

November 15, 2019

**Status: Release**

## 1 Introduction


Analog Devices<sup>1</sup> (ADI) is a supplier of high-performance analog integrated circuits (ICs) to major OEM and tier 1 automotive customers worldwide. Our commitment to the automotive industry is underscored by our exemplary dedication to quality, reliability and customer service:

- We were among the first semiconductor companies to become certified to the TS16949 standard, a distinction we have maintained since 2003.
- We are committed to the goal of zero defects as required by automotive customers.
- We have developed proprietary automotive process flows. To learn more about our unique automotive flows, please contact the sales office in your area., see <http://www.analog.com/en/about-adi/contact-us.html>

This same focus has been applied to the recently published ISO 26262 road vehicle functional safety standard. This document is the safety manual for the LTC2949 Gas Gauge for EV and HEVs.

### Author/Approver

| Document Authors    | Role                       |
|---------------------|----------------------------|
| Christoph Schwoerer | Design Manager             |
| Alessandro Trevisan | Design Engineer            |
| Gerd Trampitsch     | Design Engineer            |
| Patrick Wilhelm     | Application Engineer       |
|                     |                            |
| Approvers           | Role                       |
| Justin Park         | Functional Safety Engineer |

<sup>1</sup> , LT, LTC, LTM, Analog Devices and the Linear logo are registered trademarks of ANALOG DEVICES CORPORATION. All other trademarks are the property of their respective owners.

## Revision History

| Previous Version | Current Version | Version Description                            | Authors                              | Date       |
|------------------|-----------------|--|--------------------------------------|------------|
| N/A              | D1              | Initial Document                               | Gerd Trampitsch, Christoph Schwoerer | 12/22/2017 |
| D1               | D2              |  | Christoph Schwoerer                  | 01/11/2018 |
| D2               | D3              |  | Christoph Schwoerer                  | 03/13/2018 |
| D3               | D4              |  | Christoph Schwoerer                  | 04/05/2018 |
| D4               | D5              |  | Christoph Schwoerer                  | 04/23/2018 |
| D5               | D6              |  | Christoph Schwoerer                  | 05/24/2018 |
| D6               | D7              | After Review with J. Park, A. Trevisan         | Christoph Schwoerer                  | 07/06/2018 |
| D7               | D8              | After Further Review with J. Park, A. Trevisan | Christoph Schwoerer                  | 08/09/2018 |
| D8               | D9              | Apps review, ATSR3 Implementation guide        | Christoph Schwoerer, Patrick Wilhelm | 6/21/2019  |
| D9               | 1.0             | Release version                                | Christoph Schwoerer, Patrick Wilhelm | 10/31/19   |

## Change Log

| Release | Section | Change Description  |
|---------|---------|---|
| D1      | All     | Initial Creation  |
| D2      | 10.3    | Corrected Numbering of FMs and SMs and Figures  |
| D2      | 4.0     | Added Die Partitioning Block Diagram  |
| D2      | 10.4    | Updated FUSA Registers Descriptions   |
| D2      | 10      | Detailed SM7  |
| D3      | 3       | Updated Block Diagram   |
| D4      | 5.3     | Added Safety Mechanisms for Latent Faults Detection in List of SM   |
| D5      | 3       | Updated Block Diagram with Functional Block Numbers (FBx)   |
|         | 5.3     | Completed Safety Mechanisms   |
| D6      | 9.2     | Detailed Over-Current Comparator Open Wire Check  |
| D6      | 9.1     | Detailed Oscillator Check   |
| D7      | 9       | Changed SM5 Naming from Data Path Check to Redundant HW Multiplier<br>Changed SM17 from Filter Test to Data Path Check and Changed description<br>Added Description of Safety Mechanisms for Latent Fault |
| D8      | 4       | Removed Section Tailoring and Work Products   |
|         | 1       | Moved Authors and Approvers to section 1  |
|         | 3       | Updated Block Diagram   |
|         | 9       | Added Description of Latent Fault Checks  |
|         | 11      | Added Metric Values   |
|         | all     | Changed Part Naming   |

|    |     |   |
|----|-----|---|
| D9 | all | <p>Added optional usage of external EEPROM</p> <p>Added ATSR3 Implementation guide</p> <p>Added AoU5</p> <p>Added note about TrigBISTs (TBs) to be executed in STANDBY mode only</p> <p>Added UserDIAG (UD) equivalent for every TrigBIST (TB) that needs to be executed within the FTTI</p> <p>Used register address naming convention from datasheet (p1.0xYY)</p> <p>Added Open-Wire detection for current inputs</p> <p>Fixed register name / address mismatches</p> <p>Adjusted register bit names to those of the datasheet</p> <p>Added description of SHORT_OCC.</p> <p>Fixed SM21 / SM22 mismatch</p> <p>Added Gain correction factors</p> <p>Added Appendix Slow Channel registers</p> <p>Added Appendix SoftBypass</p> |
|----|-----|---|

|     |     |  |
|-----|-----|--|
| 1.0 | All | <p>SM15 (2<sup>nd</sup> reference measurement) tolerance for measurement of VREF2 changed to 1.16%</p> <p>SM17 (data path check) tolerance for measurement of VREF2 via internal 250k resistor changed to 1.6%</p> <p>Resistance range of internal filter resistors set to 40Ω-100Ω</p> <p>Current range of internal pull-up / pull-down sources set to ±150 μA to ±330μA</p> <p>Small typo / spelling fixes, simplification</p> <p>Replaced 'Figure 2/2b' with latest version from datasheet</p> <p>Changed description of above to reflect new figures</p> <p>Fixed typo: ASTRx change to ATSRx</p> <p>RSL_ATSR2 bit change to RW (from RO)</p> <p>Removed ISORPT (not required for safety)</p> <p>ATSR2: '...Over Current Comparator is within ±20mV...' (not ±10mV)</p> <p>Described ATSR2 as a safety mechanism used to detect over current failure events. Changed diagnostics for ATSR2 to latent fault check</p> <p>Updated execution times of TB SM11</p> <p>Added column 'failure mode' to SM table.</p> <p>Added SM4.1, SM4.2, SM4.3 UD's to replace original SM4 TB.</p> <p>Fixed names in SM table to be consistent with titles of detailed descriptions</p> <p>Changed SM10 to AB (the original TB is replaced via UVLO checks via STATUS register)</p> <p>Changed SM16 to AB/UD (AB via TBERR, UD via TBx check, UD required to check for latent fault)</p> <p>Clear separation between internal and pin/external leakage</p> <p>Handshake Byte (SM23): Added alternative by checking for changed result.</p> <p>Marked SM23 as required for ATSR3.</p> <p>Added notes to SM table</p> <p>Updated description of HSR1-3</p> <p>SM1: added note about open wire check of current sense inputs</p> <p>Added missing absolute / relative error limits for several safety mechanisms</p> <p>Updated description of SM3, SM4, SM15, SM16 for UD implementation</p> <p>Removed ADC clock test (replaced by SM16), digital filter test (replaced by SM17)</p> <p>Updated description of SM17 to reflect implementation example</p> <p>Added missing constants (SM_AA_FP_FAC, SM_55_FP_FAC, MAF)</p> <p>Note about SM17 being superset of SM15</p> <p>SM14: Added check of gain trims. Clarify link between slow channel and 128 fast measurements.</p> <p>ATSR3 implementation guide: Correctly named SM17; note about SM23</p> <p>HWMBISTEXEC description</p> <p>ESERERRINJ: Alternative injector within safety controller recommended</p> <p>Show efficient latent fault check via setting all required injectors at the same time</p> |
|-----|-----|--|

|  |  |  |
|--|--|--|
|  |  | Added 'AoU6', 'AoU7'<br>Removed pre-release note |
|--|--|--|

Prepared for Intipower  
Analog Devices Confidential

## 2 CONTENTS

|       |   |    |
|-------|---|----|
| 1     | Introduction .....  | 1  |
| 2     | CONTENTS .....  | 6  |
| 3     | Product Overview .....  | 8  |
| 4     | Typical Usage Assumptions .....                                   | 10 |
| 5     | Assumed System Safety Goals .....                                 | 12 |
| 5.1   | Assumed Technical Safety Requirements .....                       | 12 |
| 5.2   | Assumptions of Use .....  | 12 |
| 5.3   | Safety Mechanisms .....   | 13 |
| 6     | Design Development Process .....                                  | 15 |
| 7     | Evaluation and Verification Process .....                         | 17 |
| 8     | Supporting Processes .....  | 17 |
| 9     | Architecture Diagnostics / Integration Manual .....               | 18 |
| 9.1   | Hardware Safety Requirements for ATSR1 .....                      | 18 |
| 9.1.1 | V1-V12 & VBATP, VBATM input Pins .....                            | 18 |
| 9.1.2 | AUX Multiplexer .....   | 20 |
| 9.1.3 | AUX Buffer, AUX ADC including Filter, Reference, Oscillator ..... | 20 |
| 9.1.4 | Digital Processing and Storage .....                              | 23 |
| 9.1.5 | Serial Interfaces .....   | 26 |
| 9.1.6 | Voltage Regulators .....  | 28 |
| 9.1.7 | Oscillator .....  | 28 |
| 9.2   | Hardware Safety Requirements for ATSR2 .....                      | 29 |
| 9.3   | Hardware Safety Requirements for ATSR3 .....                      | 32 |
| 9.4   | Safety Mechanisms for Latent Faults .....                         | 35 |
| 9.4.1 | Latent fault checks example .....                                 | 35 |
| 9.5   | Fusa related Registers .....                                      | 36 |
| 9.6   | ATSR3 Implementation guide .....                                  | 39 |
| 9.6.1 | 4 FSSHT SPI transactions .....                                    | 40 |
| 9.6.2 | Cycling between FCM and open wire check and FSSHTs .....          | 42 |
| 9.6.3 | Checks for latent faults .....                                    | 43 |
| 9.6.4 | Expected error on coulomb counting .....                          | 45 |
| 9.7   | Appendix Slow Channel registers .....                             | 46 |
| 9.8   | Appendix SoftBypass .....   | 46 |

|      |   |    |
|------|---|----|
| 10   | Dedicated Measures – AEC-Q100 Qualification and Automotive Flow ..... | 49 |
| 11   | Quantitative Analysis Results .....                                   | 49 |
| 11.1 | Metric Calculations .....   | 49 |
| 12   | Functional Safety Considerations .....                                | 50 |

Prepared for Infypower  
Analog Devices Confidential

### 3 Product Overview

The LTC2949 is a high precision charge and energy meter for electrical and hybrid vehicles. It infers charge and energy flowing in and out of the battery pack by monitoring simultaneously the voltage drop over up to two sense resistors and the battery pack voltage. Low-offset  $\Delta\Sigma$  ADCs ensure accurate measurement of voltage and current with insignificant power loss. The LTC2949 uses instantaneous multiplication of voltage and current at a high sampling rate to infer accurately power even in presence of fast load variations. Continuous integration of current and power ensures lossless tracking of charge and energy delivered or received by the battery pack.

All measured quantities (voltages, currents, charge, power, energy and temperature) are stored in internal registers accessible via the onboard SPI interface. The LTC2949 features programmable high and low thresholds for all measured quantities to reduce digital traffic with the host. The LTC2949 can supervise state and on resistance of contactors, measure remote temperatures using NTCs, and track the isolation resistance of the battery pack to chassis ground.

In sleep mode, current consumption is reduced to around 10 $\mu$ A<sup>1</sup>. The LTC2949 can be powered directly from the battery or from an isolated supply. The built in SPI interface can be configured for isolated high speed, RF-immune, long distance communication which allows fully transformer isolated operation of the LTC2949.

The LTC2949 was designed to work with Linear Technology's LTC68xx Multicell Battery Monitors and includes numerous self-checking features for ensuring its proper operation.

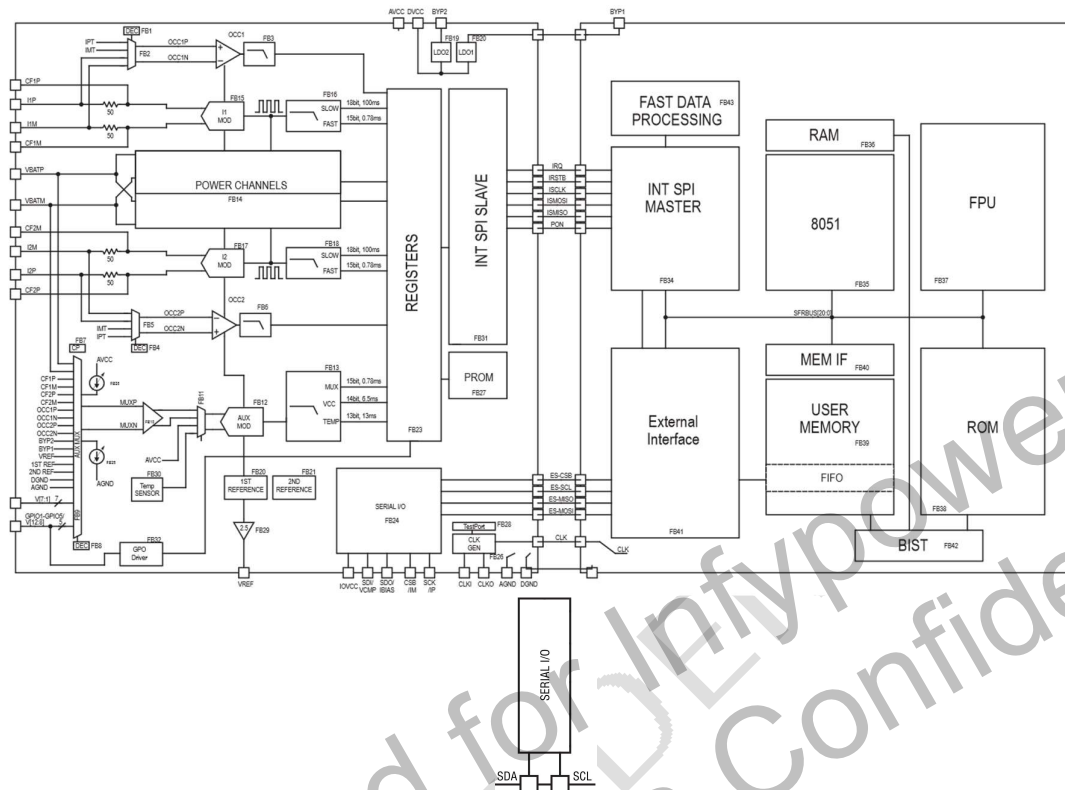
Additional features include fast modes for each ADC, a precision 3V reference voltage output, up to 12 auxiliary voltage sensing inputs and 5 general purpose I/O lines.

Figure 1 shows the major functional blocks of the LTC2949

---

<sup>1</sup> All datasheet parameters are preliminary and subject to change.





**Figure 1: Block Diagram of the LTC2949**

Two differential data acquisition channels (I1P, I1M and I2P, I2M) can be used to measure the voltage drop over sense resistors for determination of current. A third and fourth acquisition channel using inputs I1P, I1M and I2P, I2M as well as VBATP, VBATM can be used to measure either input power or only voltage between VBATP and VBATM. A fifth acquisition channel allows voltage measurements between VBATP and VBATM, any combination of V1 to V12, as well as other input signals for diagnostic by means of a Multiplexer (AUX MUX). The last measurement channel, referred to as AUX Channel, consists of AUX MUX, AUX Buffer, and AUX Modulator. ADC results are processed by an internal ALU to compensate for external resistor divider ratio imperfections and then stored in RAM accessible by the user via a serial interface. An external non-volatile EEPROM can be connected to an I2C interface to allow storage of those gain correction factors.

The LTC2949 is realized as a two-die solution with an analog front-end die (also referred as analog die throughout this document) and a digital die.

The analog front-end contains circuitry to convert analog input signals to digital values and store them. Furthermore, it contains a one-time programmable fuse PROM to store trim values determined and stored at final test. Also, the power supply generation, the isoSPI/SPI interface and the GPO Control is realized on the analog front-end die.

The analog and the digital dies communicate via an internal SPI interface. The digital die contains an 8051 microcontroller, a program ROM containing firmware, a floating-point unit as well as a RAM accessible to the user via the external interface.

## 4 Typical Usage Assumptions

The LTC2949 is a Commercial Off-The-Shelf (COTS) product, designed to be part of our broad product catalog. The LTC2949 was developed with ISO 26262 in mind as a Safety Element out of Context (SEooC).

The assumed typical use of the LTC2949 is a battery data acquisition system in Electric and Hybrid Electric Vehicles (EV/HEV). Figure 2 is a high-level schematic view of such an application.

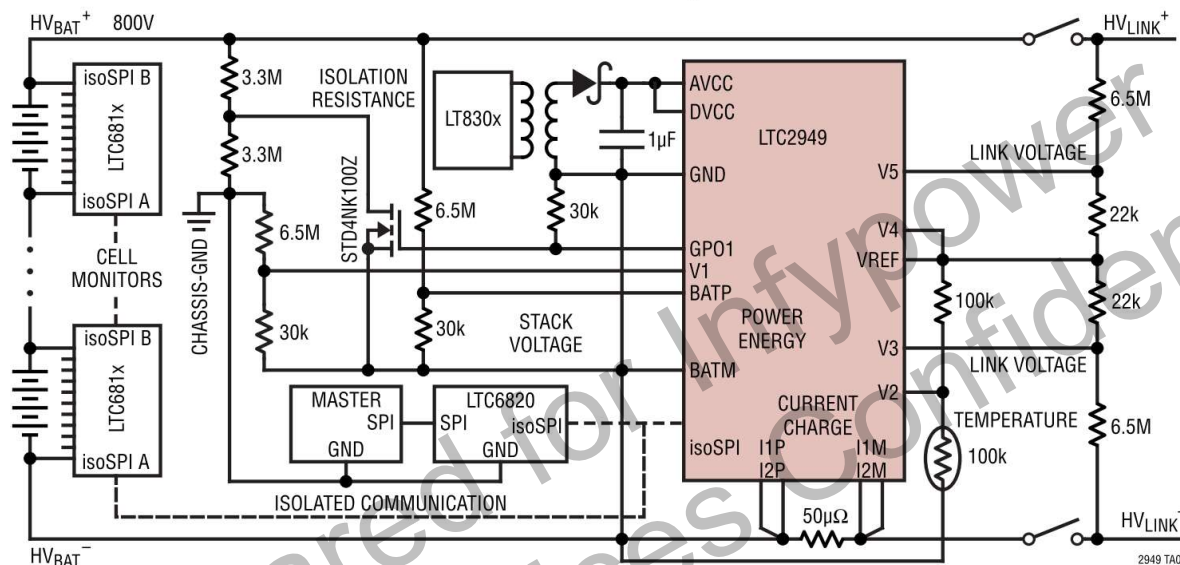


Figure 2: Schematic of a typical application using the LTC2949

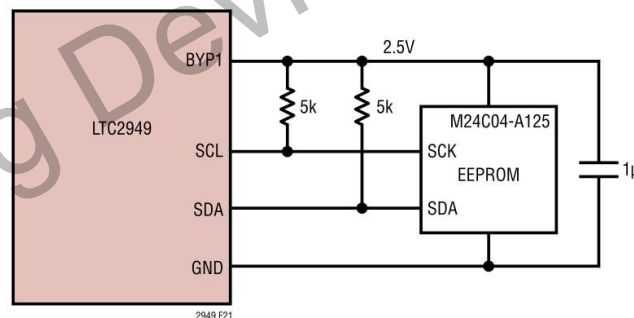


Figure 2b: Optional external EEPROM can be connected to the LTC2949 to store gain correction factors

The LTC2949 is powered from one of LT830x series isolated flyback converter and measures the voltage drop on one (or two in series) current sense resistors located at one battery terminal simultaneously by two ADCs. Alternatively, two sense resistors, measuring the current into two different paths, can be connected to the two current sense inputs.

Additionally, it measures the total battery stack voltage via the resistive divider connected to VBATM and VBATP and various other signals in the system, such as contactor voltages or NTC temperatures, via auxiliary inputs V1-V12. An optional 4MHz quartz oscillator provides the time reference to accurately calculate the charge and energy flowing in and out of the battery. A SPI or isoSPI (isolated SPI via single transformer coupled differential pair) communication enables

reliable communication to a microcontroller, even in the presence of large and fast varying potential differences of several kV. The LTC2949 can be connected in parallel with the isoSPI bus of cell monitors (LTC681x / ADBMS68xx) or on top of a daisy chain of cell monitors. Calibration parameters of the board, such as current sense resistor and resistive divider gain trims, can be stored to an external EEPROM and can be read after power-up.

## 5 Assumed System Safety Goals

A battery monitoring system (BMS) is the assumed top-level item where safety goals are defined. It is assumed that the BMS utilizes the LTC2949 to measure voltages and currents and detects over-current failure events to achieve its safety goal(s).

The LTC2949 is an element of a BMS; its functional safety requirements are determined by the BMS. If proper operation of the LTC2949 is not required for the system to achieve a safety goal, then the LTC2949 is not a safety element and the requirements of ISO 26262 do not apply.

If proper operation of the LTC2949 is required for the system to meet a safety goal, the ASIL of the system will cascade down to the LTC2949. The LTC2949 was designed, developed, produced and tested for ISO 26262 compliant systems.

### 5.1 ASSUMED TECHNICAL SAFETY REQUIREMENTS

From the typical usage assumptions on safety critical measurements the following Technical Safety Requirements (ATSR) are deduced:

- **The Assumed Technical Safety Requirement ATSR1 is to ensure that the voltage measurement error of Auxiliary Data Acquisition Channel (AUX Channel) in fast mode should be smaller than  $\pm 50\text{mV}$  or  $\pm 2\%$ , whichever is larger.**
- **The Assumed Technical Safety Requirement ATSR2 is to ensure that an over current failure event is detected by means of the Over-Current-Comparator within  $\pm 20\text{mV}$  of the programmed threshold and an in case of an exceeding input signal the heartbeat at GPO5 is stopped.**
- **The Assumed Technical Safety Requirement ATSR3 is to ensure that the current measurements in fast mode are within a minimum accuracy of  $\pm 200\mu\text{V}$  or  $\pm 2\%$ , whichever is larger, if both current measurement channels are connected to the same sense resistor or to two sense resistors in series.**

### 5.2 ASSUMPTIONS OF USE

AoU1: All inputs and outputs of the LTC2949 are used within operating range as specified in the LTC2949 datasheet.

AoU2: The listed safety mechanism will be used at least once within each FTTI or MPFDI defined in the item level.

AoU3: The differential input voltage to the Auxiliary Measurement Channel is derived from a differential source (e.g. resistor divider) with a differential output impedance smaller than  $100\text{k}\Omega$  to achieve ATSR1

AoU4: If 2 sense resistors in series are used for ATSR3, the common node is connected to IM1 and IM2.

AoU5: If an external EEPROM is used to store gain correction factors, it is supplied via BYP1 and connected to SDA, SCL with  $4.7\text{k}\Omega$ - $10\text{k}\Omega$  pull-up resistors connected to BYP1. If no EEPROM

is used, SDA must be pulled high to BYP1 via a 4.7k-10k resistor to ensure the HWBIST is executed on power-up (see bit HWBIST in FAULTS register and HWMBISTEXEC in EXTFaults registers).

AoU6: Any data written to the LTC2949 (incl. values written via SoftBypass) must be read back to verify data was written correctly.

AoU7: For ATSR2, a single shunt resistor connected to both current sense inputs shall be used and both OCC control registers (OCCxCTRL) shall be configured the same.

### 5.3 SAFETY MECHANISMS

The LTC2949 has several safety mechanisms which are categorized into one of the followings:

- AutoBIST (AB): Built-in self-test transparent to the user, result bits must be checked
- TrigBIST (TB): Built-in self-test triggered by the user, result bits must be checked
- UserDIAG (UD): Diagnostic triggered by the user, measurement results must be analyzed

Table below summarizes the safety mechanisms implemented in the LTC2949.

| #     | Description  | ATSR  | Type  | Failure mode |
|-------|--|-------|-------|--------------|
| SM1   | Open Wire Check                                    | 1,2,3 | UD    | SPF / LF (4) |
| SM2   | Leakage Current Check via 2 <sup>nd</sup> Pin Pair | 1     | UD    | SPF          |
| SM3   | Reverse Polarity Check                             | 1     | UD    | SPF          |
| SM4   | Accuracy Check                                     | 1,3   | UD    | SPF / LF (5) |
| SM4.1 | Internal Leakage Current Check                     | 1,3   | UD    | SPF / LF (5) |
| SM4.2 | Offset Check                                       | 1,3   | UD    | SPF / LF (5) |
| SM4.3 | INL Check  | 1     | UD    | SPF          |
| SM5   | Redundant Multiplier                               | 1     | AB    | SPF          |
| SM6   | Fuse PROM Error Check                              | 1,2,3 | AB    | SPF / LF (4) |
| SM7   | Memory Check                                       | 1,2,3 | TB/AB | SPF / LF (4) |
| SM8   | Internal Communication Check                       | 1,2,3 | AB    | SPF / LF (4) |
| SM9   | External Communication Check                       | 1,2,3 | UD/AB | SPF / LF (4) |
| SM10  | Voltage Regulator Check                            | 1,2,3 | AB    | SPF / LF (4) |
| SM11  | Over Current Comparator Check                      | 2     | TB    | LF           |
| SM12  | Over Current Comparator Open Wire Check            | 2     | UD    | LF           |
| SM13  | Redundant Current Measurement                      | 3     | UD    | SPF          |
| SM14  | Redundant Current Computation                      | 3     | UD    | SPF          |
| SM15  | 2nd Reference Check                                | 1,2,3 | UD    | SPF / LF (4) |
| SM16  | Oscillator Test                                    | 1,3   | AB/UD | SPF          |
| SM17  | Data Path Check                                    | 1,3   | UD    | SPF          |
| SM18  | Fault Injection to Verify SM6 for Latent Fault     | 1,2,3 | TB    | LF           |
| SM19  | Fault Injection to Verify SM8 for Latent Fault     | 1,2,3 | TB    | LF           |
| SM20  | Fault Injection to Verify SM9 for Latent Fault     | 1,2,3 | TB    | LF           |
| SM21  | Fault Injection to Verify SM5 for Latent Fault     | 1     | TB    | LF           |
| SM22  | Fault Injection to Verify SM7 for Latent Fault     | 1,2,3 | TB    | LF           |
| SM23  | RDCV Hand-Shake Check                              | 1,3   | UD    | SPF          |

|      |                       |     |    |     |
|------|-----------------------|-----|----|-----|
| SM24 | External EEPROM Check | 1,3 | TB | SPF |
|------|-----------------------|-----|----|-----|

## Notes:

1. SPF: Single point fault; LF: Latent fault
2. All TB type safety mechanisms must be executed in STANDBY state only which means they are typically executed once at start-up. An example is that the external EEPROM is read only once at power-up.
3. SM11, SM18-SM22 are latent fault checks, executed within the MPFDI, which is typically once after power-up / start-up.
4. For ATSR2, the safety mechanisms SM1, SM6-SM10, SM15 are latent fault checks, executed within the MPFDI, which is typically once after power-up / start-up.
5. For ATSR3, all SM4.x safety mechanisms are latent fault checks of the AUX ADC which serves as a safety mechanism for the current ADCs. Those SMs can be executed within the MPFDI, which is typically once after power-up / start-up.



## 6 Design Development Process

ADI's quality management system for the development and manufacture of high-performance integrated circuits is certified by an accredited registrar to be in accordance with ISO/TS 16949 and ISO 9001. ADI's design Product Development Process (design PDP) is a series of five distinct phases that provide structure and review milestones for the development activities. The design PDP also describes methods in which the design tools are verified. Detailed below are the phases and key milestones of the process:

### 1. DEFINITION PHASE Identify and develop the new product concept

- Define key features and specifications
- Identify required design and development resources
- Develop detailed schedule

### 2. DESIGN PHASE

- Schematic design
- ESD plan approval
- Chip design review (schematics only)
- Test inputs review
- Product review meeting
- Intellectual property (IP) review

### 3. MASK DESIGN PHASE

- Mask design
- Layout review

### 4. DEVELOPMENT PHASE

- Bench evaluation
- Correlation of ATE test measurements
- Characterization
- Qualification tests
- Complete initial stocking quantities

### 5. RELEASE PHASE

- Released products listing launch meeting
- Prepare press release and marketing material
- With the adoption of the ISO 26262 functional safety standard, ADI has included the following items in the design PDP for the LTC2949:
  - Established a functional safety team and safety manager
  - Incorporated safety critical items into the ADI product tracking system (PTS) such that the PTS is the safety plan
  - Tailoring analysis of ISO 26262 to determine which deliverables are applicable to the LTC2949
  - Systems level safety usage assumptions

- Failure modes effects and diagnostic analysis (FMEDA)

Prepared for Intipower  
Analog Devices Confidential



## 7 Evaluation and Verification Process

The evaluation of any ADI device (hardware unit, SEooC) follows a comprehensive evaluation plan. The design PDP details how to develop an evaluation plan, design and process FMEAs, process capability studies and copies of our quality certifications which demonstrate an effective quality system.

The evaluation process is described in the design PDP. As appropriate, safety mechanisms are verified by placing the device in various test modes. The inclusion of test modes is incorporated in the design phase of the device. Documentation of findings and methods of evaluation are maintained during the process.

Evaluation review meetings are periodically scheduled to review any findings. An errata list and revision list are maintained throughout the evaluation process. A completed evaluation checklist and summary is a work product of the evaluation process.

For allowing the LTC2949 to be used in ISO 26262 compliant systems, testing of safety related functions is summarized in the LTC2949 Verification and Validation Report.

## 8 Supporting Processes

ADI designs commercial off-the-shelf (COTS) high-performance linear integrated circuits. Within the scope of ISO 26262, these are hardware unit safety element out of context (SEooC). The safety-related characteristics of ADI devices are defined in ADI's design PDP and ensured in manufacture and test through our quality systems.

Safety requirement inputs for test hardware and test program development are outputs of the design PDP. All electrical testing requirements are determined in the design PDP and are an input into Test Engineering's test Product Development Process (test PDP). Test hardware development and qualification are accomplished per the test PDP.

Production verification measures and acceptance criteria are based on the outputs from the design PDP. All supporting process requirements are implemented and maintained as part of ADI's ISO/TS 16949 certified quality system.

ADI is ISO/TS 16949 certified in the design, manufacture, and test of high-performance integrated circuits. Included in our Production Part Approval Process (PPAP) documentation are design records, control plans, design and process FMEAs, process capability studies and copies of our quality certifications which demonstrate an effective quality system.

## 9 Architecture Diagnostics / Integration Manual

This section provides a description of functional blocks shown in Figure 1 relevant to meet the assumed functional technical safety requirements (ATSRs). For each block there are Hardware Safety Requirements (HSR) needed to meet the ATSRs. Each HSR results in a diagnostic, or safety mechanism (SM). This section also describes how to use the diagnostics.

Some of the safety mechanisms are fully self-contained while others require some processing of measurements in the host controller. All registers involved in fault reporting and controlling safety mechanisms are described at the end of this chapter. Besides some special registers like ISO0-ISO2, register description can also be found in the latest LTC2949 datasheet. Register addresses are given in the same naming convention as in the datasheet, p1.0xYY for registers on PAGE1, 0xYY for registers on PAGE0.

### 9.1 HARDWARE SAFETY REQUIREMENTS FOR ATSR1

The Assumed Technical Safety Requirement ATSR1 is to ensure that the voltage measurement error of Auxiliary Data Acquisition Channel (AUX Channel) in fast mode should be smaller than  $\pm 50\text{mV}$  or  $\pm 2\%$ , whichever is larger. The AUX Channel is composed of Input Pins, AUX Multiplexer, AUX Buffer, AUX Modulator, Reference, Oscillator, Digital Filters, Digital Gain Correction and Result Registers.

#### 9.1.1 V1-V12 & VBATP, VBATM input Pins

**HSR1: The LTC2949 shall allow to detect open wires at Pins connected to Aux Multiplexer Input**

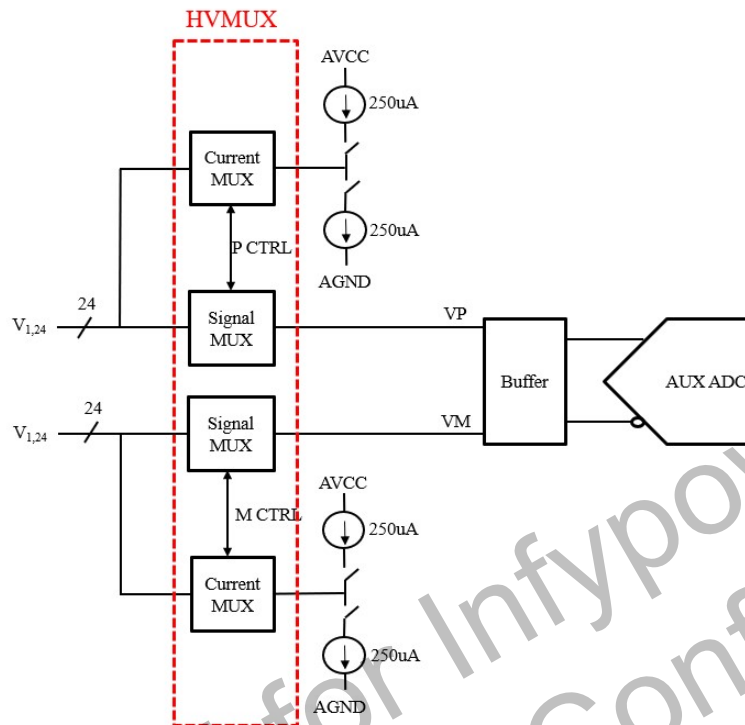
**HSR2: The LTC2949 shall allow to detect leakage of pins causing measurement errors of more than  $\pm 50\text{mV}$  or  $\pm 2\%$ , whichever is larger**

HSR1 is fulfilled by the Open Wire Check (OWC) and HSR2 by measuring the same voltage through another separate input pair.

#### Open Wire Check (SM1)

The AUX-MUX (FB9) can connect all analog input pins to the input of the AUX Buffer. To confirm that all pins are connected to the multiplexer input, an Open Wire Check has been designed into the LTC2949 that uses onboard programmable pull-up and pull-down current sources ( $\pm 250\mu\text{A}$ ) as shown in Figure 3. For open wire detection, first take measurement on a specific channel, selected by the Fast Channel Multiplexer Control Registers at FAMUXN and FAMUXP, then re-run this measurement while pull up current source is enabled and verify the excursion corresponding to the pull up current. Then disable the pull up current source and verify with a subsequent measurement that voltage returns to initial value. Finally repeat this procedure with the pull-down current source enabled and compare results of the measurements. Current sources are controlled via control bits in the FCURGPIOCTRL register.

Note that a write to the FCURGPIOCTRL register does not take effect immediately. Instead any changes will only become active once FGPIOTCTRL is written. Thus, it is recommended to always write both registers in a single burst.



**Figure 3: Open Wire Detection Circuitry.**

Depending on the external connection not all described measurements with pull-up and pull-down current sources are necessary and different combinations of enable pull-up / pull-down current sources on the positive and negative MUX inputs are possible. The AUX-MUX can also be connected to CF1P, CF1M, CF2P and CF2M which allows to do an open-wire check on the current sense resistor inputs. Current source and MUX share the same control line which means that current can be only switched to the currently selected MUX input.

Thresholds for this check depend on the external source impedance and on the variation of the current sources. In case of open wire check at the current sense inputs, also on the variation of the internal  $50\Omega$  filter resistors. For the absolute values of the current sources, a variation of  $150\mu A$  to  $330\mu A$  and for the internal filter resistors  $40\Omega$  to  $100\Omega$  shall be assumed.

The current sources can be diagnosed by measuring the voltage drop across an external  $4k$  resistor connected between VREF (3V reference output voltage) and one of V1 ... V12 with the AUX ADC, leading to a nominal differential voltage of  $\pm 1V$ . Other external sources with known impedance can also be used, as long as  $V_{SRC} \pm 250\mu A \cdot R_{SRC}$  is between  $1V$  and  $4V$  and the supply voltage of LTC2949 is at least  $3V$  above that voltage ( $V_{SRC}$  source voltage,  $R_{SRC}$  source resistance).

**Leakage Current Check via 2<sup>nd</sup> Pin Pair (SM2)**

Leakage current of input pins can impact measurement result from high impedance sources, like resistive dividers. By adding an external series resistance in front of the pins, leakage current of pins can be detected by measuring the same input on 2 different pin pairs and comparing the results. Having two redundant sensors, e.g. two NTCs for temperature measurements connected to two different inputs out of V1 to V12, allows to detect leakage errors in the pins and the sensors.

**9.1.2 AUX Multiplexer**

**HSR3: The LTC2949 shall allow to detect if multiplexer channels are skipped, or multiple channels measured simultaneously.**

HSR3 can be verified by Reverse Polarity Measurement, Open Wire Check and Leakage Current Check

**Reverse Polarity Check (SM3)**

Any input of the MUX can be switched to either the positive or negative input of the AUX Buffer by the Fast Channel Multiplexer Control Registers (FAMUXN, FAMUXP) allowing to perform redundant measurements with opposite polarity to ensure the proper operation of the MUX and ADC. If the MUX works properly the two measurement results only differ in sign and offset if the input signal is stable during the 2 measurements.

To cope with linearly varying input signals, instead of making only two measurements it is possible to perform three successive measurements with normal sign, inverted sign and again normal sign (+, -, +) and compare the average of the two non-inverted measurements with the inverted value of the inverted measurement. A fault shall be reported if the measurements differ by more than 37.5mV. However, this check can still fail if the input signal varies in a non-linear way between the first and the last measurement.

The reverse polarity measurement can be implemented as a UD by triggering two or three fast single shot AUX measurements that are compared by the host controller.

**9.1.3 AUX Buffer, AUX ADC including Filter, Reference, Oscillator**

**HSR4: Minimum accuracy of AUX input measurements in fast mode shall be verifiable through diagnostics.**

Accuracy of AUX input measurements can be impaired by leakage in front of the input buffers, offset of the input buffer and the ADC gain error, noise or INL of the ADC as well as an error in the ADC clock frequency. HSR4 can be satisfied by several safety mechanisms grouped within the Accuracy Check (SM4) and by SM15 and SM17 that ensure accuracy of the AUX channel in fast conversion mode.

**Accuracy Check (SM4)**

The Accuracy Check consists of following UD safety mechanisms (SM4.1- SM4.3)

**Internal Leakage Current Check (SM4.1)**

Internal leakage currents at the input of the AUX Buffer are measured through an internal 250kOhm resistor on MUX input 23. The test is done by measuring with MUXP set to 22 and MUXN to 23.

Leakage current on MUXN will flow through the 250kOhm resistor and cause a non-zero differential voltage. The test is repeated with MUXP set to 23 and MUXN set to 22 for detecting leakage current on MUXP. An Error shall be flagged if the leakage current is above 40nA (10mV).

For efficient integration of SM4.1 into SM17, three separate measurements  $A = V_{22} - GND$ ,  $B = GND - V_{23}$  and  $C = V_{23} - GND$  can be performed and the leakage error can be calculated by  $A+B$  and  $A-B$ .

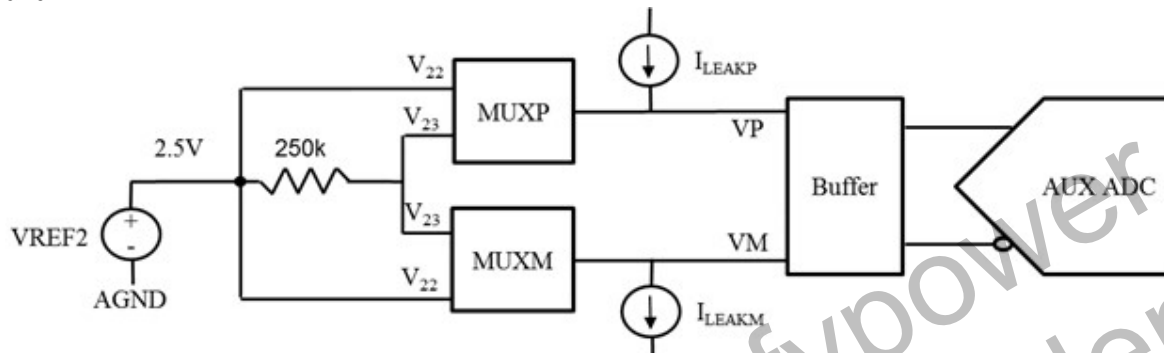


Figure 4: Leakage Detection Circuitry

This safety mechanism is implemented as a UD that makes 2 fast single shot AUX measurements.

#### Offset Check (SM4.2)

AGND vs AGND is measured to obtain offset of the Aux Channel and verified to be smaller than 37.5mV. Alternatively, offset can also be measured via VREF2 vs. VREF2 (MUX setting 22 for FAMUXP und FAMUXN).

This safety mechanism is implemented as a UD that makes 1 or 2 fast single shot AUX measurement.

For ATSR3, offset errors > 37.5mV are detected intrinsically by SM17, see above note about integrating SM4.1 into SM17.

#### INL Check (SM4.3)

VREF is a multiple of the internal 1<sup>st</sup> reference. Measuring VREF allows to verify Integral Non-Linearity of the AUX ADC by means of a 3-point measurement of GND-GND, VREF2-GND, VREF-GND. VREF must be connected to one of the V1-V12 inputs for this safety mechanism. Any other known external voltage can also be used for this check. An error is detected if the INL is larger than 25mV.

This safety mechanism is implemented as a UD that makes 3 fast single shot AUX measurements.

**Summary**

- 1) Measure offset error via
  - a)  $\text{auxOffset} = \text{AUX ADC measurement with MUX set to GND (0) vs. GND (0)}$
- 2) Measure gain error via
  - a)  $\text{vrefvx} = \text{AUX ADC measurement with MUX set to Vx (x) vs. GND (0), where Vx is connected to VREF pin.}$
  - b)  $\text{gain} = 3.0\text{V} / (\text{vrefvx} - \text{auxOffset})$
- 3) Calculate INL error via
  - a)  $\text{vref2} = \text{AUX ADC measurement with MUX set to VREF2 (22) vs. GND (0)}$
  - b)  $\text{inl} = (\text{vref2} - \text{auxOffset}) * \text{gain} - \text{VREF2\_NOMINAL (2.39V)}$
  - c) Check that  $|\text{inl}| < 25\text{mV}$

**2nd Reference Check (SM15)**

The gain of AUX Measurement is verified by measuring the voltage of a 2<sup>nd</sup> reference block (VREF2) provided for redundancy. To avoid common cause failures, the two references use completely different architectures. The core circuitry of the 1<sup>st</sup> reference is built by NPN transistors whereas the 2<sup>nd</sup> reference is based on a PNP core. An error is detected, if the measurement of the 2<sup>nd</sup> reference deviates from the nominal value by more than 1.16%. For the VREF2 measurement, the AUX multiplexer shall be set to either VREF2 (FAMUXP = 22) – GND (FAMUXN = 0) or the reverse polarity GND (FAMUXP = 0) – VREF2 (FAMUXN = 22).

ADC noise is tested by measuring the 2<sup>nd</sup> reference 3 times and verifying that the results vary by **3 LSB max**.

For ATSR3, the test for noise errors of the AUX ADC is a latent fault check done intrinsically over time by all SMs for ATSR3 that use the AUX ADC. Noise of the current ADC is tested by the redundant current measurements (SM13).

Note: SM17 is a superset of SM15 and can replace SM15.



#### 9.1.4 Digital Processing and Storage

**HSR5: The digital controlling, processing and storage shall be testable in the item to ensure they are fault free.**

HSR5 is achieved by a DSP data path check, memory checks and verifying the Hand-Shake byte of the RDCV command.

##### Data Path Check (SM17)

The conversion result of the AUX Channel in fast mode is multiplied by device-specific trim parameters and user programmable gain correction values at p1.0xB0 to p1.0xCF (e.g. the BAT gain correction register BATGC at address p1.0xB9) as shown below.

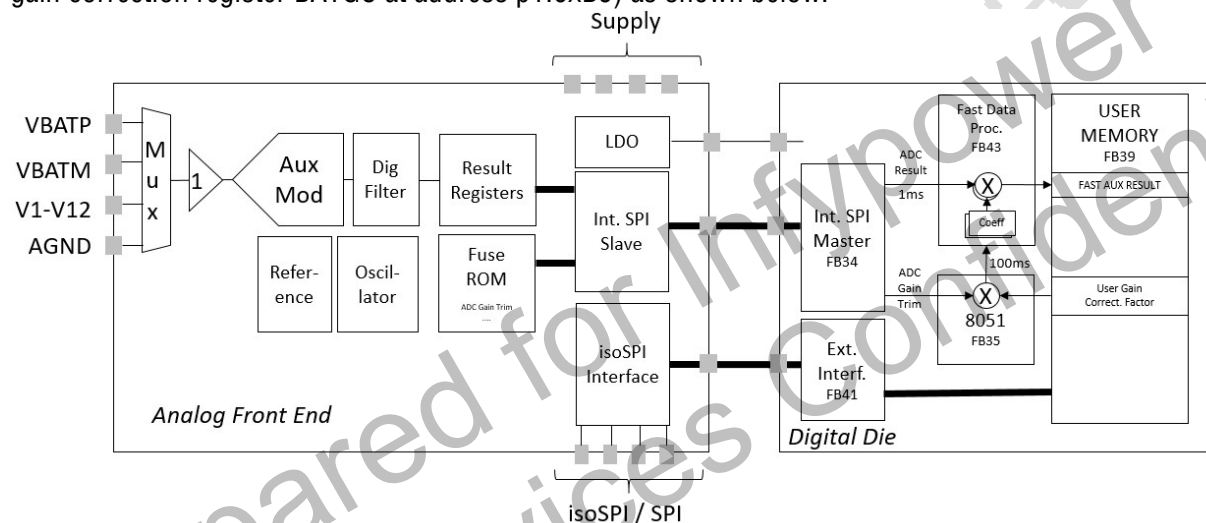


Figure 5: Block Diagram of the LTC2949

This Multiplication is done in two steps; first the ADC Trim Coefficients (Factory Trim, AXTRIM for AUX ADC, see table below) are multiplied with the User Gain Coefficients (one of MUXxGC depending on MUXPSETx/MUXNSETx, x=1..4) by the Microcontroller 8051 to yield a factor called "Coeff" in the picture above, which is then multiplied by a hardware multiplier with the actual ADC result.

To verify that the multiplications are performed correctly for any input signal and any user gain coefficient, a fast AUX conversion of -VREF2 using a Coeff of SM\_55\_FP\_FAC = 1.3330078125 and a fast AUX conversion of +VREF2 with a Coeff of SM\_AA\_FP\_FAC = 1.666015625 are compared with the expected values. As  $\text{Coeff} = \text{AXTRIM} * \text{UserGain}$ , the UserGain to be programmed becomes  $\text{UserGain} = \text{Coeff} / \text{AXTRIM}$ . The AXTRIM is calculated from register GCV which must also be compared to the value read via SoftBypass (see Appendix SoftBypass). The two AUX conversions must be compared to  $\text{SM\_55\_FP\_FAC} / \text{AXTRIM} * (-\text{VREF2})$  and  $\text{SM\_AA\_FP\_FAC} / \text{AXTRIM} * \text{VREF2}$ . The tolerance for the check when measuring VREF via MUX channel 22 is equal to that of SM15. For MUX channel 23 (VREF2 via internal 250k resistor) the tolerance shall be set to 1.6%.

The two's complement representation of the AUX-ADC output value -VREF2 and VREF2 together with the normal single LSB noise of the ADC, ensures that the multiplier is checked with two very distinguishable bit patterns leading to a high coverage. Similarly, the coefficients (Coeff) SM\_55\_FP\_FAC and SM\_AA\_FP\_FAC after conversion from float to integer appear as checkerboard pattern at the input of the multiplier. To achieve this, the user gain correction factors SM\_55\_FP\_FAC/AXTRIM and SM\_AA\_FP\_FAC/AXTRIM are used and written to MUXxGC. To allow two different user gain correction factors for -VREF2 and +VREF2, two MUX settings (FAMUXP, FAMUXN and MUXPSETx, MUXNSETx) must be used that do not only differ by polarity. This is possible by using channels 22 (VREF2) and 23 (VREF2 via 250k). For details see the ATSR3 Implementation guide.

Table gc1. AUX ADC factory trim gain correction factors.

| ADDRESS | SYMBOL        | OPERATION  |
|---------|---------------|--|
| 0x9F    | GCV           | Internal gain correction factor of the AUX ADC (8-bit signed integer)<br>$AXTRIM = GCV * 0.0009765625 + 1.0$ |
| 0x0E6   | F24GCV[23:16] | Same as GCV but in F24 format (24-bit floating point). For information only, not used for diagnostic checks. |
| 0x0E7   | F24GCV[15:8]  |  |
| 0x0EA   | F24GCV[7:0]   |  |

Beside verifying the multiplication of ADC results with Gain and Trim Coefficients, SM17 verifies the whole chain from the Analog Front End, the Internal Serial Interface down to the User Memory and the External Interface. This test can be used to replace the digital filter test of the Accuracy Check (SM4).

To fulfill ATSR3, it is still necessary to have a redundant way of checking the multiplier and internal common data paths, which is provided by the slow channel as described in 'Redundant Current Measurements (SM13)' and within the ATSR3 Implementation guide.

If SM17 is executed, SM15 is not required.



**Redundant Multiplier (SM5)**

To ensure the correct multiplication of “Coeff” (see Figure 5) with the AUX-ADC result in fast mode operation, two independent DSP channels have been implemented and their result is compared. Furthermore, every 100ms the calculation of the last fast AUX measurement is performed by the LTC2949’s internal microprocessor. If the result of the two channels or the MCU calculation do not match, an error is signaled by bit FAERR in the FAULTS register. FAERR is also set if a fast AUX Channel result is not provided within 1.5ms.

The SM5 can be verified by fault injection activated by setting bit FCAERRINJ in the ISO0 register causing FAERR in the FAULTS register to be set. This test can be completely replaced by SM17 for AUX measurements, see above.

**Fuse PROM Error Check (SM6)**

To detect an error in the fuse PROM storing trim values programed at factory test, the LTC2949 has an error code correction algorithm capable of correcting single bit errors and flagging double bit errors. When an error is detected, it is signaled by setting bit PROMERR in the FAULTS register. The SM6 can be verified by setting bit FERR in the **ISO0** register which causes PROMERR being set if ECC works correctly.

**Memory Check (SM7)**

A BIST of all memories is initiated at power up or after a reset, but only if pin SDA pulled high (see datasheet). All the volatile memories are then checked against a standard March-C algorithm. The ROM is tested against a CRC16-CCITT. The memory checks take roughly 30ms –their completion is signaled by setting bit HWMBISTEXEC in the EXTFAULTS register. A fault from the Memory BIST is signaled by bit HWBIST in the FAULTS register; bits ROMERR, MEMERR, FCAERR, XRAMERR, IRAMERR within register EXTFAULTS signal the result separately for each memory by ‘0’ for pass and ‘1’ for fail – see register description at the end of this chapter.

The SM7 can be verified by fault injection. Fault injection is provoked by setting bit HWBISTINJ in the ISO0 register.

The gain coefficients of all ADCs stored in the internal RAM are protected by CRC16. The CRC signature is calculated after each change in the configuration register triggered by ADJUP (see Operation Control Register, OPCTRL) and the coefficients are checked against this signature every 100ms. An error is signaled by setting bit CRCCFG in the FAULTS register.

Beside the memory BIST at power up, the user accessible registers (FB39) are protected by CRC16 during operation. The CRC signature is calculated at every transition to the SLEEP state and the registers are verified against this signature when the device moves to the STANDBY state. An error is signaled by setting bit CRCMEM in the FAULTS register. This test of the user registers is only relevant, if the SLEEP state of the LTC2949 is used and the memory is not initialized by the host controller after recovering from the SLEEP state.

The CRC protection of the registers and the internal RAM can be verified by fault injection by means of bits CRCCFGINJ and MEMERRINJ of register ISO0.

**RDCV Hand-Shake Check (SM23)**

To verify the execution of a new measurement by the AUX channel in fast mode, the LTC2949 acknowledges successful conversion and data processing by the hand-shake (HS) byte of the

RDCV command. See the datasheet of the LTC2949 on how to process the HS-byte with the host controller.

If the sequence of fast measurements guarantees that consecutive measurements are distinguishable from each other, the hand shake byte may not be evaluated. Instead the host shall verify the data of at least one ADC (I1, I2, BAT or AUX) changed between consecutive measurements.

### 9.1.5 Serial Interfaces

**HSR6: Data communications within the LTC2949 and between the LTC2949 and other devices shall be free from undetectable errors. The undetected bit error rate for external communication shall be comparable to that of CAN communications.**

#### Internal Communication Check (SM8)

The LTC2949 is realized by two dies with an internal serial interface. The interface communication is protected by checksum. Furthermore, all data written from the digital die to the analog die is read back and verified. Finally, to prevent unintended data write by the 8051 in the digital die to the analog front-end die, a key lock is implemented requiring an upfront immediate adjacent write transaction of an 8-bit key word before effectively writing data to the analog die from 8051.

A detected error is signaled by setting bit INTCOMMERR in the FAULTS register.

The correct working of the checksum algorithm can be verified by fault injection. Fault injection is provoked by setting bit ISERERRINJ in the ISO0 register.

Note: The above described Data Path Check (SM17) contributes to the fault detection coverage of the internal interface as the transfer of different values is verified.

#### External Communication Check (SM9)

Serial interface blocks consist of the standard SPI port, the isoSPI drivers, isoSPI receivers, and isoSPI bias circuit including the IBIAS generator. The functionality of these blocks is verified by the addition of a packet error code (PEC) on every serial transmission. The PEC must be used to verify the serial communication data. The external communication verification is completed by writing data to the LTC2949, reading it back and verifying its consistency and the PEC value.

The packet error code (PEC) is a 15-bit cyclic redundancy check (CRC) value calculated for all the bits in a register group in the order they are passed, using the initial PEC value of 00000000010000 and the following characteristic polynomial:

$$x^{15} + x^{14} + x^{10} + x^8 + x^7 + x^4 + x^3 + x + 1$$

The PEC prevents erroneous serial data from being interpreted as valid data and guarantees safe communication. A failure in the serial I/O port block will cause erroneous data to be sent and/or will prevent data from being received. In the case of a write to the device, if the calculated PEC does not match, the received data will be discarded. Data should be read back from the device after write operations to confirm that the write was correctly received. If the data read back does not match what was written, the write should be attempted again. When data is read from the LTC2949, if the PEC calculated by the MCU does not match the transmitted PEC, the data is invalid. In that case, additional reads should be performed until the PEC matches.

The characteristic polynomial is the same as that used in CAN bus communications. For the LTC2949, the data packet length is configurable. The Hamming Distance for the CAN polynomial

with 48 bits of packet data is 6, which means that any data corruption of 5 bits or fewer would be detected by the CRC (see Koopman, P.; Chakravarty, T., "Cyclic redundancy code (CRC) polynomial selection for embedded networks," Dependable Systems and Networks, 2004 International Conference on, pp. 145-154, 28 June-1 July 2004).

A typical physical layer interface has primary failure modes of stuck-at-0, stuck-at-1, or floating. The PEC with a nonzero initial seed detects stuck-at-0 and stuck-at-1 faults. Floating data lines and interference can produce random data errors. Because of the large Hamming Distance of the CAN polynomial, the probability of undetected errors is extremely low. The specific probability of undetected errors depends on the system's communications rate, the bit-error-rate (BER) induced by interference, and the bit error profile.

When the physical interface communicates over cables (like with a twisted pair), then it is assumed that the interface used is robust against electromagnetic interference. The isoSPI interface, for example, typically has zero-bit errors even under high levels of bulk current injection (BCI).

Systems using the LTC2949 should be tested for susceptibility to interference. The measured BER can then be used to estimate the probability of undetected data errors. If the chosen physical layer interface does not store or interpret the data passing through it, then the assumptions given above is true. This will be the case for the typical interfaces used with the LTC2949: direct connection to a microcontroller (voltage-mode SPI); digital isolators (e.g., opto-couplers); or isoSPI (e.g., LTC6820).

The correct working of the PEC error detection on the host side can be verified by injecting single bit errors at the low-level SPI driver of the attached safety micro-controller. A single bit flip in any data or PEC byte must be flagged with a PEC error.

For debugging purposes, the correct working of the PEC on the host side can also be verified by a fault injector within the LTC2949. Fault injection is provoked by setting by ESERERRINJ in the ISO0 register. Memory lock must be used to set the bit ESERERRINJ and read-back to ensure it was set. After releasing memory lock, the error injector is activated and the LTC2949 will send data with PEC errors until the bit is cleared again via some write command to ISO0 register.

#### **External EEPROM Check (SM24)**

An external non-volatile EEPROM can be connected to the LTC2949's I2C master interface to allow storage of gain correction factors. The EEPROM will be supplied by the 2.5V output from the BYP1 pin. The content of the EEPROM and the I2C communication is protected by two CRC-16 CCITT, one per memory page of the LTC2949 (240 bytes).

Operation of the external EEPROM is controlled via the EEPROM Control Register EEPROMCTRL. During the execution of the RESTORE command, the LTC2949 reads 512 bytes (two pages, 240 bytes each, two 16-bit CRCs and two 14 bytes signatures which are not relevant for this SM) and checks that the CRC-16 CCITT calculated from the EEPROM contents and saved by the SAVE command, matches the calculated one for each memory page that was read.

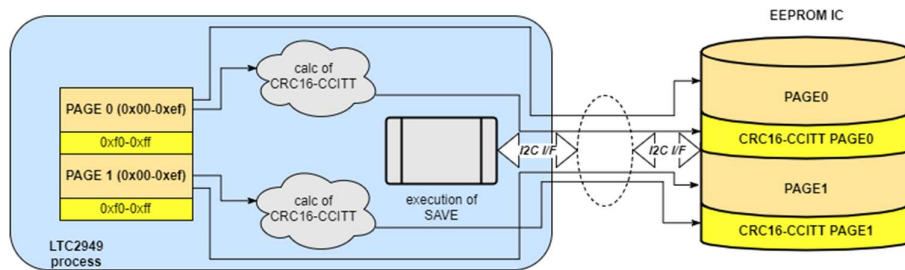


Figure 6: EEPROM SAVE operation

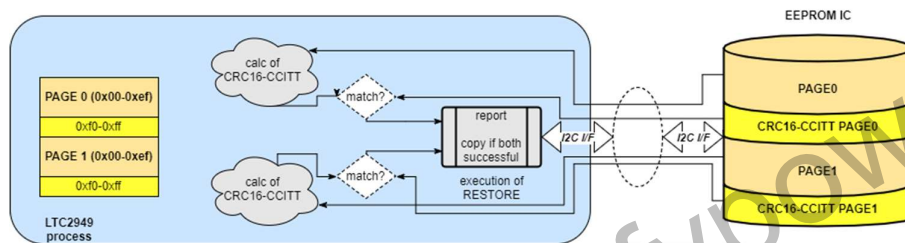


Figure 7: EEPROM RESTORE operation

If any two CRC values don't match the calculated one, the RESTORERSL bit is not set at the end of the operation indicated by clearing of the RESTORE bit.

### 9.1.6 Voltage Regulators

**HSR7:** It shall be possible to verify that the internal voltage regulators are within required limits.

#### Voltage Regulator Check (SM10)

Internal supply voltages are monitored by comparators and an under-voltage event is reported in the STATUS register. If any of the supplies drops below its under-voltage threshold, a power-on reset occurs, resetting all registers to their default value leading UVLOA, PORA, UVLOSTBY and UVLO in the STATUS register to be set. To detect these register bits to be set after a POR, they must have been cleared after the previous power up.

### 9.1.7 Oscillator

**HSR8:** It shall be possible to verify that the Frequency of the internal fast Oscillator is within required limits.

#### Oscillator Check (SM16)

The LTC2949 compares the oscillator driving the ADCs with either an internal low frequency precision oscillator or an external reference clock if available. If the ADC oscillator becomes faster than normal by more than 32%, the bit TBERR in the STATUS register is asserted. Additionally, the deviation of the ADC oscillator frequency from its nominal value shall be checked by pulling the update time from any of the time registers TB1-TB4 and comparing it to the nominal value of 100ms using a reference clock in the master controller. This way, also latent faults of the TBERR reporting via STATUS register are checked.

## 9.2 HARDWARE SAFETY REQUIREMENTS FOR ATSR2

The Assumed Technical Safety Requirement ATSR2 is to ensure that the effective trip point of the Over Current Comparator is within  $\pm 20\text{mV}$  of the programmed threshold and in case of an exceeding input signal the heartbeat at GPIO5 is stopped.

**HSR9: It shall be possible to verify the functionality, offset and threshold of the overcurrent comparator (OCC)**

### Over Current Comparator Check (SM11)

After configuring and enabling the Over Current Comparators, a write '1' to the ATSR2 bit in the ISO1 register activates the Over Current Comparator Self-Test (OCCST). Within this self-test, the Over Current Comparators are disconnected from the IxP and IxM pads and for all possible thresholds a signal 6mV above and 6mV below the threshold is applied and it is verified that the comparator tripped for an input signal exceeding the threshold but not for an input signal below the threshold. The applied input signal is further read back via the AUXADC. Heartbeat (GPIO5/V12) while toggling and stopped is measured via AUXADC. At the end of the Over Current Comparator Self-test, bit ATSR2 in the ISO1 register is cleared.

The result of the self-test is communicated by setting bit RSL\_ATSR2 in register ISO1 if the test passed. A failing self-test is indicated by bit RSL\_ATSR2 remaining 0.

If bit ATSR2 in register ISO1 is set together with bit SHORT\_OCC in register ISO2, only the thresholds and deglitch times configured via registers OCC1CTRL and OCC2CTRL are tested as explained above. If the bit SHORT\_OCC is not set, all possible configurations (threshold and deglitch times) are tested. The duration in the latter scenario is significantly longer (~4s versus ~0.3s). Please note: Deglitch times are only tested for a pulse longer than a programmed time window to trigger the comparator, but it is not tested that a pulse shorter than that does not trigger the comparator.

To verify that the correct thresholds are used by the over current comparators, register OCC1CTRLSHDW (0xD9) must be compared against OCC1CTRL (0xDE) for channel 1 and register OCC2CTRLSHDW (0xDA) must be compared against OCC2CTRL (0xDF) for channel 2. Registers OCC1CTRLSHDW and OCC2CTRLSHDW are shadow locations, on which the values written upfront in the real configuration registers OCC1CTRL and OCC2CTRL are copied. For those copies to be effective, it can take up to 20ms, according to the operation of the device. This copy is done always, not only during the internal built in self-test, thus it can be checked at any time after running the BIST. The time it takes, to copy the data to the shadow registers, is included in the execution durations of the long and short test mentioned above.

The Over Current condition in the system is a first fault, which makes the Over Current Comparator Self Check a latent fault check. This allows the SM11 to be executed within the MPFDI.



### Summary

- 1) Write OCC1CTRL = OCC1EN | ... and OCC2CTRL = OCC2EN | ... (enable OCC with custom threshold / deglitch configuration)
- 2) Read ISO1 and ISO2 and verify default state (all 0x00)
- 3) Write ISO2 = SHORTOCC (optional, reduces execution time to 0.3s instead of 4s)
- 4) Write ISO1 = AFSR2 (start built-in self-test)
- 5) Poll for (timeout 0.3s or 4s +5%)
  - a) ISO1 = AFSR2RSL AND
  - b) OCCxCTRLSHDW = OCCxCTRL
- 6) Write ISO1 = 0

### Over Current Comparator Open Wire Check (SM12)

During the Over Current Comparator Self-test, the comparators are disconnected from the input pads IxP and IxM and reconnected at the end of the test. To verify the functionality of the responsible switch, an Open Wire Check shall be performed by connecting the inputs of the over-current comparators to the inputs of the AUX-ADC while activating 250µA pull-up or pull-down current sources (see Figure 3) and verify that after pulling the inputs of the over current comparator apart from each other, they return into the expected voltage range. This check shall be executed right after SM11(both OCCs must have been enabled there, which means bits OCC1EN/OCC2EN in OCC1CTRL/OCC2CTRL must be set).

First apply IPT (14) and IMT (13) to the AUX Multiplexer by writing 13 to register FAMUXN and 14 to register FAMUXP. Going into SoftBypass (see Description in Appendix SoftBypass) and setting then bit 3 in analog die register 0x4F, connects I1P and I1M to the AUX Multiplexer input IMT and IPT, while setting bit 4 of analog die register 0x4F connects I2P and I2M to AUX Multiplexer inputs IMT and IPT. After leaving SoftBypass, enable the pull-up current source at MUXP and the pull-down current source at MUXN by setting bits MUXPCURPOL, MUXPCUREN and MUXNCUREN of register FCURGPIOCTRL and run a fast AUX ADC measurement. Verify the result is higher than 0.5V.

Directly after the previous measurement, within 1ms, disable all current source by clearing all current source control bits of FCURGPIOCTRL and trigger again a fast AUX-ADC measurement. Verify the result is within the expected input range for IxP-IxM and compare to the IADCx result. Then, enable the pull-down current source at MUXP and the pull-up current source at MUXN by setting bits MUXPCUREN, MUXNCURPOL and MUXNCUREN of register FCURGPIOCTRL and run another fast AUX-ADC measurement. Verify the result is below -0.5V. Again, directly after the previous measurement, within 1ms, disable all current source by clearing all current source control bits of FCURGPIOCTRL and trigger another fast AUX-ADC measurement. Verify the result is within the expected input range for IxP-IxM and compare to the IADCx result.

To proceed with normal operation after the test, again go to SoftBypass, clear bits 3 and 4 of analog die register 0x4F and leave SoftBypass.

### Summary

The SM12 shall be executed right after SM11.

1. Execute SM11 (both OCCs must have been enabled there, which means bits OCC1EN/OCC2EN in OCC1CTRL/OCC2CTRL must be set)

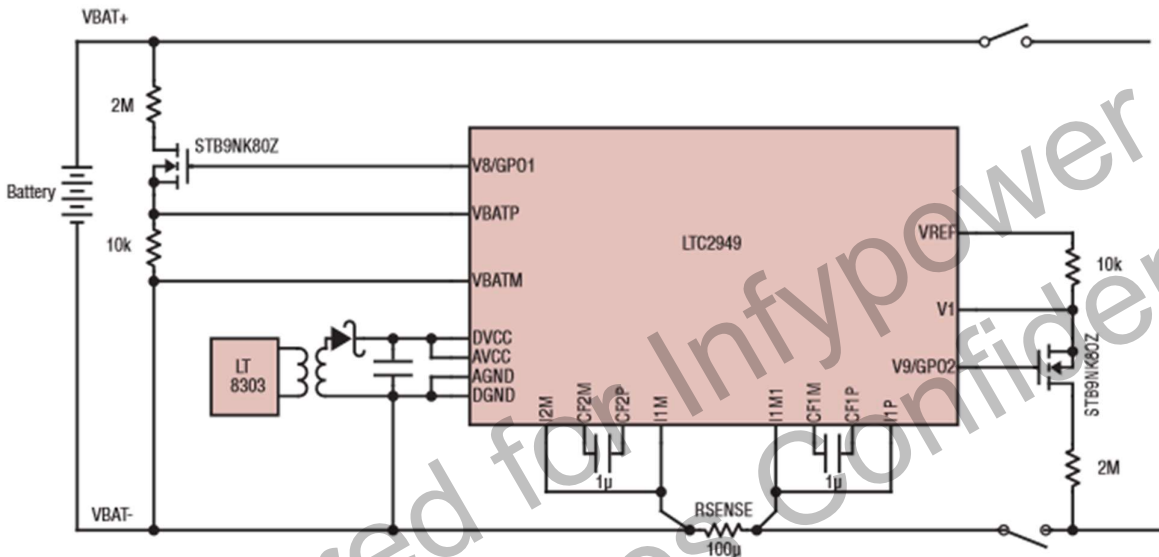
2. Optional: Adjust ADCCONF as required by application and make ADJUPD
3. OPCTRL = CONT (Enable Continuous measurement mode)
4. Wait for first update of slow channel (e.g. poll for TB1 changed)
5. FACTRL = FACHA | FACH2 (enable fast measurement for AUX and I2)
6. FAMUXP = IPT (14), FAMUXN = IMT (13)
7. Go into SoftBypass
8. Write 0x18 (only bits 3, 4 set) to AnaDie register 0x4F
9. Leave SoftBypass
10. Now, following connections are activated
  - a. AUX ADC connected to IPT (pos.) and IMT (neg.)
  - b. I1P, I2P connected to IPT
  - c. I1M, I2M connected to IMT
11. FCURGPIOCTRL = MUXPCURPOL | MUXPCUREN | MUXNCUREN (pull-up at MUXP, pull-down at MUXN; must be a 2-byte write burst including register FGPIOTRL)
12. ADCV, wait 1.2 ms (trigger fast single shot conversion)
13. RDCV (read conversion results)
  - a. Check  $AUX > 0.5V$
14. FCURGPIOCTRL = 0 (disable current sources; must be 2-byte write burst including register FGPIOTRL)
15. ADCV, wait 1.2 ms (trigger fast single shot conversion)
  - a. Time between 13. and 15.  $< 1ms$
16. RDCV (read conversion results)
  - a. Check  $|I2ADC-AUX| < 2mV$  and  $|AUX| < 124mV$
17. FCURGPIOCTRL = MUXNCURPOL | MUXPCUREN | MUXNCUREN (pull-down at MUXP, pull-up at MUXN; must be 2-byte write burst including register FGPIOTRL)
18. ADCV, wait 1.2 ms (trigger fast single shot conversion)
19. RDCV (read conversion results)
  - a. Check  $AUX < -0.5V$
20. FCURGPIOCTRL = 0 (must be 2-byte write burst including register FGPIOTRL)
21. ADCV, wait 1.2 ms (trigger fast single shot conversion)
  - a. Time between 19. and 21.  $< 1ms$
22. RDCV (read conversion results)
  - a. Check  $|I2ADC-AUX| < 2mV$  and  $|AUX| < 124mV$
23. Go into SoftBypass
24. Write 0x00 to AnaDie register 0x4F
25. Leave SoftBypass

The duration of this safety mechanism is ~0.4s.

The Over Current condition in the system is a first fault, which makes the Over Current Comparator Self Check a latent fault check. This allows the SM12 to be executed within the MPFDI.

### 9.3 HARDWARE SAFETY REQUIREMENTS FOR ATSR3

The Assumed Technical Safety Requirement ATSR3 is to ensure that the current measurement in fast mode should detect any measurement error larger than  $\pm 200\mu\text{V}$  or  $\pm 2\%$ , whatever is larger, if both current measurement channels are connected to the same sense resistor or two sense resistors in series.



The measurement results of each Current Acquisition Channel (I<sub>x</sub> Channel) can be impaired by the following failure modes (FM):

- FM1: Broken Input connection
- FM2: Offset Error
- FM3: Gain Error
- FM4: Noise Error
- FM5: Linearity Error
- FM6: Oscillator Error
- FM7: Digital Filter
- FM8: Internal Interface Error
- FM9: Fuse PROM Error
- FM10: Digital Computing Error
- FM11: Storage Error
- FM12: External Serial Interface Error
- FM13: Supply / LDO Error

Connecting both channels to sense the same input voltage introduces redundancy in all independent failure modes (FM1, FM2, FM4, FM5, FM7), leaving only multiple failures to remain undetected. These independent failure modes will therefore not be considered for in the following



failure mode analysis.

**HSR10: Broken Input Connection, Offset, Noise Error, Linearity Error and Digital Filter and Read-out Logic Error shall be verifiable**

If there is no redundant connection to the sense pins of the shunt, to detect a broken wire to the sense resistor, an open wire detection can be performed as described in the 'Open Wire Check (SM1)'. Instead of doing the check on e.g. pins V1-V12 use the MUX inputs CF1P/CF1M and CF2P/CF2M. It is not necessary to check both channels. Connect the CF2P/CF2M channel for open-wire check with the current measurement on the CF1P/CF1M channel remaining undisturbed. See ATSR3 Implementation guide for details.

**Redundant Current Measurement (SM13)**

Connecting both current measurement channels (I1ADC and I2ADC) to the same sense resistor or to two separate sense resistors and configuring both in fast mode by setting bits FACH1 and FACH2 in register FACTRL allows to perform redundant current measurements. By reading the results either via a RDCV command or from the current channel FIFOs, FIFOI1 and FIFOI2, and comparing them against the specified limit of the LTC2949 allows to detect errors due to independent faults. As SM14 requires to read packages of 128 fast samples to perform the redundant computation, it is recommended to compare the average of 128 samples between both channels, instead of every sample.

The threshold of the difference shall be set to 1% which gives enough margin to the relative error of the safety target. The absolute error limit shall be set to  $\pm 100\mu\text{V}$  which also gives enough margin to the absolute error of the safety target. Only if both absolute and relative error checks fail, the master shall flag a safety violation. If two shunt resistors are used, they must to be calibrated in an end-of-line production test and the drift of their ratio over temperature is part of the error budget of 1%.

The following failure modes are not independent:

- FM3: Gain Error due to Reference
- FM6: Oscillator Error
- FM8: Internal Interface Error
- FM9: Fuse PROM Error
- FM10: Digital Computing Error
- FM11: Storage Error
- FM12: External Serial Interface Error
- FM13: Supply / LDO Error

As both current ADCs use the same reference block (VREF1) and the same oscillator, failures of these blocks are common point of failures. Both failure modes are covered by the Accuracy Check (SM4), Fuse PROM Errors are covered by Fuse PROM Error Detection (SM6), Storage Errors are covered by the Memory Checks (SM7), Internal Serial Interface Errors are covered by SM8, External Serial Interface Errors are covered by SM9 and Supply/LDO Errors are covered by the Voltage Regulator Check (SM10), all described in the context of ATSR1.

The ADC readings of the Current Channels in fast mode are corrected by ADC gain trim values stored at final test in a one-time programmable ROM. The multiplication of the ADC gain trim values with the ADC reading is done by a hardware multiplier. As both current channels use the same data path for this digital computation, FM10 is a fault of shared resource, which is not directly detected by any existing safety mechanism integrated into the LTC2949.

**HSR11: The multiplication of current ADC results with gain trim and user programmable gain values shall be verifiable.**

The multiplication of current ADC results with gain trim and user programmable gain values is performed as described previously for the AUX-ADC. Therefore, the Data Path Check (SM17) can be used to ensure correct data processing. Additionally, to detect faults of shared resources in other internal data paths, the correct multiplication must be verified by computing the data in the host as described in the following.

#### Redundant Current Computation (SM14)

To enable the verification of the digital computation, the LTC2949 stores the raw quantities of the ADC gain trim and every 100ms (128 fast continuous samples) the raw output data of the current ADCs in the user accessible memory. Following table describes the related registers.

**Table 1. Current RAW and gain trim registers.**

| Address | Name         | Description   |
|---------|--------------|---|
| 0x79    | I1RAW[23:16] | Unsigned 3-byte of raw output data of current channel 1 |
| 0x7A    | I1RAW[15:8]  |   |
| 0x7B    | I1RAW[7:0]   |   |
| 0xAF    | GCI1[7:0]    | Signed single byte gain trim of current channel 1       |
| 0x7C    | I2RAW[23:16] | Unsigned 3-byte of raw output data of current channel 2 |
| 0x7D    | I2RAW[15:8]  |   |
| 0x7E    | I2RAW[7:0]   |   |
| 0xE0    | GCI2[7:0]    | Signed single byte gain trim of current channel 2       |

Note: The signed single byte gain trim is converted to float by following equation:  $GCI_x * GCLSB + 1.0$  with  $GCLSB = 0.0009765625$

The gain trim factors shall be verified by reading them via SoftBypass mode, see Appendix SoftBypass.

Accessing this data, the user can verify the digital computation done by the LTC2949 by calculating the product of the raw current ADC values, the gain trim, the user programmed factor (RS1GC, RS2GC) and the fixed 'magic factor' (MAF = 1.000473) and compare it to the average of 128 fast current channel samples. See following equation that shall be verified:

$$I_x = I_{xFIFOAVG} = (I_{xRAW} - 2^{17}) * (GCI_x * GCLSB + 1.0) * MAF * RS_{xGC}$$

$$GCLSB = 0.0009765625$$

Notes: x = 1 for I1 or 2 for I2;  $I_{xFIFOAVG}$  is the average of 128 samples read from the FIFO;  $I_x$  is the reading from slow channel registers I1 or I2.

The accuracy of this comparison is limited to 0.1% by the precision of the fixed-point arithmetic of the overall gain adjustment done in the fast channel. The threshold of the comparison shall be set to 1% which gives enough margin to the relative error of the safety target. The absolute error limit can be set to  $\pm 20\mu\text{V}$  which also gives enough margin to the absolute error of the safety target. Only if both, the absolute and relative error, checks fail, the master shall flag a safety violation.

In the implementation of this safety mechanism it is important to correctly match the conversion results read from the FIFO and those read from the slow channel (I1RAW, I2RAW). Once the fast continuous mode has started (write FACTRL with FACONV bit set), the host has to collect every slow channel update (I1RAW, I2RAW) and read and count every sample from the FIFO. Then, every slow channel result must be compared to every set of 128 fast samples (the average of those) in the order as they are read. Values reported by the slow channel are always delayed by some milliseconds after a complete set of 128 fast samples were acquired.

## 9.4 SAFETY MECHANISMS FOR LATENT FAULTS

The LTC2949 features several Safety Mechanisms, which run fully transparent to the user (e.g. communication CRC) and a detected failure is reported by the respective registers (AutoBIST).

**HSR12: The LTC2949 shall allow to verify execution and reporting of user transparent safety mechanisms.**

The LTC2949 implements following safety mechanisms for latent faults associated with AB-type safety mechanisms:

**SM18: Fault Injection to Fuse PROM Error Detection (SM6)**

**SM19: Fault Injection to Internal Communication Verification (SM8)**

**SM20: Fault Injection to External Communication Verification (SM9)**

**SM21: Fault Injection to Redundant Multiplier (SM5)**

**SM22: Fault Injection to Memory Check (SM7)**

The register **IS00** at p1.0x70 allows to inject faults to verify that the safety mechanisms (SM) provided by the LTC2949 are executed correctly:

| BIT | SYMBOL     | TYPE | DEFAULT | OPERATION  |
|-----|------------|------|---------|--|
| 0   | FERR       | SO   | 0       | Inject fault in PROM reading (SM6) and triggers PROM read out  |
| 1   | RESERVED   | RW   | 0       | Reserved bit, always write 0.  |
| 2   | ISERERRINJ | RW   | 0       | Inject Parity Error into Internal Serial Communication (SM8)   |
| 3   | ESERERRINJ | RW   | 0       | Optional external serial interface fault injector (injects faults in the slave to master stream). Memory must be locked before this bit can be set and read-back. See SM9 for details. |
| 4   | FCAERRINJ  | SO   | 0       | Inject fault in Redundant Multiplier and start SM5   |
| 5   | HWBISTINJ  | SO   | 0       | Inject random faults in Memory (SM7) and start SM7   |
| 6   | CRCCFGINJ  | SO   | 0       | Inject fault at gain coefficient memory CRC check  |
| 7   | MEMERRINJ  | SO   | 0       | Inject fault at CRC of user accessible register check  |

### 9.4.1 Latent fault checks example

It is recommended to perform all the latent fault checks directly after power-up / reset.

Before injecting the errors, the STATUS, EXTFAULTS and FAULTS registers shall be checked for the default values, see register description (the values are 0x0F, 0x80, 0x00).

Following fault injectors can then be set at the same time:

ISO0: 0xF5 (1111 0101b = FERR | ISERERRINJ | FCAERRINJ | HWBISTINJ | CRCCFGINJ | MEMERRINJ)

Once the error injection was performed, the related SO bits will be cleared, besides the bit ISERERRINJ which is RW and must be cleared manually. The time from setting the error injectors to the SO bits being cleared is less than 50ms. After that time, the register ISO0 must be read as 0x04 (ISERERRINJ). As result of the error injection, the following error flags will be set:

EXTFAULTS: 0xBE (1011 1110b = ROMERR | MEMERR | FCAERR | XRAMERR | IRAMERR | HWMBISTEXEC)

FAULTS: 0xF5 (1111 0101b = PROMERR | INTCOMMERR | FAERR | HWBIST | CRCCFG | CRCMEM)

The error flags are guaranteed to be set within 50ms after the error injection. Once verified, STATUS, EXTFAULTS and FAULTS can be cleared using the memory lock feature.

Note: There is no error injector for thermal shutdown (TSD). TSD is not safety relevant. If the LTC2949 resets because of a thermal shutdown, also the STATUS register will report 0x0F, indicating the reset event. The bit TSD adds additional information only, that the reset happened because of a thermal shutdown.

Note: Fault injection via ISERERRINJ is not required, see 'External Communication Verification (SM9)' for details.

#### Summary

1. Initial condition is power-up / reset
2. Read EXTFAULTS, FAULTS, STATUS and ISO0 and check for default values
  - a. Note: SDA must have been high and HWMBISTEXEC within EXTFAULTS must be set
3. Write 0xF5 to ISO0 (inject all errors the same time)
4. Wait 50 ms
5. Read ISO0 and check for value 0x04 (all injectors, besides ISERERRINJ cleared)
6. Read EXTFAULTS, FAULTS and check for error flags (0xBE, 0xF5)
7. Lock the memory
8. Write 0x00 to EXTFAULTS, FAULTS, STATUS, ISO0
9. Unlock the memory

## 9.5 FUSA RELATED REGISTERS

The LTC2949 has several registers reporting the status of the device and faults occurring during operation as well as dedicated register to trigger safety mechanisms, report their results and verify their functionality by fault injection.

The STATUS and FAULTS registers, also described in section “Register Description” of the LTC2949 datasheet, report the status of the device and faults occurring during normal operation. They are cleared by writing the corresponding bits to 0. To avoid missing any alert reporting, the memory must be locked before writing. If an alert condition occurs while the memory is locked, the LTC2949 will set the corresponding bit after the memory is unlocked by the host. See Datasheet section “Register Control Register” for more details about the memory locking.

**Table 2. STATUS (0x80)**

| BIT | SYMBOL   | TYPE | DEFAULT | OPERATION  |
|-----|----------|------|---------|--|
| 0   | UVLOA    | RW   | 1       | 1: Undervoltage in the analog domain or ADCs during a conversion                                       |
| 1   | PORA     | RW   | 1       | 1: Power-on reset has occurred due to undervoltage in the analog domain                                |
| 2   | UVLOSTBY | RW   | 1       | 1: Undervoltage in the standby domain  |
| 3   | UVLOD    | RW   | 1       | 1: Undervoltage in the digital domain  |
| 4   | UPDATE   | RW   | 0       | 1: Result registers have been updated  |
| 5   | ADCERR   | RW   | 0       | 1: The ADC conversion is not valid due to undervoltage during a conversion                             |
| 6   | TBERR    | RW   | 0       | 1: Overflow of the internal time base register. The values of accumulated result registers are invalid |

The STATUS register reports the status of register updates, under-voltage lockout, and reference clock errors. On power up, all under-voltage lockouts and the power-on reset are set to 1. After exit from shutdown, bits UVLOA and UVLOD are set. UPDATE is set to 1 when the LTC2949 has finished a measurement cycle and updated the result registers, the accumulation registers, and the tracking registers. Measurement completion in single shot mode can be observed by polling the bit UPDATE.

ADCERR is set to 1 if the supply voltage at AVCC is too low for a proper operation of the ADCs. TBERR is set to 1 if the internal time base overflows. This indicates an incorrect setting of the values PRE and DIV with respect to the external clock at CLKI. The values of accumulated results registers should be discarded if TBERR is set.

Failures of all supply voltages can be observed by reading the corresponding bits in the Status register.

**Table 3. FAULTS (0xDD)**

| BIT | SYMBOL     | TYPE | DEFAULT | OPERATION  |
|-----|------------|------|---------|--|
| 0   | PROMERR    | RW   | 0       | 1: Error in trim values stored in internal PROM              |
| 1   | TSD        | RW   | 0       | 1: Shutdown due to over temperature                          |
| 2   | INTCOMMERR | RW   | 0       | 1: Parity check of internal communication failed             |
| 3   | EXTCOMMERR | RW   | 0       | 1: PEC error in external communication (SPI/isoSPI) occurred |
| 4   | FAERR      | RW   | 0       | 1: Fast mode error   |
| 5   | HWBIST     | RW   | 0       | 1: Hardware Memory BIST Error                                |
| 6   | CRCCFG     | RW   | 0       | 1: Internal RAM gain coefficient CRC Error                   |
| 7   | CRCMEM     | RW   | 0       | 1: User accessible Register CRC Error                        |



The LTC2949 has an error code correction algorithm capable of correcting single bit errors and flagging double bit errors of the internal fuse PROM. When an error is detected, it is signaled by setting bit PROMERR in the FAULTS register.

When the LTC2949 die temperature exceeds roughly 175°C degrees, the bit TSD in the FAULTS register is set and the part resets.

The die-to-die communication of the LTC2949 is protected by a parity bit. Furthermore, each write from the digital die to the analog die is verified by a subsequent read and compare (SM8: Internal Communication Verification). If either the parity bit or the write/read comparison fails, the LTC2949 sets bit INTCOMMERR in the FAULTS register.

Communication to the host is secured by a cyclic redundancy check (CRC) (SM9: Communication Verification). When receiving data, the LTC2949 calculates the expected packet error code (PEC) and compares with the PEC received. Mismatches are reported by setting bit EXTCOMMERR in the faults register. However, EXTCOMMERR would not be set if the command bytes were altered due to some disturbance in a way, that they are treated as an unknown command by the LTC2949. In that case, the LTC2949 would just ignore the sequence. Thus, any data written to the LTC2949 must be read back and verified. In case of the fast trigger command (ADCV) the execution of the command is confirmed indirectly, by reading the conversion results via RDCV (see SM23).

The FAERR is set if the Redundant Multiplier Check (SM5) performed continuously during fast measurements of the auxiliary path, fails.

**Table 4. EXTFAULTS (0xDC)**

| BIT | SYMBOL      | TYPE | DEFAULT | OPERATION   |
|-----|-------------|------|---------|---|
| 0   | HD1BITERR   | RW   | 0       | 1: 1-bit error in trim values stored in internal PROM, corrected.   |
| 1   | ROMERR      | RW   | 0       | 1: CRC of ROM failed  |
| 2   | MEMERR      | RW   | 0       | 1: March C BIST of Register failed  |
| 3   | FCAERR      | RW   | 0       | 1: March C BIST of FIFO failed  |
| 4   | XRAMERR     | RW   | 0       | 1: March C BIST of XRAM failed  |
| 5   | IRAMERR     | RW   | 0       | 1: March C BIST of IRAM failed  |
| 6   |             |      |         |   |
| 7   | HWMBISTEXEC | RW   | 1       | 1: Hardware Memory BIST (SM7) was executed (only if SDA is pulled high to BYP1 via a 4.7k-10k resistor. Otherwise this bit will report 0) |

To control the over-current comparator latent fault checks and report the result, the LTC2949's registers ISO1 and ISO2 are provided:

The register **ISO1** at p1.0x71 allows to trigger the LTC2949's build in self-tests and read the results

| BIT         | SYMBOL    | TYPE | DEFAULT | OPERATION   |
|-------------|-----------|------|---------|---|
| 2           | ATSR2     | SO   | 0       | 1: Start the Over-Current Comparator BIST (SM11) for ATSR2<br>0: Over-Current Comparator BIST completed (after being started) |
| 6           | RSL_ATSR2 | RW   | 0       | 0: Over-Current Comparator BIST error<br>1: Over-Current Comparator BIST ok   |
| 0,1,3,4,5,7 | RESERVED  | RW   | 0       | Reserved bits, always write 0.  |

The register **ISO2** at p1.0x72 allows to configure the Overcurrent Comparator build in self-tests.

| BIT   | SYMBOL    | TYPE | DEFAULT | OPERATION   |
|-------|-----------|------|---------|---|
| 1     | SHORT_OCC | RW   | 0       | 0: Overcurrent comparator BIST (SM11) run for all thresholds (duration ~4s)<br>1: Overcurrent comparator BIST (SM11) run for current settings of OCC1CTRL and OCC2CTRL only (duration ~0.3s). |
| 0,2-7 | RESERVED  | RW   | 0       | Reserved bits, always write 0.  |

## 9.6 ATSR3 IMPLEMENTATION GUIDE

The following table summarizes the minimum set of diagnostic measurements that are required for ATSR 3 ("safe current measurement").

This table also includes an NTC measurement via one fast single shot (FSSHT) measurement typically needed for functionality only, meaning temperature measurement is not safety-related.

| Type/SM                            | Description   |
|------------------------------------|---|
| SM17                               | <p>Data Path Check:<br/>Measurement of VREF2_250k (23) – GND (0) and the reverse polarity GND (0) – VREF2_250k (23). Measurement of VREF2 (22) – GND (0). In total 3 FSSHT. For having also a data path check, two user gain correction factors will be used with above measurements: MUX1GC=SM_AA_FP_FAC/PROM_AUX_TRIM, MUX2GC=SM_55_FP_FAC/PROM_AUX_TRIM and MUX[NP]SET1=GND(0), VREF2 (22), MUX[NP]SET2=GND(0), VREF2_250k (23)</p> <p>This safety mechanism also covers:</p> <ul style="list-style-type: none"> <li>• Measure of 2nd reference (SM15)</li> <li>• Reverse Polarity Measurement (SM3)</li> <li>• Leakage check as part of Accuracy Check (SM4)</li> <li>• Voltage Regulator Check (SM10): See also below note about SM10</li> </ul> <p>In total <b>3 FSSHT</b><br/>SM_AA_FP_FAC = 1.666015625<br/>SM_55_FP_FAC = 1.3330078125</p> |
| SM1                                | <p>Open Wire Check:<br/>Measure CF2P – CF2M at least with both current sources enabled (sourcing current on the positive and the negative input).<br/>In case of an open wire it takes some milliseconds for the current sources to charge the external filter capacitors connected to CFxP and CFxM. To give enough time, the current sources will be enabled already during the fast-continuous measurement (FCM) phase, 50-100 ms before the fast-single shot (FSSHT) phase.<br/>Measurement results of the open wire check will be read via RDCV commands during the FCM phase. The current sources will be disabled before the FSSHT phase.</p>  |
| NTC meas.<br>(not safety relevant) | <p>NTC temp measurement:<br/>This is done by <b>one FSSHT</b> measurement, that will be incorporated into above set of FSSHTs.</p>  |

| Type/SM  | Description   |
|--|---|
|  | Notes: <ol style="list-style-type: none"> <li>1. Instead of measuring the actual value of VREF, the nominal value (3V <math>\pm 1\%</math>) can be used to calculate the NTC resistance and its temperature.</li> <li>2. Instead of the nominal VREF value, it is also possible to use the value reported by register VREF. Still, this value is measured only once when the slow continuous mode is enabled. Thus, changes of VREF depending on temperature or load are not reflected here.</li> </ol>   |
| <i>Note: In total 4 FSSHTs need to be performed periodically with a period defined by the FTTI, here 1 second. For best performance and least impact on the coulomb counting, the fast-continuous mode should be interrupted after 9*128 samples were read from each channel, to perform those FSSHTs.</i> |   |
| SM6  | Fuse PROM Error Detection: Done at bootup, result stored to FAULTS.   |
| SM7  | Memory Checks: All memories checked via internal memory BIST at power-up. User accessible registers are checked periodically via CRC calculation. Results must be checked via FAULTS register. Periodic execution of memory BIST is not necessarily due to full redundancy of FIFO I1 and FIFO I2.  |
| SM8  | Internal Communication Verification: periodic check of STATUS register  |
| SM9  | External Communication Verification via PEC and report of PEC errors via STATUS register  |
| SM10   | Replaced by VREF2 conversion and check of STATUS register. Any under voltage event is flagged there.  |
| SM13   | Redundant Current Measurements: It is enough to compare the average of 128 FIFO samples between CH1 and CH2. That means it is not necessary to compare every fast (FIFO) sample from CH1 with CH2. The difference between the two approaches (compare every sample or average of FIFOs) is only the section's FTTI (~ 1 ms vs. 100 ms). As the overall FTTI is defined by repetition time of above FSSHT set and by SM14 which is executed only once per 128 samples, there is no disadvantage in case of comparing only the FIFO average values. |
| SM14   | Redundant Computation in Host. This also checks that the PROM values (SM6) are applied correctly. Additionally, the PROM values are read via SoftBypass directly from the analogue die to check them redundantly.   |
| SM16   | Oscillator Test: Replaced by checking delta between TBx (one of TB1-4) updates. TBx is anyway checked periodically for changes to indicate update of slow channel results. The safety mechanism is to check that the difference between two subsequent TBx readings is in the range 100 ms $\pm 5\%$ .  |

#### 9.6.1 4 FSSHT SPI transactions

| ID    | MOSI / MISO                                       | Description                               |
|-------|---|---|
| CONTO | MOSI: FEF5EB50400EE4C6<br>MISO: XXXXXXXXXXXXXXXXX | Write to FACTRL to disable FCM (FACONV=0) |



| ID      | MOSI / MISO   | Description  |
|---------|---|--|
| MUX0    | MOSI: FEF3C7984500013D6E<br>MISO: XXXXXXXXXXXXXXXXXX  | Write two bytes to FAMUXN to select V1 vs. GND (NTC)   |
| ADCV    | MOSI: FB60FADE<br>MISO: XXXXXXXX  | ADCV to trigger conversion   |
| RDCV    | MOSI: F8040970FFFFFFFFFFFFFFFFFFFFFFFFFFFFF<br>MISO: XXXXXXXX010000000000FE4AE8180F0F0F0FC602 | RDCV to read conversion results  |
| MUX1    | MOSI: FEF3C798450016C1BA<br>MISO: XXXXXXXXXXXXXXXXXX  | Write two bytes to FAMUXN to select VREF2 vs. GND  |
| ADCV    | MOSI: FB60FADE<br>MISO: XXXXXXXX  | ADCV to trigger conversion   |
| RDCV    | MOSI: F8040970FFFFFFFFFFFFFFFFFFFFFFFFFFFFF<br>MISO: XXXXXXXX000000000000C212610F0F0F0F0F76B6 | RDCV to read conversion results  |
| MUX2    | MOSI: FEF3C7984517007512<br>MISO: XXXXXXXXXXXXXXXXXX  | Write two bytes to FAMUXN to select GND vs. VREF2_250k   |
| ADCV    | MOSI: FB60FADE<br>MISO: XXXXXXXX  | ADCV to trigger conversion   |
| RDCV    | MOSI: F8040970FFFFFFFFFFFFFFFFFFFFFFFFFFFFF<br>MISO: XXXXXXXX000000000000C212E7180F0F0F0F1A78 | RDCV to read conversion results  |
| MUX3    | MOSI: FEF3C7984500174A88<br>MISO: XXXXXXXXXXXXXXXXXX  | Write two bytes to FAMUXN to select VREF2_250k vs. GND   |
| ADCV    | MOSI: FB60FADE<br>MISO: XXXXXXXX  | ADCV to trigger conversion   |
| RDCV    | MOSI: F8040970FFFFFFFFFFFFFFFFFFFFFFFFFFFFF<br>MISO: XXXXXXXX000000000000C2121AE70F0F0F0F7C26 | RDCV to read conversion results  |
| MUXCONT | MOSI: FEF3C79845111294A6<br>MISO: XXXXXXXXXXXXXXXXXX  | Write two bytes to FAMUXN to select CF2P vs. CF2M (necessary to do the open wire check during the FCM) |
| CONT1   | MOSI: FEF5EB50400F6FF4<br>MISO: XXXXXXXXXXXXXXXXXX  | Write to FACTRL to enable FCM (0x0F = FACONV, FACHA, FACH1, FACH2)                                     |

## Notes:

1. MISO data is only shown as an example and the actual data may change. X means don't care.
2. Between any ADCV and RDCV there must be a delay of  $\geq 1.2\text{ms}$  to be sure the conversion result is valid.
3. The order of the MUX input is chosen on purpose this way to make sure consecutive measurement are different. See following table:

| # | Input | MUX[12]GC Gain correction  | Example measurement | Diagnostic Check Range                                       |
|---|-------|----------------------------|---------------------|--|
| A | MUX0  | 1.0                        | 1.545004            | 0..3 V   |
| B | MUX1  | SM_AA_FP_FAC/PROM_AUX_TRIM | 2.392167            | 2.39V $\pm 2\%$  |
| C | MUX2  | SM_55_FP_FAC/PROM_AUX_TRIM | -2.389916           | -2.39V $\pm 2\%$   |
| D | MUX3  | SM_55_FP_FAC/PROM_AUX_TRIM | 2.391792            | 2.39V $\pm 2\%$  |
| E | B+C   | see above                  | 0.0                 | $\pm 5 \cdot 8\text{nA} \cdot 250\text{k} = \pm 10\text{mV}$ |
| F | B-D   | see above                  | 0.0                 | see above  |

## Notes:

1. The example measurements / nominal ranges were already scaled with the inverse of the "MUX[12]GC Gain correction".
2. E, F are calculated out of measurements B, C, D and allow to detect leakage errors.  
Compared to the leakage test done internally as part of the Accuracy Check (SM4), with this the measurements are also affected by ADC noise and changes over time (the latter being negligible as the fast measurements happen right after each other, meaning within <4 ms).  
For this reason, the tolerance at which an error is flagged, is increased to 40nA (factor 5).

### 9.6.2 Cycling between FCM and open wire check and FSSHTs

ATSR3 requires both current channels to operate in fast continuous mode (FCM). For the open wire check, also the AUX channel will be enabled for FCM. The input will be set to CF2P (18) vs. CF2M (17) before entering FCM.

The FCM will be interrupted periodically (period defined by FTTI) to perform above listed fast single shot (FSSHT) measurements.

The current sources used for the open wire check will be enabled in the last 100 ms period before the FCM will be interrupted, allowing enough time for external filter capacitors to be charged by the current sources. Conversion result of the open-wire check will be read via RDCVs during this part of the FCM period. After the last RDCV necessary for the open wire check the current sources will be disabled again. Following table lists the timing, current source configuration and diagnostic checks for the open wire detection.

| t<br>[ms] | FCURGPIOCTRL  | Diagnostic check                   |
|-----------|---|------------------------------------|
| 15        | Pull-up MUXP, Pull-down MUXN (FCURGPIOCTRL = 0x10 0x20 0x80 written at t=0)       | AUX = I2 = I1 + RISRC<br>±RISRCTOL |
| 30        | Pull-up MUXN, Pull-down MUXP (FCURGPIOCTRL = 0x20 0x40 0x80 written at t=15ms)    | AUX = I2 = I1 – RISRC<br>±RISRCTOL |
| 45        | Pull-up MUXP, Pull-up MUXN (FCURGPIOCTRL = 0x10 0x20 0x40 0x80 written at t=30ms) | AUX = I2 = I1<br>±RISRCTOL         |
| 60        | Pull-down MUXP, Pull-down MUXN (FCURGPIOCTRL = 0x20 0x80 written at t=45ms)       | AUX = I2 = I1<br>±RISRCTOL         |
| 75        | All disabled (FCURGPIOCTRL = 0x00 written at t=60ms)                              | AUX = I2 = I1 ±AUXTOL              |

## Notes:

1. FCURGPIOCTRL must always be written together with the following register FGPICTRL in a single 2-byte write burst. If not GPIOs are used the second byte is always 0.
2. RISRC is the nominal voltage drop across the internal filter resistors due to enabled current source:  $RISRC = 0.5 * (RISRCL + RISRCU) = 19.5 \text{ mV}$
3. RISRCL is the minimum value for RISRC:  $RISRCL = 150e-6 * 40 = 6 \text{ mV}$
4. RISRCU is the maximum value for RISRC:  $RISRCU = 330e-6 * 100 = 33 \text{ mV}$
5.  $RISRCTOL = 0.5 * (RISRCU - RISRCL) = 13.5 \text{ mV}$  or 5% of Abs(I2) whichever is bigger.
6. LSBAUX is 375.183  $\mu\text{V}$ .
7.  $AUXTOL = 5 * LSBAUX$  or 5% of Abs(I2) whichever is bigger.

8. For all measurement it shall be checked that AUX is within current ADCs full scale range  $\pm(\text{RISRCU}-\text{RISRCL})$ . The AUX measurement will for sure exceed this range in case of an open wire.
9. All measurements are done during the FCM. Current source configuration is written at t-15ms and conversion result is read via RDCV at t. This way every current source configuration is static for 15 ms giving enough settling time.
10. Once the current sources are enabled, the comparison between I1 and I2 of the 100ms values done with SM14 will very likely fail and should be skipped.
11. Current sources are enabled on I2 input only. Due to the internal  $2 \times 50\Omega$  filter resistors the current measurements via I2 are biased during that time, see table above. The correct current measurement is still available via I1, also during open wire check. I1 measurement is only altered by the value of the current source itself, 250 $\mu$ A typical. In most applications this current value is far below the resolution e.g. 19mA (slow) / 152mA (fast) for a 50 $\mu\Omega$  shunt and can be neglected.
12. All calculations above in volts or LSBs. In case the checks are done in real physical quantities, the voltage values must be scaled with 1/Rsns. E.g.  $\text{RISRCTOL} = 0.5 * (\text{RISRCU} - \text{RISRCL}) = 13.5\text{mV}$  in amps becomes  $13.5\text{mV} / \text{Rsns}$ .

### 9.6.3 Checks for latent faults

The actual safety mechanisms need to be verified for correct functionality. This needs to be done either only once per initialization phase. Or it happens intrinsically via the safety mechanism or due to full redundancy.

| Type/SM | Description   |
|---------|---|
| SM1     | <p>The current sources, which are used for the open wire detection, can be checked once at boot-up via the external NTC: (assuming the NTC is connected to V1)</p> <ol style="list-style-type: none"> <li>1. Perform NTC measurement (V1 vs. GND, no current sources active) to acquire voltage across NTC (VNTC) and NTC resistance (RNTC) <ol style="list-style-type: none"> <li>a. Check that the NTC voltage is &gt;0V and &lt;VREF (sanity check, the NTC is never 0R nor infinite resistance in normal operation)</li> </ol> </li> <li>2. Calculate effective resistance of parallel combination RPAR of RNTC and RRef (for RRef=10k and NTC=10k at room temperature this will be 5k. Worst case the resistance will be 10k at lowest temperature or 220<math>\Omega</math> at highest temperature (taking the R(T) of some typical 10k NTC), depends on used NTC.</li> </ol> <p>Example:</p> <pre>&gt;&gt;&gt; RRef=10e3;RNTC=10e3;RPAR=RRef*RNTC/(RRef+RNTC)       ➔ RPAR=5000 &gt;&gt;&gt; RNTC=222.6...       ➔ RPAR=218 &gt;&gt;&gt; RNTC=969e3...       ➔ RPAR=9898</pre> <ol style="list-style-type: none"> <li>3. Measure V1 vs. GND, pull-up current source on positive input enabled (pull-up MUXP: FCURGPIOCTRL = 0x10 0x20, VNTCa)</li> </ol> |

| Type/SM  | Description   |
|--|---|
|  | <ul style="list-style-type: none"> <li>a. Check <math>VNTCa = VNTC + ISRC * RPAR</math></li> <li>4. Measure V1 vs. GND pull-down current sources on positive input enabled (pull-down MUXP: FCURGPIOCTRL = 0x20, VNTCb) <ul style="list-style-type: none"> <li>a. Check <math>VNTCb = VNTC - ISRC * RPAR</math></li> </ul> </li> <li>5. Measure GND vs. V1, pull-up current source on negative input enabled (pull-up MUXN: FCURGPIOCTRL = 0x40 0x80, VNTCc) <ul style="list-style-type: none"> <li>a. Check <math>VNTCc = -VNTC - ISRC * RPAR</math></li> </ul> </li> <li>6. Measure GND vs. V1, pull-down current source on negative input enabled (pull-down MUXN: FCURGPIOCTRL = 0x80, VNTCd) <ul style="list-style-type: none"> <li>a. Check <math>VNTCd = -VNTC + ISRC * RPAR</math></li> </ul> </li> </ul> <p>Notes:</p> <ul style="list-style-type: none"> <li>1. The nominal current source value is <math>ISRC=250e-6</math>.</li> <li>2. For all measurements the abs. max. limit is the ADC's full scale and the minimum is 0V which means, the expected values must be clipped in case they exceed those limits.</li> <li>3. The absolute tolerance for the measurements is <math>(330e-6 - 150e-6) * RPAR</math> or 5% of the measured value, whichever is bigger.</li> </ul> |
| SM6  | Done within SM14 via comparison of PROM values read from analog die with those reported in the memory.  |
| SM7  | No check needed, due to redundancy at the higher level safety mechanism (comparison of FIFO I1 with FIFO I2)  |
| SM8  | Error injector available (SM19)   |
| SM9  | Send PEC error command, check for error in STATUS<br>The other way is (Slave to Master) not necessary (must be ensured on user side)  |
| SM13   | Due to full redundancy of the two current channels, a latent fault check for SM13 is not necessary.   |
| SM14   | Latent faults either discovered by SM13 or intrinsically by the comparison of the slow channel versus the fast channel.   |
| SM16   | Latent faults are checked intrinsically (here: The register value must change, for the check to be successful)  |
| Set of FSSHT to combine relevant sections of SM4, SM15, SM17 | Not necessary as checked intrinsically via alternating and well-known input signals that are measured, here -VREF2, VREF2.  |
| SM23   | Hand Shake Byte:<br>Check of successful FSSHT measurements within SM17 is done intrinsically as subsequent measurements (even the non-safety relevant NTC meas. due to the used gain correction values 1.66... and 1.33...) are expected to be different.   |

#### 9.6.4 Expected error on coulomb counting

During FSSHT measurements the coulomb counting is suspended which leads to errors. The requirement is <3% error for 1h intervals. Below table shows the number of FSSHT measurements allowed per slow channel updates (multiple of 100ms) without exceeding this error: In case of internal oscillator 5 FSSHTs can be performed for an FTTI of <1000ms.

Still it is possible to compensate for the error over such long-time intervals like 1h:

1. Take the average current that was measured by the LTC2949 in a certain, long time, e.g. 1h  
( $\text{Current\_Average\_2949} = \text{Charge\_2949} / \text{Time\_2949}$ )
2. I take the reference time that was measured by the host controller / master module  
( $\text{Time\_Master}$ ) and calculate the time the LTC2949 was not accumulating
  - a.  $\text{Time\_Master} - \text{Time\_2949} = \text{Time\_blind}$
3. Add the missing charge
  - a.  $\text{Charge\_corrected} = \text{Charge\_2949} + \text{Time\_blind} * \text{Current\_Average\_2949}$

Note: The intermediate step of calculating the average current can be skipped. The following formula directly calculates the corrected charge:

$$\text{Charge\_corrected} = \text{Charge\_2949} * \text{Time\_Master} / \text{Time\_2949}$$

With  $\text{Time\_Master} / \text{Time\_2949} = \text{ChargeGC}$  the correction factor.

This correction is statistically correct if the time is long enough, as the time the LTC2949 was not accumulating / blind is very small and equally distributed over the full accumulation time. One hour should be long enough for this correction to be valid.

calculation of max allowed fast SSHT time:

|   |       |    |
|---|-------|----|
| FTTI target                                 | 1000  | ms |
| max allowed err for CC                      | 3.00% |    |
| max allowed ADCV induced CC err for int osc | 1.50% |    |
| max allowed ADCV induced CC err for ext osc | 2.25% |    |
| assumed time per ADCV                       | 2     | ms |
| assumed offset time for all ADCVs           | 2     | ms |

| N  | CC time [ms] | tADCV int [ms] | N-ADCVs int | FTTI int [ms] | tADCV ext [ms] | N-ADCVs ext | FTTI ext [ms] |
|----|--------------|----------------|-------------|---------------|----------------|-------------|---------------|
| 4  | 400          | 6.1            | 2           | 406.1         | 9.2            | 3           | 409.2         |
| 5  | 500          | 7.6            | 2           | 507.6         | 11.5           | 4           | 511.5         |
| 6  | 600          | 9.1            | 3           | 609.1         | 13.8           | 5           | 613.8         |
| 7  | 700          | 10.7           | 4           | 710.7         | 16.1           | 7           | 716.1         |
| 8  | 800          | 12.2           | 5           | 812.2         | 18.4           | 8           | 818.4         |
| 9  | 900          | 13.7           | 5           | 913.7         | 20.7           | 9           | 920.7         |
| 10 | 1000         | 15.2           | 6           | 1015.2        | 23.0           | 10          | 1023.0        |

Note:

The right correction factor can only be calculated after the measurement was running for some time. To be able to estimate the correction factor better than 0.1%, roughly 1000 measurement periods of 100ms each are needed, leading to a time of 100 seconds. To overcome this initial



phase till the right factor can be calculated, it's possible to just use the estimated factor. E.g. if the fast continuous mode is nominally interrupted 930 ms after FAONV was set and the FSSHT phase takes 10 ms the coulomb counter was enabled 900 ms but the actual time (MCU time) was 940 ms, which leads to a nominal correction factor of  $\text{ChargeGC} = 0.94/0.9 = 1.0444\dots$  To avoid steps in the correction factor, when going from the nominal one to the real calculated one, it's also possible to implement a smooth transition over a certain period.

It's also possible to implement another charge accumulator inside the host MCU which is incremented by the delta between two reads of the LTC2949's charge register (after being compensated according to above note). To avoid errors due to charge over- or under-flow, the LTC2949's overflow status bits (C1OVF to C3OVF) can be used to indicate when the charge register crossed 90% of its minimum or maximum value. Once that happens it is recommended to reset the charge and time register to:

$$\text{NEW\_Charge\_2949} = \text{Charge\_2949} / \text{Time\_2949} * 1000 * 0.1 \text{ seconds} / \text{ChargeGC}$$

$$\text{NEW\_Time\_2949} = 1000 * 0.1 \text{ seconds} / \text{ChargeGC}$$

$$\text{NEW\_Time\_Master} = 1000 * 0.1 \text{ seconds}$$

The new register values for Cx and TBx (e.g. typically C1 and TB1 for single shunt systems) must be written in one single burst to the LTC2949 to ensure coherency.

The new starting time (and thus charge) is chosen, to have the calculated average current better than 0.1% as the new starting value for the compensation (note: See compensation factor

**Time\_Master / Time\_2949** which becomes exactly ChargeGC after those values were written).

This way the LTC2949's charge register can be treated totally independent from the charge register that is stored in the host controller and the charge compensation is always better than 0.1% once 100 seconds have elapsed since the LTC2949 was reset / initialized.

## 9.7 APPENDIX SLOW CHANNEL REGISTERS

It is important to mention that the update of all slow channel results including STATUS/FAULTS/EXTFAULTS registers is linked to the slow channel current EOC. That means e.g. in the scenario where both channels are fast and the user makes fast single shot measurements without any interruption of more than 100ms, there will never be any slow channel EOC, thus also STATUS/FAULTS/EXTFAULTS registers (e.g. reporting UVLOs, EXTCOMERR, INTCOMERR and other fault bits) will never be updated.

## 9.8 APPENDIX SOFTBYPASS

The LTC2949 features a communication channel to the internal serial interface of the analog die via the external serial interface, allowing to access registers within the analog die. This communication channel is called the SoftBypass. As default the access is locked and can be unlocked via writing following unlock code:

```
// debug register TST
#define LTC2949_REG_DBG_TST 0xFE
// Enter SoftBypass
void LTC2949_EnterDebug()
{
    LTC2949_WRITE(LTC2949_REG_DBG_TST, (byte)'d'); // write 0x64 to 0xFE
    LTC2949_WRITE(LTC2949_REG_DBG_TST, (byte)'e'); // write 0x65 to 0xFE
    LTC2949_WRITE(LTC2949_REG_DBG_TST, (byte)'B'); // write 0x42 to 0xFE
}
```



To leave the SoftBypass, close the communication channel and lock it again following code is used:

```
// Leave SoftBypass
void LTC2949_ExitDebug()
{
    LTC2949_WRITE(LTC2949_REG_DBG_TST, (byte)'B'); // write 0x42 to 0xFE
    LTC2949_WRITE(LTC2949_REG_DBG_TST, (byte)'e'); // write 0x65 to 0xFE
    LTC2949_WRITE(LTC2949_REG_DBG_TST, (byte)'d'); // write 0x64 to 0xFE
}
```

Once SoftBypass is enabled, the master can read and write registers of the analog die.

Read operation:

```
// debug register FW
#define LTC2949_REG_DBG_FW 0xD0
// debug register DBG_0 on page 1
#define LTC2949_REG_DBG_0 0x140
// debug register DBG_3 on page 1
#define LTC2949_REG_DBG_3 0x143
// read register from analog die
uint8_t LTC2949_SwBypRead(uint8_t address, uint32_t * ui24)
{
    uint8_t data[3];
    uint8_t error;
    LTC2949_WRITE(LTC2949_REG_DBG_3, address); // write analog die register address
    LTC2949_WRITE(LTC2949_REG_DBG_FW, 4); // write operation code to DBG_FW: 4=READ
    error = LTC2949_SwBypPollDone(); // poll for end of operation
    error |= LTC2949_READ(LTC2949_REG_DBG_0, 3, data); // read result of operation (analog die's register content)
    // put all bytes into single 24 (32) bit word
    *ui24 = (((uint32_t)data[0]) << 16) | (((uint32_t)data[1]) << 8) | ((uint32_t)data[2]));
    return error;
}
```

Write operation:

```
// write to analog die register
uint8_t LTC2949_SwBypWrite(uint8_t address, uint32_t ui24)
{
    uint8_t data[4];
    // data to be written (24 bit)
    data[0] = ui24 >> 16;
    data[1] = ui24 >> 8;
    data[2] = ui24;
    // analog die register address
    data[3] = address;
    LTC2949_WRITE(LTC2949_REG_DBG_0, 4, &data[0]); // write data and address via 4-byte write burst
    LTC2949_WRITE(LTC2949_REG_DBG_FW, 3); // write operation code to DBG_FW: 3=WRITE
    return LTC2949_SwBypPollDone(); // poll for end of operation
}
```

The host needs to poll for completion within the Read and Write operations shown above. This is done with following code:

```
// Poll for SoftBypass operation done
static inline uint8_t LTC2949_SwBypPollDone()
{
    // operation is done once 0xD0 is read as 0x00
    return LTC2949_PollReg(LTC2949_REG_DBG_FW, 0x00);
}
```

As an example, to read the factory trims of AUX, I1 and I2 ADCs following code can be used:

```
// read factory trims (gain correction factors) for AUX, I1, I2 ADCs
uint8_t LTC2949_SwBypGetTrims(int8_t * gcaux, int8_t * gcil, int8_t * gcil2)
{
    uint32_t ui24; uint8_t error;
    error = LTC2949_SwBypPROMReadEnable(); // enable access to PROM
    error |= LTC2949_SwBypRead(0x50, &ui24); // read AUX GC
    *gcaux = (int8_t)((ui24 & 0xff0000) >> 8) & 0xff;
    error |= LTC2949_SwBypRead(0x51, &ui24); // read I1 GC
    *gcil = (int8_t)((ui24 & 0xff0000) >> 8) & 0xff;
    error |= LTC2949_SwBypRead(0x52, &ui24); // read I2 GC
    *gcil2 = (int8_t)((ui24 & 0xff0000) >> 8) & 0xff;
    error |= LTC2949_SwBypPROMReadDisable(); // disable access to PROM
    return error;
}
// enable access to PROM
static inline uint8_t LTC2949_SwBypPROMReadEnable()
{
    // set READ ENABLE of the PROM
    return LTC2949_SwBypWrite(
        0x60, // address of PROM CONTROL REG (96=0x60)
        0x01 // bit[0] is READ ENABLE BIT
    );
}
```

```

}
// disable access to PROM
static inline uint8_t LTC2949_SwBypPROMReadDisable()
{
    // clear READ ENABLE of the PROM
    return LTC2949_SwBypWrite(
        0x60, // address of PROM CONTROL REG (96=0x60)
        0x00 // bit[0] is READ ENABLE BIT
    );
}

```

To clear / set dedicated bits of an analog die register a read-modify-write operation must be used:

```

// set bits within analog die register
uint8_t LTC2949_SwBypSetBits(uint8_t address, uint32_t bits)
{
    uint32_t ui24; uint8_t error;
    error = LTC2949_SwBypRead(address, &ui24);
    ui24 |= bits;
    error |= LTC2949_SwBypWrite(address, ui24);
    return error;
}

// clear bits within analog die register
uint8_t LTC2949_SwBypClrBits(uint8_t address, uint32_t bits)
{
    uint32_t ui24; uint8_t error;
    error = LTC2949_SwBypRead(address, &ui24);
    ui24 &= ~bits;
    error |= LTC2949_SwBypWrite(address, ui24);
    return error;
}

```

## 10 Dedicated Measures – AEC-Q100 Qualification and Automotive Flow

ADI has qualified this product based on reliability stress tests defined by the Automotive Electronics Council (AEC). Linear guarantees that this product meets or exceeds current AEC-Q100 requirements (qualification guidelines for integrated circuits) by conducting additional device and package level stress tests. These additional tests may include but are not limited to: power cycling, liquid- to-liquid thermal shock, instant solder shock, extended duration autoclave tests, and 100% oxide stress tests. These unique and rigorous tests validate the robustness of ADI's products prior to production release.

Additional dedicated measures are available for lots identified for automotive flow. Our automotive flows were designed to eliminate test escapes and infant mortality failures. Automotive flows incorporate additional screening and inspections and/or tightened criteria in wafer fab, assembly, electrical test, and reliability monitor.

## 11 Quantitative Analysis Results

Evaluation of the hardware architectural metrics assumes that the LTC2949 is used within the scope of the stated assumptions. The system must utilize the diagnostic features of the LTC2949 to ensure that the reported measurements are not in error. A FMEDA was the quantitative analysis chosen.

### 11.1 METRIC CALCULATIONS

Using the assumptions as outlined in section 5.2, the following metrics were calculated. The total FIT was calculated using IEC 62380:

| FSR  | Total Fault/FIT<br>$\lambda$ | Safety-Related Fault/FIT<br>$\lambda_{SR}$ | SPF, Residual Fault/FIT<br>$\lambda_{SPF}, \lambda_{RF}$ | Latent MPF/FIT<br>$\lambda_{MPF,I}$ | Detected MPF/FIT<br>$\lambda_{MPF,det}$ | SPFM/% | LFM/% | PMHF<br>( $T_{Lifetime} = 15 \text{ yrs}$ ) |
|------|------------------------------|--|--|-------------------------------------|---|--------|-------|---|
| FSR1 | 33.13                        | 21.68                                      | 0.19   | 0.003                               | 18.5                                    | 99.1   | 99.9  | 0.2   |
| FSR2 | 33.13                        | 13.72                                      | 0.09   | 0.003                               | 10.4                                    | 99.3   | 99.9  | 0.09  |
| FSR3 | 33.13                        | 16.1                                       | 0.11   | 0.006                               | 12.1                                    | 99.3   | 99.9  | 0.1   |

## 12 Functional Safety Considerations

In addition to this safety manual, an LTC2949 Level 3 PPAP document is also available to customers under an NDA. The Level 3 PPAP document includes AEC-Q100 qualification data, quality systems compliance in design, operations, and manufacturing of ICs.

Please contact your local ADI sales office for further information on ISO 26262 documentation.

<http://www.analog.com/en/about-adi/contact-us.html>

Prepared for Intipower  
Analog Devices Confidential