ANALOG DEVICES

ADA4558 Hardware Reference Manual

UG-1487

One Technology Way • P.O. Box 9106 • Norwood, MA 02062-9106, U.S.A. • Tel: 781.329.4700 • Fax: 781.461.3113 • www.analog.com

ADA4558 Sensor Signal Conditioner for Bridge Sensors

SCOPE

This reference manual provides a detailed description of the functionality and features of the ADA4558.

Full specifications on the ADA4558 are available in the ADA4558 data sheet. Consult the data sheet in conjunction with this reference manual when working with this device.

TABLE OF CONTENTS

REVISION HISTORY

5/2019—Revision 0: Initial Version

ADA4558 OVERVIEW

The ADA4558 is an automotive grade, signal conditioner IC that interfaces to a range of strain gauge sensors, based on a Wheatstone bridge output. The ADA4558 senses the resistive changes in a Wheatstone bridge sensor that are proportional to the force applied to the sensor. A typical application for the ADA4558 is pressure sensing, where the sensor forms a diaphragm type pressure transducer. The measurable signal from the sensor contains errors due to mechanical, thermal, tolerance, and gauge factor errors.

The ADA4558 is a configurable device that nulls out the non-ideal behaviors of the sensor. The device provides signal conditioning to compensate for sensor offset errors, voltage span variation, sensitivity linearity, and temperature dependency. To provide temperature compensation, the ADA4558 can be configured to use either an internal temperature sensor or an external resistance temperature detector (RTD). The ADA4558 provides signal conditioning to compensate for first- or second-order error correction of temperature and sensitivity. Figure 2 shows a simplified block diagram of the ADA4558. The analog front end (AFE) contains circuitry to apply filters, gain, and offset adjust to the input signal. The digital block contains a correction algorithm to apply the linearization correction for temperature and sensitivity. The digital block provides the local interconnect network (LIN) protocol for interface logic for reading the corrected signal measurements.

The corrected measurements are mapped to a digital 12-bit code, where strain gauge minimum to maximum measurements can be mapped to Code 0 to Code 4095. The system accuracy is 0.1% full-scale range (FSR).

Configuration mode allows calibration and initial setup tasks where an end of line calibration can be run. Perform an end of line (EOL) calibration where the strain gauge sensor is connected to the ADA4558. A calibration routine generates coefficients that are stored in electrically erasable programmable read only memory (EEPROM), and used by the correction algorithm.

In normal operating mode, the ADA4558 applies coarse signal conditioning to maximize analog-to-digital converter (ADC) range, and uses a digital correction algorithm to apply a fine signal conditioning and linearization. The device operates as a LIN slave device and communicates the bridge sensor and temperature measurements over the LIN interface to the master engine control unit (ECU).

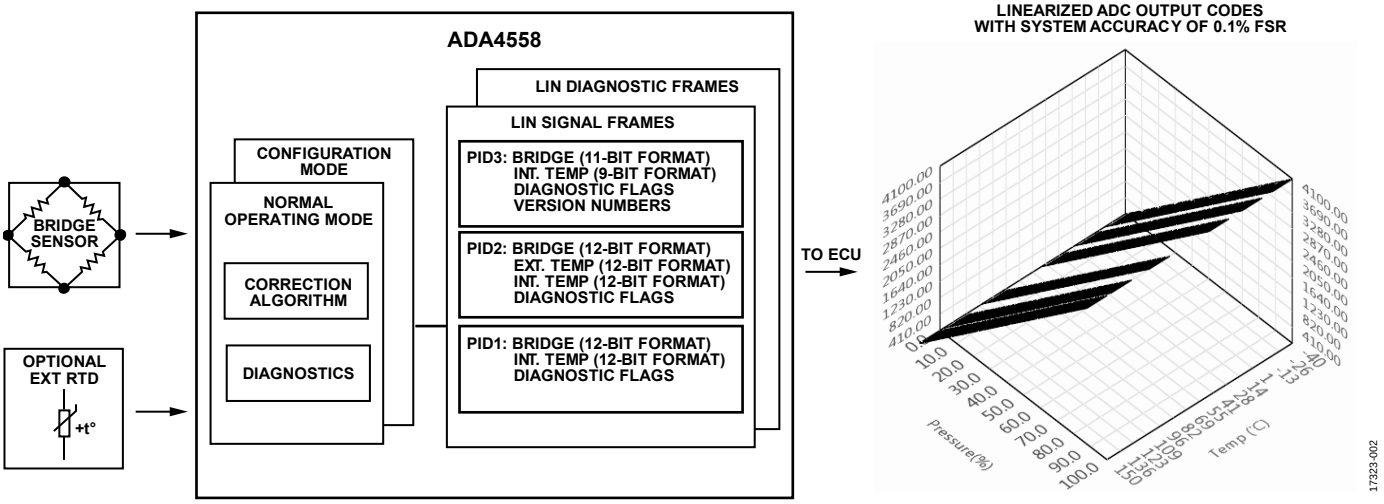

Figure 1. ADA4558 Operational Overview

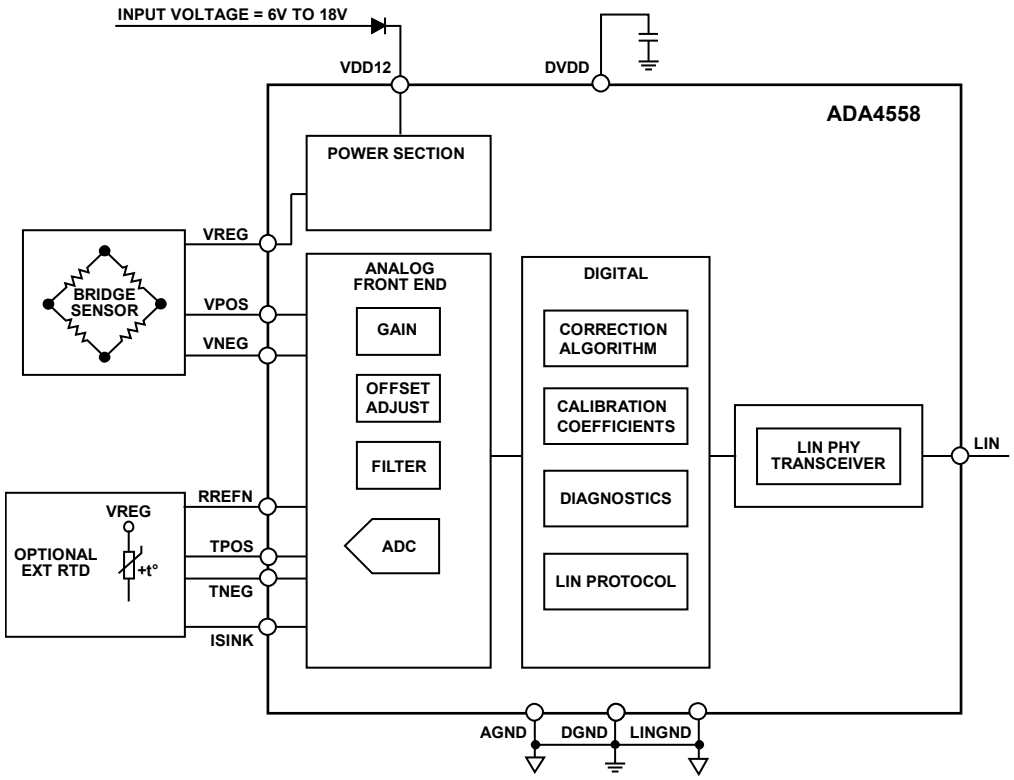

Figure 2. Simplified Block Diagram

BRIDGE SENSOR INPUT

The differential signal from the bridge sensor is measured across the VPOS pin and VNEG pin of the ADA4558 (see Figure 3). The ADA4558 includes a 4 V regulator output at the VREG pin. The VREG pin supplies power to the bridge sensor, the external temperature sensor, and the ADC reference This supply voltage creates a fully ratiometric measurement system.

Figure 2 shows a simplified block diagram of the ADA4558 AFE with the bridge sensor input. The bridge sensor input consists of the following components:

- Electromagnetic interference (EMI) filters
- Optional polarity switching
- Programmable gain amplifier (PGA) gain with offset correction
- Antialiasing filters
- Signal buffers

EMI FILTERS

The ADA4558 features IC level resistor-capacitor (RC) filters on the PGA inputs. The differential filter has a 3 dB cutoff frequency (f_{-3} dB) of 550 kHz and the common-mode filter has an f_{-3} dB of 580 kHz. The filters provide 20 dB per decade of roll-off on injected EMI signals.

INVERTING SWITCH OPERATION

An inverting switch connects the PGA inputs, the VPOS pin, and VNEG pin. When enabled in EEPROM, the switch reverses the PGA connection between the VPOS pin and the VNEG pin. This feature provides flexibility in the bonding of the bridge to the printed circuit board (PCB) to simplify connection and improve PCB layout.

PGA

A three-stage PGA at the front end provides offset correction and gain of the bridge sensor signal. The range of the coarse PGA gain and offset settings allows accurate operation with a wide range of bridge sensitivities. Configure the PGA for system offset voltage, V_{CM}, and PGA gain with the following settings in the EEPROM:

- BRG_VOS_TRIM1[6:0]: System offset voltage trim for PGA Stage 1 (S1).
- BRG_VOS_TRIM2[6:0]: System offset voltage trim for PGA Stage 2 (S2).
- PGA_DAC_5[6:0]: System offset voltage trim for PGA Stage 3 (S3).
- BRG_VCM_TRIM[6:0]: Common-mode voltage trim.
- PGA_GAIN[4:0]: PGA gain setting.

Offset Correction

During EOL calibration, the bridge sensor is set to the zero strain signal level and the system offset voltage is measured. Use PGA DACs to apply biasing to adjust the offset voltage to the desired levels. These PGA DAC codes are saved to the EEPROM. The PGA DAC codes used must be greater than the listed minimum PGA DAC codes shown in Table 2. This is a coarse offset adjust for common mode, and removes positive and negative bridge offsets.

Gain Settings

The PGA has 24 programmable gain settings, and can be set with PGA_GAIN [4:0] in the EEPROM. See Table 1 for the values used to configure the PGA gain.

The first column of Table 1 shows the settings necessary to program to PGA_GAIN bits in the EEPROM. The second column in Table 1 shows the selected gain of the PGA. The third column of Table 1 shows the maximum allowed differential input voltage to the VPOS pin and the VNEG pin for each gain setting.

The output of the PGA is the gained signal from the bridge sensor with sensor offset correction and V_{CM} offset correction. The VDIFF PGA output is applied to the ADC. Select the necessary PGA gain to maximize the dynamic range of the ADC.

The PGA gain settings provide a coarse gain adjust for the ADC input. Fine gain adjustments and linearization are performed by the digital correction algorithm.

Table 1. PGA Gain Settings and Range

PGA Gain Settings, PGA_GAIN[4:0]	Nominal PGA Gain (V)	Maximum Sensor Input Span (V)
00000	2.94	1.253805
00001	3.8	0.970109
00010	5.0	0.740447
00011	6.3	0.579944
00100	8.1	0.455537
00101	11.2	0.327793
00110	11.8	0.310175
00111	15.1	0.240339
10000	20.1	0.180348
10001	25.0	0.144312
10010	32.5	0.110424
10011	44.9	0.079112
10100	58.9	0.059942
10101	75.4	0.046314
10110	100.3	0.034592
10111	125.2	0.027282
11000	162.52	0.020524
11001	224.72	0.014165
11010	274	0.011194
11011	349.1	0.008343
11100	473.5	0.005731
11101	573.0	0.004252
11110	722.3	0.002910
11111	971.10	0.001582

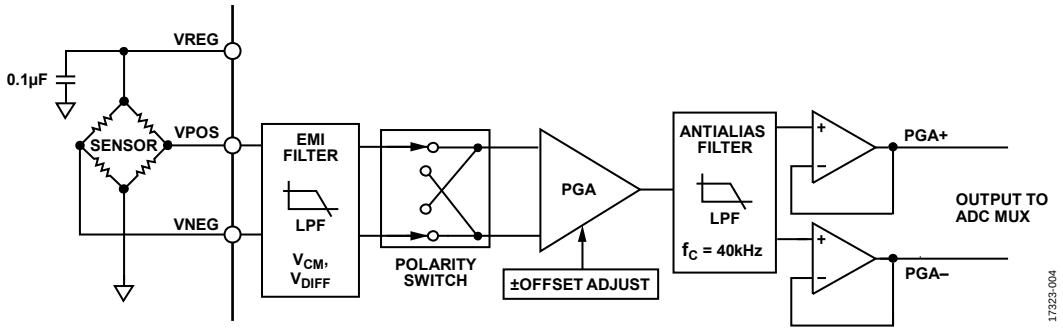

Figure 3. AFE for Bridge Sensor Input

The PGA output on the ADA4558 uses chopper stabilization to improve offset performance. To ensure that the PGA output does not saturate at the rails, select the PGA gain such that the output (including the gained up, residual, unchopped offset) always remains within the power supply rails. The residual, unchopped offset is approximately 1 mV.

ANTIALIAS LOW-PASS FILTER

The output of the PGA is fed through an antialias, low-pass filter with an f_{-3} dB of 40 kHz, which limits the bandwidth of the sampled analog signal at the ADC.

This output is buffered and provided to the ADC input through PGA+ and PGA−. The PGA− input to the ADC is fixed at 6.25% regulated 4.0 V output at pin 11(V_{REG}). Note that the ADC converts the difference between the PGA+ and PGA− when estimating the ADC output for a given PGA input voltage.

Table 2. PGA DAC Minimum Usable Code

PGA DAC	Minimal DAC Code
PGADAC1	15
PGADAC2	15
PGADAC3	12
PGADAC4	12
PGADAC5	2

INTERNAL TEMPERATURE SENSOR

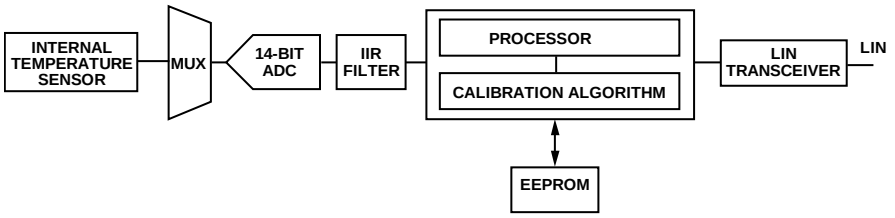

Figure 4. Internal Temperature Sensor

INTERNAL TEMPERATURE SENSOR FOR BRIDGE SENSOR CALIBRATION

The ADA4558 has an on-board, silicon diode-based temperature sensor that reads back the die temperature. The internal temperature sensor block is read by the ADC at a 50 kHz sample rate, and then filtered through a 125 Hz bandwidth. Further filtering is possible, as described in the User Configurable Filtering section. This temperature channel, raw ADC data is saved to the SYSREG Address 0x91 (ADC_DATA_TEMP).

The ADA4558 uses the die temperature reading to approximate the bridge sensor temperature in applications where the sensor and IC are in close proximity. Configure the ADA4558 to use the ADC_DATA_TEMP value for the bridge sensor correction of temperature dependent effects by writing 0 to the TEMP_SEL bit in the EEPROM Address 13 (0x0D).

INTERNAL TEMPERATURE SENSOR OUTPUT TO LIN SIGNAL FRAME

A 9-bit or 12-bit representation of the die temperature reading is available over the LIN signal frames where the ADC code represents a maximum temperature range of −40°C to +150°C. To calibrate the internal temperature reported via LIN, use the INT_TCOEFF0 and INT_TCOEFF1 values stored in the EEPROM. The following equation is the transfer function for the internal temperature output to LIN signal frames. Note that this calibration is not required for nonlinear correction of the bridge sensor channel input.

$$Internal\ Temperature = (INT\_TCOEFF0 + INT\_TCOEFF1 \times ADC\_DATA\_TEMP \times 4095 \quad (1)$$

where:
INT_TCOEFF0 = Q0.12.
INT_TCOEFF1 = Q3.9.
ADC_DATA_TEMP is normalized in the 0 to 1 range.

Both INT_TCOEFF0 and INT_TCOEFF1 are twos complement. The Qx.y format represents integer (x bits) + fraction (y bits) in twos complement format.

For example, the 14-bit ADC_DAT_TEMP register varies in ADC codes from 1720 to 8840 over the −40°C to +150°C range. Use the following equations to calculate the internal temperature coefficients for LIN signal frame:

$$INT\_TCOEFF1\ Floating\ Point = 2^{14} \div (ADC\ High\ Code - ADC\ Low\ Code) \quad (2)$$

$$INT\_TCOEFF0\ Floating\ Point = -(ADC\ Low\ Code \div 2^{14}) \times INT\_TCOEFF1\ Floating\ Point \quad (3)$$

$$INT\_TCOEFF1 = \text{floor}\ ((INT\_TCOEFF1\ Floating\ Point \times 511) + 0.5) \quad (4)$$

$$INT\_TCOEFF0 = \text{floor}\ ((INT\_TCOEFF0\ Floating\ Point \times 4095) + 0.5) \quad (5)$$

where (for this example):
INT_TCOEFF1 Floating Point = 2.30112359
INT_TCOEFF0 Floating Point = −0.24157303
INT_TCOEFF1 = 1176 decimal (0x498)
INT_TCOEFF0 = −989 decimal (0xC23)
ADC Low Code = 1720
ADC High Code = 8840

The EVAL-ADA4558EBZ software provides a tool for applying further examples of calculating the INT_TCOEFF0 and INT_TCOEFF1 values.

EXTERNAL TEMPERATURE SENSOR

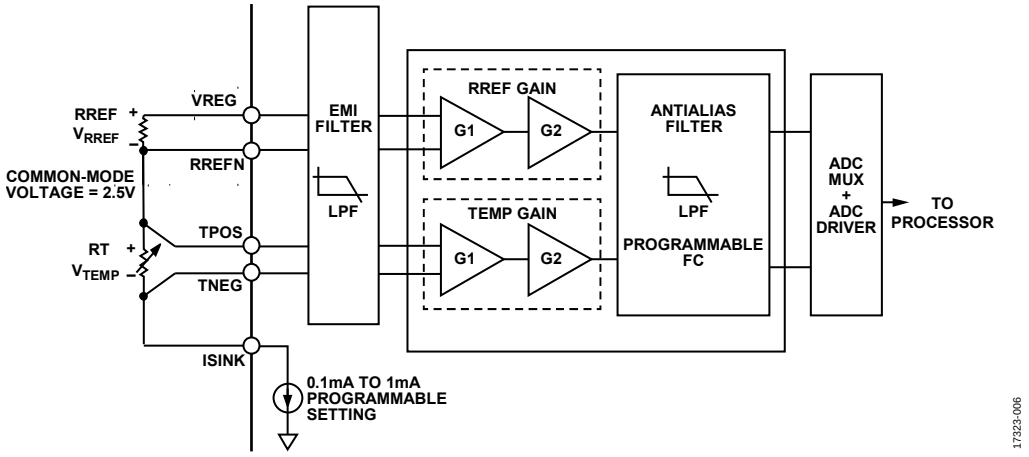

Figure 5. External Temperature Sensor

The ADA4558 has an external temperature sensor input to allow the temperature compensation of remote sensors. Write 1 to the TEMP_SEL bit in EEPROM Address 13 (0x0D) to configure the ADA4558 to use the external temperature sensor value to temperature compensate for the bridge sensor.

Remote RTDs are measured across the TPOS pin and the TNEG pin, as shown in Figure 5. A local reference resistor is measured in series to provide a reference temperature. The voltage across the RREF resistor (V_{RREF}) and the voltage across the RT resistor (V_{TEMP}) are amplified, filtered, and applied to the ADC of the IC via the on-chip multiplexer (MUX).

The ADA4558 supports the use of the platinum 100 (100 Ω at 0°C, PT100) RTD or platinum 1000 (1000 Ω at 0°C, PT1000) RTD. The external temperature sensor block is read by the ADC at a 50 kHz sample rate and then filtered through a 125 Hz bandwidth. Further filtering is described in the User Configurable Filtering section. This external temperature channel saves raw ADC data to the SYSREG Address 0x92 (ADC_DATA_EXTTS).

An on-chip sink current (I_{SINK}) generates the voltages across the RTD and reference resistor. The I_{SINK} is programmed to ensure a wide signal range for the RTD over the expected temperature range. Let I_{SINK} = 1 mA for PT100, and let I_{SINK} = 0.2 mA for PT1000. See Table 3 for the full list of programmable sink currents available.

The V_{RREF} voltage is used in on-chip calculations to reduce the dependence on the internal bias current drift over temperature. It is critical for system accuracy that the RREF reference resistor is a precision resistor with stable temperature characteristics. Choose a location for the RREF reference resistor on the PCB that avoids heating of the RREF reference resistor caused by power dissipation in the application.

The RTD is remotely located to accurately sense the target temperature. Connect the TPOS pin and the TNEG pin to the RTD using the Kelvin method (matched local connections) to ensure maximum accuracy.

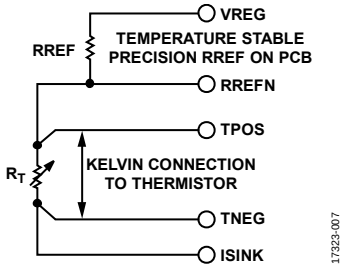

Figure 6. External Temperature Sensor Connection Guideline

As shown in Figure 5, the RREF reference resistor and RTD inputs have a programmable gain setting where the total gain is G1 × G2 for each channel. See Table 3 for the full list of programmable gain settings. Select the gain setting of G1 to maximize the signal without saturating the ADC. Select the gain setting of G2 to further maximize the signal without saturation. The recommended ADC input range is 0.1 V to V_{REG} (4 V).

Calibrate and scale the external temperature sensor input to allow the calibration routine to produce five coefficients that must be saved to the EEPROM. See the Calibration Procedure section for details of the calibration routine. The calibrated external temperature data is reported back to the application in an LIN signal frame.

Table 3. External Temperature Front End Settings

Parameter	Description	Programmable Values
EXTTS_ISNK	Programmable I_{SINK}	3-bit binary value (3'b) 0xx = 0 mA 3'b 100 = 0.1 mA 3'b 101 = 0.2 mA 3'b 110 = 0.5 mA 3'b 111 = 1 mA
EXTTS_REF_GAIN1	Reference resistor, first stage gain	3'b 000 = 1 V/V 3'b 001 = 1.5 V/V 3'b 010 = 2 V/V 3'b 011 = 3 V/V 3'b 100 = 4 V/V 3'b 101 = 6 V/V 3'b 110 = 8 V/V 3'b 111 = 12 V/V
EXTTS_REF_GAIN2	Reference resistor, second stage gain	2-bit binary value (2'b) 00 = 1 V/V 2'b 01 = 1.5 V/V 2'b 10 = 2 V/V 2'b 11 = 3 V/V
EXTTS_TEMP_GAIN1	RTD, first stage gain	3'b 000 = 1 V/V 3'b 001 = 1.5 V/V 3'b 010 = 2 V/V 3'b 011 = 3 V/V 3'b 100 = 4 V/V 3'b 101 = 6 V/V 3'b 110 = 8 V/V 3'b 111 = 12 V/V
EXTTS_TEMP_GAIN2	RTD, second stage gain	2'b 00 = 1 V/V 2'b 01 = 1.5 V/V 2'b 10 = 2 V/V 2'b 11 = 3 V/V

ADC OPERATION

The ADA4558 features a 14-bit successive approximation register (SAR) ADC. The ADC converts the output of the PGA, the internal and external temperature sensor, $V_{DD} \div 2$, bridge input V_{CM}, on-chip coarse band gap voltage, internal 1.8 V analog and digital supply (DVDD), and other miscellaneous signals. The bridge sensor channel is sampled at 200 kHz, the temperature sensor is sampled at 50 kHz, and all remaining ADC time slots are allocated for self checks, fault detection, and other functions.

The ADC output for the bridge sensor is filtered through a 500 Hz cutoff digital infinite impulse response (IIR) filter in the datapath. Further filtering is described in the User Configurable Filtering section.

The differential measurement for the bridge sensor PGA Stage 1 is stored in the SYSREG Address 0x94 (ADC_DATA_PRES_S1). PGA Stage 3 outputs a single-ended measurement for the bridge sensor, which is saved to the SYSREG Address 0x90 (ADC_DATA_PRES). These bridge sensor registers are not subject to the digital correction algorithm.

The digital correction algorithm is applied by the processor when a LIN request is received. The corrected bridge sensor measurement is sent to the system master over the LIN compliant interface

The ADC output for various channels is accessible via the LIN interface by using the user defined diagnostic frames. The ADC operates in both single-ended mode and differential mode. All channels on the ADA4558 are single-ended, apart from ADC_DATA_PRES_S1, which is differential.

In single-ended mode, the 14-bit output varies between 0x0000 and 0x3FFF as the analog input moves from 0 V to V_{REG}.

In differential mode, zero differential voltage at the input corresponds to 0x2000. Table 4 shows the range of ADC_DATA_PRES_S1 ADC codes.

Table 4. ADC_DATA_PRES_S1 Code Range

ADC_DATA_PRES_S1 Code	VPOS	VNEG
0x0000	0 V	V_{REG}
0x2000	V_{CM}	V_{CM}
0x3FFF	V_{REG}	0 V

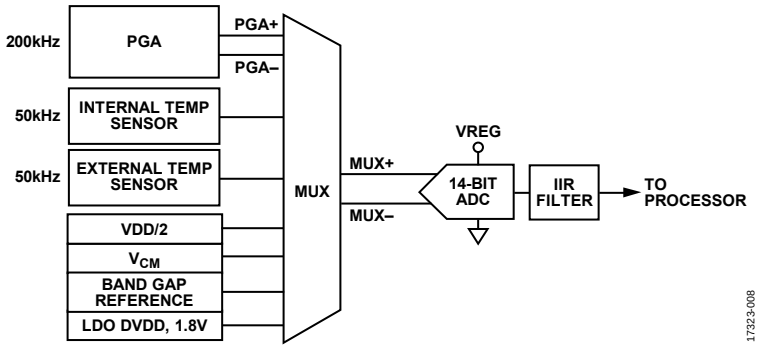

Figure 7. ADC Mux Operation

USER CONFIGURABLE FILTERING

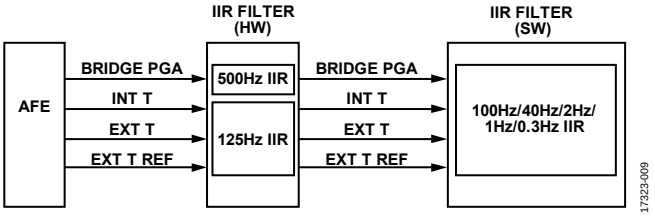

Figure 8. Hardware Filters and Configurable Software Filters for the AFE Signals

The ADA4558 has a user configurable IIR filter for each of the following ADC channels:

- PGA for the bridge sensor input
- Internal temperature (INT T)
- External temperature (EXT T)
- External temperature reference resistor (EXT T REF)

Each channel has a hardware (HW) IIR filter and a software (SW) IIR filter. In the event of a reset, the device defaults to the hardware filter where bridge sensor input has a 500 Hz, low-pass filter, while the temperature channel inputs have a 125 Hz low-pass filter. Select software IIR filters with lower cutoff frequencies using the temperature low-pass filter (LPF) and bridge low-pass filter (LPF) registers at EEPROM Address 12. The device switches to other selected filters after a settling time.

The ADA4558 implements a Chebyshev Type II fourth-order IIR filter for 100 Hz and 40 Hz pass band, and implements a first-order IIR filter for the 2 Hz, 1 Hz or 0.3 Hz pass band. The sample frequency is 4 KHz.

Write to the bridge LPF at EEPROM Address 12 (0x0C) to configure the IIR filter cutoff frequency to 500 Hz, 100 Hz, 40 Hz, 2 Hz, 1 Hz, or 0.3 Hz.

Write to the temperature LPF at EEPROM Address 12 (0x0C) to configure the IIR filter cutoff frequency for the temperature channels. Available cutoff frequencies are 125 Hz, 100 Hz, 40 Hz, 2 Hz, 1 Hz, or 0.3 Hz.

See Table 24 and Table 25 for IIR filter setting values.

OTP CODE STORAGE MEMORY

The ADA4558 has one time programmable (OTP) memory to hold the program code and the Analog Devices, Inc., data stored at final test. To improve reliability, the data in the OTP memory is encoded using Hamming code. The error correction code (ECC) performs the correction of single-bit errors and detection of 2-bit errors using seven ECC bits for every 32 bits of data. This ECC ensures reliability of the memory and that any potential field failures are detectable. The OTP memory is preprogrammed.

The data stored in the OTP include trims and configuration bits together with traceability data, such as wafer lot number and wafer X/Y coordinates, to provide full traceability of the silicon. The software revision identification is also encoded when the software is programmed into OTP memory.

All traceability data can be read back using the LIN commands described in the LIN Diagnostic Frames section.

EEPROM

The ADA4558 includes EEPROM memory to store factory trim coefficients, user calibration data, manufacturing data, and end customer serialization data. The ADA4558 EEPROM is based on a charge storage technology, and uses a differential bit cell for improved retention.

To improve reliability, the data in the EEPROM memory is encoded using Hamming code. The ECC performs the correction of single-bit errors and the detection of 2-bit errors using seven ECC bits for every 32 bits of data. This ECC ensures reliability of the memory and that any potential field failures are detectable.

After the EEPROM is programmed, read back the EEPROM status register at SYSREG Address Map 0xBB to verify that the correct data has been programmed to the EEPROM. See the SYSREG Map section for more information.

Table 5 lists the ADA4558 EEPROM map. See Table 6 to Table 44 for EEPROM bit descriptions.

Table 5. EEPROM Map

	Bit																															
Addr	**31**	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
31	Reserved																															
30	Reserved		FAULT_STATE[29:0]																													
29	Reserved															FAULT_ID1				FAULT_NUM1				FAULT_ID0					FAULT_NUM0		
28	Reserved															FAULT_ID3				FAULT_NUM3				FAULT_ID2					FAULT_NUM2		
27	Function ID																Variant ID									INT_TCOEFF1[11:4]						
26	Serial ID[31:0]																															
25	Reserved								LINPID8[7:0]								Supplier ID															
24	LINPID3[7:0]								LINPID2[7:0]								LINPID1[7:0]								Reserved							
23	Reserved																								INT_TCOEFF0[11:4]							
22	Reserved																															
21	Reserved																															
20	Reserved																															
19	Reserved																															
18	Reserved																								BS_SW				BS_HW			
17	Reserved																															
16	Reserved																															
15	FAULT_LOG_EN																															
14	FAULT_EN																															
13	NAD								INAD								TEMP_COEFF_K [10:3]								IDLE_SLP	Reserved		TEM-PSEL	TEMP_COEFF_K [2:0]		LIN CONFIG	
12	TEMP_LPF			Bridge LPF			Reserved				NORMAL_FACTOR									EXTTS_ISNK		EXTTS_TEMP_GAIN1		EXTTS_TEMP_GAIN2			EXTTS_REF_GAIN1		EXTTS_REF_GAIN2	
11	PGA_ADD_PULL-UP_150	PGA_REVE-RSE_IN		PGA_GAIN[4:0]				Rese-rved		INT_TCOEFF1 [3:0]				FAULT_TOLERANCE_NUM						TCORR_COEFF2[15:0]												
10	TCORR_COEFF1[15:0]																															
9	Reserved					BRG_VCM_TRIM						BRG_VOS_TRIM1							BRG_VOS_TRIM2								PGA_DAC_5					
8	CORR_COEFF14 [1:0]		CORR_COEFF13 [1:0]		CORR_COEFF12 [1:0]		CORR_COEFF11 [1:0]		CORR_COEFF10 [1:0]		CORR_COEFF9 [1:0]		CORR_COEF-F8[1:0]		Reserved		INT_TCOEFF0 [3:0]				ADC_RREF_INV_CONST												
7	CORR_COEFF14[17:2]																CORR_COEFF7 [1:0]		CORR_COEFF6 [1:0]		CORR_COEFF5 [1:0]		CORR_COEFF4 [1:0]		CORR_COEFF3 [1:0]			CORR_COEFF2 [1:0]		CORR_COEFF1 [1:0]		CORR_COEFF [1:0]	
6	CORR_COEFF13[17:2]																CORR_COEFF12[17:2]															
5	CORR_COEFF11[17:2]																CORR_COEFF10[17:2]															
4	CORR_COEFF9[17:2]																CORR_COEFF8[17:2]															
3	CORR_COEFF7[17:2]																CORR_COEFF6[17:2]															
2	CORR_COEFF5[17:2]																CORR_COEFF4[17:2]															
1	CORR_COEFF3[17:2]																CORR_COEFF2[17:2]															
0	CORR_COEFF1[17:2]																CORR_COEFF0[17:2]															

EEPROM BIT DESCRIPTIONS

Table 6. EEPROM Address 0

Bit No.	Bit Name	Description
[15:0]	CORR_COEFF0[17:2]	Bridge Sensor Correction Coefficient 0, Bits[17:2]
[31:16]	CORR_COEFF1[17:2]	Bridge Sensor Correction Coefficient 1, Bits[17:2]

Table 7. EEPROM Address 1

Bit No.	Bit Name	Description
[15:0]	CORR_COEFF2[17:2]	Bridge Sensor Correction Coefficient 2, Bits[17:2]
[31:16]	CORR_COEFF3[17:2]	Bridge Sensor Correction Coefficient 3, Bits[17:2]

Table 8. EEPROM Address 2

Bit No.	Bit Name	Description
[15:0]	CORR_COEFF4[17:2]	Bridge Sensor Correction Coefficient 4, Bits[17:2]
[31:16]	CORR_COEFF5[17:2]	Bridge Sensor Correction Coefficient 5, Bits[17:2]

Table 9. EEPROM Address 3

Bit No.	Bit Name	Description
[15:0]	CORR_COEFF6[17:2]	Bridge Sensor Correction Coefficient 6, Bits[17:2]
[31:16]	CORR_COEFF7[17:2]	Bridge Sensor Correction Coefficient 7, Bits[17:2]

Table 10. EEPROM Address 4

Bit No.	Bit Name	Description
[15:0]	CORR_COEFF8[17:2]	Bridge Sensor Correction Coefficient 8, Bits[17:2]
[31:16]	CORR_COEFF9[17:2]	Bridge Sensor Correction Coefficient 9, Bits[17:2]

Table 11. EEPROM Address 5

Bit No.	Bit Name	Description
[15:0]	CORR_COEFF10[17:2]	Bridge Sensor Correction Coefficient 10, Bits[17:2]
[31:16]	CORR_COEFF11[17:2]	Bridge Sensor Correction Coefficient 11, Bits[17:2]

Table 12. EEPROM Address 6

Bit No.	Bit Name	Description
[15:0]	CORR_COEFF12[17:2]	Bridge Sensor Correction Coefficient 12, Bits[17:2]
[31:16]	CORR_COEFF13[17:2]	Bridge Sensor Correction Coefficient 13, Bits[17:2]

Table 13. EEPROM Address 7

Bit No.	Bit Name	Description
[1:0]	CORR_COEFF0[1:0]	Bridge Sensor Correction Coefficient 0, Bits[1:0]
[3:2]	CORR_COEFF1[1:0]	Bridge Sensor Correction Coefficient 1, Bits[1:0]
[5:4]	CORR_COEFF2[1:0]	Bridge Sensor Correction Coefficient 2, Bits[1:0]
[7:6]	CORR_COEFF3[1:0]	Bridge Sensor Correction Coefficient 3, Bits[1:0]
[9:8]	CORR_COEFF4[1:0]	Bridge Sensor Correction Coefficient 4, Bits[1:0]
[11:10]	CORR_COEFF5[1:0]	Bridge Sensor Correction Coefficient 5, Bits[1:0]
[13:12]	CORR_COEFF6[1:0]	Bridge Sensor Correction Coefficient 6, Bits[1:0]
[15:14]	CORR_COEFF7[1:0]	Bridge Sensor Correction Coefficient 7, Bits[1:0]
[31:16]	CORR_COEFF14[17:2]	Bridge Sensor Correction Coefficient 14, Bits[17:2]

Table 14. EEPROM Address 8

Bit No.	Bit Name	Description
[11:0]	ADC_RREF_INV_CONST	Refer to the external temperature sensor calibration routine in the Calibration Procedure section
[15:12]	INT_TCOEFF0[3:0]	Internal Temperature Sensor Coefficient 0, Bits[3:0]
[17:16]	Reserved	Reserved
[19:18]	CORR_COEFF8[1:0]	Bridge Sensor Correction Coefficient 8, Bits[1:0]
[21:20]	CORR_COEFF9[1:0]	Bridge Sensor Correction Coefficient 9, Bits[1:0]
[23:22]	CORR_COEFF10[1:0]	Bridge Sensor Correction Coefficient 10, Bits[1:0]
[25:24]	CORR_COEFF11[1:0]	Bridge Sensor Correction Coefficient 11, Bits[1:0]
[27:26]	CORR_COEFF12[1:0]	Bridge Sensor Correction Coefficient 12, Bits[1:0]
[29:28]	CORR_COEFF13[1:0]	Bridge Sensor Correction Coefficient 13, Bits[1:0]
[31:30]	CORR_COEFF14[1:0]	Bridge Sensor Correction Coefficient 14, Bits[1:0]

Table 15. EEPROM Address 9

Bit No.	Bit Name	Description
[6:0]	PGA_DAC_5	Third stage bridge offset correction DAC
[13:7]	BRG_VOS_TRIM2	Final PGA Offset 2
[20:14]	BRG_VOS_TRIM1	Final PGA Offset 1
[27:21]	BRG_VCM_TRIM	Final common-mode trim value (see the Calibration Procedure section)
[31:28]	Reserved	Reserved

Table 16. EEPROM Address 10

Bit No.	Bit Name	Description
[15:0]	TCORR_COEFF0[15:0]	External Temperature Sensor Correction Coefficient 0
[31:16]	TCORR_COEFF1[15:0]	External Temperature Sensor Correction Coefficient 1

Table 17. EEPROM Address 11

Bit No.	Bit Name	Description
[15:0]	TCORR_COEFF2[15:0]	External Temperature Sensor Correction Coefficient 2
[19:16]	FAULT_TOLERANCE_NUM	Sets the number of consecutive faults, N, that must occur before the fault is flagged to the system, or logged to EEPROM
[23:20]	INT_TCOEFF1[3:0]	Internal Temperature Sensor Coefficient 1, Bits[13:0]
24	Reserved	Reserved
[29:25]	PGA_GAIN[4:0]	Selects the PGA gain (see Table 1 for PGA gain settings)
30	PGA_REVERSE_IN	Enable to reverses the PGA input 0 = PGA input normal 1 = PGA input reversed
31	PGA_ADD_PULUP_150	Additional PGA pull-down current to detect open bridge 0 = 300 nA 1 = 450 nA

Table 18. EEPROM Address 12

Bit No.	Bit Name	Description
[1:0]	EXTTS_REF_GAIN2	External temperature sensor reference resistor, second stage gain (see Table 19)
[4:2]	EXTTS_REF_GAIN1	External temperature sensor reference resistor, first stage gain (see Table 20)
[6:5]	EXTTS_TEMP_GAIN2	External temperature sensor RTD, second stage gain (see Table 21)
[9:7]	EXTTS_TEMP_GAIN1	External temperature sensor RTD, first stage gain (see Table 22)
[12:10]	EXTTS_ISNK	External temperature sensor programmable I_{SINK} (see Table 23)
[22:13]	NORMAL_FACTOR	See the Calibration Routine for External Temperature Sensor section
[25:23]	Reserved	Reserved
[28:26]	Bridge LPF	Bridge sensor low-pass filter configuration (see Table 24)
[31:29]	TEMP_LPF	Temperature low-pass filter configuration (see Table 25)

Table 19. External Temperature Sensor Reference Resistor Second Stage Gain Settings

EXTTS_REF_GAIN2	Second Stage Gain
00	1
01	1.5
10	2.0
11	3.0

Table 20. External Temperature Sensor Reference Resistor First Stage Gain Settings

EXTTS_REF_GAIN1	First Stage Gain
000	1
001	1.5
010	2.0
011	3.0
100	4.0
101	6.0
110	8.0
111	12.0

Table 21. External Temperature Sensor RTD Second Stage Gain Settings

EXTTS_TEMP_GAIN2	Second Stage Gain
00	1
01	1.5
10	2.0
11	3.0

Table 22. External Temperature Sensor RTD First Stage Gain Settings

EXTTS_TEMP_GAIN1	First Stage Gain
000	1
001	1.5
010	2.0
011	3.0
100	4.0
101	6.0
110	8.0
111	12.0

Table 23. External Temperature Sensor Programmable Sink Current Settings

EXTTS_ISNK	Sink Current (mA)
0xx	0
100	0.1
101	0.2
110	0.5
111	1.0

Table 24. Bridge Sensor Low-Pass Filter Settings

Bridge LPF	Filter Bandwidth (Hz)
000	500
001	100
010	40
011	2
100	1
101	0.3
110	500
111	500

Table 25. Temperature Low-Pass Filter Settings

TEMP_LPF	Filter Bandwidth (Hz)
000	125
001	100
010	40
011	2
100	1
101	0.3
110	125
111	125

Table 26. EEPROM Address 13

Bit No.	Bit Name	Description
[1:0]	LINCONFIG	Select LIN configuration. 00 = LIN 2.1 (default). 01 = LIN 2.0. 10 = LIN 1.3. 11 = reserved.
[4:2]	TEMP_COEFF_K[2:0]	External temperature compensation constant, K.
5	TEMP_SEL	Select internal or external temperature sensor for calibration of the bridge sensor channel. 0 = internal temperature sensor for calibration of the bridge sensor channel. 1 = external temperature sensor for calibration of the bridge sensor channel.
6	Reserved	Not Available
7	IDLE_SLP	Enable or disable auto sleep mode. 0 = enable auto sleep mode. 1 = disable auto sleep mode.
[15:8]	TEMP_COEFF_K[10:3]	External temperature compensation constant, K.
[23:16]	INAD	Initial node address (INAD) for slave node.
[31:24]	NAD	Node address (NAD) for slave node.

Table 27. EEPROM Address 14

Bit No.	Name	Description
[31:0]	FAULT EN	Enable faults. Selects which faults are enabled based on the fault ID code. Bit n = 0: fault detection is disabled, where n is the fault ID (see Table 82). Bit n = 1: fault detection enabled, where n is the fault ID (see Table 82).

Table 28. EEPROM Address 15

Bit No.	Bit Name	Description
[31:0]	FAULT_LOG_EN	Enable fault logging. Select which faults are logged to the EEPROM based on the fault identification code. See the Calibration Procedure section. Bit n = 0: fault logging disabled, where n is the fault identification. Bit n = 1: fault logging enabled, where n is the fault identification.

Table 29. EEPROM Address 16

Bit No.	Name	Description
[31:0]	Reserved	Reserved

Table 30. EEPROM Address 17

Bit No.	Bit Name	Description
[31:0]	Reserved	Reserved

Table 31. EEPROM Address 18

Bit No.	Bit Name	Description
[3:0]	BS_HW	User defined version number for bridge sensor hardware, value reported in LIN signal frame.
[7:4]	BS_SW	User defined version number for bridge sensor software, value reported in LIN signal frame.
[31:8]	Reserved	Reserved

Table 32. EEPROM Address 19

Bit No.	Bit Name	Description
[31:0]	Reserved	Reserved

Table 33. EEPROM Address 20

Bit No.	Bit Name	Description
[31:0]	Reserved	Reserved

Table 34. EEPROM Address 21

Bit No.	Bit Name	Description
[31:0]	Reserved	Reserved
[31:8]	Reserved	Reserved

Table 35. EEPROM Address 22

Bit No.	Bit Name	Description
[31:0]	Reserved	Reserved

Table 36. EEPROM Address 23

Bit No.	Bit Name	Description
[7:0]	INT_TCOEFF0[11:4]	Internal Temperature Sensor Coefficient 0
[31:8]	Reserved	Reserved

Table 37. EEPROM Address 24

Bit No.	Bit Name	Description
[7:0]	Reserved]	Reserved.
[15:8]	LINPID1[7:0]	LIN Protected Identifier 1. If unused, set these bits to 0x40.
[23:16]	LINPID2[7:0]	LIN Protected Identifier 2. If unused, set these bits to 0x40.
[31:24]	LINPID3[7:0]	LIN Protected Identifier 3. If unused, set these bits to 0x40.

Table 38. EEPROM Address 25

Bit No.	Bit Name	Description
[15:0]	Supplier ID	LIN supplier ID
[16:23]	LINPID8[7:0]	Protected Identifier 8
[24:31]	Reserved	Reserved

Table 39. EEPROM Address 26

Bit No.	Bit Name	Description
[31:0]	Serial ID	Customer specific identification data

Table 40. EEPROM Address 27

Bit No.	Bit Name	Description
[7:0]	INT_TCOEFF1[11:4]	Internal Temperature Sensor Coefficient 0
[15:8]	Variant ID	LIN variant ID defined in LIN Version 2.0 to the current LIN version
[31:16]	Function ID	LIN function ID defined in LIN Version 2.0 to the current LIN version

Table 41. EEPROM Address 28

Bit No.	Bit Name	Description
[3:0]	FAULT_NUM2	Number of occurrences of the third fault (among the enabled faults) detected by the device
[8:4]	FAULT_ID2	ID of the third fault (among the enabled faults) detected by the device
[12:9]	FAULT_NUM3	Number of occurrences of the fourth fault (among the enabled faults) detected by the device
[17:13]	FAULT_ID3	ID of the fourth fault (among the enabled faults) detected by the device
[31:18]	Reserved	Reserved

Table 42. EEPROM Address 29

Bit No.	Bit Name	Description
[3:0]	FAULT_NUM0	Number of occurrences of the first fault (among the enabled faults) detected by the device
[8:4]	FAULT_ID0	ID of the first fault (among the enabled faults) detected by the device
[12:9]	FAULT_NUM1	Number of occurrences of the second fault (among the enabled faults) detected by the device
[17:13]	FAULT_ID1	ID of the second fault (among the enabled faults) detected by the device
[31:18]	Reserved	Reserved

Table 43. EEPROM Address 30

Bit No.	Bit Name	Description
[29:0]	FAULT_STATE[29:0]	Indicates the state of the enabled faults
[31:30]	Reserved	Reserved

Table 44. EEPROM Address 31

Bit No.	Bit Name	Description
[31:0]	Reserved	Reserved

EEPROM CACHING

EEPROM values are cached either during startup or periodically. If the value is not periodically cached, the change in value using the write EEPROM LIN user defined diagnostic frame does not affect the system behavior. These values update into the system only during power-up (power-on reset).

Table 45. EEPROM Map with Behavior During EEPROM Caching

EEPROM Variable Name	Caching Behavior
TEMP_SEL	Cached during normal operation
BRG_VCM_TRIM	Cached during normal operation
BRG_VOS_TRIM1	Cached during normal operation
BRG_VOS_TRIM2	Cached during normal operation
PGA_DAC_5	Cached during normal operation
TCORR_COEFF0 to TCORR_COEFF2	Cached during normal operation
CORR_COEFF0 to CORR_COEFF14	Cached during normal operation
PGA_GAIN	Cached during normal operation
LINCONFIG	Cached at power-up only
NAD	Cached at power-up only
INAD	Cached at power-up only
LINPID1 to LINPID3, and LINPID8 (see Table 37 and Table 38)	Cached at power-up only
Supplier ID	Cached at power-up only
Serial ID	Cached at power-up only
Function ID	Cached at power-up only
Variant ID	Cached at power-up only
IDLE_SLP	Cached at power-up only
FAULT_STATE	Cached at power-up only
FAULT_ID0 to FAULT_ID3	Cached at power-up only
FAULT_NUM0 to FAULT_NUM3	Cached at power-up only
TEMP_COEFF_K	Cached during normal operation

LIN INTERFACE

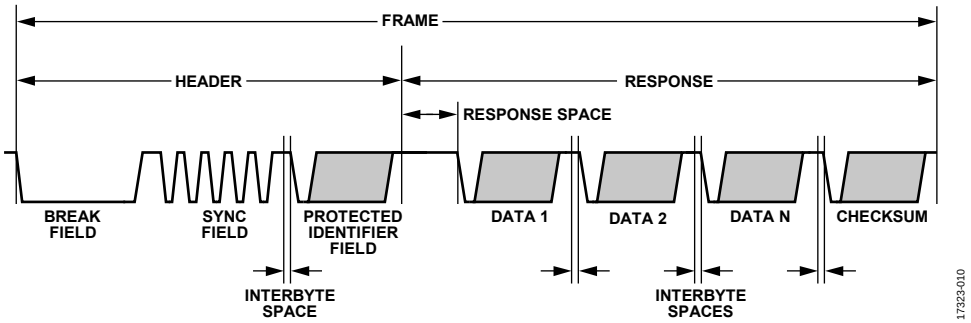

Figure 9. LIN Frame Structure

Figure 10. Byte Field Structure

The ADA4558 incorporates a 12 V LIN physical transceiver and LIN protocol stack. The ADA4558 supports LIN 1.3, LIN 2.0, and LIN 2.1 compatibility modes. Select the correct LIN compatibility mode using the LINCONFIG bits at EEPROM Address 13. See Table 26 for setting values.

The device operates as an LIN slave device where it replies to a header frame from the LIN master. The ADA4558 response frame supports single, unconditional frames that allow readings of the measurements, diagnostics, and programming of the EEPROM.

Figure 9 shows the LIN frame structure consisting of a header (from the LIN master) and response (from the ADA4558).

The header frame generated by the master node is constructed of the following:

- Break Field: wakes the ADA4558 from sleep mode and indicate the start of a new frame.
- Sync Field: a byte field with the data value of 0x55. The sync field is used to define the baud rate where the ADA4558 monitors the bit transition timing.
- Interbyte Space: the time between the end of the stop bit of the preceding field and the start bit of the following byte.
- Protected Identifier (PID) Field: requests from the LIN master. The PID carries the frame ID and the parity bits. The PID is split into values with an ID of 0 to 59 (0x0B) that are signal carrying frames, and values with an ID of 60 (0x3C) or 61 (0x3D) that carry diagnostic and configuration data.

The response frame from the ADA4558 is defined by the header PID field value. The data and checksum fields are transmitted as serial bytes, as shown in Figure 10, where each field transmission contains a dominant start bit, 8 data bits, and a recessive stop bit.

The response frame shown in Figure 9 is constructed of the following:

- Response Space: the bus idle time between the header field and the response field.
- Byte Fields: the corresponding data related to the PID field. The LSB is the first bit sent and the MSB is the last bit sent (little endian), as shown in Figure 10.
- Checksum: calculate the checksum by adding each byte value, as well as any carry over to the 8-bit result. The result is then inverted. Supported checksums include classic checksum, where all data bytes are used for calculation, and enhanced checksum, where all data bytes and PID are used for calculation. The master node manages the use of either classic or enhanced checksum.

Table 46. Checksum Description

Checksum Model	LIN Compatibility Mode
Classic	LIN 1.3, and Frame Identifier 60 (0x3C) to Frame Identifier 61 (0x3D)
Enhanced	LIN 2.0, LIN 2.1, and LIN 2.2

LIN INITIALIZATIONS

The following EEPROM data is used to initialize the LIN:

- PID values (EEPROM Address 24 and 25)
- Idle sleep mode (EEPROM Address 13)
- Node address for diagnostic (NAD) and initial NAD (INAD) (EEPROM Address 13)
- LIN compatibility mode (EEPROM Address 13)
- Product identification, supplier ID, variant ID and function ID (EEPROM Address 25 and Address 27)

FRAME TYPES

The ADA4558 supports LIN signal frames and LIN diagnostic frames.

LIN signal frames always carry signals (data). An LIN master initiates a signal frame by sending a PID.

LIN diagnostic frames always carry diagnostic data, and can be used to configure the ADA4558 or read memories.

LIN Signal Frame

The ADA4558 supports the signal frames shown in Table 51. Table 47 shows the definitions of the signal frame fields. The signal frames transmit calibrated bridge sensor measurements, internal temperature, external temperature data, and fault handling data. Each signal frame has a unique PID field that must be configured in the EEPROM at Address 24 and Address 25. Unused PIDs should be assigned a value of 0x40.

Table 47. Signal Frame Descriptions

Signal Name	Description
BRIDGE_SENSOR_11	The 11 MSBs of the 12-bit calibrated bridge output
BRIDGE_SENSOR_12	12-bit calibrated bridge output
TEMPERATURE_9	The 9 MSBs of the 12-bit internal temperature sensor output
TEMPERATURE_12	12 bits of internal temperature sensor output
EXT_TEMPERATURE_12	External temperature sensor output
BS_SENSORERROR	Available only in LIN-U3, for more information, see Table 83
BS_GLOBALFAULT	Available in LIN-U1 to LIN-U2, for more information, see Table 83
BS_CHECKSUMERROR	Sets when the checksum of the frame is incorrect, caused by a corrupted frame or using the wrong checksum model in the frame
BS_RESPONSEERROR	Sets when an error in the frame response occurs
BS_BITERROR	Sets when the bit error is reported
TNCTOC	Fault flag bit for incomplete IIR filter settling during power-up (see the Fault Handling, Diagnostics, and Logging Feature section)
HW	Indicates the 4-bit value in the EEPROM Address 18 named BS_HW
SW	Indicates the 4-bit value in the EEPROM Address 18 named BS_SW
SLFCHK_INIT	Allows the subordinate node to perform a power-up check during normal operation when set to 1
Unused	All unused PIDs are set to 0x40

Table 48. READ_P12_T12_STATUS (LIN-U1) Frame

Byte Field	Bits	Description
PID	No bits	LINPID1
DATA0	[7:0]	BRIDGE_SENSOR_12[7:0]
DATA1	[3:0]	BRIDGE_SENSOR_12[11:8]
	[7:4]	Unused
DATA2	[7:0]	TEMPERATURE_12[7:0]
DATA3	[3:0]	TEMPERATURE_12[11:8]
	4	BS_GLOBALFAULT
	5	BS_CHECKSUMERROR
	6	BS_RESPONSEERROR
	7	TNCTOC

Table 49. READ_P12_ET12_T12_STATUS (LIN-U2) Frame

Byte Field	Bits	Description
PID	No bits	LINPID2
DATA0	[7:0]	BRIDGE_SENSOR_12[7:0]
DATA1	[3:0]	BRIDGE_SENSOR_12[11:8]
	[7:4]	EXT_TEMPERATURE_12[3:0]
DATA2	[7:0]	EXT_TEMPERATURE_12[11:4]
DATA3	[7:0]	TEMPERATURE_12[7:0]
DATA4	[3:0]	TEMPERATURE_12[11:8]
	4	BS_GLOBALFAULT
	5	BS_CHECKSUMERROR
	6	BS_RESPONSEERROR
	7	TNCTOC

Table 50. READ_P11_T9_STATUS_HW_SW (LIN-U3) Frame

Byte Field	Bits	Description
PID	No bits	LINPID3
DATA0	[7:0]	BRIDGE_SENSOR_11[7:0]
DATA1	[2:0]	BRIDGE_SENSOR_11[10:8]
	[7:3]	TEMPERATURE_9[4:0]
DATA2	[3:0]	TEMPERATURE_9[8:5]
	4	BS_SENSORERROR
	5	BS_RESPONSEERROR
	6	BS_CHECKSUMERROR
	7	BS_BITERROR
DATA3	[3:0]	HW[3:0]
	[7:4]	SW[3:0]

Table 51. SELFCHECK (LIN-U8) Frame

Byte Field	Bits	Description
PID	No bits	LINPID8
DATA0	0	SLFCHK_INIT
	[7:1]	Unused
DATA1 to DATA4	No bits	Unused

LIN Diagnostic Frames

The LIN diagnostic frame is used to configure the ADA4558 and to read status of the device. The LIN master issues the diagnostic command request. Table 52 shows this LIN frame format. Table 53 shows the ADA4558 response frame format. The frames are a fixed length of eight data bytes and are sent after the LIN header frame.

Diagnostic frames always use the classic checksum. The classic checksum is transited at the end of the diagnostic frame. The PID value in the LIN header frame indicates which type of diagnostic frame is in use. A PID value of 0x3C indicates a master request frame. A PID value of 0x7D indicates the ADA4558 response frame. Table 54 shows a description of each byte field in the diagnostic frame. Table 55 shows the LIN diagnostic frames supported by the ADA4558.

Table 52. Diagnostic Frame Master Request PDU (Packet Data Unit) Response Field

NAD	PCI	SID	DATA1	DATA2	DATA3	DATA4	DATA5

Table 53. Diagnostic Frame Slave Response PDU Response Field

NAD	PCI	RSID	DATA1	DATA2	DATA3	DATA4	DATA5

Table 54. Diagnostic Frame Response Field

PDU Byte	Description		
NAD	Node address of a logical slave node.		
PCI	Protocol control information (PCI), flow control information.		
	Bits[7:4]	**Bits[3:0]**	**Type**
	0	Length	Single frame (SF), single PDU, maximum five data bytes.
	1	Length/256	First frame (FF), indicates start of multiple PDU messages, following frames followed by consecutive frame (CF).
	2	Frame counter	CF, the frame counter wraps around and continues starting at zero when there are more than 15 CF PDUs.
SID	Service identifier (SID), specifies the request that is to be performed by the slave node.		
RSID	Response service ID (RSID), specifies the contents of the response.		
DATA1 to DATA5	Data bytes. The value is set to 255 (0xFF) when this byte is unused.		

Table 55. Diagnostic Frames Supported by ADA4558 with Different LIN Compatibility Modes

Diagnostic Services	Diagnostic Frames	LIN Compatibility Mode
Node Configuration Services	Assign NAD	2.0, 2.1
	Assign frame identifier range	2.1
	Assign frame identifier	2.0
	Read by identifier	2.0, 2.1
	Save configuration	2.1

Assign NAD

The assign NAD is a node configuration diagnostic frame that assigns a NEW_NAD value to the ADA4558. There is a response for the assign NAD request frame that is only sent when the NAD, the supplier ID, and the function ID match. The PID for the assign NAD positive response is 0x7D.

Future node configuration diagnostic frames and user defined diagnostic frames use this NEW_NAD value. The NEW_NAD value is not saved to the EEPROM and is lost after a POR. This frame is not supported when the function ID is 0x0001.

The PID for the assign NAD request frame is 0x3C. Note that the NAD is the initial NAD[7:0], and INAD is the NAD from the EEPROM.

There is a response for the assign NAD positive response frame that only occurs when the NAD assignment is complete (see Table 57). The PID for the assign NAD positive response is 0x7D. See Table 58 for the assign NAD negative response.

To protect the loss of slave due to network corruption, the ADA4558 always recognizes an assign NAD command with the broadcast of NAD 0x7F. This recognition occurs regardless of the actual NAD of the subordinate when the assign NAD is transmitted.

Assign Frame Identifier Range

The assign frame identifier range is a node configuration diagnostic frame that changes the PIDs of the signal frame that is supported. This frame is supported in LIN 2.1 mode only (see Table 59).

Assign Frame Identifier Range Response (LIN 2.1)

In LIN 2.1 mode when the request is with the proper PCI and proper index, the ADA4558 sends a positive response, as shown in Table 60. See Table 61 for the negative response.

Table 56. Assign NAD Request Frame

NAD	PCI	SID	DATA1	DATA2	DATA3	DATA4	DATA5
INAD[7:0]	0x06	0xB0	SUPPLIER_ID[7:0]	SUPPLIER_ID[15:8]	FUNCTIONID[7:0]	FUNCTIONID[15:8]	NEW_NAD[7:0]

Table 57. Assign NAD Positive Response Frame

NAD	PCI	RSID	DATA1	DATA2	DATA3	DATA4	DATA5
INAD[7:0]	0x01	0xF0	0xFF	0xFF	0xFF	0xFF	0xFF

Table 58. Assign NAD Negative Response

NAD	PCI	RSID	DATA1	DATA2	DATA3	DATA4	DATA5
INAD[7:0]	0x03	0x7F	0xB0	0x11	0xFF	0xFF	0xFF

Table 59. Assign Frame Identifier Range Request (LIN 2.1)

NAD	PCI	SID	DATA1	DATA2	DATA3	DATA4	DATA5
NAD[7:0]	0x06	0xB7	Start index	PID (index)	PID (index + 1)	PID (index + 2)	PID (index + 3)

Table 60. Assign Frame ID Positive Response (LIN 2.1)

NAD	PCI	RSID	DATA1	DATA2	DATA3	DATA4	DATA5
NAD[7:0]	0x01	0xF7	0xFF	0xFF	0xFF	0xFF	0xFF

Table 61. Assign Frame ID Negative Response (Only in LIN2.0 and FUNCTIONID = 0x0001)

NAD	PCI	RSID	DATA1	DATA2	DATA3	DATA4	DATA5
NAD[7:0]	0x03	0x7F	0xB7	0x11	0xFF	0xFF	0xFF

Table 62. Index Values for Each Signal Frame

Frame Index	Frame	PID
0	Reserved	Reserved
1	READ_P12_T12_STATUS	LINPID1
2	READ_P12_ET12_T12_STATUS	LINPID2
3	READ_P11_T9_STATUS_HW_SW	LINPID3
4	Self check	LINPID8

Assign Frame Identifier (LIN 2.0)

The assign frame identifier is a node configuration diagnostic frame that changes the PID of the signal frames that are supported. This frame is supported in LIN 2.0 mode only. There is no response for this frame, and PID = 0x3C.

The assign frame identifier command saves the new PIDs to the EEPROM after the correct assignment of the NAD, indicated by the positive response frame.

Assign Frame Identifier Response (LIN 2.0)

When the frame ID assignment is complete, there is a response for the assign frame identifier, and PID = 0x7D. See Table 64 and Table 65 for the assign frame ID positive and negative responses. See Table 66 for the message ID for the assign frame identifier request.

Read by Identifier

The read by identifier is a node configuration diagnostic frame that allows a read of the supplier identity and other properties from the slave node. Read by identifier is supported in LIN 2.0 and LIN 2.1 compatibility modes only. The identifiers listed in Table 67 are supported. See Table 68 and Table 69 for the identifier request and response, respectively.

Table 63. Assign Frame Identifier Request (LIN 2.0)

NAD	PCI	SID	DATA1	DATA2	DATA3	DATA4	DATA5
NAD[7:0]	0x06	0xB1	Supplier ID[7:0]	Supplier ID[15:8]	Message ID[7:0]	Message ID[15:8]	PID

Table 64. Assign Frame ID Positive Response

NAD	PCI	RSID	DATA1	DATA2	DATA3	DATA4	DATA5
NAD[7:0]	0x01	0xF1	0xFF	0xFF	0xFF	0xFF	0xFF

Table 65. Assign Frame ID Negative Response

NAD	PCI	RSID	DATA1	DATA2	DATA3	DATA4	DATA5
NAD[7:0]	0x03	0x7F	0xB1	0x12	0xFF	0xFF	0xFF

Table 66. Message ID for Assign Frame Identifier Request

Message ID	Frame	PID
0x0000	Reserved	Reserved
0x0001	READ_P12_T12_STATUS	LINPID1
0x0002	READ_P12_ET12_T12_STATUS	LINPID2
0x0003	READ_P11_T9_STATUS_HW_SW	LINPID3
0x0004	Self check	LINPID8

Table 67. Supported Read by Identifiers

Identifier	Description	Length of Response
0	LIN product identification	5 + RSID
1	Serial number	4 + RSID
33 (0x21)	Analog Devices defined	2 + RSID
2 to 32, 34 to 255	Identifier not supported	Not Applicable

Table 68. Read by Identifier Request

NAD	PCI	SID	DATA1	DATA2	DATA3	DATA4	DATA5
NAD[7:0]	0x06	0xB2	Identifier	Supplier ID[7:0]	Supplier ID[8:15]	Function ID[7:0]	Function ID[8:15]

Table 69. Read by Identifier Response

Frame	NAD	PCI	RSID	DATA1	DATA2	DATA3	DATA4	DATA5
Read by Identifier Response, ID = 0	NAD[7:0]	0x06	0xF2	Supplier ID[7:0]	Supplier ID[8:15]	Function ID[7:0]	Function ID[8:15]	Variant ID
Read by Identifier Response, ID = 1	NAD[7:0]	0x05	0xF2	Serial 0	Serial 1	Serial 2	Serial 3	0xFF
Read by Identifier Response, ID = 33 (0x21)	NAD[7:0]	0x03	0xF2	Stored NAD EEPROM	Stored status frame ID in EEPROM	0xFF	0xFF	0xFF
Read by Identifier Response, Negative	NAD[7:0]	0x03	0x7F	0xB2	0x12	0xFF	0xFF	0xFF

Data Dump

The data dump diagnostic frame is not supported by the ADA4558. The slave sends a negative response when a data dump frame is received. There is no response for this frame in LIN 1.3 compatibility mode.

Table 70. Data Dump Request Frame

NAD	PCI	SID	DATA1	DATA2	DATA3	DATA4	DATA5
NAD[7:0]	0x06	0xB4	User defined	User defined	User defined	User defined	User defined

Table 71. Data Dump Negative Response Frame

NAD	PCI	RSID	DATA1	DATA2	DATA3	DATA4	DATA5
NAD[7:0]	0x03	0x7F	0xB4	0x11	0xFF	0xFF	0xFF

Save Configuration

The save node configuration diagnostic frame updates the EEPROM location with the configured values. The command always receives the same response regardless of the EEPROM write outcome.

Table 72. Save Configuration Request Frame

NAD	PCI	SID	DATA1	DATA2	DATA3	DATA4	DATA5
NAD[7:0]	0x01	0xB6	0xFF	0xFF	0xFF	0xFF	0xFF

Table 73. Save Configuration Positive Response LIN 2.1

NAD	PCI	RSID	DATA1	DATA2	DATA3	DATA4	DATA5
NAD[7:0]	0x01	0xF6	0xFF	0xFF	0xFF	0xFF	0xFF

Table 74. Save Configuration Negative Response LIN 2.1 (LIN 2.0 Only and Function ID = 0x0001)

NAD	PCI	RSID	DATA1	DATA2	DATA3	DATA4	DATA5
NAD[7:0]	0x03	0x7F	0xB6	0x11	0xFF	0xFF	0xFF

Response for Unsupported SIDs

For diagnostic services and commands that are not supported (0x00 to 0x21, 0x23 to 0xAF, and 0xB8 to 0xFF), the ADA4558 provides responses as listed in Table 75 to Table 77.

Table 75. Received Configured NAD

Received NAD	Node Configuration SID	Remainder of All Possible SIDs
Configured NAD	ADA4558 responds as specified by the standard	ADA4558 does not respond

Table 76. Received Configured NAD = 0x80

Received NAD	SID B0 to SID B7	Remainder of the SID
Configured NAD = 0x80	ADA4558 does not respond	ADA4558 provides negative response

Table 77. Unknown Service ID Response Frame (LIN 2.0 Only and Function ID = 0x0001)[1]

NAD	PCI	RSID	DATA1	DATA2	DATA3	DATA4	DATA5
NAD[7:0]	0x03	0x7F	SID in the request	0x13	0xFF	0xFF	0xFF

[1] Negative response only if the function ID = 0x0001.

LIN UDD Frames

Use the following user defined diagnostic (UDD) frames for reading and writing to memory mapped registers, software variables, and the EEPROM. These frames read traceability data from OTP memory and use the LIN master request and slave response PIDs, but respond to requests where NAD is NAD + 0x80.

When the register write is performed to system registers and the EEPROM, place the ADA4558 in configuration mode to disable the nonvolatile memory (NVM) caching. Otherwise, the NVM caching overwrites the registers with the EEPROM values. To place the ADA4558 in configuration mode, send the following LIN UDD frames:

- Test mode password frame, where the five data fields are set to 0x01, 0x23, 0x45, 0x67, or 0x89.
- Test mode select frame, where TESTMODE_SEL[7:0] is 0x01.

This sequence of instructions is intended to prevent accidental writes. To return the ADA4558 to normal operating mode, send the test mode password frame again followed by the test mode select frame where TESTMODE_SEL[7:0] = 0x00

When reading or writing registers, ensure that the correct frame is used for the register width. For example, a negative response is returned when a 16-bit register is written using a 32-bit write command.

Table 78. User Defined Diagnostic Frames, NAD = NAD + 0x80

Read/Write Request/Response[1]	PID	NAD	PCI	SID/RSID	DATA1	DATA2	DATA3	DATA4	DATA5
Write SYSREG 4-Byte Request	0x3C	NAD + 0x80	0x06	0x01	ADDR[7:0]	DATA[7:0]	DATA[15:8]	DATA[23:16]	DATA[31:24]
Write SYSREG 4-Byte, Positive Response	0x7D	NAD + 0x80	0x01	0x41	0xFF	0xFF	0xFF	0xFF	0xFF
Write SYSREG 4-Byte, Negative Response	0x7D	NAD + 0x80	0x01	0x01	0xFF	0xFF	0xFF	0xFF	0xFF
Write SYSREG 2-Byte, Request	0x3C	NAD + 0x80	0x04	0x09	ADDR[7:0]	DATA[7:0]	DATA[15:8]	0xFF	0xFF
Write SYSREG 2-Byte, Positive Response	0x7D	NAD + 0x80	0x01	0x49	0xFF	0xFF	0xFF	0xFF	0xFF
Write SYSREG 2-Byte, Negative Response	0x7D	NAD + 0x80	0x01	0x09	0xFF	0xFF	0xFF	0xFF	0xFF
Read SYSREG 16-Bit, Request	0x3C	NAD + 0x80	0x02	0x02	ADDR[7:0]	0xFF	0xFF	0xFF	0xFF
Read SYSREG 16-Bit, Positive Response	0x7D	NAD + 0x80	0x03	0x42	DATA[7:0]	DATA[15:8]	0xFF	0xFF	0xFF
Read SYSREG 16-Bit, Negative Response	0x7D	NAD + 0x80	0x01	0x02	0xFF	0xFF	0xFF	0xFF	0xFF
Read SYSREG 32-Bit Request	0x3C	NAD + 0x80	0x02	0x03	ADDR[7:0]	0xFF	0xFF	0xFF	0xFF
Read SYSREG 32-Bit, Positive Response	0x7D	NAD + 0x80	0x05	0x43	DATA[7:0]	DATA[15:8]	DATA[23:16]	DATA[31:24]	0xFF
Read SYSREG 32-Bit, Negative Response	0x7D	NAD + 0x80	0x01	0x03	0xFF	0xFF	0xFF	0xFF	0xFF
Write EEPROM Request	0x3C	NAD + 0x80	0x06	0x04	ADDR[7:0]	DATA[7:0]	DATA[15:8]	DATA[23:16]	DATA[31:24]
Write EEPROM, Positive Response	0x7D	NAD + 0x80	0x03	0x44	ADDR[7:0]	EEPROM_STATUS[7:0]	0xFF	0xFF	0xFF
Write EEPROM, Negative Response	0x7D	NAD + 0x80	0x03	0x04	ADDR[7:0]	EEPROM_STATUS[7:0]	0xFF	0xFF	0xFF
Read EEPROM, Request	0x3C	NAD + 0x80	0x02	0x05	ADDR[7:0]	0xFF	0xFF	0xFF	0xFF
Read EEPROM, Positive Response	0x7D	NAD + 0x80	0x06	0x45	ADDR[7:0]	DATA[7:0]	DATA[15:8]	DATA[23:16]	DATA[31:24]
Read EEPROM, Negative Response	0x7D	NAD + 0x80	0x01	0x05	ADDR[7:0]	EEPROM_STATUS[7:0]	0xFF	0xFF	0xFF
Test Mode Select	0x3C	NAD + 0x80	0x02	0x07	TESTMODE_SEL[7:0]	0xFF	0xFF	0xFF	0xFF
Test Mode Select, Positive Response	0x7D	NAD + 0x80	0x02	0x47	TESTMODE_SEL[7:0]	0xFF	0xFF	0xFF	0xFF
Test Mode Select, Negative Response	0x7D	NAD + 0x80	0x01	0x07	0xFF	0xFF	0xFF	0xFF	0xFF
Test Mode Password	0x3C	NAD + 0x80	0x06	0x08	PASSWD[7:0]	PASSWD[15:8]	PASSWD[23:16]	PASSWD[31:24]	PASSWD[39:32]
Test Mode Password, Positive Response	0x7D	NAD + 0x80	0x01	0x48	0xFF	0xFF	0xFF	0xFF	0xFF

Read/Write Request/Response[1]	PID	NAD	PCI	SID/RSID	DATA1	DATA2	DATA3	DATA4	DATA5
Test Mode Password, Negative Response	0x7D	NAD + 0x80	0x01	0x08	0xFF	0xFF	0xFF	0xFF	0xFF

[1] SYSREG is the system register.

LIN SLEEP/WAKE-UP MODE

When the IDLE_SLP bit is set to 0 in EEPROM Address 13, the ADA4558 monitors the LIN bus and goes to sleep after 10 sec or 4 sec, depending on the current LIN compatibility mode selected. The ADA4558 also enters sleep mode on reception of the sleep command. The ADA4558 disables sleep mode when the IDLE_SLP bit is set to 1.

SYSREG MAP

A set of SRAM registers are available to be accessed by the LIN UDD frames (register read/write command) as described in Table 79.

Table 79. SYSREG Command Usage Guide

Request Command	Condition
SYSREG R/W 2-Byte Command	Width ≤16 bits
SYSREG R/W 4-Byte Command	Width ≥16 bits

Table 80. SYSREG Map

SYSREG Address	Register Name	Bits	R/W	Description
0x0	Bridge Correction Coefficient 0	[17:0]	R/W	Correction coefficients for the bridge sensor input.
0x4	Bridge Correction Coefficient 1	[17:0]	R/W	
0x8	Bridge Correction Coefficient 2	[17:0]	R/W	
0xC	Bridge Correction Coefficient 3	[17:0]	R/W	
0x10	Bridge Correction Coefficient 4	[17:0]	R/W	
0x14	Bridge Correction Coefficient 5	[17:0]	R/W	
0x18	Bridge Correction Coefficient 6	[17:0]	R/W	
0x1C	Bridge Correction Coefficient 7	[17:0]	R/W	
0x20	Bridge Correction Coefficient 8	[17:0]	R/W	
0x24	Bridge Correction Coefficient 9	[17:0]	R/W	
0x28	Bridge Correction Coefficient 10	[17:0]	R/W	
0x2C	Bridge Correction Coefficient 11	[17:0]	R/W	
0x30	Bridge Correction Coefficient 12	[17:0]	R/W	
0x34	Bridge Correction Coefficient 13	[17:0]	R/W	
0x38	Bridge Correction Coefficient 14	[17:0]	R/W	
0x3C	TEMP Correction Coefficient 0	[15:0]	R/W	External temperature sensor correction coefficients.
0x3E	TEMP Correction Coefficient 1	[15:0]	R/W	
0x40	TEMP Correction Coefficient 2	[15:0]	R/W	
0x42	Bridge Out CALD	[11:0]	R	Calibrated bridge sensor output.
0x44	Temperature Out CALD	[11:0]	R	Calibrated temperature output.
0x60	INT_TCOEFF0	[11:0]	RW	Internal Temperature Correction Coefficient 0.
0x62	INT_TCOEFF1	[11:0]	RW	Internal Temperature Correction Coefficient 1.
0x64	PGA_VICM	[13:0]	R	Read back the PGA_VICM ADC.
0x66	TEMP_COEFF_K	[10:0]	RW	External temperature constant, K.
0x90	ADC_DATA_PRES	[15:0]	R	ADC raw data bridge sensor channel.
0x91	ADC_DATA_TEMP	[15:0]	R	ADC raw data internal temperature sensor.
0x92	ADC_DATA_EXTTS	[15:0]	R	ADC raw data external temperature sensor.
0x93	ADC_DATA_EXTTS_REF	[15:0]	R	ADC raw data external temperature reference.
0x94	ADC_DATA_PRES_S1	[15:0]	R	ADC raw data at bridge sensor PGA Stage 1 output.
0x95	ADC_DATA_SLFCHK1	[15:0]	R	ADC raw data Self Check Channel 1.
0x96	ADC_DATA_SLFCHK2	[15:0]	R	ADC raw data Self Check Channel 2.

SYSREG Address	Register Name	Bits	R/W	Description
0xB0	PGA_CTRL	1	RW	PGA float detect current.
				0, floating detect current = 300 nA pull-down current.
				1, floating detect current = 450 nA pull-down current.
		0	RW	PGA reverse input.
				0, PGA inputs directly connected.
				1, PGA inputs reversed.
0xB1	PGA_GAIN	[4:0]	RW	PGA gain.
0xB2	PGA_DAC_1_2	[14:8]	RW	PGADAC2, first stage bridge offset correction; note that Bit 7 is not used and must be set to 0 at all times.
		[6:0]	RW	PGADAC1, first stage bridge offset correction DAC.
0xB3	PGA_DAC_3_4	[14:8]	RW	PGADAC4, second stage bridge offset correction DAC+.
		[6:0]	RW	PGADAC3, second stage bridge offset correction DAC−.
0xB4	PGA_DAC5	[6:0]	RW	Third stage bridge offset correction DAC.
0xB5	EXTTS_GAIN			External temperature sensor gain.
		[9:7]	RW	EXTTSTEMPGAIN, first stage gain.
		[6:5]	RW	EXTTSTEMPGAIN, second stage gain.
		[4:2]	RW	EXTTSREF GAIN, first stage gain.
		[1:0]	RW	EXTTSREFGAIN, second stage gain.
0xB6	EXTTS_ISNK	[2:0]	RW	External temperature sensor I_{SINK} (see Table 23).
0xBB	EEPROM_STATUS	[7:0]	R	Reads back the EEPROM status.
		0	R	Programming status.
		1	R	ECC use. When set, this register indicates ECC is being used.
		2	R	ECC fail. When set, this register indicates ECC failed.
		3	R	Read timeout. When set, this register indicates an EEPROM read timeout occurred.
		4	R	Programming timeout. When set, this register indicates an EEPROM programming timeout occurred.
		5	R	IPEN timeout. When set, this register indicates an IPEN timeout.
		6	R	Program done. When set, this register indicates the EEPROM programming is done.
		7	R	Busy. When set, this register indicates that the EEPROM is busy.
0xBC	Fault Status	[31:0]	R	Reads back the status of enabled faults (see the Fault Handling, Diagnostics, and Logging Feature section for more information).
0xBD	FAULT_EN	[31:0]	RW	Enable the fault (see the Fault Handling, Diagnostics, and Logging Feature section for more information).
0xBE	FAULT LOG EN	[31:0]	RW	Enable fault logging (see the Fault Handling, Diagnostics, and Logging Feature section for more information).
0xD0	Silicon Revision ID	[15:0]	R	Reads back the silicon revision ID.

FAULT HANDLING, DIAGNOSTICS, AND LOGGING FEATURE

The ADA4558 features extensive self check diagnostic and fault detection, as well as system integrity checks to prevent erroneous output information and system damage. Faults are flagged to the LIN master with fault flags in signal frames and user defined diagnostic frames (readback fault status register at SYSREG Address 0xBC). See Table 81 for the behavior of the ADA4558 during fault conditions monitored by the ADA4558 and the ADA4558 fault ID numbers.

Each individual fault detection is enabled by setting the corresponding bit in FAULT_ENx at EEPROM Address 14. When a fault triggers with FAULT_ENx set high, the BS_GLOBALFAULT or BS_SENSORERROR flag is set, indicating a fault detection. This detection also sets the corresponding bit in the LIN fault status register at SYSREG Address 0xBC to high.

The ADA4558 running time faults are checked every 50 ms. The fault detection time is set by the FAULT_TOLERANCE_NUM bit (EEPROM Address 11), where FAULT_TOLERANCE_NUM is N, and fault detection time is N × 50 to (N + 1) × 50 ms. To filter out transient interference for some faults, the fault handler flags them as real faults only after N consecutive occurrences. Depending on the application requirement, select the value of N to optimize fault detection time and filtering. A smaller N gives a faster fault detection time and lower filtering than a larger N.

To enable each individual EEPROM fault logging, set the corresponding bit in FAULT_LOG_EN at EEPROM Address 15. With fault logging enabled, the corresponding bit in FAULT_STATE at EEPROM Address 30 is set high upon fault detection. The ADA4558 logs the fault ID and number of fault occurrences for the first four faults in FAULT_IDx and FAULT_NUMx at EEPROM Address 28 and Address 29. Fault log counters log the number of fault occurrences. Each fault log counter logs up to 15 fault occurrences, for the 16th fault occurrence and beyond, the EEPROM fault write stops to avoid write disturbance.

For subsequent faults after the first four faults, the number of fault occurrences is not recorded, only logged in the FAULT_STATE as soon as the fault occurs. When the subsequent faults are repeated, the EEPROM fault write stops to avoid write disturbances.

Writing fault information to the EEPROM requires that the supply operate above 5.1 V with a stable DVDD supply. Evaluate the fault logging feature as part of the qualification of a module to ensure robust operation. Enable the fault logging feature during module qualification but not during mass production.

Table 82 summarizes all the fault diagnostics in the ADA4558 with the corresponding FAULT_ID, FAULT_EN, and FAULT_LOG_EN bits in the EEPROM, and the LIN frame response for a given fault.

Table 81. ADA4558 Fault Types and Behavior

	Fault Information			
Type	**Occurrence**	**Description**	**Detection**	**Fault Handling**
Hardware (HW)				
Type 1A (HW)	Startup and running	Processor or memory fault, LIN stack processing cannot be trusted	Hardware detected faults, hardware trigger device reboot	Processor halts, LIN communication idles, log to the EEPROM when enabled, reboot device after timeout[1]
Type 1B (HW/SW)	Startup and/or running[2]	Processor or memory fault, LIN stack processing cannot be trusted	Software detected faults, hardware trigger device reboot	Processor halts, LIN communication idles, log to the EEPROM when enabled, reboot device after timeout[1]
Software (SW)				
Type 2A	Startup or LIN demand[3]	System and analog checks	Software detected faults	Set fault flag in LIN frame, set FAULT_STATUS bit in LIN FAULT_STATUS register (0xBC), log to the EEPROM when enabled[4]
Type 2B	Running	System and analog checks	Software detected faults	Set fault flag in LIN frame, set FAULT_STATUS bit in LIN FAULT_STATUS register (0xBC), log to the EEPROM when enabled[4]

[1] Timeout depends on recovery mode. Regular recovery mode is 130 ms, fast recovery mode is 0 ms.
[2] The watchdog timer fault is checked at both start-up and normal running operation.
[3] LIN-U8 frame with SLFCHK_INT bit is set, and slave node performs power-up fault check in normal operation.
[4] LIN-U1 to LIN-U2 fault flag, BS_GLOBALFAULT and LIN-U3 fault flag, BS_SENSORERROR defined (see Table 47).

Table 82. ADA4558 Fault Diagnostics Supported[1]

Fault ID	Fault Name[2,3]	Fault Type	Log Enable (NVM) FAULT_LOG_EN, Bits[29:0]	FAULT_EN, Bits[29:0] (NVM)	Filtering[4,5,6]	Processor Reset	Fault Reset, Fault Gone
30	VPOS underrange and floating	Type 2B	Bit 29	Bit 29	N check	No	Yes
29	VPOS overrange	Type 2B	Bit 28	Bit 28	N check	No	Yes
28	VNEG underrange and floating	Type 2B	Bit 27	Bit 27	N check	No	Yes
27	VNEG overrange	Type 2B	Bit 26	Bit 26	N check	No	Yes
26	ADC timeout	Type 2B	Bit 25	Bit 25	One time	No	Yes
25	ALU[7] crosscheck	Type 2B	Bit 24	Bit 24	N check	No	Yes
24	VBAT out of range	Type 2B	Bit 23	Bit 23	N check	No	Yes
23	DVDD out of range	Type 2B	Bit 22	Bit 22	N check	No	Yes
22	AHI5 out of range	Type 2B	Bit 21	Bit 21	N check	No	Yes
21	LHI5 out of range	Type 2B	Bit 20	Bit 20	N check	No	Yes
20	IBIAS[8] out of range	Type 2B	Bit 19	Bit 19	N check	No	Yes
19	Bridge VCM [9]out of range	Type 2B	Bit 18	Bit 18	N check	No	Yes
18	Internal temperature sensor, voltage out of range	Type 2B	Bit 17	Bit 17	N check	No	Yes
17	Internal temperature sensor circuit, self check	Type 2A	Bit 16	Bit 16	N check	No	Yes
16	ADC self check	Type 2B	Bit 15	Bit 15	N check	No	Yes
15	PGA gain	Type 2A	Bit 14	Bit 14	Average 16	No	Yes
14	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
13	PGA offset DAC	Type 2A	Bit 12	Bit 12	Average 16	No	Yes
12	bridge sensor out of range	Type 2B	Bit 11	Bit 11	N check	No	Yes
11	PGA input floating	Type 2A	Bit 10	Bit 10	N check	No	Yes
10	PGA input short	Type 2A	Bit 9	Bit 9	Average 16	No	Yes
9	Watchdog timer timeout	Type 1B	Bit 8	Bit 8	One time	Yes	Yes
8	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
7	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
6	EEPROM timeout	Type 1A	Bit 5	Bit 5	One time	Yes	Yes
5	EEPROM ECC	Type 1A	Bit 4	Bit 4	One time	Yes	Yes
4	Processor error	Type 1A	Bit 3	Bit 3	One time	Yes	Yes
3	General register error	Type 1A	Bit 2	Bit 2	One time	Yes	Yes
2	OTP ECC	Type 1A	Bit 1	Bit 1	One time	Yes	Yes
1	SRAM ECC	Type 1A	Bit 0	Bit 0	One time	Yes	Yes
N/A	IIR filter not settled during power-up:						
	Bridge sensor input IIR not settled, internal or external temperature IIR settled	N/A	N/A	N/A	N/A	No	Yes
	Internal or external temperature IIR not settled, bridge sensor input IIR settled or not settled	N/A	N/A	N/A	N/A	No	Yes

[1] N/A means not applicable.
[2] VPOS/VNEG underrange faults are also used to check for VPOS or VNEG floating in normal running mode operation.
[3] For N check faults, fault detection time is N × 50 ms to (N + 1) × 50 ms. See the Fault Handling, Diagnostics, and Logging Feature section for more information.
[4] N check: to filter out transient interference for some faults, the fault handler flags them as real faults only after N consecutive occurrences, where N is the fault tolerance number, available as FAULT_TOLERANCE_NUM in the EEPROM.
[5] Average = 16: the fault flag is triggered when the parameter is sampled 16 times and then averaged and compared to the expected value.
[6] One-time faults: the fault flag is triggered on the first occurrence of the fault.
[7] ALU is the arithmetic logic unit.
[8] I_{BIAS} is the bias current.
[9] V_{CM} is the bridge input common-mode voltage.

Table 83. ADA4558 LIN Frames During Fault Condition

		LIN-U1 and LIN-U2 Frames During Fault Condition							LIN-U3 Frame During Fault Condition					
Fault ID	**Fault Name**	**Bridge Sensor Input**	**External Temperature Input**	**Internal Temperature Value**	**BS_ GLOBALFAULT**	**BS_ CHECKSUM-ERROR**	**BS_ RESPONSEERROR**	**TNCTOC**	**Bridge Sensor Input**	**Internal Temperature**	**BS_ SENSORERROR**	**BS_ RESPONSE-ERROR**	**BS_ CHECKSUM-ERROR**	**BS_ BITERROR**
30	VPOS underrange and floating	0xFFF	Normal	Normal	0	0	0	0	Normal	Normal	1	0	0	0
29	VPOS overrange	0xFFF	Normal	Normal	0	0	0	0	Normal	Normal	1	0	0	0
28	VNEG underrange and floating	0xFFF	Normal	Normal	0	0	0	0	Normal	Normal	1	0	0	0
27	VNEG overrange	0xFFF	Normal	Normal	0	0	0	0	Normal	Normal	1	0	0	0
26	ADC timeout	Normal	Normal	Normal	1	0	0	0	Normal	Normal	1	0	0	0
25	ALUs crosscheck	Normal	Normal	Normal	1	0	0	0	Normal	Normal	1	0	0	0
24	VBAT out of range	Normal	Normal	Normal	0	0	0	0	Normal	Normal	1	0	0	0
23	DVDD out of range	Normal	Normal	Normal	1	0	0	0	Normal	Normal	1	0	0	0
22	AHI5 out of range	Normal	Normal	Normal	1	0	0	0	Normal	Normal	1	0	0	0
21	LHI5 out of range	Normal	Normal	Normal	1	0	0	0	Normal	Normal	1	0	0	0
20	IBIAS out of range	Normal	Normal	Normal	1	0	0	0	Normal	Normal	1	0	0	0
19	Bridge V_{CM} out of range	0xFFF	Normal	Normal	0	0	0	0	Normal	Normal	1	0	0	0
18	Internal temperature sensor circuit, voltage out of range	0xFFF	Normal	0xFFF	0	0	0	0	Normal	Normal	1	0	0	0
	External temperature sensor circuit, voltage out of range	Normal	0xFFF	Normal	0	0	0	0	Normal	Normal	1	0	0	0
17	Internal temperature sensor circuit, self check	0xFFF	Normal	0xFFF	0	0	0	0	Normal	Normal	1	0	0	0
16	ADC self check	Normal	Normal	Normal	1	0	0	0	Normal	Normal	1	0	0	0
15	PGA gain	0xFFF	Normal	Normal	1	0	0	0	Normal	Normal	1	0	0	0
14	Reserved	Reserved												
13	PGA offset DAC	0xFFF	Normal	Normal	1	0	0	0	Normal	Normal	1	0	0	0
12	Bridge sensor out of range	0xFFF	Normal	Normal	0	0	0	0	Normal	Normal	1	0	0	0
11	PGA input floating	0xFFF	Normal	Normal	0	0	0	0	Normal	Normal	1	0	0	0
10	PGA input short	0xFFF	Normal	Normal	0	0	0	0	Normal	Normal	1	0	0	0
9	Watchdog timer timeout	No LIN response							No LIN response					
8	Reserved	Reserved							Reserved					
7	Reserved	Reserved							Reserved					
6	EEPROM timeout	No LIN response							No LIN response					
5	EEPROM ECC	No LIN response							No LIN response					
4	Processor error	No LIN response							No LIN response					
3	General register error	No LIN response							No LIN response					

Fault ID	Fault Name	LIN-U1 and LIN-U2 Frames During Fault Condition							LIN-U3 Frame During Fault Condition					
		Bridge Sensor Input	External Temperature Input	Internal Temperature Value	BS_GLOBALFAULT	BS_CHECKSUMERROR	BS_RESPONSEERROR	TNCTOC	Bridge Sensor Input	Internal Temperature	BS_SENSORERROR	BS_RESPONSEERROR	BS_CHECKSUMERROR	BS_BITERROR
2	OTP ECC	No LIN response							No LIN response					
1	SRAM ECC	No LIN response							No LIN response					
N/A[1]	IIR filter not settled during power up													
	Bridge sensor input IIR not settled, INT/EXT temperature IIR settled	0xFFE	Normal	Normal	0	0	0	1	Normal	Normal	0	0	0	0
	INT/EXT temperature IIR not settled, bridge sensor input IIR settled or not settled	0xFFE	0xFFE	0xFFE	0	0	0	1	Normal	Normal	0	0	0	0

[1] N/A means not applicable.

FAULT RECOVERY

The ADA4558 has two fault recovery modes, fast recovery mode and regular recovery mode. In fast recovery mode, the recovery time is 0 ms. In regular recovery mode, the recovery time is 130 ms. Fast recovery mode is active when all 32 bits in EEPROM Address 15 are set to 0 (FAULT_LOG_EN). Regular recovery mode is active when any of the 32 bits are set to 1 for FAULT_LOG_EN.

FAULT DESCRIPTIONS

ECC Faults (SRAM ECC, OTP ECC, EEPROM ECC)

The SRAM, OTP, and EEPROM are ECC protected to improve the reliability of the data. The ECC performs correction of single-bit errors and detection of 2-bit errors using seven ECC bits for every 32 bits of data. If a 1-bit error occurs, the ECC block corrects the error without any impact of function and no fault triggered. If a 2-bit error occurs, the error is detected, and a fault is triggered. If an error that is more than 2 bits occurs, the error is not detected, and no fault is triggered.

General Register Fault

In case of a parity check failure for register data, the register block declares a fault to the fault handler.

Processor Fault

In case of any processor related errors, the processor declares a fault to the fault handler.

EEPROM Timeout

If a timeout occurs during a read or write access to the EEPROM, the EEPROM controller declares a fault to the fault handler.

Watchdog Timer Fault

If the software execution time exceeds the defined time limit of 1.5 sec set by the watchdog clock, the timer declares a fault to the fault handler.

PGA Input Short

This start-up test checks for a short between the VPOS pin and the VNEG pin of the bridge sensor inputs. During a preshort test, the VPOS and VNEG inputs are muxed into the ADC through the voltage short positive switch (S_{VSP}) and the voltage short negative switch (S_{VSN}). The VPOS pin and the VNEG pin are checked by the processor. A postshort test consists of closing the positive shunt (S_{PSHT}) switch and the negative shunt (S_{NSHT}) switch to pull the VPOS and VNEG inputs to the ground and then sample them. The processor checks the voltage difference between the preshort and postshort VPOS and VNEG pins, and determines whether there is a short between the VPOS and VNEG inputs (see Figure 11). The input short detection resistance maximum value is not Analog Devices factory production tested, but all sub blocks contributing to this specification value are production tested, ensuring the functionality.

PGA Input Floating

The PGA input floating startup test checks for a broken bond wire at the bridge sensor inputs, VPOS and VNEG. During this fault check, the S_{FLT} switch enables a pull-down current (5 µA) on VPOS and VNEG. VPOS and VNEG are monitored by the comparator against the low reference voltage (V_{ILO}). If the inputs are floating, VPOS and VNEG are pulled low and the comparator triggers either the positive float voltage (V_{PFLT}) output or the negative float voltage (V_{NFLT}) output. This trigger indicates a floating flag. If flagged N times, the processor declares a fault to the fault handler. Flag count N is defined by FAULT_TOLERENCE_NUM in EEPROM Address 11 [19:16]

(see Figure 12). When the inputs are floating during normal running operation, lower pull-down currents are used and a VPOS and VNEG underrange fault is triggered. See the Overrange and Underrange for VPOS and VNEG section. The input open detection resistance (during startup and normal) maximum value is not Analog Devices factory production tested, but all the subblocks (pull-down current and comparator thresholds) contributing to this specification value are production factory tested, ensuring the functionality.

Bridge Sensor Input Out of Range

The processor monitors the bridge sensor signal filtered from the ADC output. If the output is out of range (0% or 100% of ADC full range) for N consecutive times, the processor declares a fault to the fault handler.

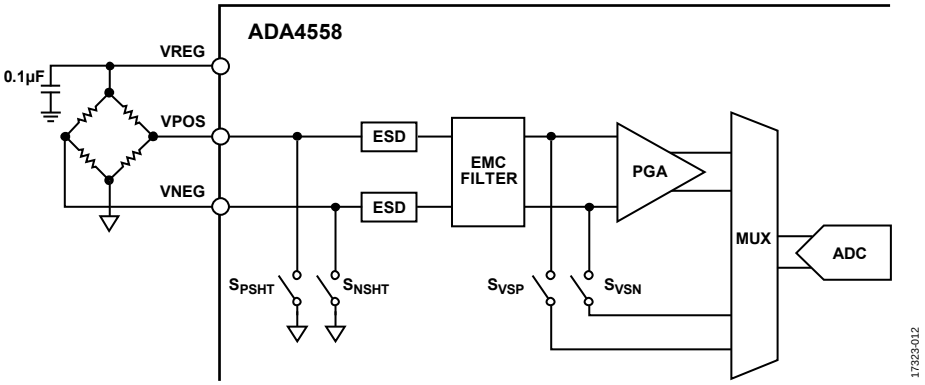

Figure 11. PGA Input Short

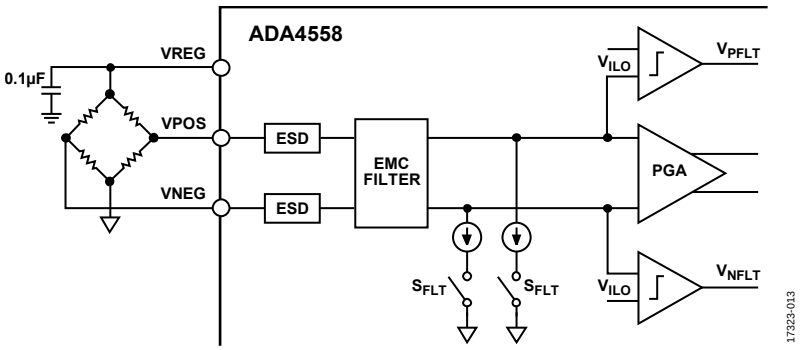

Figure 12. PGA Input Floating

PGA Offset DAC

The PGA offset digital-to-analog converter (DAC) start-up test checks the gross failures in the offset cancellation DACs for the bridge sensor signal input. During this fault test, the PGA inputs are disconnected from the VPOS and VNEG bridge sensor inputs, and are connected to an internal test DAC. An input of 0 is sent to the PGA, and the PGA loads the customer configured offset trim values. The ADC samples the PGA output multiple times and averages the samples. The processor then checks the average value against the expected value. If the check fails, the processor declares a fault to the fault handler. See Figure 13, where TRIMDACs represents the PGA offset DACs under test.

PGA Gain

The PGA gain startup test checks the gross failure in the PGA gain stages. During this fault, the PGA inputs are disconnected from the VPOS and VNEG bridge sensor signal inputs, and are connected to an internal test DAC. A near full-scale input is applied to the PGA, and the PGA loads the customer configured gain settings. The ADC samples the PGA output multiple times and averages the samples. The processor then checks the average value against the expected value. If the check fails, the processor declares a fault to the fault handler.

ADC Self Check

The ADC self check tests the gross failure of the ADC. The internal, 8-bit DAC cycles through an output voltage for digital test codes from zero to full scale. This DAC test code output voltage is drives the ADC. For each DAC test code, the processor checks the ADC output. If the check fails for N consecutive times, the processor declares a fault to the fault handler.

Internal Temperature Sensor Circuit Self Check

The internal temperature sensor circuit self check start-up test checks for the gross failures in the internal temperature sensor amplifier. During this fault, an internal DAC drives the internal temperature sensor amplifier. The DAC generates four voltage levels to cover the full temperature range. At each level, the ADC samples the amplifier output and checks the sample against the expected value. If the check fails for N consecutive times, the processor declares a fault to the fault handler. See Figure 14, where Internal V_{BE} represents the internal temperature diode junctions base emitter voltage, and R_{TRM} DAC represents the internal DAC used to generate the four test voltages.

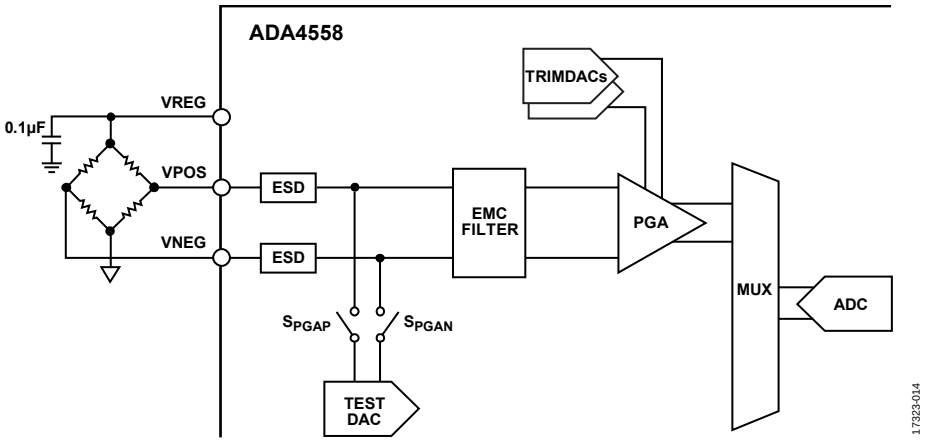

Figure 13. PGA Offset DAC and PGA Gain

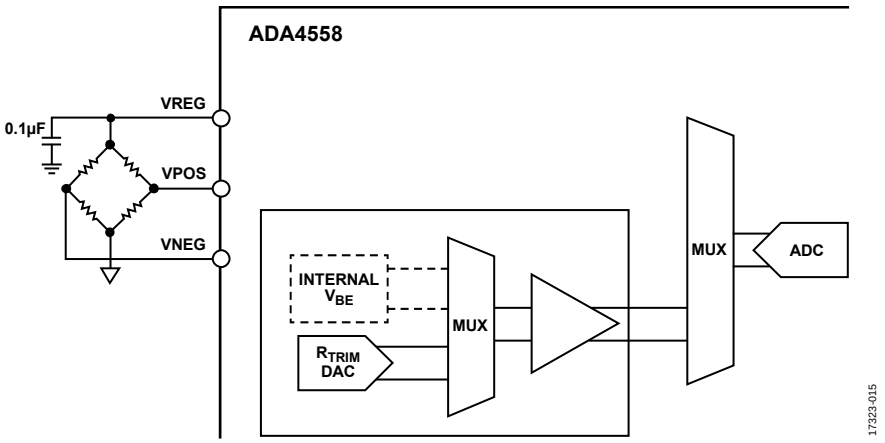

Figure 14. Simplified Internal Temperature Sensor Self Check

ALU Crosscheck

The processor ALU and the IIR filter ALU calculate the results of a predefined equation. If the two results are different for N consecutive times, the processor declares a fault to the fault handler.

ADC Timeout Fault

If the 14-bit ADC does not complete a conversion in time, the ADC controller generates a timeout signal. The processor monitors the timeout signal. If the timeout signal is triggered once, the processor declares a fault to the fault handler.

Overrange and Underrange for VPOS and VNEG

Four comparators check the VPOS and VNEG PGA inputs against V_{ILO} and the high reference voltage (V_{IHI}). The processor monitors the output of the comparators. If either input is out of range for N consecutive times, the processor declares a fault to the fault handler.

In normal running mode operation, the underrange fault for the VPOS and VNEG inputs also tests if the VPOS and VNEG pins are floating. Internal pull-down currents are switched to the VPOS and VNEG signal paths, as shown in Figure 15. When PGA_ADD_PULUP_150 = 0, a 300 nA internal pull-down current is generated, while if PGA_ADD_PULUP_150 = 1, a 450 nA internal pull-down current is generated. When either of the inputs are floating, the corresponding inputs are pulled low, which triggers the underrange fault.

When the PGA_INPUT reverse is set high, the faults are interchanged. The fault where the VPOS input is underrange and floating indicates that the VNEG input is also underrange and floating, and vice versa (see Figure 15).

Table 84. VPOS/VNEG Over range/Under range Fault Limits

Faults	Limit
VPOS/VNEG Overrange	>90% of V_{REG}
VPOS/VNEG Underrange	<18.75% of V_{REG}

Voltage Out of Range Test for Internal and External Temperature Sensor Circuits

The processor monitors the internal and external temperature sensor circuits. If the circuits generate voltages outside of the expected nominal values for N consecutive times, the processor declares a fault to the fault handler.

Bridge Input Common-Mode Voltage Out of Range

The processor monitors the V_{CM}. If the V_{CM} is less than 30% of V_{REG} or greater than 80% of V_{REG} for N consecutive times, the processor declares a fault to the fault handler.

I_{BIAS} Out of Range

The processor monitors the I_{BIAS} in the core bias block. If out of range for N consecutive times, the processor declares a fault to the fault handler.

LHI5 Out of Range

The processor monitors the internal 5 V digital regulated voltage, LHI5. If it is out of range for N consecutive times, the processor declares a fault to the fault handler.

AHI5 Out of Range

The processor monitors the internal 5 V analog regulated voltage, AHI5. If the voltage is out of range for N consecutive times, the processor declares a fault to the fault handler.

DVDD Out of Range

The processor monitors the internal 1.8 V digital regulated voltage. If the voltage is below 1.6 V or above 2.2 V for N consecutive times, the processor declares a fault to the fault handler.

VBAT Out of Range

The processor monitors the attenuated VDD12 voltage. If the voltage is below +5.3 V or above +20 V (±2 V) for N consecutive times, the processor declares a fault to the fault handler.

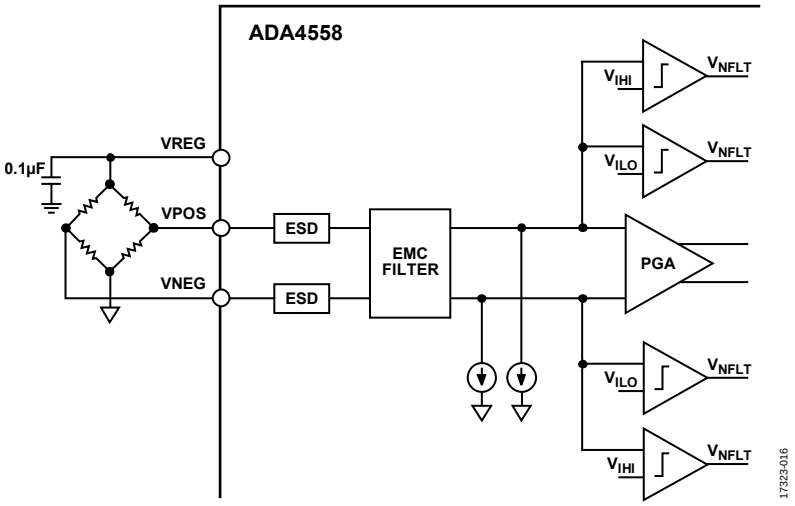

Figure 15. VPOS and VNEG Input Overrange and Underrange Faults

IIR Filter Not Settled During Power-Up

During power-up, the processor monitors IIR filters for the bridge sensor input, internal temperature, and external temperature. If the filter is not settled, a fault is triggered. The IIR filter settling time depends on the filter frequency. See Table 85 for IIR filter settling times.

The TNCTOC bit sets high when this fault is triggered, and the bridge sensor or temperature LIN readback values are clamped with one of the following two scenarios:

- When the selected temperature channel IIR is settled and the bridge sensor IIR is not settled, the LIN readback of the bridge sensor = 0xFFE and the LIN readback of the selected temperature is normal.
- When the selected temperature IIR is not settled, the LIN readback of the bridge sensor and selected temperature channel are both clamped to 0xFFE.

Table 85. IIR Filter Settling Times for the Bridge Sensor and Internal and External Temperature Channels

Filter		Settling Time	
Bridge Sensor (Hz)	**Internal and External Temperature (Hz)**	**Bridge Sensor IIR (ms)**	**Internal and External Temperature IIR (ms)**
100	100	25	25
100	40	55	55
40	100	55	25
40	40	55	55

CALIBRATION PROCEDURE

The ADA4558 is a configurable device that works with many sensor types. To optimize the sensor performance, calibrate the ADA4558 to compensate for the sensor offsets and linearity correction.

The ADA4558 requires a coarse calibration in the AFE before carrying out a fine calibration by the microcontroller. The coarse calibration maximizes the sensor signal for the ADC. The following procedures describe how to place the ADA4558 in configuration mode, and then program it for offset, gain, and temperature calibration.

PLACE THE ADA4558 IN CONFIGURATION MODE

Use the following procedures to enable writing to the EEPROM, and to place the ADA4558 in configuration mode.

Programming the PASSWD Test Mode Password Register

As per the UUD frames shown in Table 78 the LIN master sends a UUD frame to set the five PASSWD bytes to 0x01, 0x23, 0x45, 0x67, and 0x89.Table 86 shows an example master request for ADA4558 with NAD = 1.

Table 86. Example UUD Command to Program PASSWD Register

PID	NAD	PCI	SID	Data
0x3C	0x81	0x06	0x08	0x0123_4567_89

Programming the TESTMODE_SEL Test Mode Select Register

Table 87 shows an example master request for ADA4558 with NAD = 1.

Table 87. Example TESTMODE_SEL Programming

PID	NAD	PCI	SID	Data
0x3C	0x81	0x02	0x07	0x01FF_FFFF_FF

As per the UUD frames shown in Table 78, the LIN master sends a UUD frame to set the TESTMODE_SEL byte to 0x01.

The ADA4558 configuration mode enables writing to the system registers via the LIN interface, and stops the processor from regularly refreshing the register contents from the EEPROM.

To place the ADA4558 back to normal operating mode, set the TESTMODE_SEL byte to 0, or carry out a POR. Note that some registers are only loaded from the EEPROM at power-up (see Table 45).

PGA GAIN FOR THE BRIDGE SENSOR INPUT

To ensure that the ADC is not saturated by the bridge sensor signal, set the PGA gain to maximize the dynamic range of the ADC (see the Gain Settings section). Include the maximum variation of the sensor voltage when selecting the gain. Analog Devices recommends having 5% headroom at both extremes of the ADC range on the ADA4558.

For example, select the PGA gain for a sensor with a maximum span of 17 mV/V and a total 25% variation of span and offset over temperature and lifetime. In this example, the sensor voltage span is 68 mV (4 V × 17 mV/V).

Apply the 25% variation over temperature and lifetime to give a maximum voltage span of 85 mV. To calculate the target gain, use the following equation:

$$Target\ Gain = Target\ Gained\ Voltage\ Span \div Maximum\ Input\ Voltage\ Span \quad (6)$$

Due to internal chopping noise, the target gained voltage span changes depending on the PGA gain selected. Table 1 shows the target gained voltage spans allowed for each gain option. Refer to Table 1 to select the highest gain that is within the useable output span range. A gain of 32.5 V/V is the optimal gain for this example, and provides a maximum gained span of 2.79565 V/V for the 85 mV input.

CALCULATING THE COMMON-MODE VOLTAGE TRIM

This section provides the steps to calculate the BRG_VCM_TRIM[6:0] value for the V_{CM} trim.

The V_{CM} trim allows the PGA to operate over a wide range of V_{CM} and gain settings without saturating the second and third stage of the PGA. Use the 7-bit BRG_VCM_TRIM register to trim the V_{CM} of the sensor. Set BRG_VCM_TRIM to mid-scale for initial settings, and then adjust up or down for positive or negative V_{CM}.

To perform common-mode calibration, take the following steps:

1. Initial gain setting: program the PGA_GAIN (SYSREG Address 0xB1) to 2.94 V/V to ensure the PGA is not saturated.
2. Initial V_{CM} trim setting: program BRG_VCM_TRIM (EEPROM Address 0x09) initially to mid-scale (value = 0x40)
3. Read the V_{CM}: set the bridge sensor strain gauge to 0%, and then read the ADC data from the PGA_VICM (SYSREG Address 0x64) location. The read result is the V_{CM} of the bridge.
4. Note that the PGA_VICM register returns a 14-bit ADC code while the trim register BRG_VCM_TRIM is a 7-bit register. To reformat the 14-bit reading for the 7-bit trim register, right shift the data read from PGA_VICM[13:0] by seven bits. Record this V_{CM} trim value (VCM_TRIM).
5. At the end of the full offset calibration routine, program the shifted data into BRG_VCM_TRIM[6:0] at EEPROM Address 0x09[27:21] together with the other trim values.

CALCULATING THE SENSOR OFFSET VOLTAGE TRIM AT PGA STAGE 1

This section provides the steps to calculate the BRG_VOS_TRIM1[6:0] value for sensor offset voltage trim at PGA Stage 1.

The bridge sensors offset voltage is adjusted at three stages of the PGA. Figure 16 shows that PGA_DAC_1_2 is used at Stage 1, PGA_DAC_3_4 is used at Stage 2, and PGA_DAC5 is used at Stage 3.

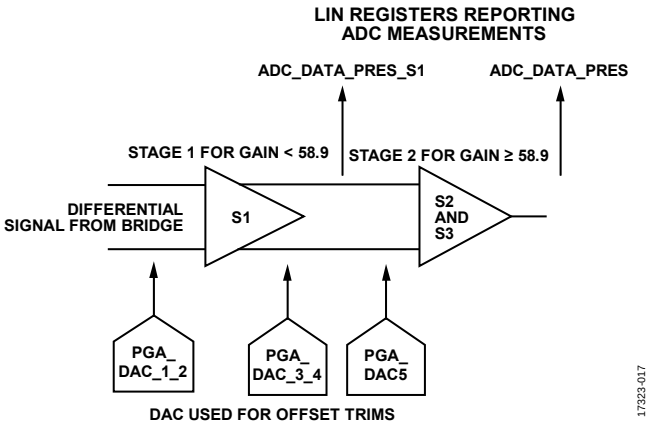

Figure 16. Configuring the PGA offset for S1, S2, and S3

PGA_DAC_1_2 is a 15-bit register at SYSREG Address 0xB2. The register contains two DAC codes to define PGA Stage 1 differential offset values.

DAC_1 is a value represented by PGA_DAC_1_2[6:0]. DAC_2 is a value represented by PGA_DAC_1_2[14:8]. Bit 7 of PGA_DAC_1_2 is not used and should be 0.

Define the trim values for PGA Stage 1 by using the following procedure:

1. The initial value for DAC_1 and DAC_2 must equal the V_{CM} measurement programmed to BRG_VCM_TRIM. To ensure that the initial values equal the V_{CM}, use the following equation:

$$PGA\_DAC\_1\_2 = (DAC\_2 << 7) + DAC\_1 \quad (7)$$

   where:
   DAC_1 = VCM_TRIM.
   DAC_2 = VCM_TRIM.
2. Set the initial PGA_DAC_3_4. PGA_DAC_3_4 is a 15-bit register at SYSREG Address 0xB3. Bits[6:0] represent DAC_3, and Bits[14:8] represent DAC_4. Set Bit 7, which is not used, to 0.
3. The initial value for DAC_3 and DAC_4 must equal the V_{CM} measured and defined in BRG_VCM_TRIM. To ensure that the initial values equal the V_{CM}, use the following equation:

$$PGA\_DAC\_3\_4 = (DAC\_4 << 7) + DAC\_3 \quad (8)$$

   where:
   DAC_3 = VCM_TRIM.
   DAC_4 = VCM_TRIM.
4. Set the initial PGA_DAC5. PGA_DAC5 is a 7-bit value. Set the PGA_DAC5 initial value to the mid-scale value of 0x40. This setting allows a positive or negative trim adjustment during calibration.
5. Set the initial PGA_GAIN. Program the PGA_GAIN (SYSREG Address 0xB1) to either 11.8 or the target gain setting if the setting is lower than 11.8. This programming ensures that the PGA is not saturated and operates at the middle gain for Stage 1.
6. Set the bridge sensor strain gauge to 0%.
7. Read back the ADC_DATA_PRES_S1 register (SYSREG Address 0x94) to read back the ADC code representing the voltage output of Stage 1.
8. The ADC_DATA_PRES_S1 is a 14-bit ADC code of a differential signal. For an input signal of 0%, the ideal value is mid-scale at 0x2000. Adjust the PGA_DAC_1_2 values to set the ADC_DATA_PRES_S1 as close to 0x2000 as possible for an input signal of 0%.
9. Carry out a binary search to find the required PGA_DAC_1_2 values. If ADC_DATA_PRES_S1 is <0x2000, set DAC_1 to VCM_TRIM + 32, and set DAC_2 to VCM_TRIM − 32. If ADC_DATA_PRES_S1 is >0x2000, set DAC_1 to VCM_TRIM − 32, and set DAC_2 to VCM_TRIM + 32.
10. Continue the binary search method to the target of ADC_DATA_PRES_S1 = 0x2000. At each trial, divide the code delta by two (that is, add or subtract 32 for Trial 1, add or subtract 16 for Trial 2, and add or subtract 8 for Trial 3). The maximum number of trials is seven.
11. To calculate the final value for BRG_VOS_TRIM1[6:0], use the following formula:

$$BRG\_VOS\_TRIM1[6:0] = (DAC2 - DAC1) \div 2 \quad (9)$$

12. Program the BRG_VOS_TRIM1[6:0] value to EEPROM Address 0x09 at the end of the trim procedure. Calculate the twos complement for negative values.

SENSOR OFFSET VOLTAGE TRIM, PGA STAGE 2

Use PGA Stage 2 offset trim for gains that are ≥58.9 V/V. Use BRG_VOS_TRIM2 to adjust the offset at PGA Stage 2. When the trim of the offset of PGA Stage 1 is completed, set the PGA gain back to the final production value. For gain settings that are <58.9 V/V, the BRG_VOS_TRIM2 value is not used and does not need to be calculated.

For gain settings that are ≥58.9 V/V, take the following steps:

1. Set the initial PGA_DAC_3_4 initial settings. Program PGA_DAC_3_4 (SYSREG Address 0xB3) using the value calculated for VCM_TRIM. To calculate the value, use the following equation:

$$PGA\_DAC\_3\_4 = (VCM\_TRIM << 7) + VCM\_TRIM \quad (10)$$

   This equation sets zero offset correction in either direction.
2. Ensure the PGA_DAC_1_2 is set up from the Calculating the Sensor Offset Voltage Trim at PGA Stage 1 section. For the initial PGA_DAC5 settings, set PGA_DAC5[6:0] to the midscale value of 0x40.
3. Set the bridge sensor strain gauge to 0%.
4. Read back the ADC_DATA_PRES register value with SYSREG Address 0x90.
5. The ADC_DATA_PRES is the 14-bit ADC code of a single-ended signal. TO adjust the ADC_DATA_PRES value, use the PGA_DAC_3_4 values. To allow for variations, add headroom at this end of scale range. An example ADC target for the 0% signal input is 15% × 2^{14}. This example allows variation of the bridge over temperature because the offset is being trimmed at a single temperature.
6. Adjust the PGA_DAC_3_4 values to set the ADC_DATA_PRES as close to the target code as possible for an input signal of 0%.
7. If ADC_DATA_PRES is less than the target code, set DAC_3 to VCM_TRIM − 32, and set DAC_4 to VCM_TRIM + 32. If ADC_DATA_PRES is greater than the target code, set DAC_3 to VCM_TRIM + 32, and set DAC_4 to VCM_TRIM − 32.
8. Continue the binary search method to the target of ADC_DATA_PRES = target. At each trial, divide the code delta by two (that is, add or subtract 32 for Trial 1, add or subtract 16 for Trial 2, add or subtract 8 for Trial 3, and so forth). The maximum number of trials is seven.
9. Calculate the final value for BRG_VOS_TRIM2[6:0] using the following formula:

$$BRG\_VOS\_TRIM2[6:0] = (DAC4 - DAC3) \div 2 \quad (11)$$

10. Program the BRG_VOS_TRIM2[6:0] value to EEPROM Address 0x09 at the end of the trim procedure together with BRG_VOS_TRIM1 and PGA_DAC5. Calculate the twos complement for negative values.

SENSOR OFFSET VOLTAGE TRIM, PGA STAGE 3

This section describes how to set the PGA_DAC5[6:0] setting for the PGA Stage 3 configuration.

PGA_DAC5 adjusts the output of the final PGA stage to ensure that there is sufficient headroom at the bottom range of the ADC. After the trim is complete, there is sufficient margin to deal with negative offset variations over temperature and lifetime. This margin ensures that the input to the ADC is not saturated.

To set the PGA_DAC5[6:0], take the following steps:

1. Read back the ADC_DATA_PRES register value with SYSREG Address 0x90. The target is the 14-bit ADC code representing the target percentage for the calibration bridge sensor level. An example target is 15% × 2^{14}.
2. Calculate the PGA_DAC5 value required to bring the readback ADC value to the target value.
3. PGA_DAC5 was set to 0x40 for BRG_VOS_ TRIM2, and can be adjusted up or down. Each LSB of PGA_DAC5 corresponds to approximately 128 ADC codes. See the EVAL-ADA4558EBZ user guide for examples of how to carry out calibration trims of V_{CM}, sensor offset, and PGA gain for the bridge sensor input.

GAIN SETTING FOR EXTERNAL TEMPERATURE SENSOR

When programming EEPROM values, set the ADA4558 to Test Mode 1 as described in Table 86 and Table 87.

The ADA4558 has an external temperature sensor input that measures the temperature of a remote PT100 or PT1000. The remote RTD is biased using an on-chip current source. An R_{REF} reference resistor is also measured in series to provide a reference temperature. See Figure 5 for the connection and block diagram.

Define the following items to set up the external temperature channel:

- Define the reference resistor RREF value.
- Select an external temperature sensor type (PT100 or PT1000).
- Select a bias current value for the selected RTD.
- Select a PGA Gain for the RTD and reference resistor.

Choose the PT type, reference resistor, and bias current so that the voltage at the AFE inputs are not close to the rails to avoid clipping due to EMI, and are large enough to minimize the effects of EMI and offset drift.

The input parameters for this example include the following:

- RTD type is PT1000
- Temperature range = −40°C to +150°C
- I_{SINK} = 0.2 mA
- Minimum resistance of RTD at −40°C = 842.7 Ω
- Maximum resistance of RTD at 150°C = 1573.1 Ω

- V_{REF} = 4 V ± 5%
- Maximum dynamic range of ADC = 0.1 V to 4.0 V
- Minimum voltage (V_{TEMP_MIN}) across the RTD = 0.2 mA × 842.7 Ω = 0.16854 V
- Maximum voltage (V_{TEMP_MAX}) across the RTD = 0.2 mA × 1573.1 Ω = 0.31462 V.

To select the temperature channel gain, choose R_{REF} to bias the TPOS pin and the TNEG pin close to the common mode of 2.5 V. Use the following equation:

$$R_{REF} = \frac{VREG - V_{CM}}{I_{SINK}} = \frac{4\text{ V} - 2.5\text{ V}}{0.2\text{ mA}} = 7.5\text{ k}\Omega \quad (12)$$

$$V_{RREF\_MAX} = I_{SINK} \times R_{REF} = 0.2\text{ mA} \times 7.5\text{ k}\Omega = 1.5\text{ V} \quad (13)$$

To select the gain for the external reference resistor input, select the EXTTS_REF channel total gain (shown as REF Channel Total Gain) to maximize the dynamic range of the ADC by using the following equations:

$$REF\ Channel\ Total\ Gain = \frac{ADC\ Maximum\ Dynamic\ Range}{V_{RREF\_MAX}} \quad (14)$$

$$REF\ Channel\ Total\ Gain = \frac{(4\text{ V} - 0.1\text{ V})}{1.5\text{ V}} = 2.6\text{ V} \quad (15)$$

To select the closest available gain that is not more than 2.6, select the total gain (G1 × G2) of 2.25, and select EXTTS_REF_GAIN1 = 1.5, and EXTTS_REF_GAIN2 = 1.5.

To select the gain for external RTD input, select the temperature channel total gain to maximize the dynamic range of the ADC. Use the following equations:

$$Temperature\ Channel\ Total\ Gain = \frac{ADC\ Maximum\ Dynamic\ Range}{V_{TEMP\_MAX}} \quad (16)$$

$$Temperature\ Channel\ Total\ Gain = \frac{(4 - 0.1)\text{ V}}{0.3146\text{ V}} = 12.39 \quad (17)$$

To select the closest available gain not greater than 12.39, select a total gain (G1 × G2) of 12.

Select EXTTS_TEMP_GAIN1 = 12, and EXTTS_TEMP_GAIN2 = 1.

CALIBRATION ROUTINE FOR EXTERNAL TEMPERATURE SENSOR

Complete the AFE setup for the external temperature sensor before carrying out this calibration routine, which includes setting the gain for the RTD, the gain for the reference resistor, and selecting the I_{SINK} for the RTD.

The calibration routine for the external temperature sensor generates five coefficients that must be programmed to the EEPROM. Table 97 shows the external temperature coefficients

The calibration routine requires collecting ADC codes of the RTD for selected temperature calibration points, and applying the ADC codes and the expected temperature points to an executable application to generate the five temperature coefficients.

When the coefficients are generated, use TEMP_COEFF_K to apply a temperature compensation constant. The final corrected temperature value is available as EXT_ TEMPERATURE _12 in LIN signal frame LIN-U2.

The calibration routine steps are as follows:

1. Select the RTD temperature to 25°C.
2. Read from the ADC_DATA_EXTTS (Register Address 0x92). See Table 89 for an example LIN command request. Read from ADC_DATA_EXTTS_RREF (Register Address 0x93). See Table 88 for an example LIN command request.
3. Calculate $ADC\_RREF\_INV\_CONST_{FLOAT}$ with the following equation:

$$ADC\_REF\_INV\_CONST_{FLOAT} = \frac{EXT\_RREF\_GAIN}{ADC\_DATA\_EXTTS\_REF / 2^{14}} \quad (18)$$

4. Calculate $ADC\_RREF\_INV\_CONST_{FIXED}$ with the following equation:

$$ADC\_RREF\_INV\_CONST_{FIXED} = (\text{floor}(((ADC\_RREF\_INV\_CONST_{FLOAT} \times 2^7) + 0.5))/2^7 \quad (19)$$

5. Change the RTD temperature to −40°C.
6. Read from ADC_DATA_EXTTS and ADC_DATA_EXTTS_REF.
7. Change the RTD temperature to 150°C.
8. Read from ADC_DATA_EXTTS and ADC_DATA_EXTTS_REF.
9. For each ADC_DATA measurement in these steps, calculate the referred to input value with the following equation:

$$ADC\_DATA_{RTI} = \frac{ADC\_DATA / 2^{14}}{GAIN} \quad (20)$$

where:
GAIN = EXT_RREF_GAIN
ADC_DATA = ADC_DATA_EXTTS_REF
GAIN = EXT_TEMP_GAIN
ADC_DATA = ADC_DATA_EXTTS
ADC_DATA_{RTI} is the referred to input calculation of the external temperature measurement.

10. Calculate the normalization factor with the following equations:

$$NF_{FLOAT} = \frac{ADC\_DATA\_EXTTS\_REF_{RTI\_25}}{ADC\_DATA\_EXTTS_{RTI\_-40}} \quad (21)$$

$$NF_{FIXED} = \left(\text{floor}\left(\left(NF_{FLOAT} \times 4\right) + 0.5\right)\right) / 4 \quad (22)$$

where:
NF_{FLOAT} is the normal factor in floating point form.
$ADC\_DATA\_EXTTS\_REF_{RTI\_25}$ is the value calculated with Equation 20 for a temperature input of 25°C.
$ADC\_DATA\_EXTTS_{RTI\_40}$ is the value calculated with Equation 20 for a temperature input of −40°C.
NF_{FIXED} is the normal factor in fixed point form.

11. Calculate A0, A1, and A2 as a solution to the system of equations shown in the External Temperature Sensor Equations section. A0 represents TCORR_COEFF0, A1 represents TCORR_COEFF1, and A2 represents TCORR_COEFF2. Note that T_{OUT_TDEGC} equals the expected LIN output code for a particular temperature, normalized to the range of 0 to 1.
12. To convert A0, A1, and A2 to fixed point coefficients, use the following equations:

$$EXTTS\_A0 = (\text{floor}\ ((A0 \times 2^{13}) + 0.5))/2^{13} \quad (23)$$

$$EXTTS\_A1 = (\text{floor}\ ((A1 \times 2^{13}) + 0.5))/2^{13} \quad (24)$$

$$EXTTS\_A2 = (\text{floor}\ ((A2 \times 2^{13}) + 0.5))/2^{13} \quad (25)$$

13. Write all calculated calibration software parameters to the EEPROM.

See the evaluation board EVAL-ADA4558EBZ user guide for an example calculation using a coefficient calculator where the ADC data is collected in a file, as shown in Figure 17.

CALIBRATION TEMPERATURES (°C)	ADC_DATA_EXTTS_REF	ADC_DATA_EXTTS	NORMALIZED TEMPERATURE
−40.1000	0x18BE	0xC82	−0.000526
55.1000	0x18BE	0x1217	0.500526
150.2000	0x18C0	0x177C	1.001053

17323-018

*Figure 17. Example Input File (**trim.dat**) Format for External Temperature Coefficient Generator*

The coefficient calculator used with the EVAL-ADA4558EBZ is an executable to calculate five temperature correction coefficients. The user must validate any coefficients to verify they are suitable for use.

To use the calculator, the executable requires an input file, **trim.dat.** Locate the **trim.dat** file in the same directory as the executable. The executable file, **extts_cal.exe**, and requires three parameters when calling the file. See Table 96 for descriptions of the parameters. An example call to generate coefficients based on an RTD gain setting of 9 and an R_{REF} gain setting of 2 is **extts_cal.exe 2 9 1**.

The executable generates the **nvm.dat** and **nvm_fl.dat** output files. The **nvm.dat** file contains the five coefficients in hex format that need to be programmed to the EEPROM. Figure 18 shows the format of the output **nvm.dat** file to identify each coefficient. Write the five calculated coefficients to the EEPROM.

To program the external temperature coefficients to the EEPROM, take the following steps:

1. Enable configuration mode as described in Table 86 and Table 87.
2. Write the ADC_RREF_INV_CONST value to EEPROM Address 0x08, write the NORMAL_FACTOR value to EEPROM Address 0x0C, write the TCORR_COEFF0 and TCORR_COEFF1 values to EEPROM Address 0x0A, and write the TCORR_COEFF2 value to EEPROM Address 0x0B (see Table 90 to Table 93).
   Define the corrected external temperature constant. The ADA4558 uses the TEMP_COEFF_K value at EEPROM Address 13 to optionally add or subtract a correction constant. The correction constant is a multiple of the difference in external and internal temperature. The multiple factor is defined in Table 94. If this correction factor is not required, set it to 0. Calculate the corrected external temperature using the following equation:

$$T_{EXT\_CORRECTED} = T_{EXT} + K \times (T_{EXT} - T_{INT}) \quad (26)$$

   where:
   $T_{EXT\_CORRECTED}$ is the corrected external temperature. This value is available over the LIN signal frame (LIN-U2) as EXT_TEMP_12.
   T_{EXT} is the external temperature. This value is the external temperature value computed using the external temperature correction coefficients.
   K is the external temperature compensation constant stored in the EEPROM at Address 13. The decoding for K is shown in Table 94.
   T_{INT} is the internal temperature sensor. This value is available over the LIN signal frame as TEMP_9 and TEMP_12.
3. Write the TEMP_COEFF_K values to the EEPROM Address 0x0D.

To verify the external temperature coefficients, take the following steps:

1. Exit configuration mode and return to normal mode. This step restarts caching of the EEPROM variables to the LIN registers in the SRAM. After this command, allow four seconds to completely refresh all EEPROM variables. Send the configuration password, as shown in Table 86, followed by the command to set normal mode, as shown in Table 95
2. Use the LIN signal frame to read and verify external temperature measurements.

Table 88. Read Reference Temperature Command with Address 0x93 in Data Field

PID	NAD	PCI	SID	Data
3C	NAD	0x02	0x02	0x93FF_FFFF_FF

Table 89. Read RTD Temperature Command with Address 0x92 in Data Field

PID	NAD	PCI	SID	Data
3C	NAD	0x02	0x02	0x92FF_FFFF_FF

Table 90. Write ADC_RREF_INV_CONST to EEPROM Address 0x08 Where XX_XXXX_XX is the ADC_RREF_INV_CONST Value

PID	NAD	PCI	SID	Data
3C	NAD	0x06	0x04	0x08XX_XXXX_XX

Table 91. Write NORMAL_FACTOR to EEPROM Address 0x0C Where XX_XXXX_XX is the NORMAL_FACTOR Value

PID	NAD	PCI	SID	Data
3C	NAD	0x06	0x04	0x0CXX_XXXX_XX

Table 92. Write TCORR_COEFF0 and TCORR_COEFF1 to EEPROM Address 0x0A

PID	NAD	PCI	SID	Data
3C	NAD	0x06	0x04	0x0AXX_XXXX_XX

Table 93. Write TCORR_COEFF2 to EEPROM Address 0x0B

PID	NAD	PCI	SID	Data
3C	NAD	0x06	0x04	0x0BXX_XXXX_XX

Table 94. Decoding for the External Temperature Compensation Constant, K

Binary	Hexadecimal	Decimal
0 0000000000	0x000	0 (zero)
0 0000000001	0x001	0.001953125
0 1111111111	0x3FF	1.998046875 (maximum)
1 0000000000	0x400	−2 (minimum)
1 0000000001	0x401	−1.998046875
1 1111111111	0x7FF	−0.001953125

Table 95. Example TESTMODE_SEL Programming for Normal Mode

PID	NAD	PCI	SID	DATA
3C	NAD	0x02	0x07	0x00FF_FFFF_FF

Table 96. Parameters for Generating Temperature Coefficients

Parameter	Description	Comments
First Parameter	Total gain value for the reference resistor (R_{REF})	Usable values 1 to 36 Example: G1 = 1.5, G2 = 1.5, total gain = 1.5 × 1.5 = 2.25
Second Parameter	Total gain value for the RTD	Usable values 1 to 36 Example: G1 = 2, G2 = 6, total gain = 2 × 6 = 12
Third Parameter	Generate square term	Always set to 1

```
0xDE22 ← TCORR_COEFF0
0x401C ← TCORR_COEFF1
0xD6A  ← TCORR_COEFF2
0x12   ← NORMAL_FACTOR
0x296  ← ADC_RREF_INV_CONST
```

17323-019

*Figure 18. Example Output File (**nvm.dat**) Format for External Temperature Coefficient Generator*

Table 97. External Temperature Sensor Coefficients and Settings

Parameter	Description	Format
TEMP_SEL	Selects internal or external temperature sensor for bridge sensor correction.	Not applicable
TCORR_COEFF2[15:0]	Second order coefficient for external temperature coefficient.	Q3.13
TCORR_COEFF1[15:0]	Gain coefficient for external temperature correction.	Q3.13
TCORR_COEFF0[15:0]	Offset coefficient for external temperature correction.	Q3.13
NORMAL_FACTOR	MAX_ACROSS_TEMP indicates the maximum value across temperature for the equation ADCRTEMP ÷ ADCRREF, where ADCRTEMP and ADCRREF are divided by the respective PGA gains. NORMAL_FACTOR limits the range of the coefficients that are needed.	Q8.2
ADC_RREF_INV_CONST[11:0]	Scaling factor for 25°C reference ADC value.	Q5.7
TEMP_COEFF_K[10:0]	External temperature constant, K.	Not applicable

External Temperature Sensor Equations

To solve for the external temperature sensor coefficients, use Equation 26 to Equation 28 (see the Calibration Routine for External Temperature Sensor section).

$$T_{OUT\_-40} = A0 + A1 \times NF \times \frac{ADC\_DATA\_EXTTS_{RTI\_-40}}{ADC\_DATA\_EXTTS\_REF_{RTI\_-40}} + \left(A2 \times NF \times \frac{ADC\_DATA\_EXTTS_{RTI\_-40}}{ADC\_DATA\_EXTTS\_REF_{RTI\_-40}} \right)^2 \quad (27)$$

$$T_{OUT\_25} = A0 + A1 \times NF \times \frac{ADC\_DATA\_EXTTS_{RTI\_25}}{ADC\_DATA\_EXTTS\_REF_{RTI\_25}} + \left(A2 \times NF \times \frac{ADC\_DATA\_EXTTS_{RTI\_25}}{ADC\_DATA\_EXTTS\_REF_{RTI\_25}} \right)^2 \quad (28)$$

$$T_{OUT\_150} = A0 + A1 \times NF \times \frac{ADC\_DATA\_EXTTS_{RTI\_150}}{ADC\_DATA\_EXTTS\_REF_{RTI\_150}} + \left(A2 \times NF \times \frac{ADC\_DATA\_EXTTS_{RTI\_150}}{ADC\_DATA\_EXTTS\_REF_{RTI\_150}} \right)^2 \quad (29)$$

where:

T_{OUT_-40} is the expected LIN output code for −40°C.
T_{OUT_25} is the expected LIN output code for 25°C.
T_{OUT_150} is the expected LIN output code for 150°C.

BRIDGE SENSOR CHANNEL CALIBRATION

Ensure that the AFE setup for the bridge sensor input has been completed, and that the bridge sensor offset and gain settings have been set.

Before carrying out the bridge sensor channel calibration routine, determine how temperature compensation will be done. For the remote RTD, perform the AFE setup and temperature calibration routine. For the on-chip temperature sensor, no temperature calibration is required for the bridge sensor channel correction, but there is an option to map the internal temperature sensor to a linear 12-bit or 9-bit scale for LIN signal frames.

The ADA4558 corrects Wheatstone bridge sensors based on the following model for V_{BRIDGE}. The six coefficients (A0 to A5) define the bridge sensor characteristic to a second-order sensitivity (P) and temperature (T).

$$V_{BRIDGE} = A0 + (A1 \times P) + (A2 \times T) + (A3 \times P^2) + (A4 \times PT) + (A5 \times T^2) \quad (30)$$

where:
A0 = Sensor offset coefficient.
A1 = Sensor first-order sensitivity coefficient.
A2 = Sensor first-order temperature coefficient.
A3 = Sensor second-order sensitivity coefficient.
A4 = Sensor sensitivity temperature coefficient.
A5 = Sensor second-order temperature coefficient.

See the EVAL-ADA4558EBZ user guide for an example correction coefficient calculation using an executable calculator. The user must validate that the coefficients are suitable for use. The evaluation board software takes input calibration ADC codes and generates outputs of up to 15 calibration coefficients based on the sensor model. Write these coefficients to the ADA4558 EEPROM.

Bridge Sensor Channel Calibration Routine

To collect the data for the **trim.dat** file shown in Figure 19, use the following steps:

1. Define the calibration points. For a second-order correction of sensitivity and temperature, a minimum of six calibration points is required, and they must be spread out over the temperature range and bridge signal. Table 98 shows an example of six calibration points for a second-order sensor model.
   For sensors with more linear characteristics, the number of calibration points can be reduced. For example, if a sensor has no A5 coefficient (second-order temperature coefficient), the number of calibration points can be reduced to 5.
   More calibration points yield higher linearization accuracy. Note that more calibration points only increase accuracy if they are appropriately selected. For example, choosing three temperature calibration points close to each other likely degrades accuracy because the curve fitting has to extrapolate out the temperature extremes.
2. For each calibration point, set the target temperature and bridge sensor conditions. Read the temperature and bridge sensor input ADC codes depending on which sensor temperature compensation is selected.
   If on-chip temperature channel is selected for sensor temperature compensation, read the 14-bit ADC measurement code (ADC_DATA_TEMP at SYSREG Address 0x91). If remote RTD temperature channel is selected for sensor temperature compensation, read the 12-bit ADC measurement code (EXT_TEMP_12 from LIN Signal Frame LIN-U2.)
3. For external temperature sensor ADC codes, multiply the ADC code value by 4 to convert the ADC code from 12-bit to 14-bit. Save the 14-bit version to a file with the other calibration data.
   Read the 14-bit ADC measurement code for the bridge sensor input (ADC_DATA_PRES at SYSREG Address 0x90).
4. Calculate the normalized bridge sensor input signal applied for each calibration point, where 10 is 100% bridge sensor input signal level, and 0 is 0% bridge sensor input signal level.
5. The **trim.dat** file contains one line per calibration point, and has four columns. The lines do not need to be in any particular sequence, but the columns must be sequenced and tab delimited, as shown in Figure 19. This example shows four temperature points and four bridge sensor input signal points.
   Copy the values to the four columns in the trim.dat file as temperature in °C, temperature in ADC code measured at the ADA4558, bridge sensor signal in normalized units of 0 to 10, and bridge sensor signal in ADC code measured by the ADA4558.

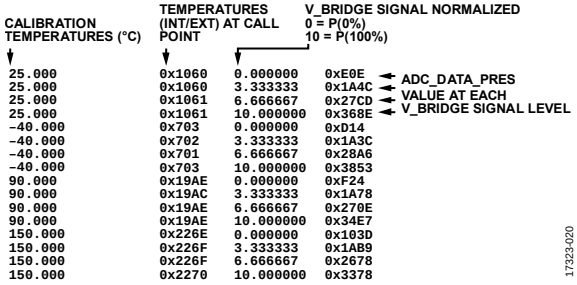

Figure 19. Example Input File to Coefficient Generator

Table 98. Example Calibration Points for Second-Order Correction

Temperature Input	Sensor Input Minimum	Sensor Input Mid-Scale	Sensor Input Maximum
Temperature Maximum	Calibration Point 5		Calibration Point 6
Temperature Mid-scale		Calibration Point 4	
Temperature Minimum	Calibration Point 1	Calibration Point 2	Calibration Point 3

Calculating Bridge Sensor Coefficients:

See the EVAL-ADA4558EBZ user guide for an example correction coefficient cautions. The user must validate that the coefficients are suitable for use. The correction coefficient calculator is an executable called **ada4558_p_cal.exe**. This executable uses the **trim.dat** input file and has eleven configurable parameters. Save the input **trim.dat** file in the same folder as the executable.

The output is the **nvm.dat** text file, which contains the bridge sensor coefficients. An example output file is shown in Figure 20, where the coefficients are shown in hex format and listed from Coefficient C0 to a maximum of 15 coefficients. Write the calculated coefficients to CORR_COEFF0 to CORR_COEFF14 in EEPROM Address range 0 to 8.

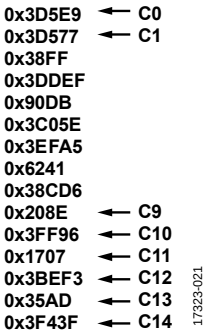

Figure 20. Example Output File from Coefficient Generator

The executable requires 11 command line parameters, which define how it performs the correction calculations.

This example calls to the coefficient generator including **ada4558_p_cal.exe 0 1 4 1 1 1 1 1 1 0 1,** **ada4558_p_cal.exe 0 1 4 1 1 1 1 1 1 1 0.01 0.98**, and **ada4558_p_cal.exe 0 1 4 1 1 1 0 0 0 0 1**.

Table 99 shows the following parameter options:

- Command Line Position 1 and Command Line Position 2: temperature settings that add additional offset or gain value to allow algorithm temperature input adjustment. Select the default values.
- Command Line Position 3: order of correction equation. Select the order of polynomial mapping used in coefficient calculations. The ADA4558 uses a fourth order polynomial equation to linearize the output measurements. To maximize the linearization, select the default value of 4.
- Command Line Position 4 to Command Line Position 9: enable the coefficients and settings for each coefficient from A0 to A5. The bridge sensor is modeled for second order nonlinearity with five coefficients. Customize the ADA4558 to use all six coefficients or selected coefficients, depending on the characteristics of the sensor enabling or disabling the specific coefficients.
- Command Line Position 10: enter a target normalized value to normalize the output settings when the bridge signal is at a minimum level of 0%. The normalized value ranges from 0 to 1. The default is 0 to indicate that an ADC value of 0x0 represents bridge signal level of 0%. To adjust the minimum value scale, set this parameter value to (target ADC code at 0% bridge signal) ÷ full scale. For example, to set the target code at 0% bridge signal to 41 ADC code, set the parameter value to 41 ÷ 4095 = 0.01.
- Command Line Position 11: enter the target normalized value for the bridge span from 0% to 100% to normalize the output span. Normalized values range from 0 to 1. The default is 1 to indicate that the full scale of ADC (0 to 4095) is the same as the full scale of the bridge signal (0% to 100%).
- To adjust the scale, set this parameter value to (target ADC code at 100% signal − target ADC code at 0% signal) ÷ full scale. For example, to set the target ADC code at 100% signal to 4054 and target ADC code at 0% signal to 41, set the parameter value to (4054 − 41) ÷ 4095 = 0.98.

Table 99. Command Line Parameters for Bridge Sensor Coefficient Calculator

Command Line Position	Default Command Line Value	Parameters	Description	Comments
1	0	TS_OFS	Normalized temperature sensor offset	Values 0 to 1
2	1	TS_GAIN	Normalized temperature sensor gain	Values 0 to 1
3	4	CORRECTION_ORDER	Define order of correction equation to use on-chip	Values 1 or 2 or 4
4	1	CORRECT_OFS	Enable correction of system offset	Boolean value 0 or 1
5	1	CORRECT_TEMPCO	Enable correction of system temperature coefficients	Boolean value 0 or 1
6	1	CORRECT_GAIN	Enable correction of system gain error	Boolean value 0 or 1
7	1	CORRECT_TSQ	Enable correction of second-order temperature coefficients	Boolean value 0 or 1
8	1	CORRECT_PT	Enable gain error temperature coefficients correction	Boolean value 0 or 1
9	1	CORRECT_PSQ	Enable second-order bridge correction	Boolean value 0 or 1
10	0	LIN_OFS	Defines normalized target output code	Values 0 to 1 (default = 0)
11	1	LIN_SPAN	Defines normalized target output span	Values 0 to 1 (default = 1)

NOTES