Users Manual
MLX90308CCC & MLX90314AB
and
Demo Software

The MLX90308CCC and MLX90314AB sensor interfaces provide a versatile programmable interface to a wide assortment of sensors and transducers. The sensor interface provides gain and offset control, linearization and temperature correction for sensor. The device contains on chip EEPROM for storage of the coefficients for making the signal corrections and adjustments. This document describes the use of the Melexis software for programming and calibrating the sensor interface. The MLX90308CCC and the MLX90314AB are identical circuits except that the MLX90314AB has four times the gain as the MLX90308CCC. The package and pin out are the same as well as the programming.

Within this document the terms ‘sensor interface’, ‘controller’, ‘chip’, and ‘ASIC’ are used interchangeably and all refer to the MLX90308CCC or the MLX90314AB.

1) Main Dialog Box

Upon starting the software the dialog box below, figure 1, will appear.

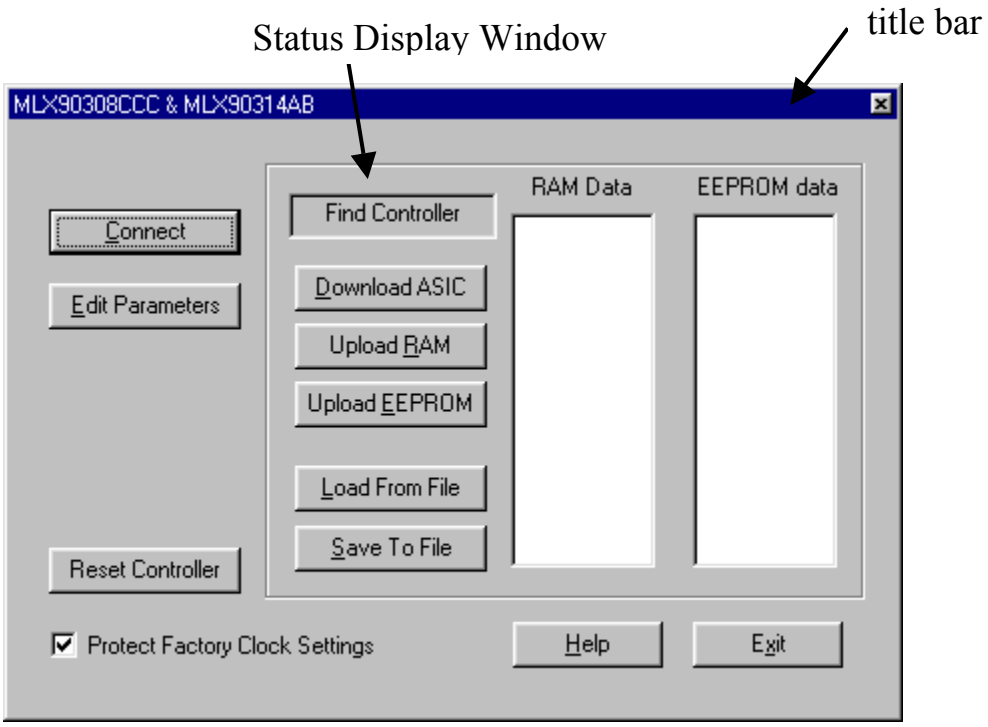

Figure 1, Main dialog box.

Software Title

The applicable part number for which the software is intended to work with is located in the title bar of this dialog box. The software and this document is intended to work with the MLX90308CCC or MLX90314AB sensor interface.

Software revision

To determine the version of software which is currently running, a right mouse click on the title bar will cause a pop up menu to be displayed. From this menu, select 'About test…'. This will cause the following pop up menu to be displayed, see figure 2.

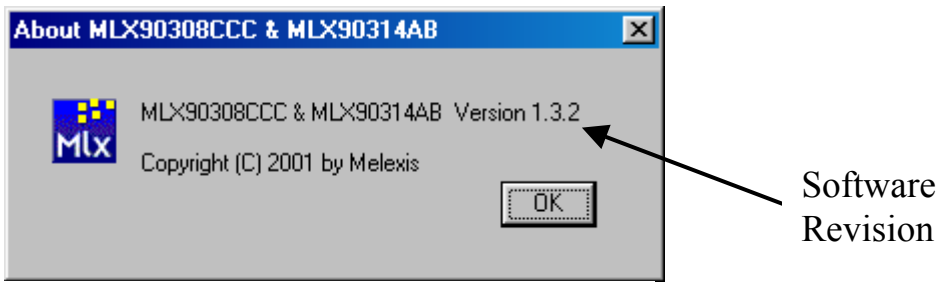

Figure 2, Software revision dialog box.

Connect

The 'Connect' button will bring up the 'Find Controller' dialog box. This is needed to establish communications between the PC and the sensor interface. A description of this dialog box is in section 2.

Edit Parameters

The 'Edit Parameters' button is used to bring up the 'Edit Parameters' dialog box. This dialog box is used for changing the operating parameters of the sensor interface. This dialog box is described in section 3.

Reset Controller

The 'Reset Controller' button is used to send a reset command to the sensor interface. This is a soft reset commanded by a read to EEPROM address 63. Address 63 of EEPROM is not a valid address so no data is returned. Upon a successful reset the sensor interface responds with a BCh. The software in turn notifies the user in the status display window of the successful completion of the reset command.

When uploading new parameters to EEPROM, the sensor interface must be reset before the changes take effect.

Protect Factory Clock Settings

Six bytes in the EEPROM are used to control the internal clock. These bytes are set by Melexis during final test of the chip. The settings are to ensure that the clock rate is such the non-turbo baud rate is as close to 2400 as possible. The software protects these bytes, 1, 32-36, and 39 from inadvertent modifications. This check box, default is checked, is used to give the user the ability change the factory set locations if needed. The datasheet describes the function of each of the factory set bytes.

Download ASIC

This button downloads all of the data from the sensor interface to the PC. This data is from both RAM and EEPROM.

The data must be downloaded from the sensor interface before it can be edited using the edit parameters dialog box

If the data from the sensor interface does not appear to be correct, click on this button several times and observe the data that is downloaded. If this data changes or is extremely slow to download then the PC is not communicating at a close enough baud rate to the baud rate of the chip.

Upload Ram

This button is used to upload only the RAM data to the sensor interface. Any changes to the data from what is already in the RAM will take effect immediately. The ram data is overwritten with data from EEPROM when the chip is reset.

Some of the locations used in RAM are for dynamic variables. This means that as soon as the user loads values into these locations the chip's firmware overwrites those locations with the current operating data.

Upload EEPROM

This button is used to upload only the EEPROM data to the sensor interface. The entire block of data, 48 bytes, is uploaded. The software automatically generates the correct checksum and places it in byte 47 when the block is uploaded. This occurs whether or not the checksum test has been selected. Changes to the EEPROM data do not take effect until the sensor interface has been reset.

Load From File

This button is used to obtain previously saved data from a file. A standard dialog box is used for locating the desired file to be loaded. Configuration data can be saved from one chip and stored in another. The protection of the factory programmed clock settings will prevent the loaded file from overwriting these values as they are unique for each chip. It is important to note that the settings from one chip will not necessarily yield the same results on another chip. This is due to the resistors implemented in silicon having about a 20% tolerance.

Save to File

This button is used to save the RAM and EEPROM data to a file. The data can be used as a starting point for calibrating another sensor-sensor interface. A standard dialog box is called to allow the user to name the file and location. The data is saved in a simple readable ASCII text file format with the addresses identified. Saved files can be edited and comments added.

Byte Editor

The Byte editor, see figure 3, allows the user to modify individual bytes in RAM or EEPROM without specifically labeling the data contained in the byte. The byte editor can access all of the bytes in RAM and EEPROM. The byte editor is accessed by double clicking on the desired byte within the data windows on the main dialog box. Once a new data value is entered, clicking OK sends the data to the sensor interface. The data is shown and set in a hex format.

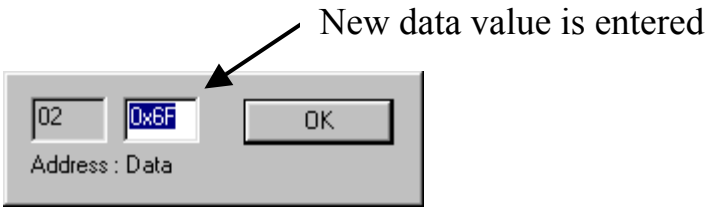

Figure 3, Byte editor.

2) Find Controller Dialog Box

The Find Controller dialog box, see figure 4, is used to establish communications between the PC and the sensor interface. Each sensor interface is set during final test to have an internal clock frequency that will result in a baud rate of 2400. Also, if the turbo bit is set, the baud rate will be around 9600.

Find Controller

The Find Controller button initiates the search for the baud rate for the PC to communicate with the sensor interface.

Serial Port

The serial port pull down menu allows the user to select the serial port the sensor interface is connected to. The options are Com1, 2, 3, and 4. The software assumes standard port addresses and IRQs are defined in Windows®. The default is Com 1.

Baud Rate

The baud rate pull down menu allows the user to select a specific baud rate or a range of baud rates for the software to search. The default is the 2400 baud rate selection.

Search Status

The search status window informs the user of the status of the search.

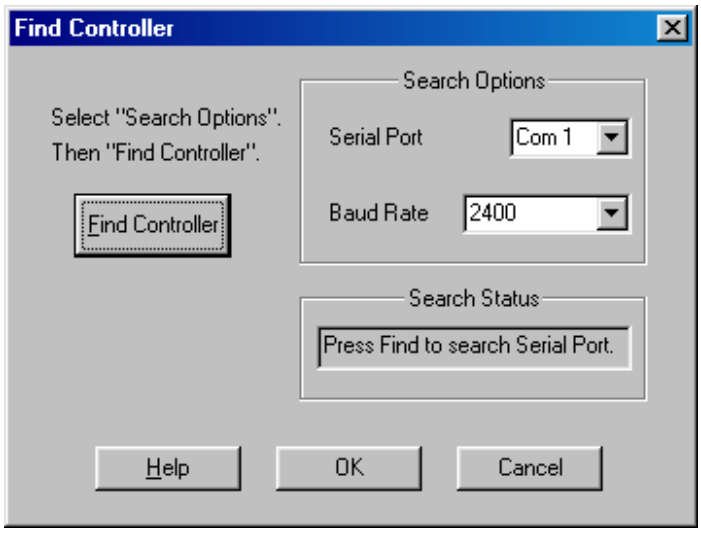

Figure 4, Find controller dialog box

3) Edit Parameters Dialog Box

The Edit Parameters dialog box is the most used dialog box of the software, see figure 5. This is where the individual operating parameters are identified from the block of EEPROM data. All user settable parameters are accessed using this dialog box. Also, important data from RAM is displayed within this box. The items in this dialog box are arranged by functions in areas and groups.

3.1) Mode

The Mode area of the dialog box contains all of the mode options contained in the mode byte, byte 0 in EEPROM.

Voltage Mode

This box enables Voltage Mode output of the sensor interface. This is used for both absolute and ratiometric configurations. Voltage mode can be used simultaneously with Current Mode. Either Voltage Mode and/or Current Mode should always be enabled.

Current Mode

This box enables Current Mode output of the sensor interface. Voltage mode can be used simultaneously with Voltage Mode. Either Voltage Mode and/or Current Mode should always be enabled.

Turbo

This box enables the Turbo mode of the sensor interface. This causes the internal clock rate to be multiplied by four. The baud rate is also multiplied by four as a result. The faster clock rate will provide wider bandwidth.

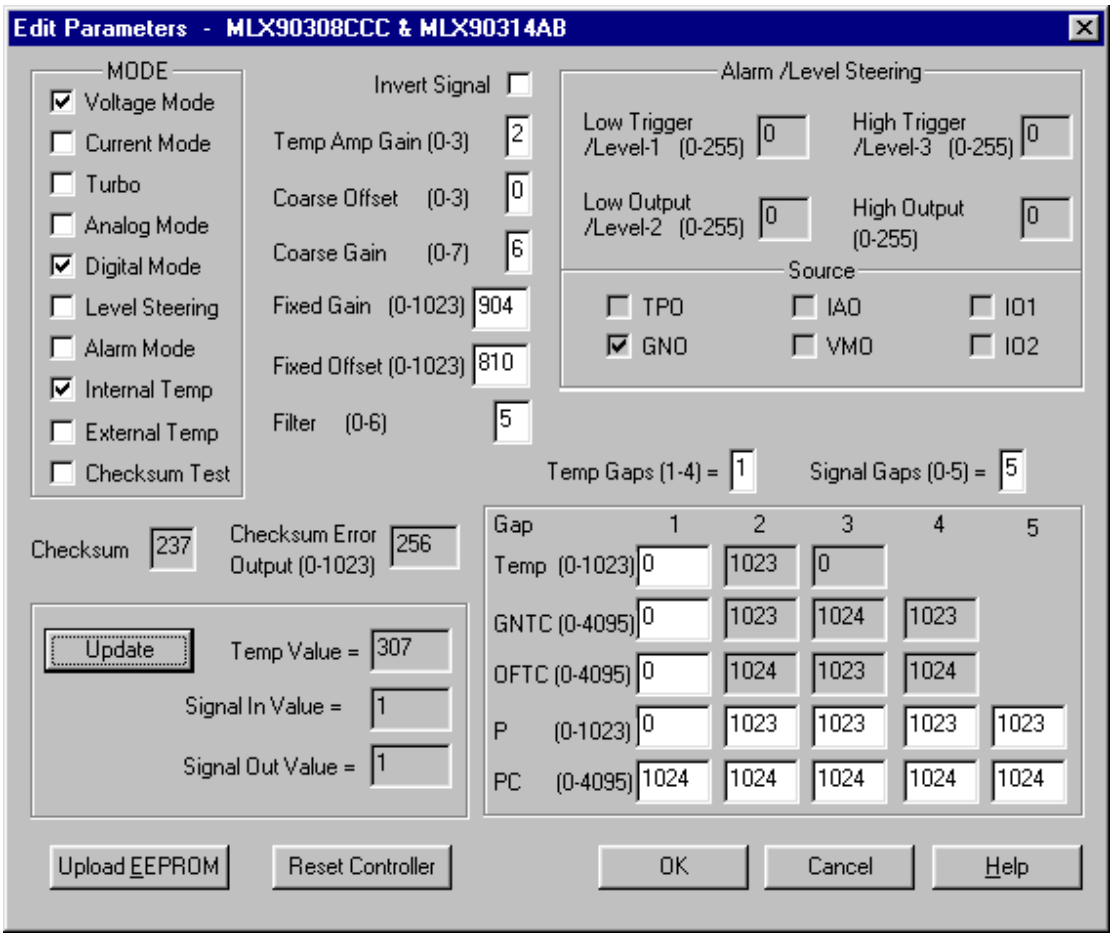

Figure 5, Edit parameters dialog box.

Analog Mode

This box enables the chip to operate with a completely analog signal path from sensor input to signal output. This provides the best accuracy however signal linearization is not available in Analog Mode. Analog Mode is mutually exclusive with Digital Mode.

Digital Mode

Digital Mode breaks the signal path by converting the signal to digital data for manipulation by the internal microcontroller. The signal is then converted back to analog before it is sent to the output stage of the chip. Both the A to D and the D to A are 10 bits wide. Digital Mode allows the processor to manipulate the data thus providing signal linearization. Digital Mode has a smaller bandwidth than Analog Mode and some small quantization noise and error is typical. Digital Mode is mutually exclusive with Analog Mode.

Level Steering

The Level Steering Mode uses the IO1 and IO2 pins as digital outputs to provide a 2 bit A to D to an internal analog signal. The analog signal is chosen by the user for the Alarm/Level Steering Source area of the dialog box. The trigger levels are also defined by the user and are contained in the same part of the dialog box. This function is mutually exclusive with Alarm Mode.

Alarm Mode

The Alarm Mode is used to force the output signal to a defined value when an input analog signal exceeds a user defined range. The input analog signal is chosen from the Alarm/Level Steering Source area of the dialog box. The defined range is between the upper trigger and the lower trigger. If the chosen signal exceeds the upper trigger then the output is forced to the value of the high output. If the chosen signal goes below the low trigger then the output is forced to the level of the low output. This function is mutually exclusive with Level Steering Mode.

Internal Temp

This box selects the internal temperature sensor for temperature compensating the sensor. The internal temperature sensor is mutually exclusive with the external temperature sensor.

External Temp

This box selects the external temperature sensor for temperature compensating the sensor. The external temperature sensor is mutually exclusive with the internal temperature sensor.

Checksum Test

This box enables the Checksum Test to ensure the integrity of the contents of the EEPROM. When enabled the checksum is calculated during a reset, either power-up or a soft reset. The sum of all of the EEPROM locations should be 0FFh for the test to pass. Location 47 in EEPROM is reserved as the checksum location. This location is set by the software to a value that forces the sum to 0FFh. The software will calculate and use this value whether or not the test is enabled. The software updates the checksum whenever the Upload EEPROM button is used. If the test detects a failure, the output is forced to a value defined by the user, 'Checksum Error Output'. The chip will regain normal function when the correct checksum is calculated.

3.2) General Signal Control

The center of the dialog box contains data and control that is needed to control the sensor's signal for both analog and digital modes as well as for voltage and current outputs. The diagram below, figure 6, illustrates the signal path within the sensor interface.

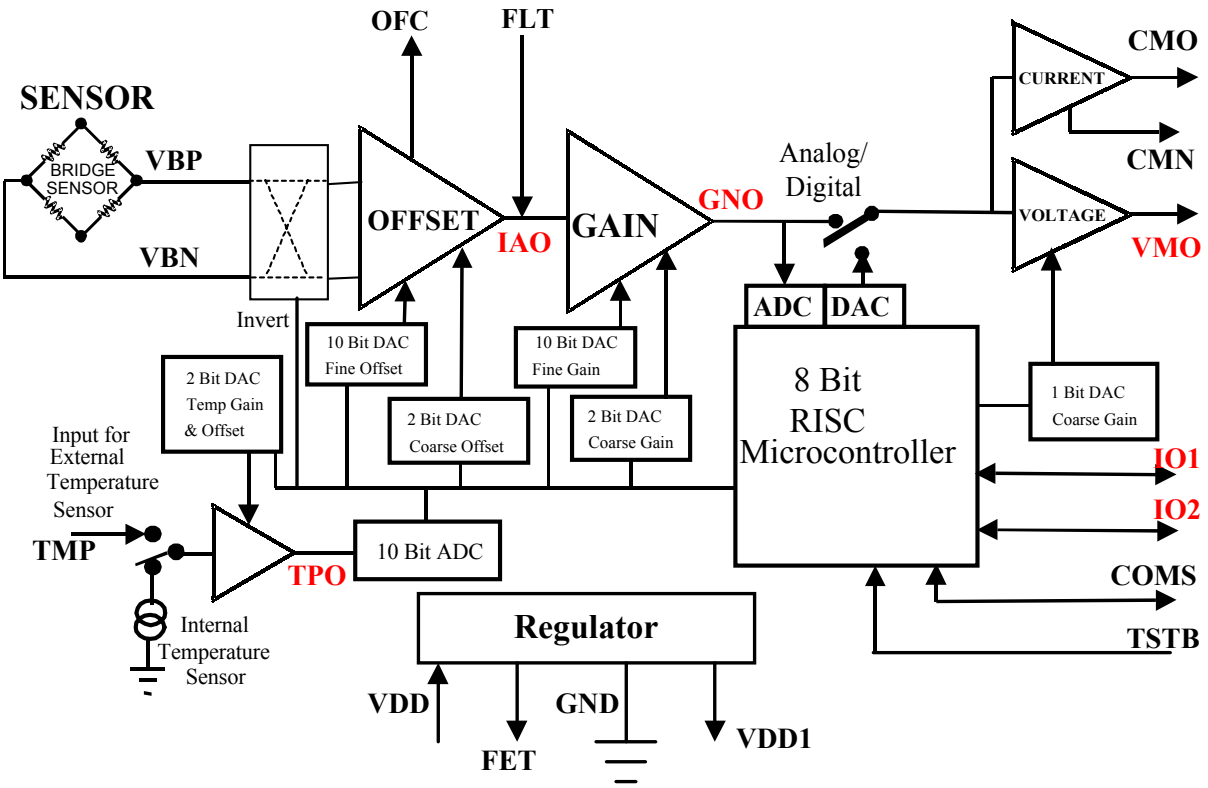

Figure 6, MLX90308CCC/MLX90314AB Block diagram.

Invert Signal

The invert signal control swaps the VBN and VBP signals at the front end of the signal path. This has the same effect as rewiring the connections from the sensor.

Temp Amp Gain

The temperature amplifier gain control (GNTP) is a 2 bit control on the gain of the amplifier for the analog temperature signal. This gain adjustment applies to both the internal or external temperature sensor.

Coarse Offset

The coarse offset (CSOF) is a 2 bit offset control on the front end amplifier. This provides a very coarse control on the offset of the signal.

Coarse Gain

Coarse gain (CSGN) controls two parts of the signal path. The most significant bit controls the output amplifier used for voltage output. The two least significant bits control the gain on the input stage.

Fixed Gain

Fixed Gain (GN) is a 10 bit value controlling the fine gain of the gain amplifier stage. This allows the user to precisely calibrate the output span.

Fixed Offset

Fixed Offset (OF) is a 10 bit value controlling the fine offset adjustment to the offset amplifier.

Filter

The filter value is used to filter the temperature value. This is a 4 bit number (0 to 6) where 0 is not filtering and 6 heaviest filtering. The heavier filtering means that it will take longer for a change in temperature to impact the filtered temperature value used in the control of the sensor signal. The 4 MSBs of the byte 25 in EEPROM are kept cleared by the software.

Checksum

The Checksum data box is for display only. The software calculates the checksum whenever an upload to EEPROM is performed. This happens whether the Checksum test is enabled or not. This field is read only.

Checksum Error

This data field defines what the output will do if the Checksum test fails. This value is sent to the D to A directed at the output, either Voltage or Current, irregardless of whether the part is in analog or digital mode. This is a 10 bit value, 0-1023.

3.3) Alarm/Level Steering

The Alarm and Level Steering part of the dialog box is used to enter the operating parameters for either function. The data fields are shared because the memory locations where the data is stored are shared. Alarm Mode and Level Steering Mode are mutually exclusive.

Alarm Mode

Alarm Mode provides the capability range limit the output, see figure 7. The trigger levels and the alarm levels are programmed by the user. These levels are programmed as 8 bit values which correlate to 8 MSBs of the digitized source signal or the 8MSBs of the output D/A. The source signal is also selected by the user. Alarm Mode is mutually exclusive with Level Steering.

Low Trigger

This value is the source signal threshold below which the low output value will be sent to output. This is programmed as an 8 bit number which is compared to the 8 MSBs of the of the digitized source signal.

Low Output

This value determines the output when the digitized source signal goes below the low trigger value. This value is programmed as an 8 bit number correlating to the 8 MSBs of the output (10 bit value).

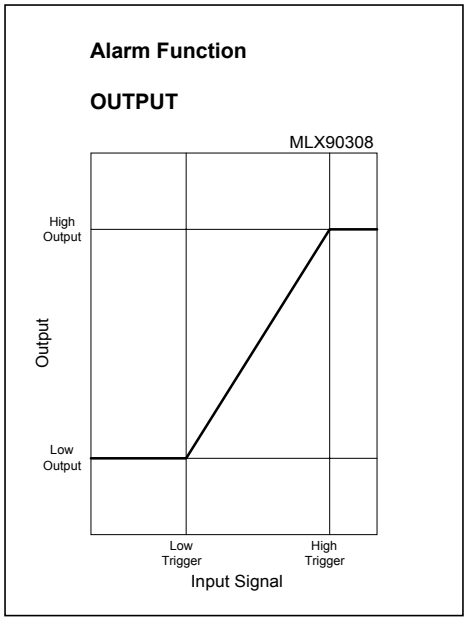

Figure 7, Alarm Function

High Trigger

This value is the source signal threshold above which the high output value will be sent to output. This is programmed as an 8 bit number which is compared to the 8 MSBs of the of the digitized source signal.

High Output

This value determines the output when the digitized source signal goes above the high trigger value. This value is programmed as an 8 bit number correlating to the 8 MSBs of the output (10 bit value).

Source

The source signal for Alarm Mode can be chosen from 6 points which the microcontroller has access to. These are four internal points; input amplifier out (IAO), gain amplifier out (GNO), temperature amplifier out (TPO), and voltage mode output (VMO) or two external points IO1 and IO2.

Level Steering

The Level Steering function provides two digital outputs acting as a two bit analog to digital converter, see figure 8. The converted analog signal is selected by the user as well as the points at which the two bits toggle. Level Steering is mutually exclusive with Alarm Mode. The transition points, levels 1,2,& 3, are defined as 8 bit values. These correspond to the 8 MSBs of the signal level displayed in the lower left corner of the edit parameters dialog box (also stored in bytes 60 and 61 of RAM).

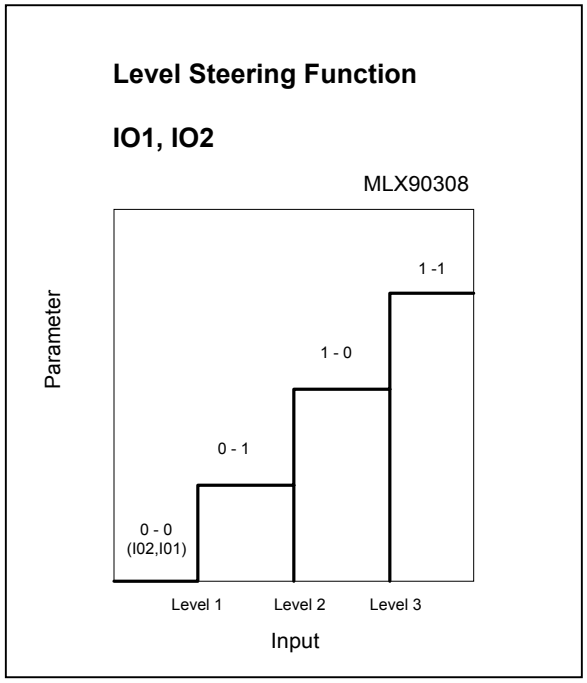

Figure 8, Level steering.

Level 1

Level 1 is the user defined point where IO1, IO2 transition from 0, 0 to 0, 1 with an increasing source signal. The opposite is true for a decreasing signal, there is no hysteresis.

Level 2

Level 2 is the user defined point where IO1, IO2 transition from 0, 1 to 1, 0 with an increasing source signal.

Level 3

Level 3 is the user defined point where IO1, IO2 transition from 1, 0 to 1, 1 with an increasing source signal.

Source

The source signal for Alarm Mode can be chosen from 4 points which the microcontroller has access to. These are four internal points input amplifier out (IAO), gain amplifier out (GNO), temperature amplifier out (TPO), and voltage mode output (VMO).

3.4) Temperature & Signal Data

The temperature and signal data is displayed for use by the user in determining the temperature and linearization coefficients.

Update

The Update button is used to request the Temperature (Temp) and Signal data be updated. The software reads the appropriate RAM locations and displays the data when this button is used. Whenever the temp or signal data is needed, the displayed data has to be updated.

Temp Value

The temperature value is displayed from RAM locations 58 and 59. The data is 10 bits and is right justified in the 16 bit field. The software shifts the data left before being displayed by the software (divided by 64). The data is only updated when the update button is pressed.

Signal In Value

The digitized value of the sensor's signal is displayed from RAM locations 46 and 47. While in Analog Mode, this value is the digitized GNO (gain amplifier output) and is proportional to the output. In Digital Mode, this data is the 'signal in value' (RAM locations 46 and 47) or the 'signal out value' (RAM locations 60 and 61). These are, respectively, the signal values before and after the linearization calculation. Setting the PC values to +1.0 (1024 decimal) means that the 'signal in value' = 'signal out value'. (Both ADC and DAC are 10 bits). The data is only updated when the update button is pressed.

3.5) Temperature Compensation

The MLX90308's temperature compensation algorithm is piece-wise with up to 4 temperature segments, see figure 9. Within each temperature segment ('gap'), the correction is a first order calculation. There are separate temperature coefficients for gain and offset.

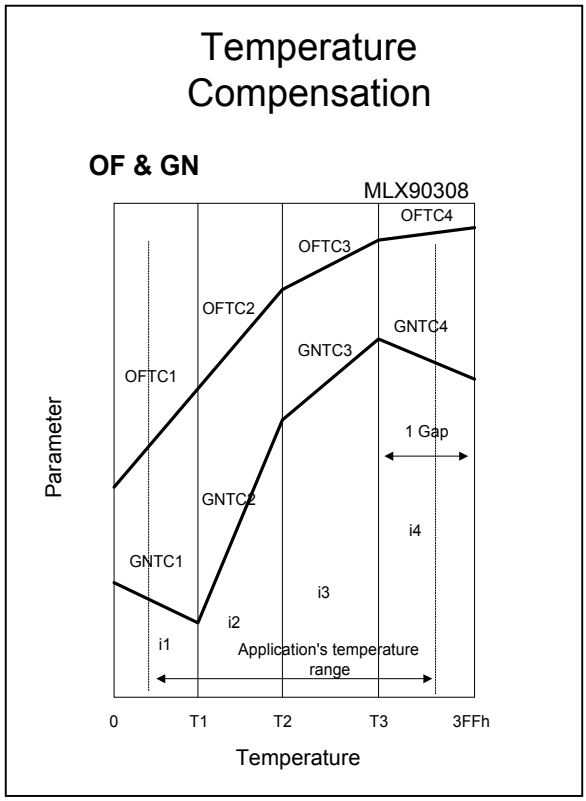

Figure 9, Temperature Compensation

Temperature Gaps

There are four temperature gaps possible with the MLX90308. The gaps are defined with respect to the filtered digitized temperature value, displayed in the 'Temp Value' box (saved in RAM locations 58 and 59). The first temperature gap is always bound at the low end by a digitized temperature value of zero. The fourth temperature gap is bound at the high end by a digitized temperature value of 1023_{10} ($3FF_{hex}$).

Whenever fewer than four gaps are used, the last gap should have an upper bound of 1023_{10}. This is to ensure that if the temperature exceeds the last point it won't enter into an undefined temperature gap.

The first temperature gap is slightly different than the other three. The compensation is based on the temperature difference between the current temp and the T1 point (the upper end of the gap). The other gaps use the lower end of the gap to measure the temperature difference. For instance in the second gap, the temperature difference is current temperature minus T1. This difference has to be accounted for in the compensation procedure.

Temperature Points

The temperature points T1, T2 and T3 are defined by the user to differentiate between the four possible temperature compensation ranges. The low endpoint T0 is defined as the minimum digitized filtered temperature value, zero. This means that the coefficients for the first gap, OFTC1 and GNTC1, will apply to the signal until the digitized filtered temperature reaches zero. The high endpoint T4 is defined as the maximum digitized filtered temperature value, 1023_{10}. This means that the coefficients for the last gap, OFTC4 and GNTC4 will apply to the signal from the T3 point and up until the digitized filtered temperature reaches its maximum, 1023_{10}.

When defining these points, the number that is used is simply copied from the 'Temp Value' box at the desired temperature (contents of ram locations 58 and 59).

GNTC

The GNTC values are the gain temperature compensation values for each of the four temperature gaps, GNCT1, GNTC2, GNTC3, and GNTC4. These values are used to adjust the signal up or down based on the relative temperature within a particular gap. Mathematically this is represented by the following equation:

$$DAC\_GAIN = CSGN * \{GN[9:0] + [GNTCi * \Delta Ti]\}$$

Where:

$\Delta Ti = (T1 - temp\_f)$ for gap 1

$\Delta Ti = (temp\_f - T_{i-1})$ for gaps 2,3,and 4

Temp_f = Filtered temperature
DAC_GAIN = The digital value that adjusts the gain on the gain amplifier.
Ti = Temperature segment point i = 2, 3, or 4
GNTCi = Gain TC for a given temperature segment i,
CSGN = Course Gain.
GN[9:0] = Fixed Gain (doesn't change with temperature)

The GNTC and OFTC values are 12 bit fixed point signed numbers, see figure 10. That means that the most significant bit is a sign bit. The next bit is one or zero (whole number). The remaining 10 bits are to the right of the decimal point. This gives the variable the range of –1.9990234 to +1.9990234. The table below shows how the fixed point numbers are represented as the binary number progresses from 0 to 4095.

Figure 10, 12 Bit fixed point numbers

Decimal Value	Hexadecimal Equivalent	Real Number Equivalent
0	0000h	+0.00
1023	3FFh	+0.9990234
1024	400h	+1.000
2047	7FFh	+1.9990234
2048	800h	-0.000
3071	0BFFh	-0.9990234
3072	0C00h	-1.000
4095	0FFFh	-1.9990234

OFTC

The OFTC values are the offset temperature compensation values for each of the four temperature gaps, OFCT1, OFTC2, OFTC3, and OFTC4. These values are used to adjust the signal's offset up or down based on the relative temperature within a particular gap. The offset adjustment is additive from one gap to the next. That is the offset adjustment for the fourth gap is added to the total offset adjustment of the third gap which is also added to the total offset adjustment of the second gap. Mathematically this is represented by the following equation:

For the first temperature gap:
DAC_OFFSET = OF[9:0] + OFTCi * (T1 – Temp_f).

For the second temperature gap:

DAC_OFFSET =OF[9:0] + OFTC2 * (Temp_f – T2)

For the third temperature gap:

DAC_OFFSET = OF[9:0] + [OFTC2 * (T2-T1)] + OFTC3 * (Temp_f – T3)

For the fourth temperature gap:

DAC_OFFSET =OF[9:0] + [OFTC2 * (T2-T1)] + [OFTC3 * (T3 – T2)] +
OFTC4 * (Temp_f – T4)

OF[9:0] = Fixed offset

3.6) Signal Linearization

The MLX90308's signal linearization algorithm is piece-wise with up to 5 signal segments, see figure 11. Within each signal segment ('gap'), the correction is first order. There are separate correction coefficients for each gap.

There are up to five signal gaps possible with the sensor interface. The gaps are defined with respect to the signal's digitized value, displayed in the 'Signal Value in' box and 'Signal value out'-box.

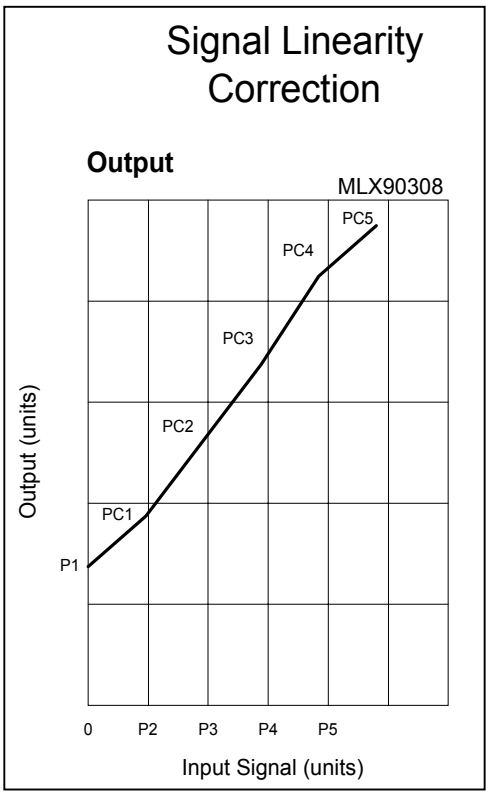

Figure 11, Signal linearization

The signal linearization is an adjustment to the output level based on the input signal level. Adjustment coefficients can be set for five different signal ranges. The adjustment is a first order correction based on the relative signal level within the particular gap plus the total correction of the previous gaps. The output is obtained by the following formula:

Output = (Signal – Pi) * Pci + Pi where
Signal = input signal measurement;
Pi = programmed signal point (i=2,3,4,5).
Pci = programmed coefficient.

The PCi coefficients are 12 bit fixed point signed numbers, the same format as the OFTC and GNTC variables. The Pi values are coded on 10 bits (0-3FFh). These values are copied from the Signal Value box.

4) Programming

4.1) Baseline Calibration

The baseline calibration involves setting the operating modes and temperature gain along with the coarse and fixed gain and offset.

Setting the operating modes Voltage and/or Current mode, turbo mode, Analog or Digital mode, Level steering or alarm mode (digital mode only), internal or external temperature sensor, and Checksum test. The checksum test, if used, should be enabled after all other settings are complete.

The invert signal bit swaps the differential inputs to the signal path. This has the same effect as swapping the connections to the VBP and VBN pins of the chip.

Temperature amplifier gain, GNTP, should be set such that the analog converter doesn't under-run or over-run at the temperature extremes. When using the internal temperature sensor over the full temperature span of the device (-40C to +140C) typically a GNTP setting of 2 will work. When using an external temperature sensor the voltage on the TMP pin must stay within the ranges as described in the datasheet. The more of the voltage range used the greater the temperature compensation adjustment range will be.

The coarse gain and fixed gain should be set before the coarse offset and fixed offset. Gain and offset are inter-related, the offset is multiplied by the gain. It is much easier to program the gain first then offset. It may be necessary to make some minor adjustment to the coarse offset settings before adjusting the gain. This is only needed if the output clips at either high end or low end. It is difficult to precisely calculate the offset and gain values. The amplifier circuitry within the chip uses resistors implemented in silicon. These resistors have around a 20% tolerance, thus the gain and offset will vary from chip to chip. Each chip is tested to provide a gain and offset adjustment capability within a specified range. For calculations a typical value can be taken.

4.2) Temperature calibration

The temperature compensation capability of the 90308/90314 is piece-wise with first order compensation in each segment (gap). The compensation is based on the difference between the current digitized filtered temperature and the appropriate temperature point. The equations have been previously described. The first temperature gap is slightly different than the other three. The first gap uses the temperature difference between the current temperature and the T1 temperature point (the upper end of the temperature gap). The second temperature gap also uses the T1 point for determining the temperature differences (the low end of the second gap). The third and fourth gaps also use the temperature point at the low end of their gap. This means that

programming the temperature compensation for the first gap is slightly different than the other gaps. The compensation coefficients for the first gap (OFTC1 and GNTC1) apply to digitized filtered temperature values from T1 down to zero. The fourth gap coefficients apply to digitized filtered temperature range from T3 to 1023 (decimal).

The temperature points T1 thru T3 should always be in increasing order from T1 to T2 to T3. If the temperature sensor has increasing signal with increasing temperature then the compensation procedure is intuitively easy. This is the case with the internal temperature sensor. If the temperature sensor has decreasing signal with increasing temperature then the compensation will start at a hotter temperature and go towards cold. The procedure below is written with regard to the filtered digitized temperature not the real physical temperature.

4.3) Procedure

1) Set the desired operating mode and number of gaps. It is also helpful to set T2 and T3 to 1023.

2) At the T1 temperature do a baseline calibration, the span then the offset by using CSGN[2:0] and GN[9:0] then CSOF[1:0] and OF[9:0]. The T1 temperature is somewhat arbitrary, typically its room temperature. If the temperature sensor goes down with increasing temperature then the T1 value will be towards the high end of the application's temperature range. Update the Temp/Signal values. Copy the digitized temperature reading to the T1 box. If one temperature gap is desired then T1 will be the highest digitized temperature value.

3) Lower the temperature to the lowest operating temperature and recalibrate the output using GNTC1 and OFTC1 only. If the temperature sensor goes down with increasing temperature then raise the temperature to its maximum value and re-calibrate the output using GNTC1 and OFTC1. Do not change the T1, OF, or GN values.

4) Raise the temperature until the second temperature point, T2. If the temperature sensor goes down with increasing temperature then lower the temperature to the T2 value. This point may be arbitrary and can be based on how far the output has deviated from it's desired value. Re-calibrate the sensor by adjusting GNTC2 and OFTC2 only.

5) Update the Temp/Signal values displayed on the screen. Copy the digitized temperature value to the T2 box. This must happen AFTER step 4.

6) Repeat steps 4 and 5 for T3 and setting GNTC3, OFTC3. T1, T2, and T3 must be in ascending order.

7) For the last temperature gap, raise the temperature to the highest operating temperature point and re-calibrate the output using GNTC4 and OFTC4. There is no T4, it is assumed to be 1023 (the maximum value of the digitized filtered temperature).

For less than four temperature gaps the last temperature point should be set to 1023. This means that the last gap extends out to the end of the temperature range.

4.4) Signal linearization

The signal linearization is only available when the device is in digital mode. This means that the entire calibration procedure should be done in digital mode. A calibration in analog mode may not yield a valid calibration in digital mode.
It is the best to first do the baseline calibration then linearize the sensor signal and then then temperature compensation. The baseline offset calibration must be perform in that way that the digitized signal has the value 0 or 1 LSB (be careful with clamping at 0 value).
The total number of linearization gaps plus the temperature gaps cannot be more than five. This is because the temperature coefficients and the linearization coefficients share the same memory locations.

1) Set all of the PC values to 1024. This corresponds to a real value of +1.000. Also, set P2 thru P5 to a value of 1023. (P1 = 0)

2) With no signal (zero pressure) find the correct value of P1 to have the correct voltage on the output.

3) Raise the signal to the signal point (pressure point) P2. This is an arbitrary value (can be based on where the output varies from the desired output by a set amount). Adjust the output to the desired level by adjusting PC1.

4) Set P2 by copying the pressure value by copying the digitized pressure value to the P2 location. This must happen after step 3.

5) Repeat steps 3 and 4 for P3, P4, and P5 setting PC2, PC3, and PC4.

6) Raise the pressure to its maximum and set PC5

If fewer than 5 pressure gaps are used the last P value (highest) should be set to 1023.