISSY_EN |
| Reset | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| UART_BAUD = This bit changes the UART baud rate to 250kb/s. Useful on VIF debug. When system sets all daisy chain devices to the 250kb/s baud rate, it slows down the response byte through the VIF to increase the robustness of the VIF for debug purposes. 0 = Default 1Mb/s 1 = 250kb/s | | | | | | | | |
| UART_MIRROR_EN = This bit enables the stack VIF communication to mirror to UART. To use this debug function, the stack device's UART TX has to be enabled first by setting [UARTTX_EN] = 1. 0 = Disable 1 = Enable | | | | | | | | |
| UART_TX_EN = Stack device, by default, has the UART TX disabled. This bit enables the UART TX to allow read/write via UART on the stack device. 0 = Disable 1 = Enable | | | | | | | | |
| USER_UART_EN = This bit enables the debug UART control bits, [UART_TX_EN] and [UART_MIRROR_EN]. 0 = The setting of the bits mentioned above has no effect. 1 = The device configures the UART per [UART_TX_EN] and [UART_MIRROR_EN] settings | | | | | | | | |
| USER_DAISSY_EN = This bit enables the debug COML and COMH control bits in the <i>DEBUG_COMM_CTRL2</i> register 0 = The setting of <i>DEBUG_COMM_CTRL2</i> register has no effect. 1 = The device configures the COML and COMH per <i>DEBUG_COMM_CTRL2</i> register setting. | | | | | | | | |

6.5.4.14.3 *DEBUG_COMM_CTRL2*

| Address | 0x0702 | | | | | | | |
|---|--------|-------|-------|-------|------------|------------|------------|------------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | | COML_TX_EN | COML_RX_EN | COMH_TX_EN | COMH_RX_EN |
| Reset | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| RSVD = Reserved | | | | | | | | |
| COML_TX_EN = Enables COML transmitter. 0 = Disable 1 = Enable | | | | | | | | |
| COML_RX_EN = Enables COML receiver. 0 = Disable 1 = Enable | | | | | | | | |
| COMH_TX_EN = Enables COMH transmitter. 0 = Disable 1 = Enable | | | | | | | | |
| COMH_RX_EN = Enables COMH receiver. 0 = Disable 1 = Enable | | | | | | | | |

6.5.4.14.4 *DEBUG_COMM_STAT*

| Address | 0x0780 | | | | | | | |
|-----------|--------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| | | | | | | | | |

| Name | RSVD | | HW_UART_DRV | HW_DAISSY_DRV | COML_TX_ON | COML_RX_ON | COMH_TX_ON | COMH_RX_ON |
|---|------|---|-------------|---------------|------------|------------|------------|------------|
| Reset (Base) | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| Reset (Stack) | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| RSVD = Reserved | | | | | | | | |
| HW_UART_DRV = Indicates the UART TX is controlled by the device itself or by MCU control. Applicable to the stack device in which the UART TX is disabled by default once a device is configured as 'STACK'. 0 = The <i>DEBUG_COMM_CTRL1[USER_UART_EN]</i> = 1. UART TX is under manual control through the <i>DEBUG_COMM_CTRL2</i> register. 1 = UART TX is controlled by the device | | | | | | | | |
| HW_DAISSY_DRV = Indicates the COML and COMH are controlled by the device itself or by MCU control. 0 = The <i>DEBUG_COMM_CTRL1[USER_DAISSY_EN]</i> = 1. COML and COMH are under manual control through the <i>DEBUG_COMM_CTRL2</i> register. 1 = COML and COMH are controlled by the device | | | | | | | | |
| COML_TX_ON = Shows the current COML transmitter status. 0 = off 1 = on | | | | | | | | |
| COML_RX_ON = Shows the current COML receiver status. 0 = off 1 = on | | | | | | | | |
| COMH_TX_ON = Shows the current COMH transmitter status. 0 = off 1 = on | | | | | | | | |
| COMH_RX_ON = Shows the current COMH receiver status. 0 = off 1 = on | | | | | | | | |

6.5.4.14.5 DEBUG_UART_RC

| Address | 0x0781 | | | | | | | |
|---|--------|-------|---------|----------|--------|-------------|----------|--------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | RC_IERR | RC_TXDIS | RC_SOF | RC_BYTE_ERR | RC_UNEXP | RC_CRC |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| RC_IERR = Detects initialization byte error in the received command frame. This may be due to the frame initialization byte has a stop error, incorrect frame type is set, or reserved command type bit is set. All bytes that follow are ignored until a communication CLEAR is received. When a communication frame is ignored, the device will not attempt to detect any communication error in the ignored frame nor counting it as valid/discard in the frame counters. 0 = No error 1 = Error detected | | | | | | | | |
| RC_TXDIS = Detects if UART TX is disabled, but the host MCU has issued a command to read data from the device. 0 = No error 1 = Error detected | | | | | | | | |
| RC_SOF = Detects a start-of-frame (SOF) error. That is, an UART CLEAR is received on the UART before the current frame is finished. 0 = No error 1 = Error detected | | | | | | | | |
| RC_BYTE_ERR = Detects any byte error, other than the error in the initialization byte, in the received command frame. All bytes that follow are ignored until a communication CLEAR is received. When a communication frame is ignored, the device will not attempt to detect any communication error in the ignored frame nor counting it as valid/discard in the frame counters. 0 = No error 1 = Error detected | | | | | | | | |

| |
|---|
| <p>RC_UNEXP = In a stack device (that is, [STACK_DEV] = 1 and [MULTIDROP] = 0), it is not expected to receive a stack or broadcast command through the UART interface. If so, this is detected as an error and this bit is set. If device is configured with [MULTIDROP] = 1, this bit will not be set. 0 = No error 1 = Error detected</p> |
| <p>RC_CRC = Detects a CRC error in the received command frame from UART. The frame will be considered as discarded frame. 0 = No error 1 = Error detected</p> |

6.5.4.14.6 DEBUG_UART_RR_TR

| Address | 0x0782 | | | | | | | |
|---|--------|-------|-------|--------|---------|--------|-------------|--------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | TR_SOF | TR_WAIT | RR_SOF | RR_BYTE_ERR | RR_CRC |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| TR_SOF = Indicates that a UART CLEAR is received while the device is still transmitting data. 0 = No error 1 = Error detected | | | | | | | | |
| TR_WAIT = The device is waiting for its turn to transfer a response out but the action is terminated because either: <ul style="list-style-type: none"> • The device receives a UART CLEAR signal. • The device receives a new command. This bit is valid when broadcast or stack read command has been issued. 0 = No error 1 = Error detected | | | | | | | | |
| RR_SOF = Indicates a UART CLEAR is received while receiving the response frame. Response frames on the UART only apply in multidrop mode. 0 = No error 1 = Error detected | | | | | | | | |
| RR_BYTE_ERR = Detects any byte error, other than the error in the initialization byte, in the received response frame. All bytes that follow are ignored until a communication CLEAR is received. When a communication frame is ignored, the device will not attempt to detect any communication error in the ignored frame nor counting it as valid/discard in the frame counters. 0 = No error 1 = Error detected | | | | | | | | |
| RR_CRC = Detects are CRC error in the received response frame from UART. The frame will be considered as a discarded frame. 0 = No error 1 = Error detected | | | | | | | | |

6.5.4.14.7 DEBUG_COMH_BIT

| Address | 0x0783 | | | | | | | |
|---|--------|-------|-------|-------|----------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | PERR | BERR_TAG | SYNC2 | SYNC1 | BIT |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| PERR = Detects abnormality of the incoming communication frame and hence, the device will forward the communication frame with [BERR] bit set. Any error bit that is set in this register will also set the [PERR] bit. However, an abnormality that isn't classified in the register can also trigger the [PERR] bit (for example, detecting missing data or wrong data order). 0 = No communication error detected, the forwarded communication frame does not have the [BERR] inserted. 1 = Detected abnormality of the received communication frame. [BERR] is asserted to the forwarded communication. | | | | | | | | |

| |
|---|
| <p>BERR_TAG = Set when the received communication is tagged with $[BERR] = 1$. 0 = Received communication frame has no $[BERR]$ 1 = Received communication frame has $[BERR]$</p> |
| <p>SYNC2 = The Preamble half-bit and the $[SYNC1:0]$ bits are detected. Device is using the timing information extracted from these bits but it is unable to detect valid data. 0 = No error 1 = Error detected</p> |
| <p>SYNC1 = Unable to detect the preamble half-bit or any of the $[SYNC1:0]$ bits. It could be the bit is missing or the signal is too distorted to be detectable. 0 = No error 1 = Error detected</p> |
| <p>BIT = The device has detected a data bit; however, the detection samples are not enough to assure a strong 1 or 0. 0 = No error 1 = Error detected</p> |

6.5.4.14.8 DEBUG_COMH_RC

| Address | 0x0784 | | | | | | | |
|--|--------|-------|---------|----------|--------|-------------|----------|--------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | RC_IERR | RC_TXDIS | RC_SOF | RC_BYTE_ERR | RC_UNEXP | RC_CRC |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| <p>RC_IERR = Detects initialization byte error in the received command frame. This may due to improper formatting of a byte such as a frame initialization byte is expected, but start-of-frame (SOF) bit is not set, or an invalid frame type is selected. Because bytes received on the COMH/COML are propagated up the stack, it is likely devices in the upper stack will also detect this error. All bytes that follow are ignored until a SOF bit is set is received. When a communication frame is ignored, the device will not attempt to detect any communication error in the ignored frame nor counting it as valid/discard in the frame counters. 0 = No error 1 = Error detected</p> | | | | | | | | |
| <p>RC_TXDIS = Valid when $[DIR_SEL] = 1$. Device detects the COMH TX is disabled but the device receives a command to read data (that is, to transmit data out). The command frame will be counted as a discard frame. 0 = No error 1 = Error detected</p> | | | | | | | | |
| <p>RC_SOF = Valid when $[DIR_SEL] = 1$. Detects a start-of-frame (SOF) error on COMH. The SOF bit is set only in the initialization frame but the SOF bit is set in the current frame that is not expected. 0 = No error 1 = Error detected</p> | | | | | | | | |
| <p>RC_BYTE_ERR = Valid when $[DIR_SEL] = 1$. Detected any byte error, other than the error in the initialization byte, in the received command frame. This error can trigger one or more error bit set in the <i>DEBUG_COMMH_BIT</i> register. 0 = No error 1 = Error detected</p> | | | | | | | | |
| <p>RC_UNEXP = If $[DIR_SEL] = 0$, but device receives command frame from COMH which is an invalid condition and device will set this error bit. 0 = No error 1 = Error detected</p> | | | | | | | | |
| <p>RC_CRC = Indicates a CRC error that resulted in one or more COMH command frames being discarded. Any other errors in the frame are not indicated as the frame was discarded. 0 = No error 1 = Error detected</p> | | | | | | | | |

6.5.4.14.9 DEBUG_COMH_RR_TR

| Address | 0x0785 | | | | | | | |
|-----------|--------|-------|---------|----------|--------|-------------|----------|--------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | TR_WAIT | RR_TXDIS | RR_SOF | RR_BYTE_ERR | RR_UNEXP | RR_CRC |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| |
|--|
| RSVD = Reserved |
| <p>TR_WAIT = The device is waiting for its turn to transfer a response out but the action is terminated because the device receives a new command. This bit is valid when broadcast or stack read command has been issued. 0 = No error 1 = Error detected</p> |
| <p>RR_TXDIS = Valid when $[DIR_SEL] = 0$, device receives a response but fails to transmit to the next device because the COMH TX is disabled. The frame is counted as a discarded frame. 0 = No error 1 = Error detected</p> |
| <p>RR_SOF = Valid when $[DIR_SEL] = 0$. Detects a start-of-frame (SOF) error on COMH. The SOF bit is set only in the initialization frame but the SOF bit is set in the current frame that is not expected. 0 = No error 1 = Error detected</p> |
| <p>RR_BYTE_ERR = Valid when $[DIR_SEL] = 0$. Detects any byte error, other than the error in the initialization byte, in the received response frame. This error can trigger one or more error bits set in the <i>DEBUG_COMMH_BIT</i> register. 0 = No error 1 = Error detected</p> |
| <p>RR_UNEXP = If $[DIR_SEL] = 1$, but device received response frame from COMH which is an invalid condition and device sets this error bit. 0 = No error 1 = Error detected</p> |
| <p>RR_CRC = Indicates a CRC error that resulted in one or more COMH response frames being discarded. Most other errors in the frame are not indicated as the frame was discarded. If $[RR_BYTE_ERR]$ is observed on the final byte of the CRC, both CRC and BERR will be indicated. 0 = No error 1 = Error detected</p> |

6.5.4.14.10 DEBUG_COML_BIT

| Address | 0x0786 | | | | | | | |
|---|--------|-------|-------|-------|----------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | PERR | BERR_TAG | SYNC2 | SYNC1 | BIT |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| <p>PERR = Detect abnormality of the incoming communication frame and the outgoing communication frame will be set with BERR. Any error bit that is set in this register will also set the $[PERR]$ bit. However, abnormality that isn't classified in the register can also trigger the $[PERR]$ bit (for example, detecting missing data or wrong data order). 0 = No communication error detected, the forwarded communication frame does not have the BERR inserted 1 = Detected abnormality of the received communication frame. BERR is asserted to the forwarded communication.</p> | | | | | | | | |
| <p>BERR_TAG = Set when the received communication is tagged with BERR. 0 = Received communication frame has no BERR 1 = Received communication frame has BERR</p> | | | | | | | | |
| <p>SYNC2 = The Preamble half-bit and the $[SYNC1:0]$ bits are detected. Device is using the timing information that is extracted from these bits but it is unable to detect valid data. 0 = No error 1 = Error detected</p> | | | | | | | | |
| <p>SYNC1 = Unable to detect the preamble half-bit or any of the $[SYNC1:0]$ bits. It could be the bit is missing or the signal is too distorted to be detectable. 0 = No error 1 = Error detected</p> | | | | | | | | |
| <p>BIT = The device has detected a data bit. However, the detection samples are not enough to assure a strong 1 or 0. 0 = No error 1 = Error detected</p> | | | | | | | | |

6.5.4.14.11 DEBUG_COML_RC

| Address | 0x0787 | | | | | | | |
|-----------|--------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |

| Name | RSVD | | RC_IERR | RC_TXDIS | RC_SOF | RC_BYTE_ERR | RC_UNEXP | RC_CRC |
|--|------|---|---------|----------|--------|-------------|----------|--------|
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| RC_IERR = Detected initialization byte error in the received command frame. This may due to improper formatting of a byte such as a frame initialization byte is expected, but start-of-frame (SOF) bit is not set, or an invalid frame type is selected. Because bytes received on the COMH/COML are propagated up the stack, it is likely devices in the upper stack will also detect this error. All bytes that follow are ignored until a SOF bit is received. When a communication frame is ignored, the device will not attempt to detect any communication error in the ignored frame nor counting it as valid/discard in the frame counters. 0 = Error not detected 1 = Error detected | | | | | | | | |
| RC_TXDIS = Valid when $[DIR_SEL] = 0$. Device detects the COML TX is disabled but the device receives a command to read data (that is, to transmit data out). The command frame will be counted as a discarded frame. 0 = No error 1 = Error detected | | | | | | | | |
| RC_SOF = Valid when $[DIR_SEL] = 0$. Detects a start-of-frame (SOF) error on COML. The SOF bit is set only in the initialization frame but the SOF bit is set in the current frame that is not expected. 0 = No error 1 = Error detected | | | | | | | | |
| RC_BYTE_ERR = Valid when $[DIR_SEL] = 0$. Detected any byte error, other than the error in the initialization byte, in the received command frame. This error can trigger one or more error bits set in the <i>DEBUG_COMML_BIT</i> register. 0 = No error 1 = Error detected | | | | | | | | |
| RC_UNEXP = If $[DIR_SEL] = 1$, but device received command frame from COML which is an invalid condition and device will set this error bit. 0 = No error 1 = Error detected | | | | | | | | |
| RC_CRC = Indicates a CRC error that resulted in one or more COML command frames being discarded. Any other errors in the frame are not indicated as the frame was discarded. 0 = No error 1 = Error detected | | | | | | | | |

6.5.4.14.12 DEBUG_COML_RR_TR

| Address | 0x0788 | | | | | | | |
|--|--------|-------|---------|----------|--------|-------------|----------|--------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | TR_WAIT | RR_TXDIS | RR_SOF | RR_BYTE_ERR | RR_UNEXP | RR_CRC |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| TR_WAIT = The device is waiting for its turn to transfer a response out but the action is terminated because the device receives a new command. This bit is valid when broadcast or stack read command has been issued. 0 = No error 1 = Error detected | | | | | | | | |
| RR_TXDIS = Valid when $[DIR_SEL] = 1$, device receives a response but fails to transmit to the next device because the COML TX is disabled. The frame is counted as a discarded frame. 0 = No error 1 = Error detected | | | | | | | | |
| RR_SOF = Valid when $[DIR_SEL] = 1$. Detects a start-of-frame (SOF) error on COML. The SOF bit is set only in the initialization frame but the SOF bit is set in the current frame that is not expected. 0 = No error 1 = Error detected | | | | | | | | |
| RR_BYTE_ERR = Valid when $[DIR_SEL] = 1$. Detects any byte error, other than the error in the initialization byte, in the received response frame. This error can trigger one or more error bits set in the <i>DEBUG_COMML_BIT</i> register. 0 = No error 1 = Error detected | | | | | | | | |

| |
|--|
| <p>RR_UNEXP = If $[DIR_SEL] = 0$, but device received a response frame from COML which is an invalid condition and device will set this error bit. 0 = No error 1 = Error detected</p> |
| <p>RR_CRC = Indicates a CRC error that resulted in one or more COML response frames being discarded. Most other errors in the frame are not indicated as the frame was discarded. If $[RR_BYTE_ERR]$ is observed on the final byte of the CRC, both CRC and BERR are indicated. 0 = No error 1 = Error detected</p> |

6.5.4.14.13 DEBUG_UART_DISCARD

| Address | 0x0789 | | | | | | | |
|--|------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | COUNT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| <p>COUNT[7:0] = UART frame counter to track the number of discard frames received or transmitted. The registers of the <i>DEBUG_UART_DISCARD</i> and <i>DEBUG_UART_VALID*</i> are latched and the related counters are reset when this register is read.</p> | | | | | | | | |

6.5.4.14.14 DEBUG_COMH_DISCARD

| Address | 0x078A | | | | | | | |
|--|------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | COUNT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| <p>COUNT[7:0] = COMH frame counter to track the number of discard frames received or transmitted. The registers of the <i>DEBUG_COMH_DISCARD</i> and <i>DEBUG_COMH_VALID*</i> are latched and the related counters are reset when this register is read.</p> | | | | | | | | |

6.5.4.14.15 DEBUG_COML_DISCARD

| Address | 0x078B | | | | | | | |
|--|------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | COUNT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| <p>COUNT[7:0] = COML frame counter to track the number of discard frames received or transmitted. The registers of the <i>DEBUG_COML_DISCARD</i> and <i>DEBUG_COML_VALID*</i> are latched and the related counters are reset when this register is read.</p> | | | | | | | | |

6.5.4.14.16 DEBUG_UART_VALID_HI/LO

DEBUG_UART_VALID_HI

| Address | 0x078C | | | | | | | |
|---|------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | COUNT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| <p>COUNT[7:0] = The high-byte of UART frame counter to track the number of valid frames received or transmitted. Counter saturates when both <i>DEBUG_UART_VALID_HI/LO</i> is 0xFF. This register is latched and the related counter is reset when <i>DEBUG_UART_DISCARD</i> is read.</p> | | | | | | | | |

DEBUG_UART_VALID_LO

| Address | 0x078D | | | | | | | |
|---|------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | COUNT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| COUNT[7:0] = The low-byte of UART frame counter to track the number of valid frames received or transmitted. Counter saturates when both <i>DEBUG_UART_VALID_HI/LO</i> is 0xFF. This register is latched and the related counter is reset when <i>DEBUG_UART_DISCARD</i> is read. | | | | | | | | |

6.5.4.14.17 *DEBUG_COMH_VALID_HI/LO*

DEBUG_COMH_VALID_HI

| Address | 0x078E | | | | | | | |
|--|------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | COUNT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| COUNT[7:0] = The high-byte of COMH frame counter to track the number of valid frames received or transmitted. Counter saturates when both <i>DEBUG_COMH_VALID_HI/LO</i> is 0xFF. This register is latched and the related counter is reset when <i>DEBUG_COMH_DISCARD</i> is read. | | | | | | | | |

DEBUG_COMH_VALID_LO

| Address | 0x078F | | | | | | | |
|---|------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | COUNT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| COUNT[7:0] = The low-byte of COMH frame counter to track the number of valid frames received or transmitted. Counter saturates when both <i>DEBUG_COMH_VALID_HI/LO</i> is 0xFF. This register is latched and the related counter is reset when <i>DEBUG_COMH_DISCARD</i> is read. | | | | | | | | |

6.5.4.14.18 *DEBUG_COML_VALID_HI/LO*

DEBUG_COML_VALID_HI

| Address | 0x0790 | | | | | | | |
|--|------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | COUNT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| COUNT[7:0] = The high-byte of COML frame counter to track the number of valid frames received or transmitted. Counter saturates when both <i>DEBUG_COML_VALID_HI/LO</i> is 0xFF. This register is latched and the related counter is reset when <i>DEBUG_COML_DISCARD</i> is read. | | | | | | | | |

DEBUG_COML_VALID_LO

| Address | 0x0791 | | | | | | | |
|---|------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | COUNT[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| COUNT[7:0] = The low-byte of COML frame counter to track the number of valid frames received or transmitted. Counter saturates when both <i>DEBUG_COML_VALID_HI/LO</i> is 0xFF. This register is latched and the related counter is reset when <i>DEBUG_COML_DISCARD</i> is read. | | | | | | | | |

6.5.4.14.19 DEBUG_OTP_SEC_BLK

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x07A0 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | BLOCK[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| BLOCK[7:0] = Holds last OTP block address where SEC occurred. Valid only when <i>FAULT_OTP[SEC_DET]</i> = 1. | | | | | | | | |

6.5.4.14.20 DEBUG_OTP_DED_BLK

| | | | | | | | | |
|--|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x07A1 | | | | | | | |
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | BLOCK[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| BLOCK[7:0] = Holds last OTP block address where DED occurred. Valid only when <i>FAULT_OTP[DED_DET]</i> = 1. | | | | | | | | |

6.5.4.15 OTP Programming Control and Status

6.5.4.15.1 OTP_PROG_UNLOCK1A through OTP_PROG_UNLOCK1D

| | | | | | | | | |
|--|-----------------------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0300 to 0x0303 | | | | | | | |
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CODE[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CODE[7:0] = The first 32-bit OTP programming unlock code is required as part of the OTP programming unlock sequence before performing OTP programming. This 32-bit code is entered in the sequence from <i>OTP_PROG_UNLOCK1A</i> to <i>OTP_PROG_UNLOCK1D</i> . These registers always read back 0. | | | | | | | | |

6.5.4.15.2 OTP_PROG_UNLOCK2A through OTP_PROG_UNLOCK2D

| | | | | | | | | |
|---|-----------------------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x0352 to 0x0355 | | | | | | | |
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | CODE[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CODE[7:0] = The second 32-bit OTP programming unlock code, required as part of the OTP programming unlock sequence before performing OTP programming. This 32-bit code is entered in the sequence from <i>OTP_PROG_UNLOCK2A</i> to <i>OTP_PROG_UNLOCK2D</i> . These registers always read back 0. | | | | | | | | |

6.5.4.15.3 OTP_PROG_CTRL

| | | | | | | | | |
|---|---------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Address | 0x030B | | | | | | | |
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | RSVD | | | | | | PAGESEL | PROG_GO |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RSVD = Reserved | | | | | | | | |
| PAGESEL = Selects which customer OTP page to be programmed. 0 = page 1 1 = page 2 | | | | | | | | |
| PROG_GO = Enables programming for the OTP page selected by <i>OTP_PROG_CTRL[PAGESEL]</i> . Requires <i>OTP_PROG_UNLOCK1*</i> and <i>OTP_PROG_UNLOCK2*</i> registers are set to the correct codes. 0 = Ready 1 = Start OTP programming | | | | | | | | |

6.5.4.15.4 OTP_ECC_TEST

| Address | 0x034C | | | | | | | | |
|--|--------|-------|-------|-------|---------|-------------|---------|--------|--|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| Name | RSVD | | | | DED_SEC | MANUAL_AUTO | ENC_DEC | ENABLE | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| RSVD = Reserved | | | | | | | | | |
| DED_SEC = Sets the decoder function (SEC or DED) to test. This bit is ignored during encoder testing. 0 = Test SEC functionality. Sets the <i>FAULT_OTP[SEC_DETECT]</i> flag and outputs test result to <i>OTP_ECC_DATAOUT*</i> registers. 1 = Test DED functionality. Sets the <i>FAULT_OTP[DED_DETECT]</i> flag and outputs test result <i>OTP_ECC_DATAOUT*</i> . Note: If SEC or DEC fault is detected, host sets <i>[RST_OTP_DATA] = 1</i> to reset the corresponding fault. Switch to run SEC test does not clear DEC fault or vice versa. | | | | | | | | | |
| MANUAL_AUTO = Sets the location of the data to use for the ECC test. 0 = Auto mode. Use the internal data for test. 1 = Manual mode. Uses data in <i>ECC_DATAIN_n</i> registers for test. Use for MPF test. | | | | | | | | | |
| ENC_DEC = Sets the encoder/decoder test to run when <i>OTP_ECC_TEST[ENABLE] = 1</i> . 0 = Run decoder test 1 = Run encoder test | | | | | | | | | |
| ENABLE = Executes the OTP ECC test configured by <i>[ENC_DEC]</i> and <i>[DED_SEC]</i> bits. 0 = Normal operation, ECC test disabled 1 = Initiate test | | | | | | | | | |

6.5.4.15.5 OTP_ECC_DATAIN1 through OTP_ECC_DATAIN9

| Address | 0x0343 to 0x034B | | | | | | | |
|---|------------------|-------|-------|-------|-------|-------|-------|-------|
| RW | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | DATA[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| DATA[7:0] = When ECC is enabled in manual mode, <i>CUST_ECC_TEST[MANUAL_AUTO] = 1</i> , <i>OTP_ECC_DATAIN1...9</i> registers are used to test the ECC encoder/decoder. If <i>CUST_ECC_TEST[ENC_DEC] = 1</i> , <i>ECC_DATAIN8</i> through <i>ECC_DATAIN1</i> are fed to the encoder. If <i>CUST_ECC_TEST[ENC_DEC] = 0</i> , <i>ECC_DATAIN9</i> through <i>ECC_DATAIN1</i> are fed to the decoder. The <i>ECC_DATAOUT0...8</i> bytes must be read back to verify functionality. | | | | | | | | |

6.5.4.15.6 OTP_ECC_DATAOUT1 through OTP_ECC_DATAOUT9

| Address | 0x0510 to 0x0518 | | | | | | | |
|---|------------------|-------|-------|-------|-------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | DATA[7:0] | | | | | | | |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| DATA[7:0] = <i>OTP_ECC_DATAOUT*</i> bytes output the results of the ECC decoder and encoder tests. If <i>CUST_ECC_TEST[ENC_DEC] = 0</i> , <i>ECC_DATAOUT8</i> through <i>ECC_DATAOUT1</i> are read to determine a successful decoder test. If <i>CUST_ECC_TEST[ENC_DEC] = 1</i> , <i>ECC_DATAOUT9</i> through <i>ECC_DATAOUT1</i> are read to determine a successful encoder test. The correct result depends on the input to the test. | | | | | | | | |

6.5.4.15.7 OTP_PROG_STAT

| Address | 0x0519 | | | | | | | |
|-----------|--------|-------|-------|-------|--------|--------|---------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | UNLOCK | OTERR | UVERR | OVERR | SUVERR | SOVERR | PROGERR | DONE |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| |
|--|
| <p>UNLOCK = Indicates the OTP programming function unlock status. After this bit is set (that is, OTP programming is enabled), the host writes to the <i>OTP_PROG_CTRL</i> register to start the OTP programming. Writing to any other register relocks the OTP programming function and clears this bit to 0. [<i>PROG_GO</i>] = 1 also clears this bit to 0. 0 = OTP programming locked 1 = OTP programming is unlocked</p> |
| <p>OTERR = Indicates the die temperature is greater than T_{OTP_PROG} and device does not start OTP programming. 0 = No fault 1 = Detected die temperature is greater than T_{OTP_PROG}. Abort OTP programming.</p> |
| <p>UVERR = Indicates an undervoltage error detected on the programming voltage during OTP programming. This bit is cleared with [<i>PROG_GO</i>] = 1. 0 = No error 1 = UV error detected</p> |
| <p>OVERR = Indicates an overvoltage error detected on the programming voltage during OTP programming. This bit is cleared with [<i>PROG_GO</i>] = 1. Information received from the device with this error must not be considered reliable. 0 = No error 1 = OV error detected</p> |
| <p>SUVERR = A programming voltage stability test is performed before starting the actual OTP programming. This bit indicates an undervoltage error is detected during the voltage stability test. This bit is cleared with [<i>PROG_GO</i>] = 1. 0 = No error 1 = UV error detected during OTP programming voltage stability test</p> |
| <p>SOVERR = A programming voltage stability test is performed before starting the actual OTP programming. This bit indicates an overvoltage error is detected during the voltage stability test. This bit is cleared with [<i>PROG_GO</i>] = 1. 0 = No error 1 = OV error detected during OTP programming voltage stability test</p> |
| <p>PROGERR Indicates when an error is detected due to incorrect page setting caused by any of the following:</p> <ul style="list-style-type: none"> • Trying to program but OTP programming [<i>UNLOCK</i>] = 0. • Trying to program a page that has [<i>TRY</i>] = 1. • Trying to program a page which has [<i>FMterr</i>] = 1. <p>This bit is cleared with [<i>PROG_GO</i>] = 1. 0 = No error or programming not attempted 1 = Error detected</p> |
| <p>DONE = Indicates the status of the OTP programming for the selected page. This bit is cleared with [<i>PROG_GO</i>] = 1. 0 = Not completed or programming not attempted 1 = Complete.</p> |

6.5.4.15.8 OTP_CUST1_STAT

| Address | 0x051A | | | | | | | |
|--|--------|---------|---------|--------|--------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | LOADED | LOADWRN | LOADERR | FMterr | PROGOK | UVOK | OVOK | TRY |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| <p>LOADED = Indicates OTP page 1 has been selected for loading into the related registers. See [<i>LOADERR</i>] and [<i>LOADWRN</i>] for error and warning status. 0 = Not selected for loading 1 = Page 1 selected and loaded</p> | | | | | | | | |
| <p>LOADWRN Indicates OTP page 1 was loaded but with one or more SEC warnings. = 0 = No warning, or no load attempted 1 = Warning</p> | | | | | | | | |
| <p>LOADERR = Indicates an error while attempting to load OTP page 1; that is, DED is detected while loading the selected page. 0 = No error, or no load was attempted. 1 = Error detected</p> | | | | | | | | |
| <p>FMterr = Indicates a formatting error in OTP page 1; that is, when [<i>UVOK</i>] or [<i>OVOK</i>] is set, but [<i>TRY</i>] = 0. Do not program if this bit is set. 0 = No error 1 = Error detected</p> | | | | | | | | |
| <p>PROGOK = Indicates the validity for loading for OTP page 1. A valid page indicates that successful programming occurred. 0 = Not valid 1 = Valid</p> | | | | | | | | |

| |
|--|
| <p>UVOK = Indicates an OTP programming voltage undervoltage condition is detected during programming attempt for OTP page 1. The OV condition may also trigger the UV as part of the shutdown process. 0 = UV condition detected. Also reads as 0 if no programming attempt is performed. 1 = No UV condition detected</p> |
| <p>OVOK = Indicates an OTP programming voltage overvoltage condition is detected during programming attempt for OTP page 1. The OV condition will trigger the UV as part of the shutdown process. The device must be taken out of service. 0 = OV condition detected. Also reads as 0 if no programming attempt is performed. 1 = No OV condition detected</p> |
| <p>TRY = Indicates a first programming attempt for OTP page 1. 0 = No first attempt made 1 = First attempt made</p> |

6.5.4.15.9 OTP_CUST2_STAT

| Address | 0x051B | | | | | | | |
|--|--------|---------|---------|--------|--------|-------|-------|-------|
| Read Only | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Name | LOADED | LOADWRN | LOADERR | FMTERR | PROGOK | UVOK | OVOK | TRY |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| <p>LOADED = Indicates OTP page 2 has been selected for loading into the related registers. See <i>[LOADERR]</i> and <i>[LOADWRN]</i> for error and warning status. 0 = Not selected for loading 1 = Page 2 selected and loaded</p> | | | | | | | | |
| <p>LOADWRN Indicates OTP page 2 was loaded but with one or more SEC warnings. = 0 = No warning, or no load attempted 1 = Warning</p> | | | | | | | | |
| <p>LOADERR Indicates an error while attempting to load OTP page 2; that is, DED is detected while loading the selected page. = 0 = No error, or no load was attempted. 1 = Error detected</p> | | | | | | | | |
| <p>FMTERR = Indicates a formatting error in OTP page 2; that is, when <i>[UVOK]</i> or <i>[OVOK]</i> is set, but <i>[TRY]</i> = 0. Do not program if this bit is set. 0 = No error 1 = Error detected</p> | | | | | | | | |
| <p>PROGOK = Indicates the validity for loading for OTP page 2. A valid page indicates that successful programming occurred. 0 = Not valid 1 = Valid</p> | | | | | | | | |
| <p>UVOK = Indicates an OTP programming voltage undervoltage condition is detected during programming attempt for OTP page 2. The OV condition may also trigger the UV as part of the shutdown process. 0 = UV condition detected. Also reads as 0 if no programming attempt is performed. 1 = No UV condition detected</p> | | | | | | | | |
| <p>OVOK = Indicates an OTP programming voltage overvoltage condition is detected during programming attempt for OTP page 2. The OV condition will trigger the UV as part of the shutdown process. The device must be taken out of service. 0 = OV condition detected. Also reads as 0 if no programming attempt is performed. 1 = No OV condition detected</p> | | | | | | | | |
| <p>TRY = Indicates a first programming attempt for OTP page 2. 0 = No first attempt made 1 = First attempt made</p> | | | | | | | | |

7 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.

7.1 Application Information

The BQ79616 device family provides high-accuracy, cell voltages and temperature measurements for 6-series to 16-series battery modules.

7.2 Typical Applications

7.2.1 Base Device Application Circuit

The following application circuit (see [Figure 7-1](#)) is based on the BQ79616 device connecting to a 16S module.

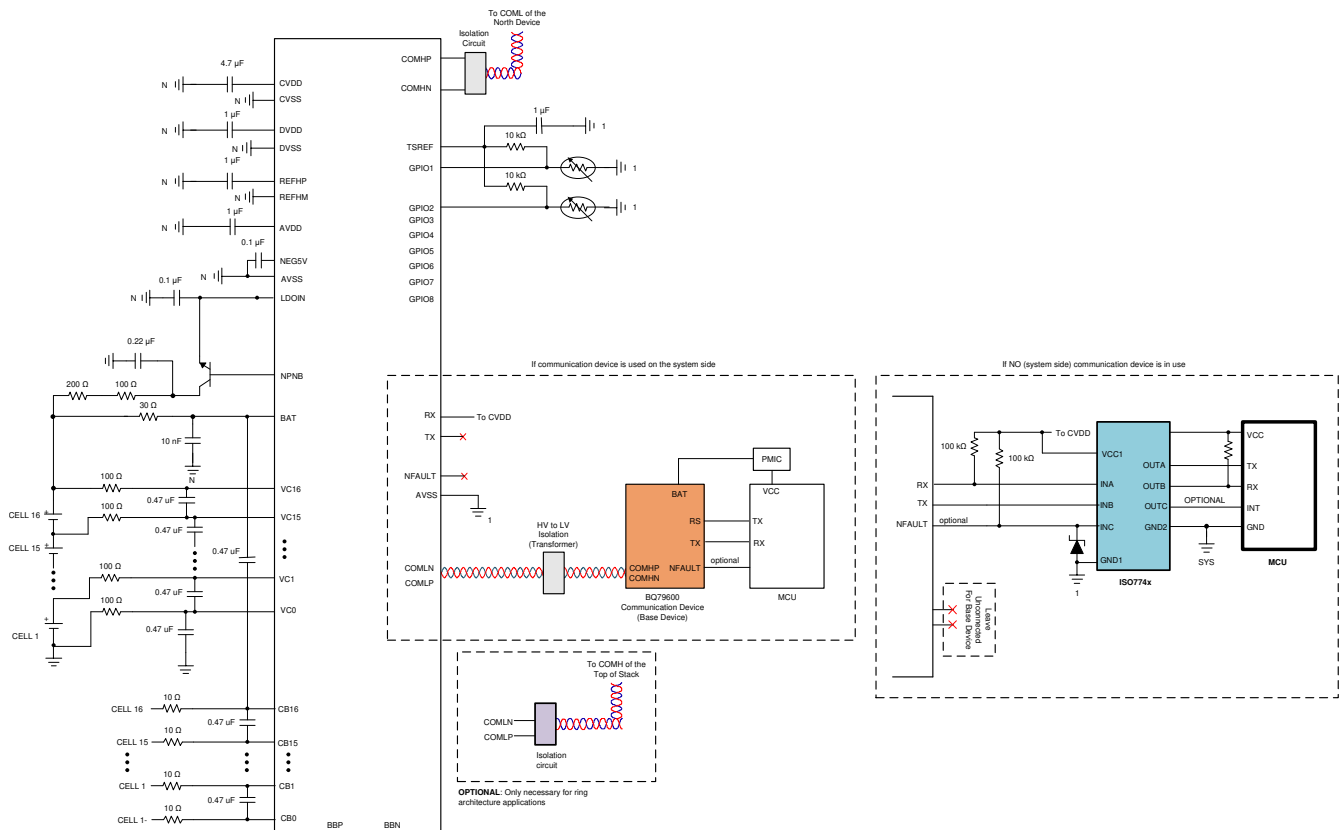


Figure 7-1. Typical Base Device with Measurement Application Circuit

7.2.1.1 Design Requirements

[Table 7-1](#) below shows the design parameters.

Table 7-1. Recommended Design Requirements

| PARAMETER | VALUE |
|---|---|
| Module Voltage Range (Voltage at the BAT pin) | 9V to 80V |
| Number of cells (single device) | 6 to 16 cells (BQ79616), 6 to 14 cells (BQ79616), 6 to 12 cells (BQ79616) |

Table 7-1. Recommended Design Requirements (continued)

| PARAMETER | VALUE |
|--------------------|----------|
| Cell voltage range | 0V to 5V |

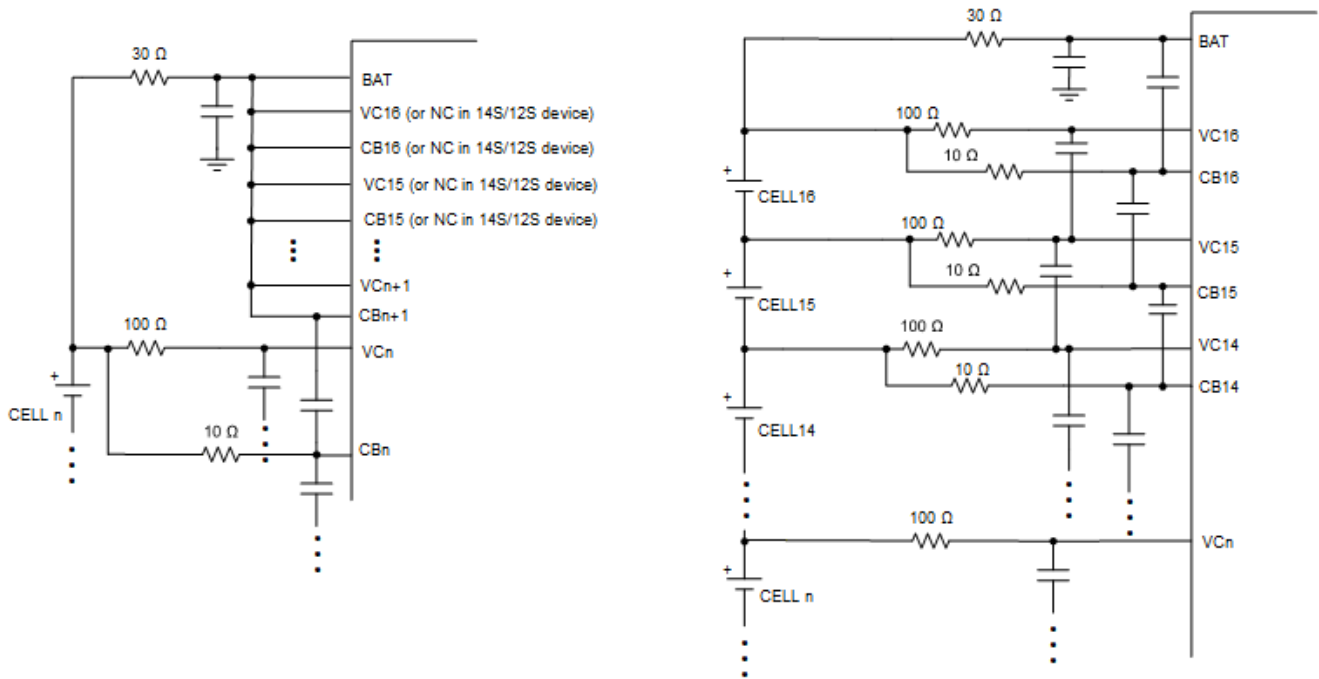
7.2.1.2 Detailed Design Procedure

7.2.1.2.1 Cell Sensing and Balancing Inputs

| Related Pins | Components | Value | Description |
|--------------|------------------|--|---|
| VC0 to VC16 | Filter resistor | 100Ω | Only differential RC filters are needed for VC channels. Besides serving for input signal filtering, these components are required to support hot-plug events during cell module insertion. Hence, it is highly recommended to use the component values as suggested. |
| | Filter capacitor | 0.47μF/16V or 1μF/16V | |
| CB0 to CB16 | Filter resistor | Depends on system's balancing current requirements | The filter resistor on CB pins sets the maximum balancing current. See Section 6.3.3 for details. Only differential RC filters are needed for CB channels. Besides serving for input signal filtering, these components are required to support hot-plug events during cell module insertion. Hence, it is highly recommended to use the component values as suggested. |
| | Filter capacitor | 0.47μF/16V or 1μF/16V | |

Cell Connections

Populate the battery cells from the bottom channels (both VC and CB channels) and up, leaving upper channels as unused channels if cell module size is smaller than the maximum channel size of the BQ79616 device. Unused channel(s) in the BQ79616 device are connected as shown in Figure 7-2. PCB Layout for open/NC pins should have minimum trace lengths and must not be connected to a wire or cable.



(a) Customized PCB for certain channels applications - Short unused pins to BAT Pin

(b) One PCB for all channels applications - For BQ79616: Configured for 18 VC and CB

Figure 7-2. Unused VC and CB Channels

7.2.1.2.2 BAT and External NPN

| Related Pins | Components | Value | Description |
|--------------|--|--|--|
| BAT | Filter resistor | 30 Ω | Single-ended RC filter, recommended values must be used for hot-plug performance. |
| | Filter capacitor | 10 nF/100 V Can use lower voltage rating based on module size | |
| NPNB | NPN (Q1) | Collector–emitter breakdown voltage 80 V to 100 V, but can use lower rating based on module size Power rating ≥ 1 W Gain > 80 at the expected load current Current handling >100 mA | The external NPN is used to form a preregulation circuit to provide a 6-V (typical) input to the LDOIN pin. The voltage rating of the NPN can be optimized by the following equation: NPN voltage rating = Max VModule – Min VLDOIN + Margin Where: Max VModule = maximum module voltage with fully charged cells Min VLDOIN = the minimum spec of the VLDOIN parameter Margin = system transient voltage + design margin per application requirement |
| | Resistor on external NPN collector (R _{NPN}) | Various based on module voltage | The resistor has a couple purposes: (a) For an RC filter for the NPN preregulation circuit (b) Share the thermal dissipation with the NPN |
| | Capacitor on external NPN collector | 0.22 μF/100 V Can use lower voltage rating based on module size | The capacitor forms the RC filter for the NPN preregulation circuit The capacitor rating is based on peak voltage spike seen on the module. For smaller module size, < 100-V rated capacitor can be used. System designer selects the optimized voltage-rated capacitor per their system tolerance and requirements. |

To reduce the power rating needed for the external NPN (Q1), system designers can put power resistors on the NPN collector to create an IR drop from the module voltage (V_{Module}). Figure 7-3 shows the current paths to power the BQ79616 device.

Typical I_{STARTUP} current, that is, inrush startup current when the device enters from SHUTDOWN to ACTIVE, is 20 mA for TI recommended components. This current is the sum of I_{BAT} + I_{LDOIN}, and is dependent on PCB board components and layout. Designers must characterize this on their end.

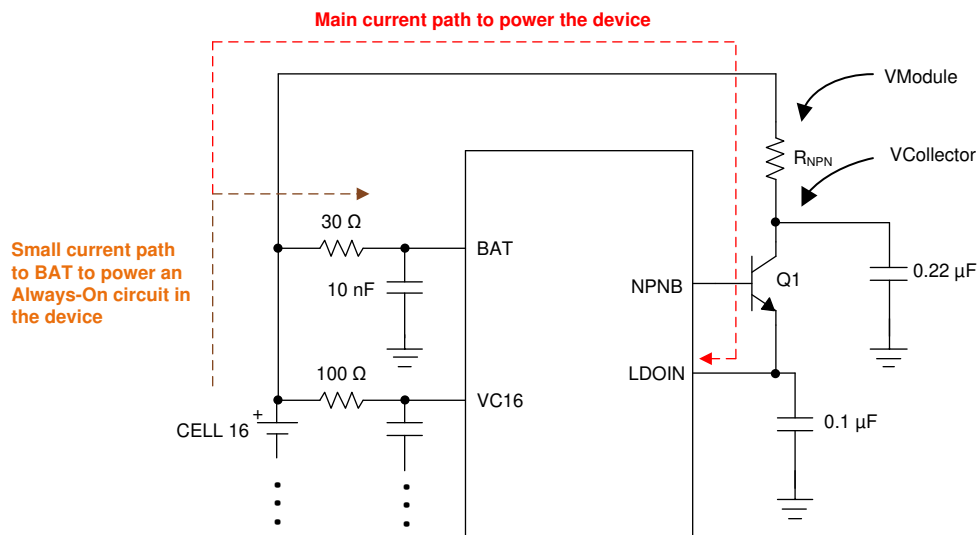


Figure 7-3. Power Consumption Paths

To ensure there is sufficient headroom to maintain 6 V (typical) regulated voltage on LDOIN pin, system designer ensures V_{Collector} has ≥ 8 V at any time with the assumption of about 2-V drop across the NPN.

Therefore, maximum allowable R_{NPN} value = ((Min V_{Module}) – (V_{Collector})) / (Max peak current)

Where:

Min VModule: based on module size and minimum cell voltage per application

VCollector: 8 V with the assumption of about 2-V drop across NPN

Max peak current: highest operation current, which is the active current with all functions turned on. Note that different communication isolation components (for example, capacitor isolation versus transformer, or the type of transformer) contribute different loading to the total power consumption.

Powering the device separately from the top of the battery stack:

The device is designed to be powered by the battery stack. If there is a need to power the device from a separate source such as in [Figure 7-4](#), the following relationship between the voltage on the BAT pin and the highest VC pin voltage (with respected to ground) is $BAT\ voltage \geq (0.5 \times \text{highest VC voltage}) + 2$.

For example, if the device is connected to a 14S module with max cell voltage of 4.2 V/cell, the highest VC pin is VC14, and the highest VC14 voltage is $(4.2\ V \times 14) = 58.8\ V$. If the BAT pin is powered separately, BAT voltage must be $\geq 31.4\ V$.

Similarly, if BBP/N channel is connected above the highest cell stack, the BBP pin will have the highest voltage (with respected to ground) than the VC pins. In this scenario, $V_{BAT} \geq [(V_{BBP}-2.5) \times 0.84] + 4.5$. The requirement applies when BAT is power separately and it is to ensure proper operation of the internal level shifter. Fail to maintain the voltage relationship will increase the ADC measurement error on the VC and BB channels.

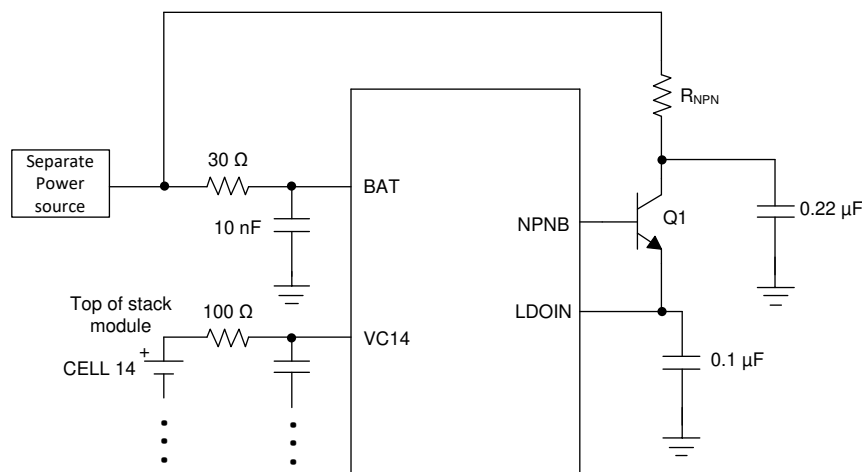


Figure 7-4. Separate Power Source to BAT

7.2.1.2.3 Power Supplies, Reference Input

| Related Pins | Components | Value | Description |
|-------------------|------------------|------------------|---|
| AVDD, DVDD, TSREF | Bypass capacitor | 1 μ F/10 V | Bypass capacitor for the internal LDOs |
| CVDD | Bypass capacitor | 4.7 μ F/10 V | Bypass capacitor for CVDD |
| NEG5V | Bypass capacitor | 0.1 μ F/10 V | Bypass capacitor for the negative charge pump |

7.2.1.2.4 GPIO For Thermistor Inputs

When using external thermistor, for ADC measurement only, there is no limitation of what type of thermistors (NTC or PTC) or the bias resistor (R1) value or whether the thermistor is placed on high side or low side with respected to the bias resistor.

However, when using with the integrated OTUT comparators, the programmable OT and UT threshold ranges are designed to work with a 103NTC (10 kΩ at 25°C) type of NTC thermistor, following the connection shown in Figure 7-5 with different options for the R1 and R2 resistors.

- Option 1: $R_1 = 10\text{ k}\Omega$, and no R_2
- Option 2: $R_1 = 10\text{ k}\Omega$, and $R_2 = 100\text{ k}\Omega$ for better linearity at cold temperature
- Option 3: $R_1 = 3.6\text{ k}\Omega$, and $R_2 = 15\text{ k}\Omega$. This base option can be used for NTC used for the OTCB feature assuming system designer allows the PCB temperature to be higher than the cell temperature during balancing. Because the device does not differentiate which NTC is used on the cells versus the PCB, NTC biasing with this option scales the NTC's hot temperature curve differently, allowing the threshold set for OT comparator to be triggered at a lower GPIO voltage. Thus, making the device to only trigger OTCB threshold on this NTC.

The device does not require external RC for temperature measurement. However, it is common for system designer to add an RC filter on the GPIO pin for the NTC circuit. System designer can select the RC values for the application need. Example: $R_{\text{GPIO}} = 1\text{ k}\Omega$, $C_{\text{GPIO}} = 0.1\text{ }\mu\text{F}$ to $1\text{ }\mu\text{F}$.

Unused GPIO must be grounded to AVSS with a 10-kΩ resistor.

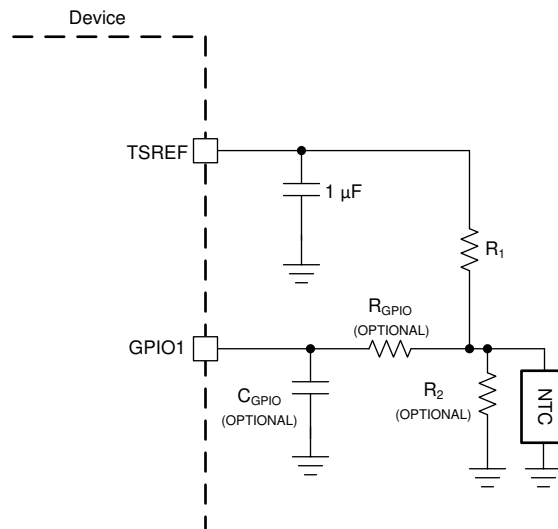


Figure 7-5. NTC Connection

7.2.1.2.5 Internal Balancing Current

When internal cell balancing is used, the max balancing current the device can support (before going into thermal pause) can vary based on the ambient temperature.

7.2.1.2.6 UART, NFAULT

If device is used as a base device, the UART interface requires the TX and RX pins are pulled up through a 10-kΩ to 100-kΩ resistor. Do not leave TX and RX unconnected. The TX must be pulled high to prevent triggering an invalid communications frame during the idle state. When using a serial cable to connect to the host controller, connect the TX pull-up on the host side and the RX pull-up to the CVDD on the device side.

If device is used as a stack device, the TX pin is disabled by default and is left floating. RX pin is shorted to CVDD.

NFAULT pin for base device, if not used, must be left floating. Otherwise, pull it up with 100-kΩ to CVDD. NFAULT pin on stack device is floating.

7.2.1.2.7 Daisy Chain Isolation

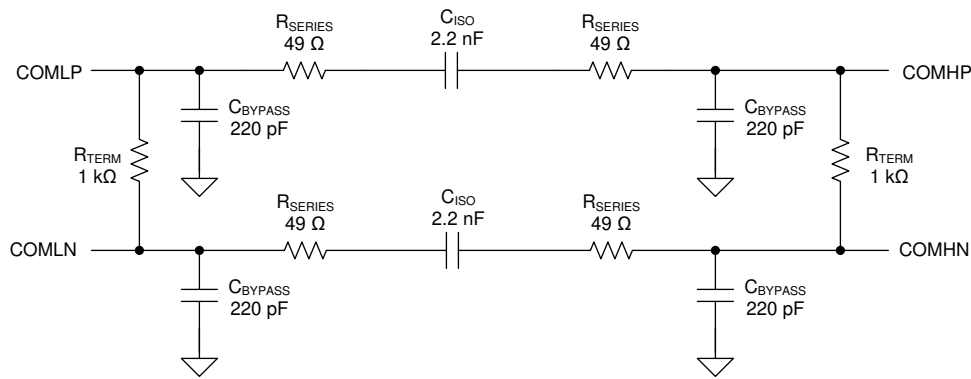
The device works with multiple daisy chain isolation types: capacitor isolation, capacitor-choke isolation, and transformer isolation. For devices that are daisy-chained on the same PCB, capacitor isolation without ESD

diode as shown in Figure 7-6 is sufficient. Unused COMLP/H or COMHP/N pins must be connected with 1-k Ω termination resistor.

7.2.1.2.7.1 Devices Connected on the Same PCB

Table 7-2. Isolation Components for Devices Connected on the Same PCB

| Components | Value | Description (Capacitor Isolation on the Same PCB) |
|---------------------|--------------|---|
| R_{TERM} | 1 k Ω | Termination resistor |
| R_{SERIES} | 49 Ω | Filter resistor and impedance matching resistor. The connection between devices must be $\sim 120 \Omega$ ($\sim 50 \Omega$ on each end of the signal connection of the device plus 10- Ω internal resistance) |
| C_{BYPASS} | 220 pF/50 V | Bypass capacitor |
| C_{ISO} | 2.2 nF | Isolation capacitor Voltage rating depends on application requirement. It is common to select 2x of module voltage rating to provide standoff margin in the event of a fault in the system. |



Components Required for Cap Coupled Daisy Chain on the same PCB

Figure 7-6. Capacitor Isolation with Devices on the Same PCB

7.2.1.2.7.2 Devices Connected on Different PCBs

For devices that are daisy-chained to different PCBs through a pair of twisted cables, all three isolation types can be used for daisy chain isolation, however it is not possible to use one type of isolation on one side of the daisy chain (for example, transformer isolation on COMLP/N to the Battery Management Unit) while using a different type of isolation for the other side of the daisy chain (for example, capacitor isolation on COMH/N to the Cell Module Unit).

Option 1: Capacitor Isolation

Table 7-3. Components for Capacitor Isolation on Different PCBs

| Components | Value | Description (Capacitor Isolation Between PCBs) |
|---------------------|--------------|--|
| R_{TERM} | 1 k Ω | Termination resistor |
| R_{SERIES} | 49 Ω | Filter resistor and impedance matching resistor. The connection between devices must be $\sim 120 \Omega$ ($\sim 50 \Omega$ on each end of the signal connection of the device plus 10- Ω internal resistance). |
| C_{BYPASS} | 220 pF/50 V | Bypass capacitor |
| C_{ISO} | 2.2 nF | Isolation capacitor Voltage rating depends on application requirement. It is common to select 2x of module voltage rating to provide standoff margin in the event of a fault in the system. |

Table 7-3. Components for Capacitor Isolation on Different PCBs (continued)

| Components | Value | Description (Capacitor Isolation Between PCBs) |
|------------|-----------|--|
| ESD diode | TVS diode | The ESD protector should provide protection to the communication interface pins during hot-plug events and also for absorption of high-voltage transients during service disconnect or reconnect. Select the ESD diodes to limit the maximum voltage on the COM* bus to below the maximum rating. A voltage rating close to the maximum voltage to provide the highest possible common-mode voltage range is recommended for best EMC performance. The capacitance must be low compared to the coupling capacitance (if using capacitor coupling). |

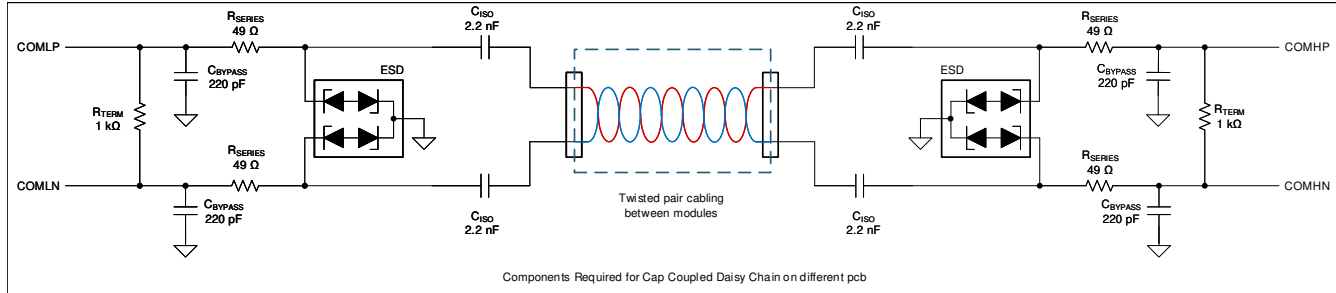


Figure 7-7. Capacitor Isolation on Different PCB

Figure 7-7 shows the capacitor isolation circuit for devices connecting between PCBs. Similar to the capacitor isolation on the same PCB case, the capacitor must be rated with a high enough voltage to provide standoff margin in the event of a fault in the system that exposes the device to a local hazardous voltage. The voltage is determined by the application requirement but it is common to select 2x of the module voltage.

The capacitance on the daisy chain bus has a direct effect on performance. All parasitic capacitances from the support components and cabling must be taken into consideration when designing for communication robustness to EMC. Capacitance from the cables, ESD diodes, bypass capacitance, and chokes form a capacitive divider with the isolation capacitors that may affect performance. Additionally, the amount of capacitance on the bus has a direct impact to the operating current during communication (the capacitor charging or discharging).

Option 2: Capacitor Plus Common-Mode Choke Isolation

Table 7-4. Components for Capacitor Plus Common-Mode Choke Isolation

| Components | Value | Description (Capacitor Plus Choke Isolation Between PCBs) |
|----------------------|------------------|--|
| R _{TERM} | 1 kΩ | Termination resistor |
| R _{SERIES} | 49 Ω | Filter resistor and impedance matching resistor. The connection between devices must be ~120 Ω (~50 Ω on each end of the signal connection of the device plus 10-Ω internal resistance) |
| C _{BYPASS} | 220 pF/50 V | Bypass capacitor |
| C _{ISO} | 2.2 nF | Isolation capacitor Voltage rating depends on application requirement. It is common to select 2x of module voltage rating to provide a standoff margin in the event of a fault in the system. |
| Common-mode choke | 100 μH to 500 μH | Common-mode choke (The inductance range 100 μH to 500 μH is a general guidance value, not a guaranteed range, as there are many parameters affecting the performance of a common-mode choke.) |
| ESD diode (optional) | TVS diode | Optional ESD protection depends on pcb level ESD requirement (Adding this diode or not is subject to the user's system level ESD requirement) |

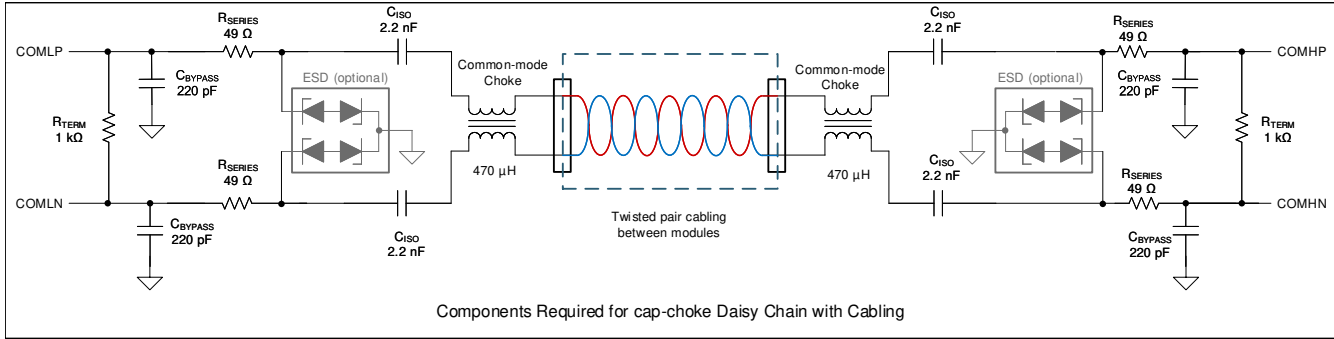


Figure 7-8. Capacitor Plus Choke Isolation

Longer cable lengths, or abnormally noisy applications may require the use of a common-mode choke filter. Capacitor plus choke isolation has better noise immunity than capacitor only. For these applications, use an automotive grade from 100-μH to 500-μH common-mode filter minimum for proper operation. To achieve the best performance in noisy environments, use dual common-mode filters (470 μH). The recommended impedance of the choke is at least 1 kΩ from 1 MHz to 100 MHz and above 300 Ω for higher frequencies.

Option 3: Transformer Isolation

Table 7-5. Components for Transformer Isolation

| Components | Value | Description (Capacitor Plus Choke Isolation Between PCBs) |
|----------------------|-------------------------------|--|
| R _{TERM} | 1 kΩ | Termination resistor |
| R _{SERIES} | 49 Ω | Filter resistor and impedance matching resistor. The connection between devices must be ~120 Ω (~50 Ω on each end of the signal connection of the device plus 10-Ω internal resistance) |
| C _{BYPASS} | 220 pF/50 V | Bypass capacitor |
| Transformer | Inductance: 150 μH to 1400 μH | The inductance range 150 μH to 1400 μH is a general guidance value, not a guaranteed range as there are many parameters affecting the performance of transformer. When coming to specific recommended part, please refer to SLVAEP4 <i>BQ79616 Daisy Chain Communications Application Report</i> . User shall perform the through test in their environment. |
| ESD diode (optional) | TVS diode | Optional ESD protection depends on pcb level ESD requirement (Adding this diode or not is subject to the user's system level ESD requirement.) |

Transformer isolation is supported. [Transformer Isolation](#) shows how it can be implemented. For example, transformer isolation can be used between the low-voltage and high-voltage boundary for galvanic isolation.

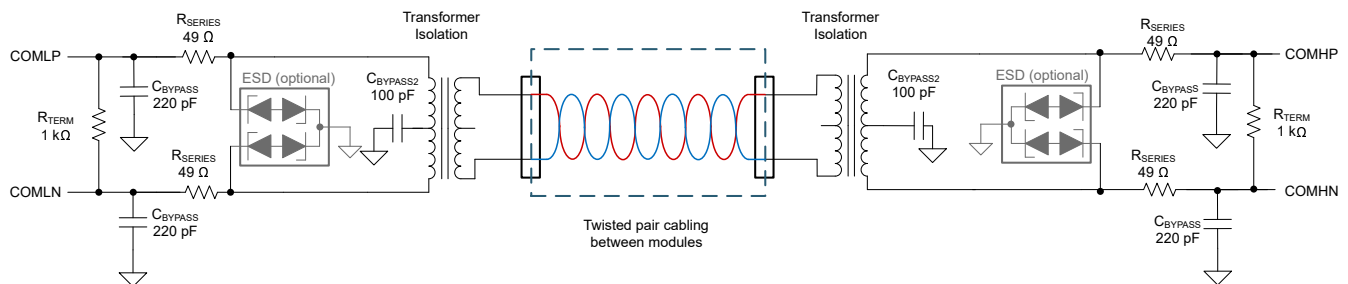


Figure 7-9. Transformer Isolation

7.2.1.3 Application Curve

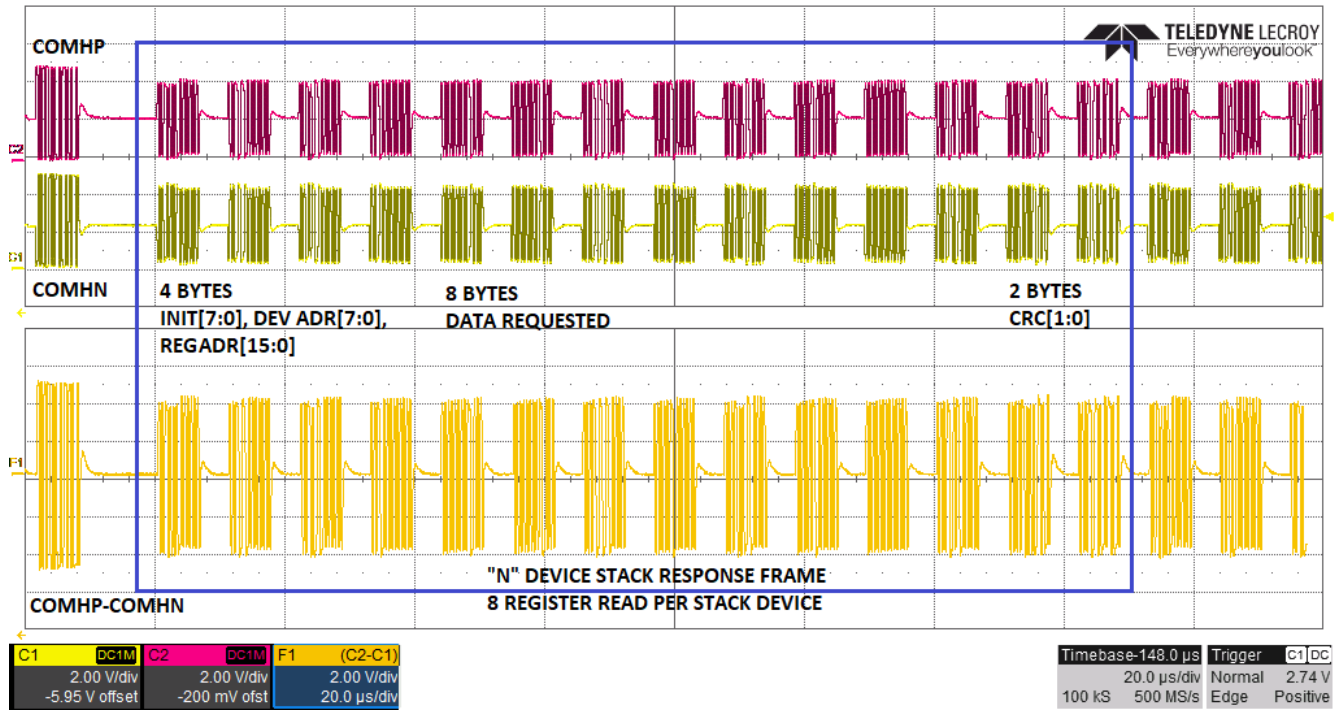


Figure 7-10. Response Frame for 8 Registers Read from Stack Devices

7.2.2 Daisy Device Application Circuit

The following application circuit (see [Figure 7-11](#)) is based on the BQ79616 device connecting to a 16S module.

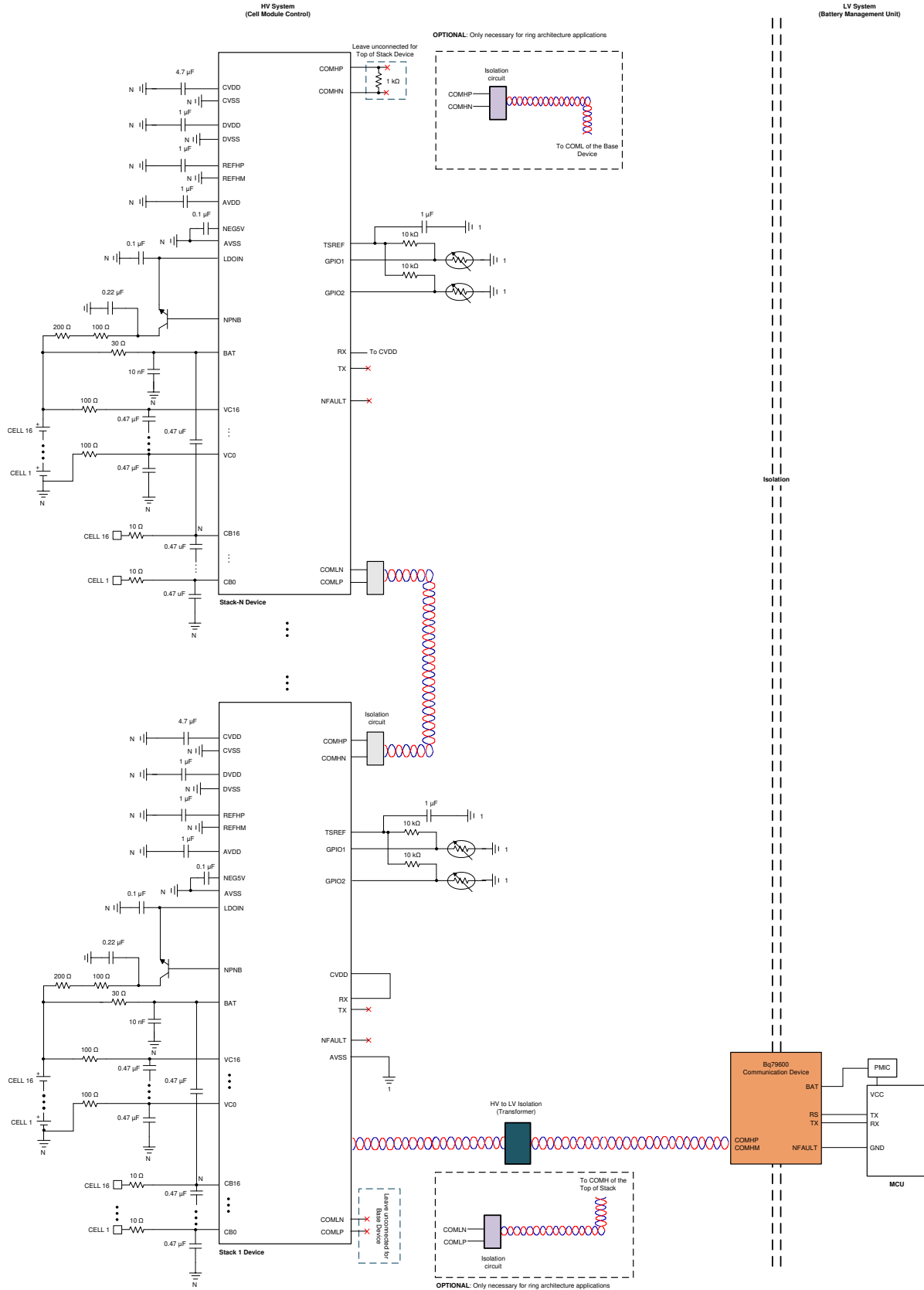


Figure 7-11. Daisy Device Application Circuit

7.2.2.1 Design Requirements

See [Section 7.2.1.1](#) section for design requirements.

7.2.2.2 Detailed Design Procedure

See [Section 7.2.1.2](#) section for detailed design procedure.

8 Power Supply Recommendations

The device is powered by BAT pin and the LDOIN pin, with which the LDOIN pin is regulated by the pre-regulation circuit formed with an external NPN. The device can be powered by a battery module as low as 9 V (without OTP programming) on the BAT pin. However, system designers must scale the R_{NPN} resistor accordingly to ensure there is sufficient headroom to have 6 V on the LDOIN pin after the IR drop across R_{NPN} and the external NPN. For example, if BAT voltage is at 9 V, the R_{NPN} reduces to 10 Ω to allow sufficient voltage at the LDOIN pin.

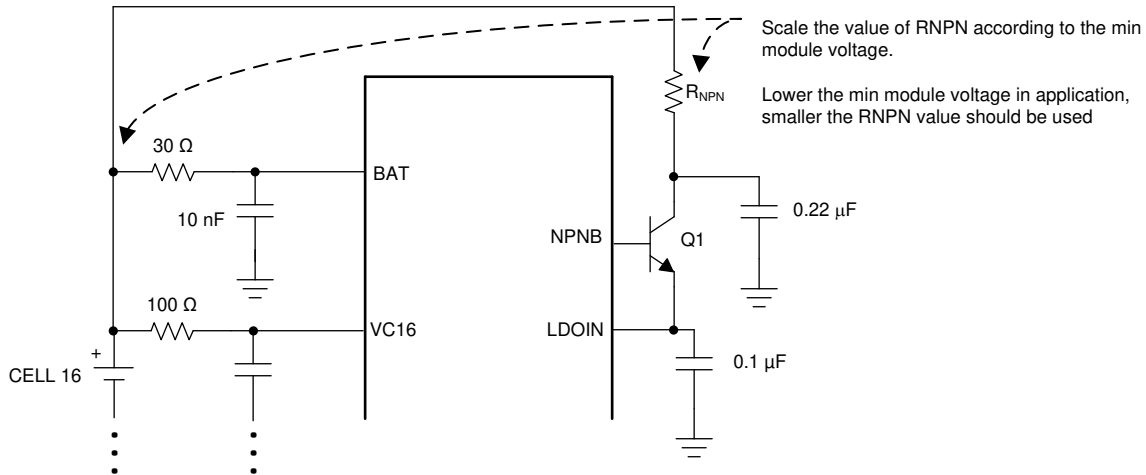


Figure 8-1. Device Powering Path

Multiple cell modules can be connected to the same device through the bus bar support of the BQ79616 family. The same power will be drawn from each of the cell modules.

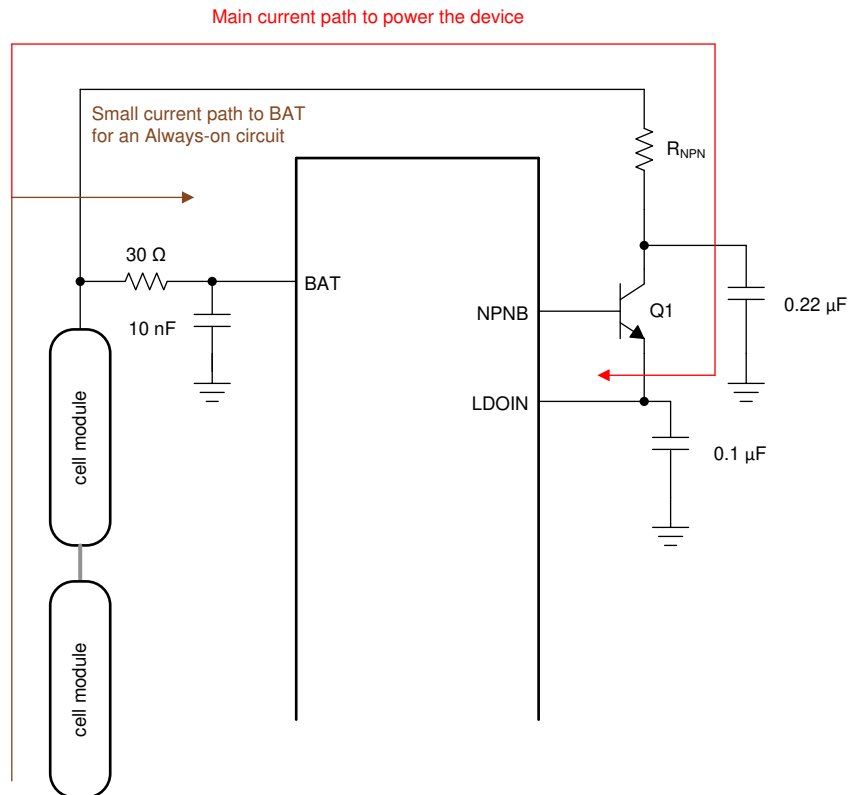


Figure 8-2. Device Can Be Powered by Multiple Cell Modules in Stack

9 Layout

The layout for this device must be designed carefully. Any design outside these guidelines can affect the ADC accuracy and EMI performance. Care must be taken in the layout of signals to and from the device to avoid coupling noise onto sensitive inputs. The layout of ground and power connections, as well as communication signals, should also be made carefully.

9.1 Layout Guidelines

9.1.1 Ground Planes

It is very important to establish a clean grounding scheme to ensure best performance of the device. There are three ground pins (AVSS, DVSS, CVSS) for the device's internal power supplies and one ground reference (REFHM) for the precision reference. There are noisy grounds and quiet grounds that must be separated in the layout initially and rejoined together in a lower PCB layer. The external components (for example, bypass capacitors) must be tied to the proper grounding group if possible to keep the separation of noisy and quiet grounds apart.

- AVSS ground:
 - Bypass capacitor for these pins: BAT, VC0, CB0, and AVDD.
 - Package power pad
- DVSS ground:
 - Bypass capacitor for DVDD.
 - GPIO filter capacitor (if used). It can also connect to AVSS ground plane, if needed.
- CVSS ground:
 - Bypass capacitor for GPIOs, CVDD, TSREF, NEG5V, LDOIN, COMHP/N, and COMLP/N.
- REFHM ground:
 - Bypass capacitor for REFHP.
 - If possible, separate out REFHM from AVSS on the signal connection layer and reconnect REFHM to AVSS ground plane in the lower layer.

Even on a PCB layer that is mainly for signal routing, it is good practice to have a small island of ground pour if possible to provide a low-impedance ground, rather than simply a via through the ground trace to an lower ground plane.

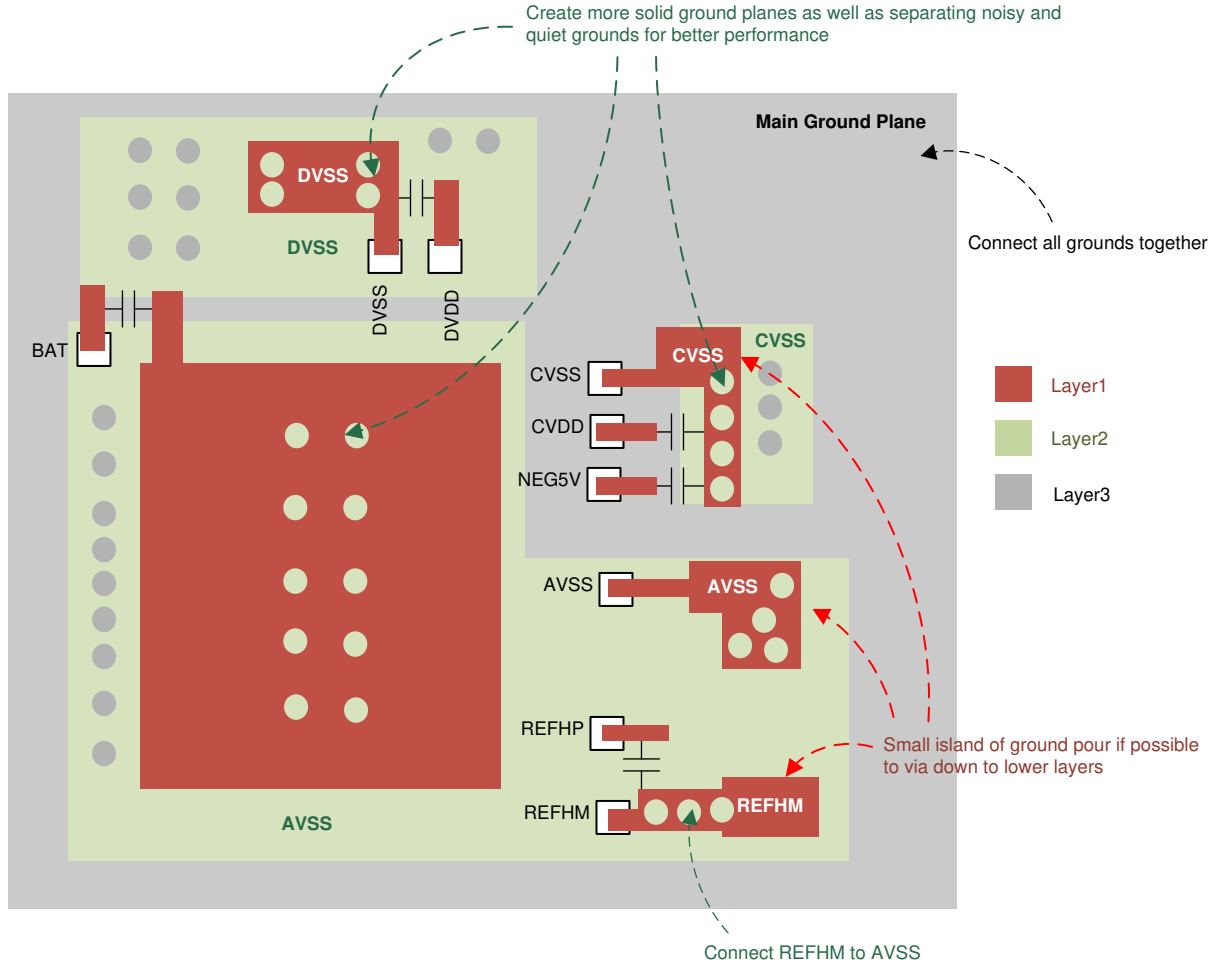


Figure 9-1. Grounding Layout Consideration

If multiple devices are placed on the same PCB, each device must have its own ground plane with proper layout clearance.

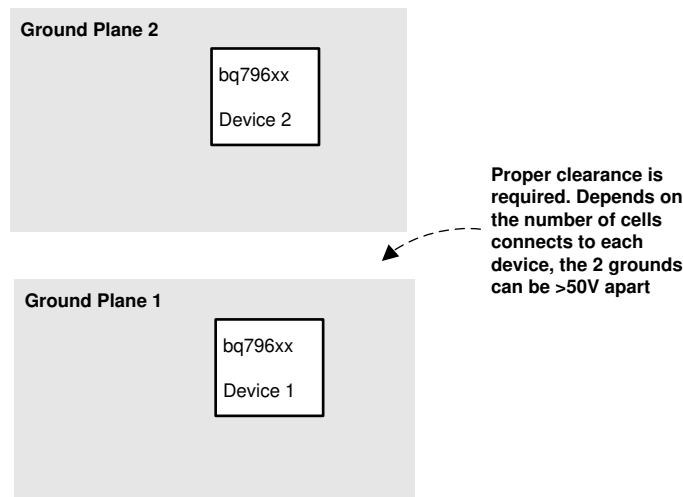


Figure 9-2. Separate Ground Planes Per Device on the Same PCB

9.1.2 Bypass Capacitors for Power Supplies and Reference

The bypass capacitors of the following pins must be placed as close to the device pins as possible to ensure proper performance, especially for the REFHP capacitor.

- REFHP, BAT, LDOIN, AVDD, DVDD, CVDD, TSREF, and NEG5V

9.1.3 Cell Voltage Sensing

Cell voltage sensing traces (VC pins and CB pins) must be placed in parallel with impedance matching. The balancing traces (CB pins) must be sized properly to carry the maximum balancing current and proper thermal performance for the application.

It is recommended to use separate cables, connect tabs, and PCB traces for the BAT pin and top VC pin connections. Same applies to AVSS and VC0 connections. This avoids the device current impact on the top and bottom cell voltage measurements.

If the same cable and connector tab is used for BAT/top VC pins connection and AVSS/VC0 pins connection, the PCB trace going to BAT/top VC pins and AVSS/VC0 pins must be separated at the connector tabs. Note the device current will still go through the cell to the PCB cable, which may introduce IR errors across the cable connection to the top and bottom cell measurements.

9.1.4 Daisy Chain Communication

It is important to have proper layout on the COMHP/N and COMLP/N circuits in order to have the best robust daisy chain communication.

- Keep differential traces as short as possible and as straight as possible. Minimize turns and avoid any looping on the traces.
- Keep the differential traces on the same layers. Run the trace in parallel with shielding and matching trace impedance.
- Place the isolation components close to the connectors.
- When using capacitive isolation, place the high-voltage capacitor of the COMxP/N pair (where x = H or L) close to each other along the parallel traces.
- Create a keep-out area (no other traces and no ground plane) around the daisy chain components in all PCB layers.

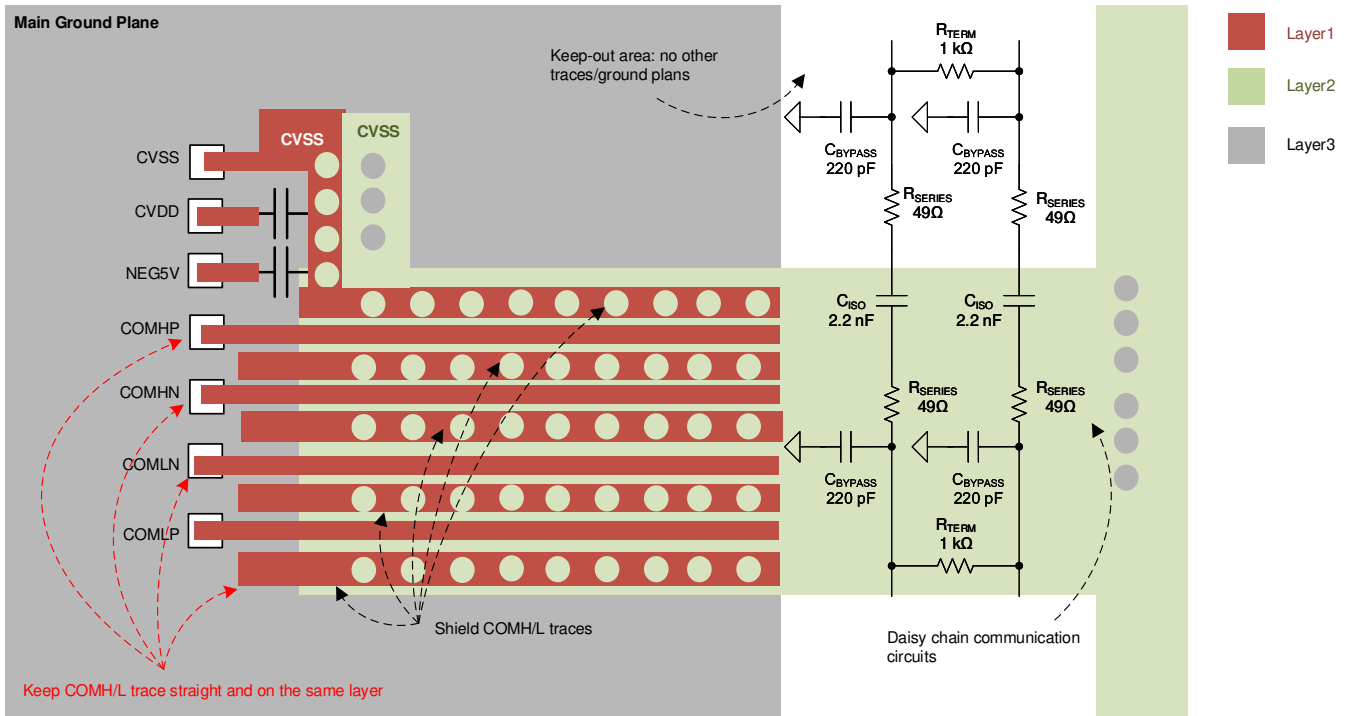


Figure 9-3. Daisy Chain Layout Consideration

9.2 Layout Example

This section presents the BQ79616 Evaluation Module (EVM) design as a layout example.

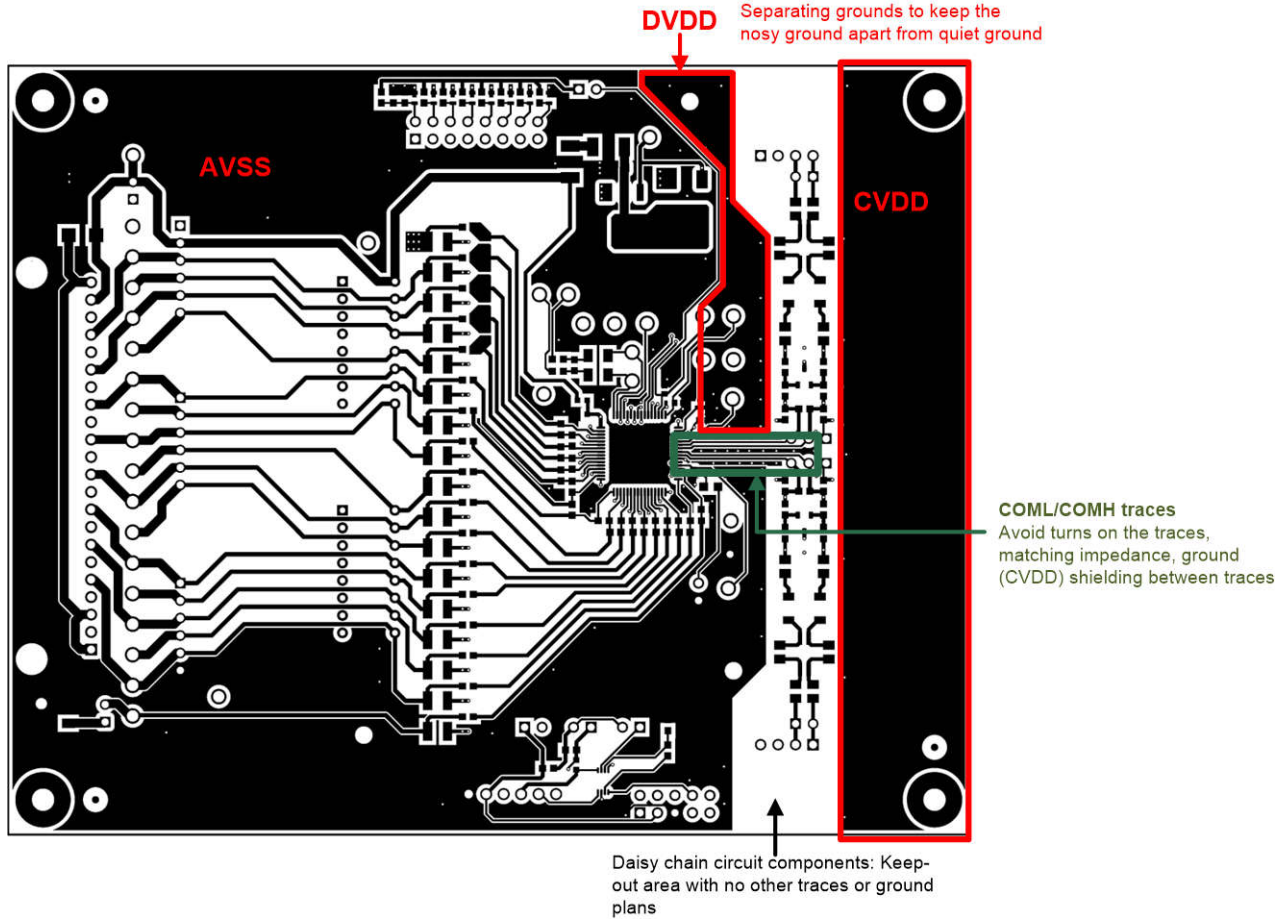


Figure 9-4. Top Signal Layer

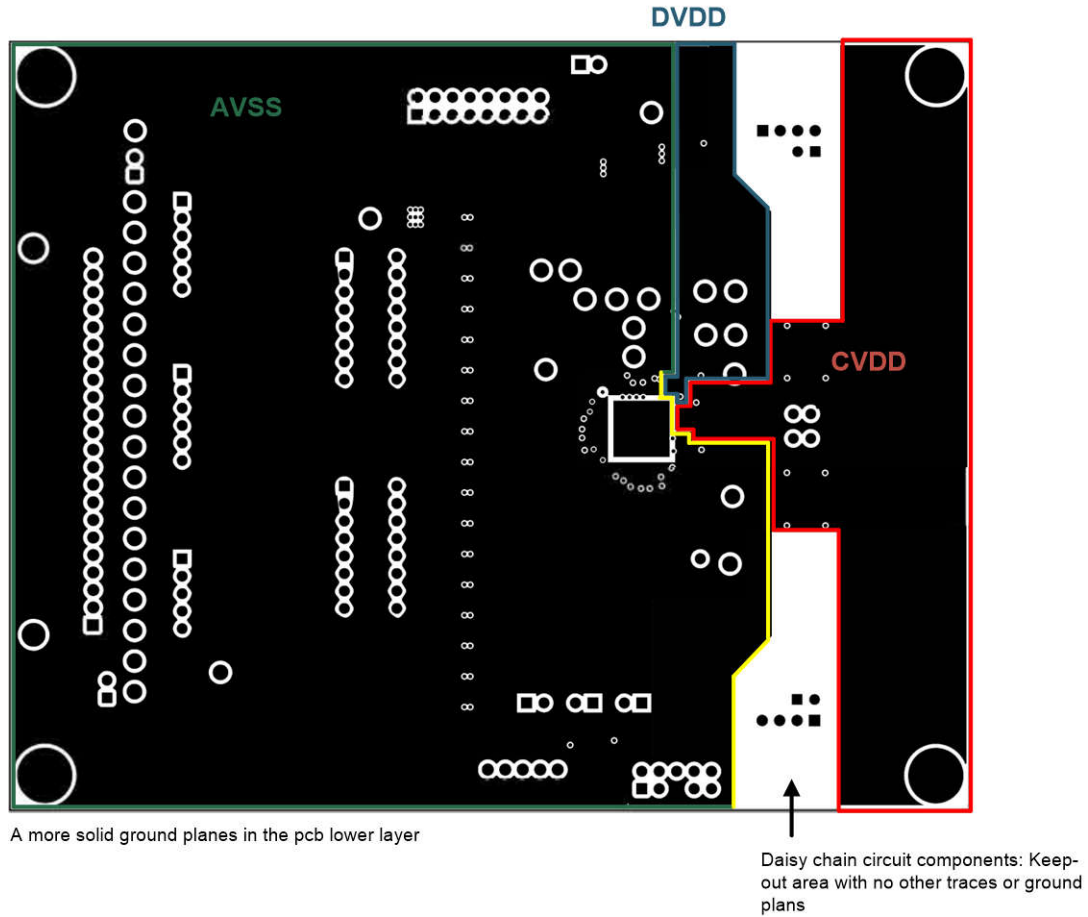


Figure 9-5. Second Layer with Solid, Separate Ground Planes

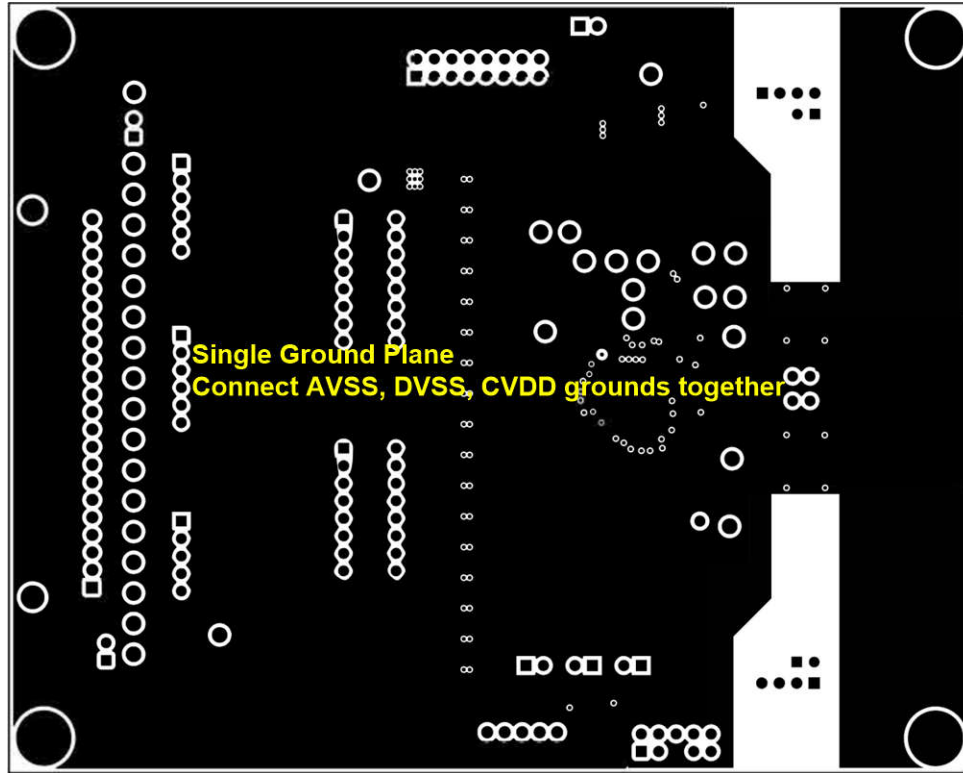


Figure 9-6. Thrd Layer with Single Ground Plane

10 Device and Documentation Support

10.1 Device Support

10.1.1 Third-Party Products Disclaimer

TI'S PUBLICATION OF INFORMATION REGARDING THIRD-PARTY PRODUCTS OR SERVICES DOES NOT CONSTITUTE AN ENDORSEMENT REGARDING THE SUITABILITY OF SUCH PRODUCTS OR SERVICES OR A WARRANTY, REPRESENTATION OR ENDORSEMENT OF SUCH PRODUCTS OR SERVICES, EITHER ALONE OR IN COMBINATION WITH ANY TI PRODUCT OR SERVICE.

10.2 Receiving Notification of Documentation Updates

To receive notification of documentation updates, navigate to the device product folder on ti.com. Click on *Notifications* 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.

10.3 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](#).

10.4 Trademarks

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

10.5 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.

10.6 Glossary

[TI Glossary](#) This glossary lists and explains terms, acronyms, and definitions.

11 Revision History

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

| Changes from Revision * (June 2023) to Revision A (June 2026) | Page |
|---|------|
| • Removed Figures (c) and (d) from Figure 7-2 | 188 |

| DATE | REVISION | NOTES |
|-----------|----------|-----------------|
| June 2023 | * | Initial Release |

12 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) |
|-----------------------------|---------------|----------------------|------------------|-----------------------|-------------|--------------------------------------|-----------------------------------|--------------|---------------------|
| BQ79616PAPR | Active | Production | HTQFP (PAP) 64 | 1000 LARGE T&R | Yes | NIPDAU | Level-3-260C-168 HR | -40 to 125 | BQ75216 |
| BQ79616PAPR.A | Active | Production | HTQFP (PAP) 64 | 1000 LARGE T&R | Yes | NIPDAU | Level-3-260C-168 HR | -40 to 125 | BQ75216 |

(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.

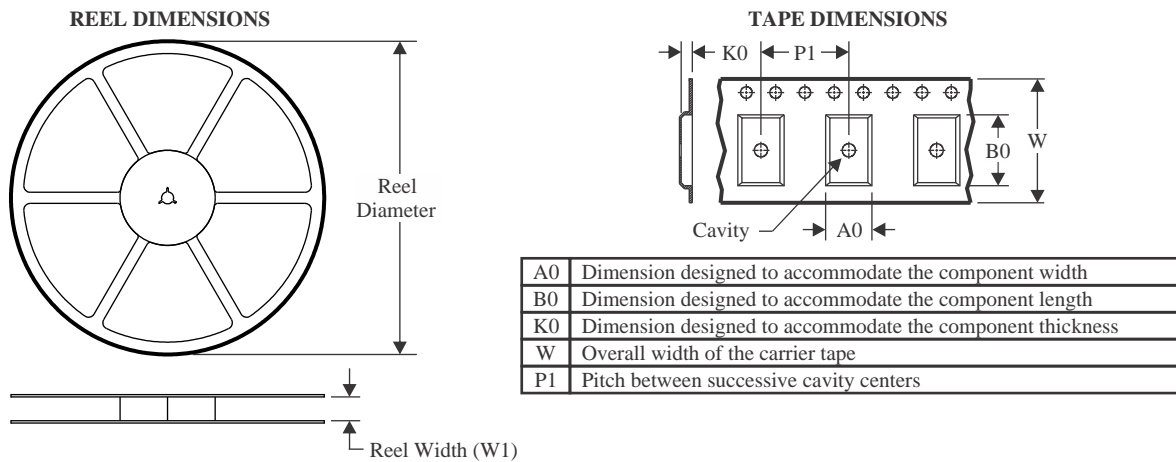
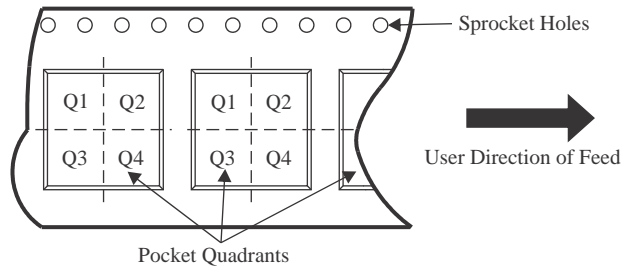
In no event shall TI's liability arising out of such information exceed the total purchase price of the TI part(s) at issue in this document sold by TI to Customer on an annual basis.

OTHER QUALIFIED VERSIONS OF BQ79616 :

- Automotive : [BQ79616-Q1](#)

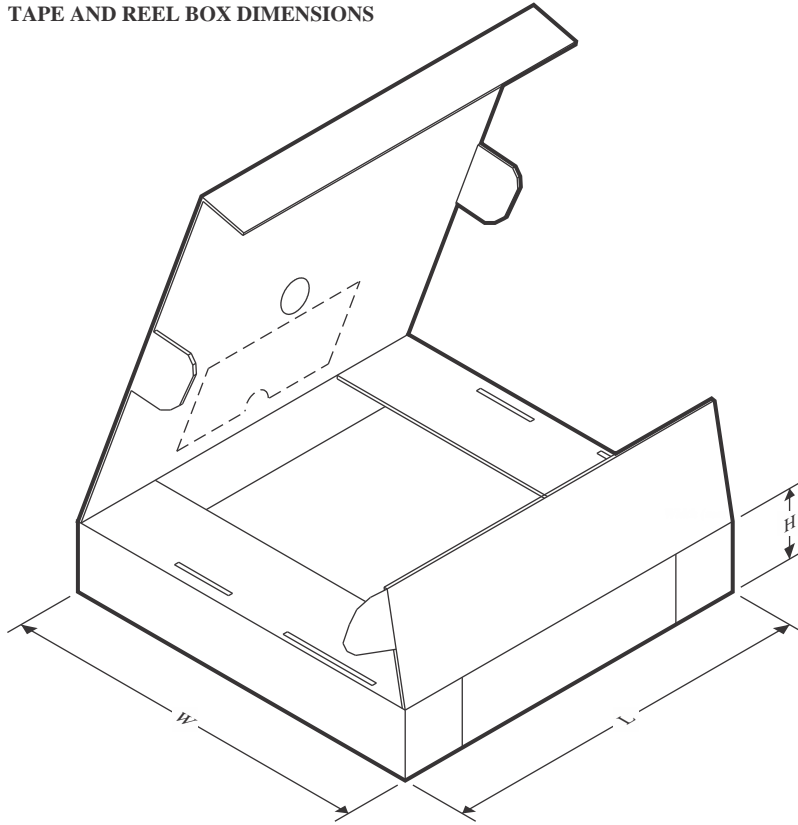
NOTE: Qualified Version Definitions:

- Automotive - Q100 devices qualified for high-reliability automotive applications targeting zero defects

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 |
|-------------|--------------|-----------------|------|------|--------------------|--------------------|---------|---------|---------|---------|--------|---------------|
| BQ79616PAPR | HTQFP | PAP | 64 | 1000 | 330.0 | 24.4 | 13.0 | 13.0 | 1.5 | 16.0 | 24.0 | Q2 |

TAPE AND REEL BOX DIMENSIONS


*All dimensions are nominal

| Device | Package Type | Package Drawing | Pins | SPQ | Length (mm) | Width (mm) | Height (mm) |
|-------------|--------------|-----------------|------|------|-------------|------------|-------------|
| BQ79616PAPR | HTQFP | PAP | 64 | 1000 | 367.0 | 367.0 | 55.0 |

GENERIC PACKAGE VIEW

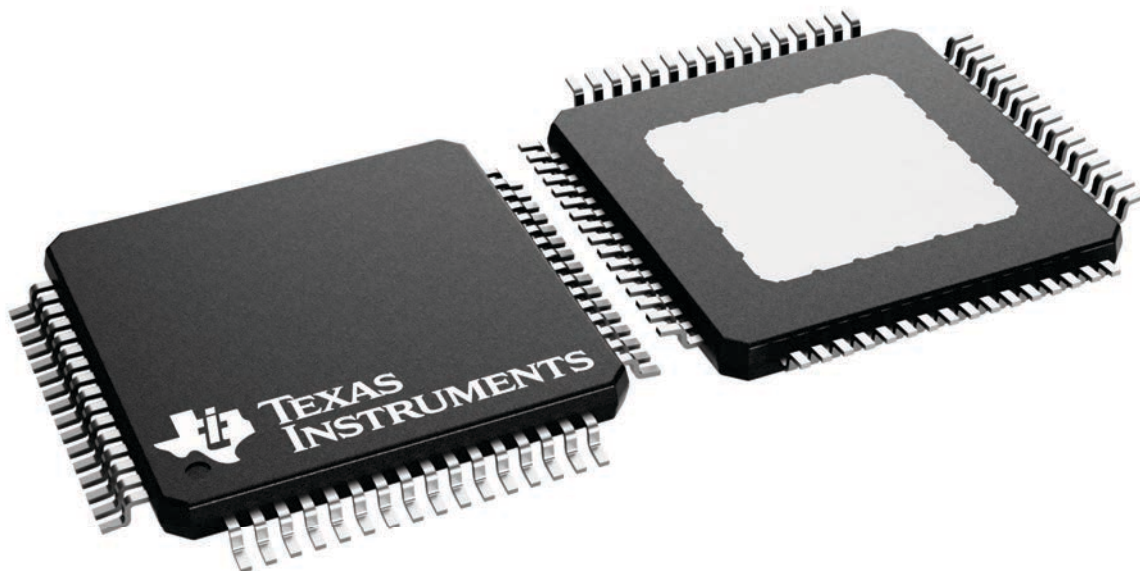
PAP 64

HTQFP - 1.2 mm max height

10 x 10, 0.5 mm pitch

QUAD FLATPACK

This image is a representation of the package family, actual package may vary.
Refer to the product data sheet for package details.



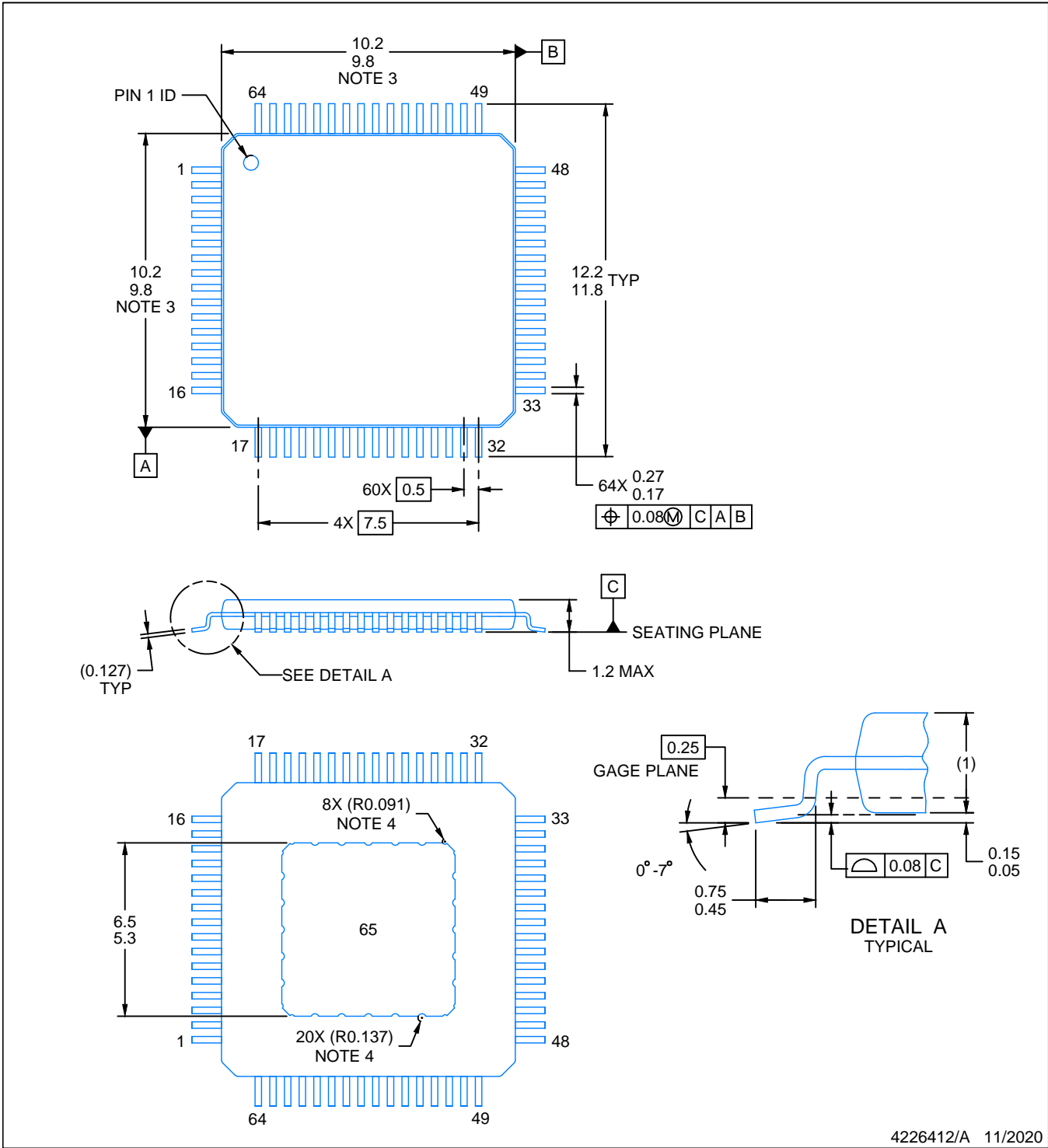
4226442/A

PACKAGE OUTLINE

PAP0064F

PowerPAD™ TQFP - 1.2 mm max height

PLASTIC QUAD FLATPACK



4226412/A 11/2020

NOTES:

PowerPAD is a trademark of Texas Instruments.

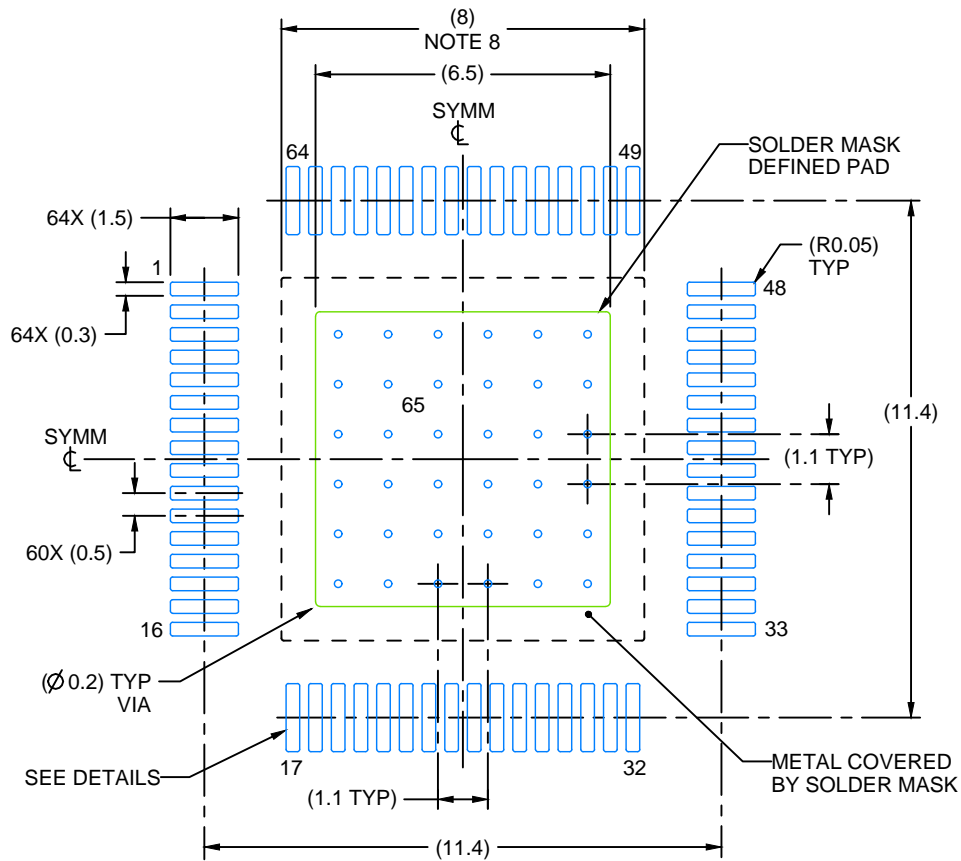
1. All linear dimensions are in millimeters. Any dimensions in parenthesis are for reference only. Dimensioning and tolerancing per ASME Y14.5M.
2. This drawing is subject to change without notice.
3. This dimension does not include mold flash, protrusions, or gate burrs.
4. Strap features may not be present.
5. Reference JEDEC registration MS-026.

EXAMPLE BOARD LAYOUT

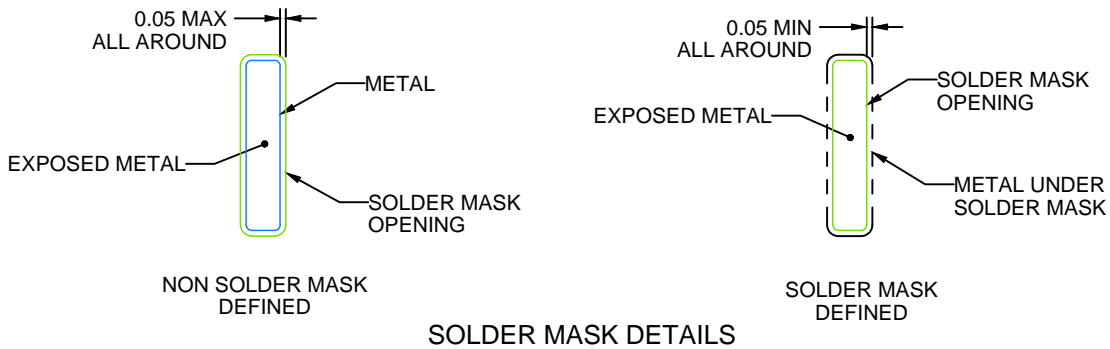
PAP0064F

PowerPAD™ TQFP - 1.2 mm max height

PLASTIC QUAD FLATPACK



LAND PATTERN EXAMPLE
EXPOSED METAL SHOWN
SCALE:6X



SOLDER MASK DETAILS

4226412/A 11/2020

NOTES: (continued)

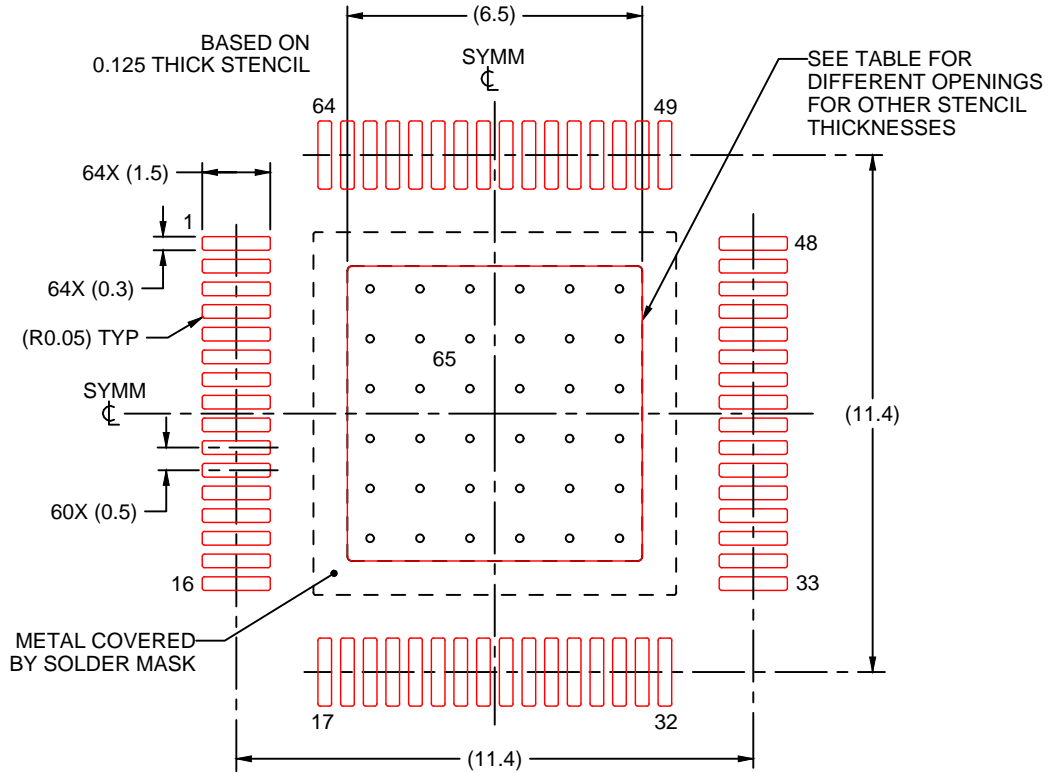
6. Publication IPC-7351 may have alternate designs.
7. Solder mask tolerances between and around signal pads can vary based on board fabrication site.
8. This package is designed to be soldered to a thermal pad on the board. See technical brief, Powerpad thermally enhanced package, Texas Instruments Literature No. SLMA002 (www.ti.com/lit/slma002) and SLMA004 (www.ti.com/lit/slma004).
9. Vias are optional depending on application, refer to device data sheet. It is recommended that vias under paste be filled, plugged or tented.
10. Size of metal pad may vary due to creepage requirement.

EXAMPLE STENCIL DESIGN

PAP0064F

PowerPAD™ TQFP - 1.2 mm max height

PLASTIC QUAD FLATPACK



SOLDER PASTE EXAMPLE
EXPOSED PAD
100% PRINTED SOLDER COVERAGE BY AREA
SCALE:6X

| STENCIL THICKNESS | SOLDER STENCIL OPENING |
|-------------------|------------------------|
| 0.1 | 7.27 X 7.27 |
| 0.125 | 6.5 X 6.5 (SHOWN) |
| 0.15 | 5.93 X 5.93 |
| 0.175 | 5.49 X 5.49 |

4226412/A 11/2020

NOTES: (continued)

11. Laser cutting apertures with trapezoidal walls and rounded corners may offer better paste release. IPC-7525 may have alternate design recommendations.
12. Board assembly site may have different recommendations for stencil design.

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