Keywords: MAX1358, MAX1359, MAX1360, Medical, Temperature

APPLICATION NOTE 4296

Measuring Temperature with the MAX1358 Data Acquisition System

Sep 16, 2008

Abstract: This application note explains how to use the internal and external temperature sensors on the MAX1358 data acquisition system. The temperature circuits for the MAX1358/MAX1359/MAX1360 are identical, therefore any reference to the MAX1358 also applies to the MAX1359 and MAX1360. A step-by-step approach guides the user through the setup, measurement, and calculation of the temperature.

Introduction

The MAX1358 data acquisition system can measure temperature using an internal or external transistor PN junction. **Figure 1** shows the internal temperature circuit and the external circuit. A constant current is supplied by the current source to produce a voltage (V_{BE}) across the transistor. The current source can be programmed to produce up to four currents. For each current the voltage drop across the transistor is measured using the integrated ADC. The ADC has inputs for TEMP+, TEMP-, AIN1, AIN2, and AGND. These measured values are used in an equation to determine the junction temperature.

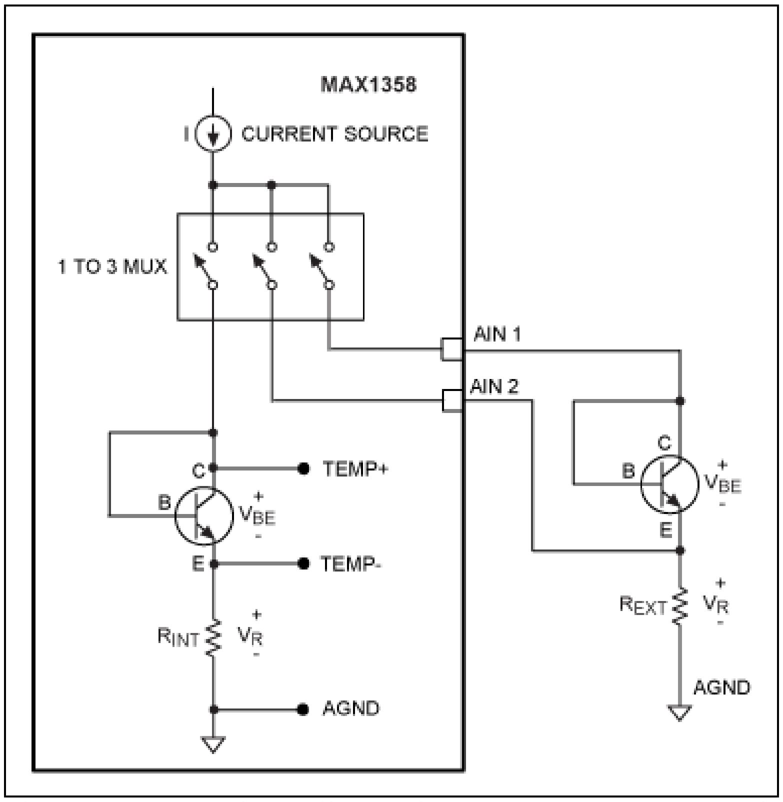

Figure 1. MAX1358 internal/external temperature measurement circuit.

Internal Four-Current Method

The internal four-current method requires eight measurements to be used in the temperature-measurement equation. Equation 1 below is used for four-current measurement.

$$T_{MEAS} = \left[\frac{q(NV_{BE3} - NV_{BE1}) - q(NV_{BE4} - NV_{BE2})}{nk\ln\left[\left(\frac{I_3}{I_1}\right)\left(\frac{I_2}{I_4}\right)\right]}\right]\left[\frac{V_{REF}}{2^{16}}\right]$$

Equation 1. Four-current temperature measurement equation.

Substituting for I_1, I_2, I_3, and I_4 in the denominator:

$$I_1 = \frac{NV_{R1}}{R_{INT}} \qquad I_2 = \frac{NV_{R2}}{R_{INT}} \qquad I_3 = \frac{NV_{R3}}{R_{INT}} \qquad I_4 = \frac{NV_{R4}}{R_{INT}}$$

Where:
T_{MEAS} = temperature in Kelvin
q = electron charge = 1.60219×10^{-19} coulombs
NV_{BE1} = ADC reading with I_1 as current source
NV_{BE2} = ADC reading with I_2 as current source
NV_{BE3} = ADC reading with I_3 as current source
NV_{BE4} = ADC reading with I_4 as current source
V_{REF} = ADC reference voltage = 1.251V (typ)
n = diode ideality = 1.000 (typ)

k = Boltzmann's constant = 1.3807 × 10^{-23} Joules/Kelvin
I_1 = Current source low setting (4µA)
I_2 = Current source high setting (60µA)
I_3 = Current source high setting (64µA)
I_4 = Current source high setting (120µA)
NV_{R1} = ADC reading with I_1 as current source
NV_{R2} = ADC reading with I_2 as current source
NV_{R3} = ADC reading with I_3 as current source
NV_{R4} = ADC reading with I_4 as current source
2^{16} = Number of ADC steps for MAX1358 16-bit ADC

To convert the measured temperature in Kelvin to degrees Celsius, the following formula is used:

°C = K - 273.15

Procedure Using the Internal Transistor

The procedure for measuring the voltages across the internal transistor and internal resistor applies a current from the current source, and measures the resulting V_{BE} and V_R.

Step 1. Enable the Reference and ADC

Enable the internal 1.251V reference and the reference buffer with a gain of 1.0 by setting REFV[1:0] bits to 0x01 in the REF_SDC register.

Enable the ADC by setting the ADCE bit in the ADC register. The internal reference and ADC are enabled. Note: The ADC is set with the default parameters of unipolar, normal polarity, single conversion, internal reference, unity gain, 10 samples per second, and normal conversion.

Step 2. Calibrate the ADC

Set the ADC conversion mode to Self Offset and Gain Calibration by setting the Mode[2:0] bits to 0x07 in the ADC register. Start an ADC conversion by setting the STRT bit in the ADC register. The ADC is now calibrated. The Mode[2:0] bits in the ADC register are automatically cleared. This returns the ADC to normal operation.

Step 3. Set the Current Source for the Internal Temperature Sensor

Set the current source for the internal temperature sensor by setting the IMUX[1:0] bits to 0x01 in the TEMP_CTRL register.

Step 4. Set the Current Source for I_1 (4µA)

Set the current source for I_1 by setting the IVAL[1:0] bits to 0x00 in the TEMP_CTRL register.

Step 5. Set the ADC Input for TEMP+ to TEMP-

Set the ADC positive input multiplexer for TEMP+ by setting MUXP[3:0] to 0x07 in the MUX register. Set the ADC negative multiplexer for TEMP- by setting MUXN[3:0] to 0x00 in the MUX register.

Step 6. Measure V_{BE1} Using the ADC

The V_{BE1} voltage is measured from the TEMP+ to the TEMP- inputs to the ADC. The ADC is already configured and only needs to convert to get the resulting V_{BE1} voltage. To start the ADC conversion, set the STRT bit in the ADC register. The ADC will do a conversion and the result will be in the DATA register. Read the DATA register value and save as a 16-bit integer named V_{BE1} for later calculation.

Step 7. Set the ADC Input for V_{R1}

Set the ADC positive input multiplexer for AGND by setting MUXP[3:0] to 0x09 in the MUX register. Set the ADC negative multiplexer for TEMP- by setting MUXN[3:0] to 0x00 in the MUX register (same as step 5). To measure the TEMP- input relative to AGND, the polarity flipper bit is used. Set the POL bit in the ADC register. The ADC is now setup with TEMP- as its positive input and AGND as its negative input.

Step 8. Measure V_{R1} Using the ADC

To start the ADC conversion, set the STRT bit in the ADC register. The ADC will do a conversion and the result will be in the DATA register. Read the DATA register value and save as a 16-bit integer named V_{R1} for later calculation.

Step 9. Set the Current Source for I_2 (60µA)

Set the current source for I_2 by setting the IVAL[1:0] bits to 0x01 in the TEMP_CTRL register.

Step 10. Set the ADC Input for TEMP+ to TEMP-

Set the polarity flipper back to normal by clearing the POL bit in the ADC register. Set the ADC positive input multiplexer for TEMP+ by setting MUXP[3:0] to 0x07 in the MUX register. Set the ADC negative multiplexer for TEMP- by setting MUXN[3:0] to 0x00 in the MUX register.

Step 11. Measure V_{BE2} Using the ADC

The V_{BE2} voltage is measured from the TEMP+ to the TEMP- inputs to the ADC. The ADC is already configured and only needs to convert to get the resulting V_{BE2} voltage. To start the ADC conversion, set the STRT bit in the ADC register. The ADC will do a conversion and the result will be in the DATA register. Read the DATA register value and save as a 16-bit integer named V_{BE2} for later calculation.

Step 12. Set the ADC Input for V_{R2}

Set the ADC positive input multiplexer for AGND by setting MUXP[3:0] to 0x09 in the MUX register. Set the ADC negative multiplexer for TEMP- by setting MUXN[3:0] to 0x00 in the MUX register (same as step 5). To measure the TEMP- input relative to AGND, the polarity flipper bit is used. Set the POL bit in the ADC register. The ADC is now setup with TEMP- as its positive input and AGND as its negative input.

Step 13. Measure V_{R2} Using the ADC

To start the ADC conversion, set the STRT bit in the ADC register. The ADC will do a conversion and the result will be in the DATA register. Read the DATA register value and save as a 16-bit integer named V_{R2} for later calculation.

Step 14. Set the Current Source for I_3 (64µA)

Set the current source for I_3 by setting the IVAL[1:0] bits to 0x10 in the TEMP_CTRL register.

Step 15. Set the ADC Input for TEMP+ to TEMP-

Set the polarity flipper back to normal by clearing the POL bit in the ADC register. Set the ADC positive input multiplexer for TEMP+ by setting MUXP[3:0] to 0x07 in the MUX register. Set the ADC negative multiplexer for TEMP- by setting MUXN[3:0] to 0x00 in the MUX register.

Step 16. Measure V_{BE3} Using the ADC

The V_{BE3} voltage is measured from the TEMP+ to the TEMP- inputs to the ADC. The ADC is already configured and only needs to convert to get the resulting V_{BE3} voltage. To start the ADC conversion, set the STRT bit in the ADC register. The ADC will do a conversion and the result will be in the DATA register. Read the DATA register value and save as a 16-bit integer named V_{BE3} for later calculation.

Step 17. Set the ADC Input for V_{R3}

Set the ADC positive input multiplexer for AGND by setting MUXP[3:0] to 0x09 in the MUX register. Set the ADC negative multiplexer for TEMP- by setting MUXN[3:0] to 0x00 in the MUX register (same as step 5). To measure the TEMP- input relative to AGND, the polarity flipper bit is used. Set the POL bit in the ADC register. The ADC is now setup with TEMP- as its positive input and AGND as its negative input.

Step 18. Measure V_{R3} Using the ADC

To start the ADC conversion, set the STRT bit in the ADC register. The ADC will do a conversion and the result will be in the DATA register. Read the DATA register value and save as a 16-bit integer named V_{R3} for later calculation.

Step 19. Set the Current Source for I_4 (120µA)

Set the current source for I_4 by setting the IVAL[1:0] bits to 0x11 in the TEMP_CTRL register.

Step 20. Set the ADC Input for TEMP+ to TEMP-

Set the polarity flipper back to normal by clearing the POL bit in the ADC register. Set the ADC positive input multiplexer for TEMP+ by setting MUXP[3:0] to 0x07 in the MUX register. Set the ADC negative multiplexer for TEMP- by setting MUXN[3:0] to 0x00 in the MUX register.

Step 21. Measure V_{BE4} Using the ADC

The V_{BE4} voltage is measured from the TEMP+ to the TEMP- inputs to the ADC. The ADC is already configured and only needs to convert to get the resulting V_{BE4} voltage. To start the ADC conversion, set the STRT bit in the ADC register. The ADC will do a conversion and the result will be in the DATA register. Read the DATA register value and save as a 16-bit integer named V_{BE4} for later calculation.

Step 22. Set the ADC Input for V_{R4}

Set the ADC positive input multiplexer for AGND by setting MUXP[3:0] to 0x09 in the MUX register. Set the ADC negative multiplexer for TEMP- by setting MUXN[3:0] to 0x00 in the MUX register (same as step 5). To measure the TEMP- input relative to AGND, the polarity flipper bit is used. Set the POL bit in the ADC register. The ADC is now setup with TEMP- as its positive input and AGND as its negative input.

Step 23. Measure V_{R4} Using the ADC

To start the ADC conversion, set the STRT bit in the ADC register. The ADC will do a conversion and the result will be in the DATA register. Read the DATA register value and save as a 16-bit integer named V_{R4} for later calculation.

Step 24. Calculate the Temperature

The temperature is calculated using the formula in Equation 1 above. The equation can be simplified by multiplying and dividing the constants in advance. The constant is applied to Equation 2 shown below.

$$T_{MEAS} = \left[\text{Const}\right]\left[\frac{(NV_{BE3} - NV_{BE1}) - (NV_{BE4} - NV_{BE2})}{\ln\left[\left(\frac{NV_{R3}}{NV_{R1}}\right)\left(\frac{NV_{R2}}{NV_{R4}}\right)\right]}\right]$$

$$\text{Where Const} = \left[\frac{qV_{REF}}{nk2^{16}}\right] = \frac{(1.60219 \times 10^{-19})(1.251 \text{ typical})}{(1.00)(1.3807 \times 10^{-23})(65{,}536)} = 0.221509354$$

Equation 2. Simplified four-current temperature measurement equation.

Four-Current Method Example Using Real Data

The temperature was measured on an evaluation (EV) kit at room temperature using the procedure above with the following results.

For $I_1 \cong 4\mu A$	NV_{BE1} = 0x81AF	NV_{R1} = 0x0350
For $I_2 \cong 60\mu A$	NV_{BE2} = 0x9048	NV_{R2} = 0x32D4
For $I_3 \cong 64\mu A$	NV_{BE3} = 0x90A5	NV_{R3} = 0x3629
For $I_4 \cong 120\mu A$	NV_{BE4} = 0x93E2	NV_{R4} = 0x615c

Using Equation 2 above and substituting the measured values:

$$T_{MEAS} = \left[\text{Const}\right]\left[\frac{(0x90A5 - 0x81AF) - (0x93E2 - 0x9048)}{\ln\left[\left(\frac{0x3629}{0x0350}\right)\left(\frac{0x32D4}{0x615C}\right)\right]}\right] = \left[\text{Const}\right]\frac{(37{,}029 - 33{,}199) - (37{,}858 - 36{,}936)}{\ln\left[\left(\frac{13{,}865}{848}\right)\left(\frac{13{,}009}{24{,}924}\right)\right]}$$

$$T_{MEAS} = \left[0.221509354\right]\left[\frac{3{,}830 - 922}{\ln\left[\left(\frac{180{,}369{,}785}{21{,}135{,}552}\right)\right]}\right] = \left[0.221509354\right]\frac{2{,}908}{\ln(8.53395194)}$$

Solving the equation for the temperature yields: T_{MEAS} = 300.19K
Converting the result from Kelvin to Celsius yields: °C = 300.19 - 273.15 = 27.04°C

This is the measured value at room temperature. A correction for gain and offset has not been applied. The gain and offset correction is detailed below.

External Four-Current Method

The external four-current method is the same as the internal four-current method, except that the internal current-source multiplexer must be changed to direct the current source out AIN1 or AIN2. The ADC input multiplexer must also be changed to use AIN1 and AIN2 as the ADC inputs.

The external components are connected as shown in Figure 1 above. The external transistor chosen for this application is a low-cost surface-mount 2N3904 from On Semiconductor^{SM}, part number MMBT2N3904LT1. Other transistors or diodes can also be selected. The resistor chosen is a low-cost surface-mount 4.02K 1% size 0805 1/8 watt from Panasonic®, part number ERJ-6ENF4021V. This value resistor was chosen to match the internal resistor, which is typically 4KΩ.

Procedure Using the External Transistor

The procedure for measuring the voltages across the external transistor and external resistor is similar to the internal four-current method, except that the current source must be selected to drive AIN1 or AIN2 and different inputs must be selected to read the external V_{BE} and V_R.

Step 1. Enable the Reference and ADC

Enable the internal 1.251V reference and the reference buffer with a gain of 1.0 by setting REFV[1:0] bits to 0x01 in the REF_SDC register.

Enable the ADC by setting the ADCE bit in the ADC register. The internal reference and ADC are enabled. Note: The ADC is set with the default parameters of unipolar, normal polarity, single conversion, internal reference, unity gain, 10 samples per second, and normal conversion.

Step 2. Calibrate the ADC

Set the ADC conversion mode to Self Offset and Gain Calibration by setting the Mode[2:0] bits to 0x07 in the ADC register. Start an ADC conversion by setting the STRT bit in the ADC register. The ADC is now calibrated. The Mode[2:0] bits in the ADC register are automatically cleared. This returns the ADC to normal operation.

Step 3. Set the Current Source for Internal AIN1

Set the current source for internal temperature sensor by setting the IMUX[1:0] bits to 0x10 in the TEMP_CTRL register.

Step 4. Set the Current Source for I_1 (4µA)

Set the current source for I_1 by setting the IVAL[1:0] bits to 0x00 in the TEMP_CTRL register.

Step 5. Set the ADC Input for AIN1 to AIN2

Set the ADC positive input multiplexer for AIN1 by setting MUXP[3:0] to 0x00 in the MUX register. Set the ADC negative multiplexer for AIN2 by setting MUXN[3:0] to 0x07 in the MUX register.

Step 6. Measure V_{BE1} Using the ADC

The V_{BE1} voltage is measured from the AIN1 to the AIN2 inputs to the ADC. The ADC is configured and only needs to convert to get the resulting V_{BE1} voltage. To start the ADC conversion, set the STRT bit in the ADC register. The ADC will do a conversion and the result will be in the DATA register. Read the DATA register value and save as a 16-bit integer named V_{BE1} for later calculation.

Step 7. Set the ADC Input for V_{R1}

Set the ADC positive input multiplexer for AGND by setting MUXP[3:0] to 0x09 in the MUX register. Set the ADC negative multiplexer for AIN2 by setting MUXN[3:0] to 0x07 in the MUX register. To measure the AIN2 input relative to AGND, the polarity flipper bit is used. Set the POL bit in the ADC register. The ADC is now setup with AIN2 as its positive input and AGND as its negative input.

Step 8. Measure V_{R1} Using the ADC

To start the ADC conversion, set the STRT bit in the ADC register. The ADC will do a conversion and the result will be in the DATA register. Read the DATA register value and save as a 16-bit integer named

V_{R1} for later calculation.

Step 9. Set the Current Source for I_2 (60µA)

Set the current source for I_2 by setting the IVAL[1:0] bits to 0x01 in the TEMP_CTRL register.

Step 10. Set the ADC Input for AIN1 to AIN2

Set the polarity flipper back to normal by clearing the POL bit in the ADC register. Set the ADC positive input multiplexer for AIN1 by setting MUXP[3:0] to 0x00 in the MUX register. Set the ADC negative multiplexer for AIN2 by setting MUXN[3:0] to 0x07 in the MUX register.

Step 11. Measure V_{BE2} Using the ADC

The V_{BE2} voltage is measured from the AIN1 to the AIN2 inputs to the ADC. The ADC is already configured and only needs to convert to get the resulting V_{BE2} voltage. To start the ADC conversion, set the STRT bit in the ADC register. The ADC will do a conversion and the result will be in the DATA register. Read the DATA register value and save as a 16-bit integer named V_{BE2} for later calculation.

Step 12. Set the ADC Input for V_{R2}

Set the ADC positive input multiplexer for AGND by setting MUXP[3:0] to 0x09 in the MUX register. Set the ADC negative multiplexer for AIN2 by setting MUXN[3:0] to 0x07 in the MUX register. To measure the AIN2 input relative to AGND, the polarity flipper bit is used. Set the POL bit in the ADC register. The ADC is now setup with AIN2 as its positive input and AGND as its negative input.

Step 13. Measure V_{R2} Using the ADC

To start the ADC conversion, set the STRT bit in the ADC register. The ADC will do a conversion and the result will be in the DATA register. Read the DATA register value and save as a 16-bit integer named V_{R2} for later calculation.

Step 14. Set the Current Source for I_3 (64µA)

Set the current source for I_3 by setting the IVAL[1:0] bits to 0x10 in the TEMP_CTRL register.

Step 15. Set the ADC Input for AIN1 to AIN2

Set the polarity flipper back to normal by clearing the POL bit in the ADC register. Set the ADC positive input multiplexer for AIN1 by setting MUXP[3:0] to 0x00 in the MUX register. Set the ADC negative multiplexer for AIN2 by setting MUXN[3:0] to 0x07 in the MUX register.

Step 16. Measure V_{BE3} Using the ADC

The V_{BE3} voltage is measured from the AIN1 to the AIN2 inputs to the ADC. The ADC is already configured and only needs to convert to get the resulting V_{BE3} voltage. To start the ADC conversion, set the STRT bit in the ADC register. The ADC will do a conversion and the result will be in the DATA register. Read the DATA register value and save as a 16-bit integer named V_{BE3} for later calculation.

Step 17. Set the ADC Input for V_{R3}

Set the ADC positive input multiplexer for AGND by setting MUXP[3:0] to 0x09 in the MUX register. Set the ADC negative multiplexer for AIN2 by setting MUXN[3:0] to 0x07 in the MUX register. To measure the AIN2 input relative to AGND, the polarity flipper bit is used. Set the POL bit in the ADC register. The ADC is now setup with AIN2 as its positive input and AGND as its negative input.

Step 18. Measure V_{R3} Using the ADC

To start the ADC conversion, set the STRT bit in the ADC register. The ADC will do a conversion and the result will be in the DATA register. Read the DATA register value and save as a 16-bit integer named V_{R3} for later calculation.

Step 19. Set the Current Source for I_4 (120µA)

Set the current source for I_4 by setting the IVAL[1:0] bits to 0x11 in the TEMP_CTRL register.

Step 20. Set the ADC Input for TEMP+ to TEMP-

Set the polarity flipper back to normal by clearing the POL bit in the ADC register. Set the ADC positive input multiplexer for AIN1 by setting MUXP[3:0] to 0x00 in the MUX register. Set the ADC negative multiplexer for AIN2 by setting MUXN[3:0] to 0x07 in the MUX register.

Step 21. Measure V_{BE4} Using the ADC

The V_{BE4} voltage is measured from the AIN1 to the AIN2 inputs to the ADC. The ADC is already configured and only needs to convert to get the resulting V_{BE4} voltage. To start the ADC conversion, set the STRT bit in the ADC register. The ADC will do a conversion and the result will be in the DATA register. Read the DATA register value and save as a 16-bit integer named V_{BE4} for later calculation.

Step 22. Set the ADC Input for V_{R4}

Set the ADC positive input multiplexer for AGND by setting MUXP[3:0] to 0x09 in the MUX register. Set the ADC negative multiplexer for AIN2 by setting MUXN[3:0] to 0x07 in the MUX register. To measure the AIN2 input relative to AGND, the polarity flipper bit is used. Set the POL bit in the ADC register. The ADC is now setup with AIN2 as its positive input and AGND as its negative input.

Step 23. Measure V_{R4} Using the ADC

To start the ADC conversion, set the STRT bit in the ADC register. The ADC will do a conversion and the result will be in the DATA register. Read the DATA register value and save as a 16-bit integer named V_{R4} for later calculation.

Step 24. Calculate the Temperature

The temperature calculation is identical to the four-current internal method. Use that method with the eight saved measurements to calculate the external temperature.

Gain and Offset Correction

The TEMP_CAL register is provided to correct for gain and offset errors in the temperature-measurement circuit. To correct for the gain and offset errors, the following formula is used.

$$T_{ACTUAL} = g(T_{MEAS}) + T_{OFFSET}$$

Equation 3. Gain and offset correction equation.

Procedure for Gain and Offset Correction

The procedure for adding the gain and offset correction is detailed below. The procedure reads the values from the gain and offset correction registers. The gain and offset amounts are then computed using two formulas. The resulting correction values are applied to Equation 3 above.

Step 1. Measure and Calculate the Temperature

Use the procedure for the internal or the external transistor and save the result as T_{MEAS} for use in the correction formula above.

Step 2. Read the TGAIN Register

Read the TEMP_CAL register and save the upper byte TGAIN[7:0] for use in the formula below. It should be saved as a type-signed integer.

Step 3. Calculate the Gain Correction Factor

Using the value saved above, apply this value to the gain formula below.

$$\text{Gain} = 0.9025 + \text{TGAIN} \times 0.000576$$

Equation 4. Gain formula equation.

Step 4. Read the TOFFS Register

Read the TEMP_CAL register and save the lower byte TOFFS[5:0] for use in the formula below. The data should be saved as a type-signed integer. Do not right-shift the data.

Step 5. Calculate the Offset Correction Factor

Using the offset value saved above, apply this value to the gain formula below.

$$\text{Offset} = -14.0 + \text{TOFFS} \times 0.3125$$

Equation 5. Offset correction equation.

Step 6. Calculate the Corrected Temperature

Use the gain and offset values calculated above in the temperature equation below.

$$T_{ACTUAL} = \text{GAIN} \times T_{MEAS} + \text{OFFSET}$$

Conclusion

The MAX1358 features internal circuitry to measure the internal die temperature. The internal circuitry can also be used with an external transistor for a low-cost remote temperature sensor. The four-current temperature measurement method described in this application note can be used to achieve the ±0.5 and ±1.0 typical accuracy specified in the MAX1358 data sheet.

Related Parts	
MAX1360	16-Bit Data-Acquisition Systems with ADC, DACs, UPIOs, RTC, Voltage Monitors, and Temp Sensor

More Information
