A17820 END-OF-LINE CALIBRATION

By Solène Bastien
Allegro MicroSystems

INTRODUCTION

The A17820 is a high-precision inductive position-sensor interface IC designed for low-speed linear or rotary applications.

This application note provides the essential information needed to use this IC by describing:

- Usage of the IC
- Compensations available
- Compensation usage strategies
- How to calibrate and program compensations

USAGE OF THE IC

Designed to support rotary and shot-linear-stroke applications, the A17820 features advanced end-of-line calibration and digital compensations that accurately solve the electrical angle.

The output gain and offset features allow for programmable movement ranges and output clamping levels that support a wide range of coil designs and target displacements.

Table of Contents

COMPENSATIONS AVAILABLE

The A17820 contains nonvolatile EEPROM that stores configuration settings. This memory is preprogrammed by Allegro with default values that minimize the need for programming by the system implementer. The required programming depends on the application and coil system used.

The A17820 compensations listed in this section are static compensations that remain fixed throughout the device lifetime, unless reprogrammed in EEPROM.

TX Driver Current: This adjusts the current in the transmitting coil to meet a defined reference voltage between the TXP and TXN pins. The reference voltage depends on target-to-coil air gap, system dimensions, and target material.

Front-End Amplifier Gain: This amplifies the received signals before analog-to-digital conversion to improve overall accuracy. The optimal code value depends on the application air gap.

Channel Direct-Coupling Compensation: This corrects the direct-coupling term of the channel offset. The direct-coupling compensation is used instead of channel offset and gain mismatch to reduce calibration time.

Channel Offset and Gain Trimming (OGT): This corrects for offset and amplitude mismatch between the RX signals. In general, OGT corrects for coil sensor imbalances in the printed circuit board (PCB). Gain-mismatch correction is applied on one RX signal to match the other RX amplitude.

Electrical Angle Harmonic Compensation (EHC): This corrects the periodic error that repeats each electrical cycle on the calculated angle.

Zero-Angle Compensation: This applies an offset to the calculated angle, so that any arbitrary target position can represent 0°.

Output Gain: This adjusts the application range to the IC full-scale output range and automatically clamps the output when not within the range for gain values > 1.

Thresholds for Magnitude Checks (Optional): The magnitude check verifies that the IC input signals are within the range defined in the datasheet. The thresholds are EEPROM programmable and depend on the front-end (FE) gain applied.

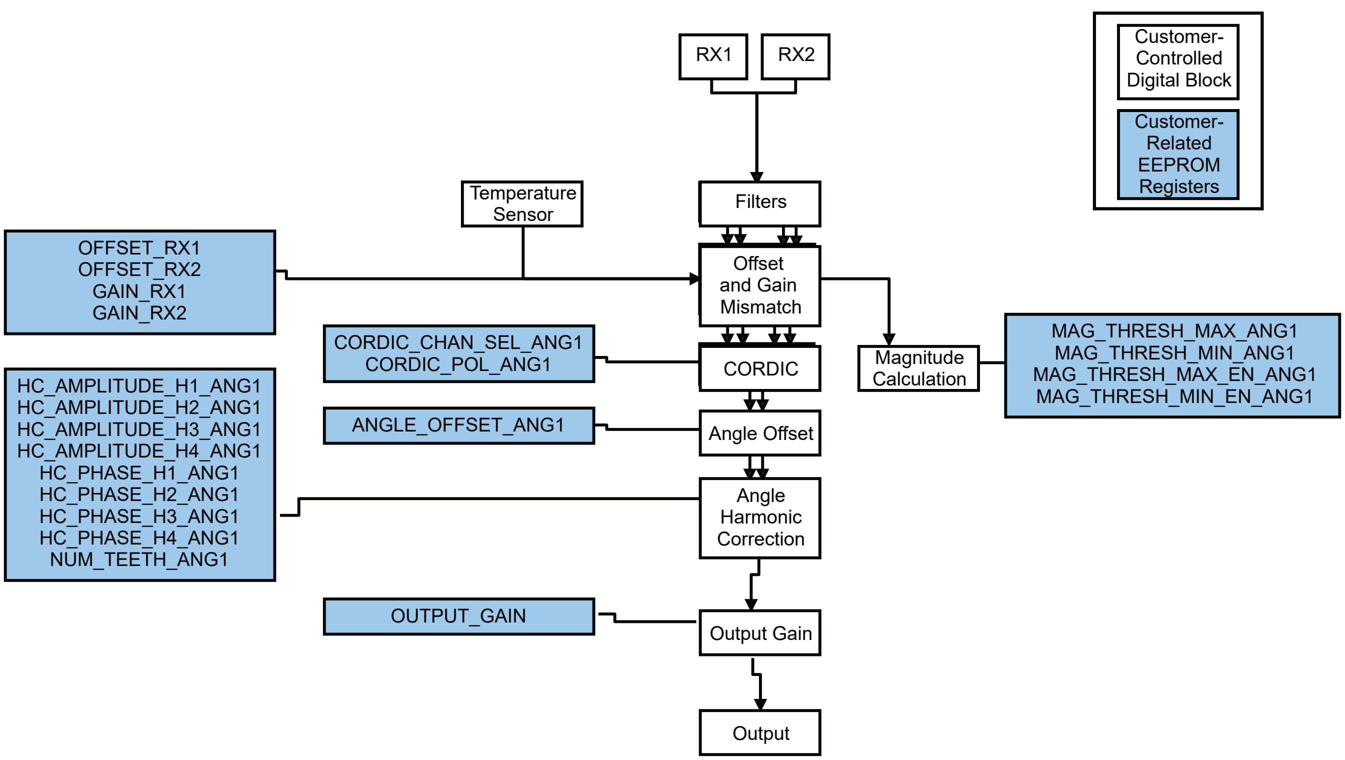


Figure 1: Compensations Flow Diagram

COMPENSATION USAGE STRATEGIES

System implementers have the choice of which compensations to use when using the A17820, depending on accuracy requirements and system tolerances.

NOTE: Errors vary greatly from design to design, so the data shown here is only a reference.

Option 1: Basic Compensation

Only TX driver current and front-end amplifier gain are used. This is the baseline performance. If only basic compensation is applied, large errors are expected.

Option 2: Fixed Calibration Across System

A one-time characterization study determines the programmable compensations to apply to all units. This is effective at eliminating systematic errors while avoiding and/or reducing a measurement-based end-of-line calibration. TX trim, FE amplifier gain trim, and direct coupling or channel offset and gain mismatch compensation could be calculated through characterization.

Option 3: Unique Calibration Per System

After end-of-line assembly, each system is measured and receives unique programmable compensations. This approach maximizes accuracy and combats errors introduced by mechanical misalignment.

Typically, electrical angle harmonics benefit the most from unique calibration, while the other compensations can often use a fixed calibration approach.

HOW TO CALIBRATE AND PROGRAM COMPENSATIONS

The following describes how to collect data and set the programmable compensations.

Presample Acquisition Procedure

The following procedure is required for any calibration method and describes TX driver and front-end gain amplifier trimming. This provides baseline performance and does not require sample acquisition.

1. Send the EEPROM access code C6189D63 to the ACCESS field in memory register 0x72.
2. Before performing a sample acquisition, trim the transmitting driver current to provide operating condition oscillations of approximately 4.5 V peak-to-peak (V_{pk-pk}). TX current is determined by EEPROM field TX_TRIM (0x1A, bits [6:0]). Adjust TX_TRIM to obtain 4.5 V_{pk-pk} between the TXP and TXN pins.

   TX current trim procedure:

   A. Set the shadow field TX_LOOP_EN = 0 (0x76, bit [24]). This disables the control loop on the TX current used for optimal performances during operation across temperature or dynamic shifts. In cases of multiple ICs connected to the same TX coil, this must be disabled on all ICs to correctly perform the TX trim.

   B. The voltage can be directly measured via the TXP and TXN pins or by use of an internal test mode. To use the test mode, set the volatile bit to SIGMON_SAR_TRIGGER_VTX = 1 in the volatile memory address (0x94, bit [24]), immediately followed by a read of the SIGMON_SAR_VALUE in the volatile memory address (0x93, bits [8:0]). The 9-bit digital value can be converted to V_{pk-pk} using:

   $$V_{PK-PK} = \left(\frac{sigmon\_sar\_value}{511} \times 3.3V - 1.65\right) \times \frac{16}{3}$$

   Then, close the test mode by sending SIGMON_SAR_TRIGGER_VTX = 0 to the volatile memory address (0x94, bit [24]).

C. If the measured TX V_{pk-pk} is less than 4.5 V_{pk-pk}, increase the TX_TRIM value in shadow (0x76, bits [6:0]); otherwise, lower its value. For cases where the TX coil is connected to two ICs, the TX_TRIM code must be updated only on one of the two ICs before the TX voltage is read; the other maintains TX_TRIM = 0.

D. If the TX V_{pk-pk} is measured via the test mode, step 2B must be repeated each time a new TX_TRIM value is programmed to update the test mode value SIGMON_SAR_VALUE.

E. Once the TX voltage has reached 4.5 V_{pk-pk}, program the final TX_TRIM value in EEPORM (0x1A, bits [6:0]) on all ICs working with the measured TX coil.

F. Ensure TX_LOOP_EN = 1 in EEPROM (0x1A, bit [24]). For cases where the TX coil is connected to two ICs, only the IC with programmed TX_TRIM has the TX current control loop enabled (TX_LOOP_EN = 1) in EEPROM while the other IC remains disabled (TX_LOOP_EN = 0).

3. Program the EEPROM fields specific to the application system:

A. CORDIC_POL_ANG1 (0x20, bit [0]) enables angle inversion: 0 Increases the angle in the clockwise direction; 1 increases the angle in the counter-clockwise direction.

B. OUTPUT_GAIN (0x2E, bits [16:1]). This adjusts the application output range to the IC full-scale range and automatically saturates the output when not within the range for gain values > 1; otherwise, the output rolls over after reaching the maximum of the range. In linear applications, this is particularly useful when the application range is smaller than the receiving coil range and still requires an angle output from zero to 360.

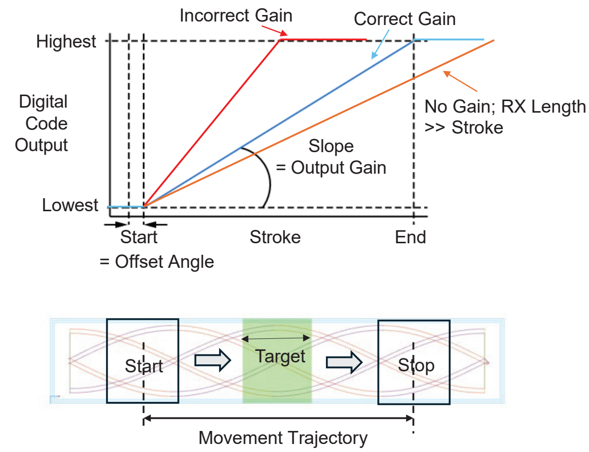


Figure 2: Example of Linear Application with Output Gain > 1

4. Read all compensation fields in the table that follows. To ensure end-of-line (EOL) calibration is performed correctly, if any compensation field does not equal 0, write it with the value 0.

Compensation	Field Name	EEPROM Address
Front-End Amplifier Gain	FE_AMP_GAIN_ANG1	0x1B, bits [5:3]
Channel Offset	OFFSET_RX1 OFFSET_RX2,	0x1C, bits [13:0] 0x1D, bits [13:0]
Channel Gain	GAIN_RX1 GAIN_RX2	0x1C, bits [25:14] 0x1D, bits [25:14]
Angle Harmonic Amplitude	HC_AMPLITUDE_H1_ANG1 HC_AMPLITUDE_H2_ANG1 HC_AMPLITUDE_H3_ANG1 HC_AMPLITUDE_H4_ANG1	0x24, bits [24:11] 0x25, bits [24:11] 0x26, bits [24:11] 0x27, bits [24:11]
Angle Harmonic Phase	HC_PHASE_H1_ANG1 HC_PHASE_H2_ANG1 HC_PHASE_H3_ANG1 HC_PHASE_H4_ANG1	0x24, bits [10:0] 0x25, bits [10:0] 0x26, bits [10:0] 0x27, bits [10:0]
Zero Angle Offset	ANGLE_OFFSET_ANG1	0x22, bits [16:0]

Front-End Amplifier Gain

To amplify the received signals before the analog-to-digital conversion, the front-end amplifier gain is trimmed as follows:

1. Read volatile complex-amplitude memory field MAGNITUDE_ANG1.
2. Compare the read value of MAGNITUDE_ANG1 with the table below, and select the appropriate gain code for the front-end amplifier gain, FE_AMP_GAIN_ANG1. If the input signals are less than a complex amplitude value of 1414.3 or higher than 29983.16, the input signals are too low or too large, respectively, for the IC to operate within the specified accuracy.

FE_AMP_GAIN_ANG Code	Acquired Value Code Range (LSB)
0	≤29983.16 and >20507.35
1	≤20507.35 and >13435.85
2	≤13435.85 and >10182.96
3	≤10182.96 and >8061.51
4	≤8061.51 and >6364.35
5	≤6364.35 and >5657.20
6	≤5657.20 and >4808.62
7	≤4808.62

3. Program the best gain code for the EEPROM field FE_AMP_GAIN_ANG1 (0x1B, bits [5:3]).

Output Gain

1. Perform a sweep along the full application range and collect the data through the IC SENT output.
2. Calculate the output gain as follows:

   output gain decimal = max[abs(ideal value in code)]/ max(collected value in code)

   The ideal value in code is related to the SENT data nibble number of bits for angle data. On the SENT output, the angle data is a signed representation from [(–2 × #bits)/2] to [(2 × #of bits/2) – 1]. The ideal value in codes is equal to [(2 × #of bits/2) – 1].
3. Convert to fixed-length binary representation:

   output gain = Allegro_fi2bin(output gain decimal, 0,16,10).
4. Write the calculated value in EEPROM field OUTPUT_ GAIN (0x2E, bits [16:1]).

Channel Direct-Coupling Compensation

Each channel offset can be found through volatile memory reads in EEPROM of TEMP_COMP_OUT_RX1 and TEMP_COMP_OUT_RX2 when a target is not present in the system. The read values correspond to the direct-coupling terms of the channel offset.

The direct-coupling compensation could replace the channel offset and gain mismatch compensation in application when end-of-line calibration must be reduced.

1. Read volatile fields TEMP_COMP_OUT_RX1 and TEMP_ COMP_OUT_RX2.
2. Convert to the fixed-length binary representation:

   OFFSET_RX1 = Allegro_fi2bin(TEMP_COMP_OUT_ RX1/2^{16},1,14,14)

   OFFSET_RX2 = Allegro_fi2bin(TEMP_COMP_OUT_ RX2/2^{16},1,14,14)
3. Write each calculated parameter in the corresponding EEPROM field:

Fields	Type (si)	WL	FL	EEPROM Address
OFFSET_RX1	1	14	14	0x1C, bits [13:0]
OFFSET_RX2	1	14	14	0x1D, bits [13:0]

Thresholds for Magnitude Checks (Optional)

To detect input-signal variation, the IC offers the option to program minimum and maximum thresholds based on signal amplitude. A representation of the signal amplitudes related to RX1/2 is given in the MAGNITUDE_ANG1 volatile memory field on 17 bits.

1. To use the detection option, the following fields must be set to 1:
   - MAG_THRESH_MAX_EN_ANG1 = (0x20, bit [3])
   - MAG_THRESH_MIN_EN_ANG1 = (0x20, bit [2])
2. Minimum and maximum thresholds can be calculated and converted to the fixed-length binary representation (see the Allegro_fi2bin function in Appendix B: Fixed-Point Binary Function):
   - MAG_THRESH_MIN_ANG1 = Allegro_fi2bin(MAG_ THRESH_MIN_ANG1,0,11,11)
   - MAG_THRESH_MAX_ANG1 = Allegro_fi2bin(MAG_ THRESH_MAX_ANG1,0,11,11)
3. Write each calculated parameter in the corresponding EEPROM field:

Fields	Type (si)	WL	FL	EEPROM Address
MAG_THRESH_MIN_ANG1	0	11	11	0x20, bits [14:4]
MAG_THRESH_MAX_ANG1	0	11	11	0x20, bits [25:15]

Sample Acquisition Procedure Using Controlled Angular Positions for A17820 Accuracy Optimization

- Mount the target in a system equipped with an accurate rotary/linear stage, depending on the application.
- Mount the sensor PCB precisely in front of the target at the nominal air gap of the application. Ensure that any nonideal tilt in the PCB and/or target is minimized.
- To accurately calculate compensations, there must be an accurate reference system with the capability to move the system target to unique, equally spaced angular positions over a complete mechanical revolution of 360°. If the intended EOL calibration uses electrical angle harmonic compensation, there must be a sufficient number of angular positions, n, to resolve the content of a fourth harmonic in each electrical period. It is recommended that n angular positions be equal to or greater than 16 times the number of teeth, NTeeth, of the target-coil system.
- To perform the calibration over continuous rotation, refer to the Simplified Calibration Using SENT Output section.
- Repower the device, and send the access codes as in the Presample Acquisition Procedure, step 1.

Channel Offset and Gain Trimming (OGT)

The channel-offset and gain-mismatch trim procedure can be avoided if the channel direct-coupling compensation method is used instead. Both channel-compensation methods cannot be used. The results of the OGT method are superior to the direct-coupling compensation method.

The A17820 compensates offset and gain nonidealities: in the balance of clockwise and counterclockwise windings; and between sine and cosine windings in the PCB coils. Compensation is performed directly at power-up with the offset and gain fields on each channel.

For linear applications, OGT must be performed across the full RX coil length, which might be larger than the measurement of interest. This is necessary to capture a minimum and a maximum peak of each RX channel.

The offset and gain fields are programmed as follows:

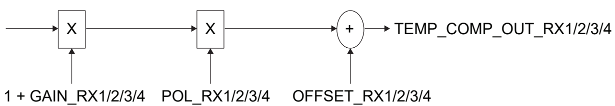
1. Sweep the position of the target across the equally spaced n angular positions. At each position:
   A. Wait 10 µs.
   B. Read each volatile memory field TEMP_COMP_OUT_RX1/2.
   C. Uncomplement the read values (see the function in Appendix A: Uncomplement Function, uncomplement2C(double(TEMP_COMP_OUT_Rx1/2),17).
   D. Store the value of each field in an array, RX1/2, at the index of the current mechanical position:
      - TEMP_COMP_OUT_RX1 = (0x82, bits [16:0])
      - TEMP_COMP_OUT_RX2 = (0x83, bits [16:0])

   NOTE: If two ICs are connected to the same TX, both ICs must be powered on during data collection.

2. The mechanical position of each sample can be stored in an array called MECHANICAL_POSITION.
3. Each RX polarity can be individually controlled using the POL_RX1/2 EEPROM fields.

Fields	EEPROM Address
POL_RX1	0x1B, bit [9]
POL_RX2	0x1B, bit [10]

4. Calculate the gain and offset parameters for all ICs that apply according to the following channel-compensation path:



   - GAIN_RX1 = (Amplitude(RX2) / Amplitude(RX1)) – 1
   - GAIN_RX2 = 0

   Offsets on normalized signals are calculated as:

   - RX1/2_NORM = RX1/2 × (GAIN_RX1/2 + 1)
   - OFFSET_RX1/2 = –mean(RX1/2_NORM)
5. Convert each offset and gain calculated value to the fixed-length binary representation (see the Allegro_fi2bin function in Appendix B: Fixed-Point Binary Function).
   - OFFSET_RX(i)_SF = Allegro_fi2bin(OFFSET_RX(i),1,14,14), i = 1 or 2
   - GAIN_RX(i)_SF = Allegro_fi2bin(GAIN_RX(i),1,12,13), i = 1 or 2
6. Write each calculated parameter in the corresponding EEPROM field:

Fields	Type (si)	WL	FL	EEPROM Address
OFFSET_RX1	1	14	14	0x1C, bits [13:0]
OFFSET_RX2	1	14	14	0x1D, bits [13:0]
GAIN_RX1	1	12	13	0x1C, bits [25:14]
GAIN_RX2	1	12	13	0x1D, bits [25:14]

Angle Offset

The objective of the angle offset is to enable the rescale of an angle according to the reference.

To calibrate the correct angle offset, sweep the target position across the equally spaced n angular positions. For linear applications, the target position must be collected across the full RX length, which might be larger than the distance of interest. This is necessary to compensate for electrical harmonics using the proposed fitting approach to obtain compensation coefficients. At each position:

1. Wait 10 μs , then:
   A. Read the volatile memory field CORDIC_ANG1.
   B. Convert to degrees as:

      CORDIC_ANG_DEG1 = modulo((cordic_ang1)/2^{17} × 360,360)

   C. Store the value of the field in an array.
2. Calculate the error in degrees as:

   CORDIC_ERROR1 = modulo(CORDIC_ANG_DEG1 – refAngle1 + 180,360) – 180

   where refAngle is the reference angle in electrical degrees).
3. Calculate the offset on the angle as:

   ANGLE_OFFSET_ANG_DEG1 = mean(modulo(CORDIC_ERROR1,360));
4. Convert the value to the fixed-length binary representation:
   - ANGLE_OFFSET_ANG1 = Allegro_fi2bin(ANGLE_OFFSET_ANG_DEG1 /360,0,17,17)
5. Write the obtained value into the respective EEPROM field:
   - ANGLE_OFFSET_ANG1 (0x22, bits [16:0])
6. Optionally, the output angle rotation direction with respect to the input signals can be changed by changing the value of the EEPROM field CORDIC_POL_ANG1 (0x20, bit [0]).

Electrical Angle Harmonic Compensation (EHC)

The A17820 can compensate for electrical harmonic distortion on the angle due to the nonidealities of coil design, coil fabrication, and field uniformity across the coil surface. Compensation for the first, second, third, and fourth electrical harmonics can be made on the electrical angle.

To calculate the harmonic compensation parameters, the previous data acquisition is used. The objective is to calculate the amplitude and phase of each harmonic to apply the following equation:

$$\begin{aligned} \text{HC\_ANG1} = \text{OFFS\_ANG1} - (&\text{HC\_AMPLITUDE\_H1\_ANG1} \times \sin(w \times (\text{OFFS\_ANG1} + \text{HC\_PHASE\_H1\_ANG1})) \\ &+ \text{HC\_AMPLITUDE\_H2\_ANG1} \times \sin(w \times (2 \times \text{OFFS\_ANG1} + \text{HC\_PHASE\_H2\_ANG1})) \\ &+ \text{HC\_AMPLITUDE\_H3\_ANG1} \times \sin(w \times (3 \times \text{OFFS\_ANG1} + \text{HC\_PHASE\_H3\_ANG1})) \\ &+ \text{HC\_AMPLITUDE\_H4\_ANG1} \times \sin(w \times (4 \times \text{OFFS\_ANG1} + \text{HC\_PHASE\_H4\_ANG1})) \\ &) \end{aligned}$$

where OFF_ANG1 is the offset-corrected angle output, and HC_ANG1 is the output angle.

1. Find parameters HC_AMPLITUDE_H_ANG and HC_PHASE_H_ANG to fit this function:

   CORDIC_ERROR1 = a0 + am1 × sind(1 × xm + pm1) + a1 × sind(1 × x + p1) + a2 × sind(2 × x + p2) + a3 × sind(3 × x + p3) + a4 × sind(4 × x + p4)

   where:

   - xm = reference angle in mechanical degrees
   - a1/2/3/4 = HC_AMPLITUDE_H1/2/3/4_ANG1
   - p1/2/3/4 = HC_PHASE_H1/2/3/4_ANG1 (±180°)
   - x = CORDIC_ANG_DEG1 – mean(modulo(CORDIC_ERROR1,360)) (= the offset-corrected angle output)
   - am1 and pm1 cannot be compensated internally because they correct for mechanical harmonics on error caused by target wobble, for example.

2. Convert the obtained values to the fixed-length binary representation and write them in EEPROM:

   - HC_AMPLITUDE_H1_ANG1 = Allegro_fi2bin(a1/45,1,14,14)
   - HC_PHASE_H1_ANG1 = Allegro_fi2bin(p1/360,1,11,11)

Fields	Type (si)	WL	FL	EEPROM Address
HC_AMPLITUDE_H1_ANG1	1	14	14	0x24, bits [24:11]
HC_AMPLITUDE_H2_ANG1	1	14	14	0x25, bits [24:11]
HC_AMPLITUDE_H3_ANG1	1	14	14	0x26, bits [24:11]
HC_AMPLITUDE_H4_ANG1	1	14	14	0x27, bits [24:11]
HC_PHASE_H1_ANG1	1	11	11	0x24, bits [10:0]
HC_PHASE_H2_ANG1	1	11	11	0x25, bits [10:0]
HC_PHASE_H3_ANG1	1	11	11	0x26, bits [10:0]
HC_PHASE_H4_ANG1	1	11	11	0x27, bits [10:0]

Figure 3: Angle Error After Harmonics Compensation

Simplified Calibration Using SENT Output

Advanced calibration requires use of the Manchester interface to read internal nodes before the angle calculation is performed at specific angle positions.

It is possible to perform a simplified calibration that does not use those nodes, by using SENT output data at specific angle positions.

Channel gain and channel offset cannot be directly compensated with this calibration, but the first and second harmonics compensation of the angle can resolve these specific errors depending on the gain mismatch and channel-offset amplitude.

1. Write EEPROM field OUTMSG_MODE = 6 to configure the SENT interface for the trigger SENT protocol, where data latch upon the falling edge of the trigger.
2. Write EEPROM field SENT_ DATA_CFG to 3. The angle become available with 16-bit resolution on the SENT frame.
3. Sweep the position of the target across the equally spaced n angular positions. Perform this sweep at each position, for each IC.
4. Using the trigger SENT message:
   A. Acquire angle data.
   B. Convert the data to degrees as:

      CORDIC_ANG_DEG1 = modulo((CORDIC_ANG1)/2^{16} × 360,360)

   C. Store the value of each field in an array.
5. Calculate the error in degrees as:

   CORDIC_ERROR1 = modulo(CORDIC_ANG_DEG1 – refAngle1 + 180,360) – 180

   where refAngle is the reference angle in electrical degrees.
6. Calculate the offset on angle as:

   ANGLE_OFFSET_ANG_DEG1 = mean(modulo(CORDIC_ERROR1,360)
7. Convert the value to the fixed-length binary representation:
   - ANGLE_OFFSET_ANG1 = Allegro_fi2bin(ANGLE_OFFSET_ANG_DEG1 /360,0,17,17)
8. Write the value from step 7 in EEPROM.
9. Using the previous acquisition, calculate the harmonic-compensation trims (amplitudes and phases).
10. Find parameters HC_AMPLITUDE_H_ANG and HC_PHASE_H_ANG to fit the function:

    CORDIC_ERROR1 = a0 + am1 × sind(1 × xm + pm1) + a1 × sind(1 × x + p1) + a2 × sind(2 × x + p2) + a3 × sind(3 × x + p3) + a4 × sind(4 × x + p4)

    where:
    - am1 and pm1 are useful if there are mechanical harmonics on the error due to target wobble (cannot be compensated).
    - xm = reference angle is in mechanical degrees
    - a1/2/3/4 = HC_AMPLITUDE_H1/2/3/4_ANG1
    - p1/2/3/4 = HC_PHASE_H1/2/3/4_ANG1 (±180°)
    - x = CORDIC_ANG_DEG1 – mean(modulo(CORDIC_ERROR1,360)) (= the offset-corrected angle output)
11. Convert the values to the fixed-length binary representation and write them in EEPROM:
    - HC_AMPLITUDE_H1_ANG1 = Allegro_fi2bin(a1/45,1,14,14)
    - HC_PHASE_H1_ANG1 = Allegro_fi2bin(p1/360,1,14,14)

APPENDIX A: UNCOMPLEMENT FUNCTION

MATLAB source code to uncomplement two's-complement signed field data. N_bits corresponds to the field number of bits.

```
function array_out = uncomplement2C(array_in, n_bits)
range = (2^n_bits);
threshold = range/2 - 1;
array_out = double(array_in);
for i = 1:length(array_in)
if array_out(i) > threshold
array_out(i) = array_out(i) - range;
end
end
end
```

APPENDIX B: FIXED-POINT BINARY FUNCTION

The MATLAB method below can be used to convert a decimal value into its fixed-point binary representation.

```
function [int_value] = Allegro_fi2bin(value,si,wl,fl)
% Convert a fixed point number to a binary value
% value - decimal value
% si - 0/1 0 = unsigned 1 = signed
% wl - The word length or number of bits
% fl - The fractional length (number of places to move the decimal)
if si
fi_max = (2^(wl-1)-1)/2^fl;
fi_min = -(2^(wl-1))/2^fl;
else
fi_max = (2^wl-1)/2^fl;
fi_min = 0;
end

if value > fi_max
value = NaN;
warning('OVERFLOW');
end

if value < fi_min
value = NaN;
warning('OVERFLOW');
end

if isnan(value)
int_value = value;
else
negative_num = 0;
if si && value < 0
value = 2^wl/2^fl+value;
negative_num = 1;
end
value = round(2^fl * value);
int_value = value;
end
end
```

Revision History

Number	Date	Description
–	July 3, 2025	Initial release
1	December 9, 2025	Modified Table of Contents, Compensations Available, Compensation Usage Strategies, Presample Acquisition Procedure, Front-End Amplifier Gain, Channel Offset and Gain Trimming (OGT), Angle Offset, and Electrical Angle Harmonic Compensation (EHC) sections (pages 1–8); and added Output Gain and Channel Direct-Coupling Compensation sections (page 5)

955 PERIMETER ROAD • MANCHESTER, NH 03103 • USA
+1-603-626-2300 • FAX: +1-603-641-5336 • ALLEGROMICRO.COM