ISL29501 Temperature and Ambient Light Error Correction

The ISL29501 has the ability to perform real time correction of distance measurements due to changing temperature and ambient light conditions. The chip can be programmed to correct for a second order error polynomial. This document explains how to collect data, generate coefficients, and program the chip to do these error corrections. Renesas has had good results calculating the coefficients once per board design so it should not be necessary to collect data for each individual board. Calibration of each board design is also not required, but in practice almost every application will need temperature calculation at a minimum. The data used in this document was generated with an ISL29501-ST-EV1Z Sand Tiger reference board, but these techniques can be applied to any alternate optical components.

Contents

1. Advanced Calibration Process . . . 2

1.1 Advanced Calibration Registers . . . 2

2. Data File Requirements . . . 2

3. Collecting Data . . . 3

3.1 Standard Calibration . . . 3

3.2 Temperature Setup Procedure . . . 3

3.3 Temperature Sweep Procedure . . . 4

3.4 Ambient Light Setup Overview . . . 4

3.4.1 Light Source and Setup . . . 5

3.4.2 Ambient Light Sweep Procedure . . . 5

3.4.3 Choosing an Ambient Light Source . . . 5

3.4.4 Setting the Sweep Range . . . 5

3.4.5 Laboratory Setup . . . 6

4. Generating Coefficients . . . 7

4.1 Using Coeff_Gen . . . 7

4.2 GUI Setup Conditions . . . 7

4.2.1 Temperature . . . 7

4.2.2 Temperature Reference Adjustment . . . 7

4.2.3 Ambient Light . . . 8

4.2.4 Alternate Light Sources . . . 8

4.2.5 Slope Correction . . . 8

4.3 Ideal Coefficient Calculations . . . 8

4.4 Ideal Coefficients Defined . . . 8

4.5 Force Exponent Adjustment . . . 8

4.6 Finalizing Coefficients . . . 9

4.7 Measurements with Advanced Calibration Coefficients . . . 9

5. Revision History . . . 10

1. Advanced Calibration Process

These are the required steps in the advanced calibration process.

1. Phase data is collected while a single independent variable is varied. Supported variable types are die temperature and ambient light. The ISL29501 has separate 8-bit sensors for each of these. Phase data is collected for each measurement along with the sensor data for temperature or ambient light.
2. Renesas provides a software tool that reads and parses this data. A least squares curve fit to a second order polynomial is generated from the data. The resultant curve has the form Ax2 + Bx + C. From this curve the A and B coefficients are calculated and converted into the correct hexadecimal format.
3. The tool returns either three or four register values (depending on whether testing temperature or ambient light) that need to be programmed into the ISL29501 each time it is power cycled. They can be left at the default (0x00), which performs no correction.
4. The final constant C comes from the standard distance calibration (for details on standard calibration, see the *ISL29501 Datasheet*). Distance calibration sets the reference point for the system. The reference point is the point where all errors including temperature and ambient light are assumed to be zero. When performing distance calibration a reference temperature needs to be recorded. It is later programmed back into the chip with the error correction coefficients. The user reads the temperature sensor (reg 0xE2) and needs to write this code value into the temperature reference input (reg 0x31). This temperature reference is also used by the tool when generating the coefficients. For high integration time settings (reg 0x10 > 0x6), the temperature changes while running the distance calibration routine. The accurate way to compensate for this difference is to average the temperature value in the distance calibration and use this value as your temperature reference. For many circuit board designs it is sufficiently close to read only a single temperature at the end of distance calibration.

1.1 Advanced Calibration Registers

By following the advanced calibration process the user calculates the error coefficients for temperature and ambient light and then programs the following registers:

Table 1. Advanced Calibration Registers

Type	Reg Name	Reg #	Description
Temp	Temp Reference	0x31	Temperature Reference Code
	Phase Temp B	0x34	First Order Temperature Mantissa
	Phase Temp A	0x39	Second Order Temperature Mantissa
Ambient Light	Phase Ambient B	0x36	First Order Ambient Light Mantissa
	Phase Ambient A	0x3B	Second Order Ambient Light Mantissa
Shared	Phase Exponent	0x33	Coefficient Exponent

2. Data File Requirements

The data file input to the coefficient generator must be in a comma-delimited CSV format. It is designed to be the same as the “Record” format produced by the evaluation GUI. The first row of the file is a header that contains the register number for the data in that column. Each register field in the header needs to be a comma-delimited 2-digit hexadecimal with no leading or trailing spaces. Upper and lower case characters are accepted.

Following the header are rows of register data. They must follow the same comma delimited 2-digit hexadecimal format with no leading or trailing spaces.

Comments are allowed providing that they follow a blank line that is after the last row of data. At a minimum the file must contain four columns (four registers) of data: 0xE2, 0xE3, 0xD8, 0xD9. Additional columns of register or other data is allowed if they have a header entry and follow the stated rules. A data file example is shown in Figure 1.

```
N,D5,D6,D7,D8,D9,E2,E3
1,44,34,dd,14,e7,56,1
2,44,33,dd,16,e7,57,1
3,44,36,dd,15,e7,57,1
4,44,34,dd,20,e7,57,1
5,44,35,dd,14,e7,58,1
6,44,33,dd,18e7,58,1
7,44,34,dd,16,e7,58,1
.
```

Figure 1. Data File Example

3. Collecting Data

Temperature and ambient light require two independent sweeps, which cannot be done at the same time. It is critical that only a single variable is changed in each sweep. When collecting temperature data, the ambient light must not change. When collecting ambient light data, the temperature must remain constant. Failure to do this results in bad coefficients and poor error correction.

3.1 Standard Calibration

The user needs to load standard calibration values before starting data collection (standard calibration is described in the *ISL29501 Datasheet*). Renesas recommends running all 3 standard calibrations with the same emitter current and integration time that are used for the data collection sweep. It is best to do this before performing the sweep. The emitter settings are different from the applications in almost all cases. For this reason, crosstalk and distance calibration should be executed using the temperature setup just before starting the sweep. The temperature code (when collecting temperature data) in 0xE2 needs to be recorded for reference later. Distance calibration is unnecessary; however, if conducted, the user can directly observe distance changes due to temperature and ambient light and before and after adding advanced calibration coefficients.

3.2 Temperature Setup Procedure

The board undergoing testing needs to be positioned to measure distance to a target. To obtain temperature measurements, testing must be conducted inside an oven. The target size and configurations are limited by the dimensions of the oven, but these are not critical. This process measures the phase shift in the emitter, photodiode and ISL29501 chip as a system, which reflects how an application reacts to changes in ambient temperature. The emitter current should be set so that the return signal is 1–10nA (for emitter current programming, see the*ISL29501 Datasheet*). A heat gun should not be used because it is unlikely to produce a representative temperature gradient. This method has been tried and the resulting coefficients were poor. The target should be sized and the board aligned so that the beam fully strikes the target. Figure 2 shows an example of an oven set up for collecting data.

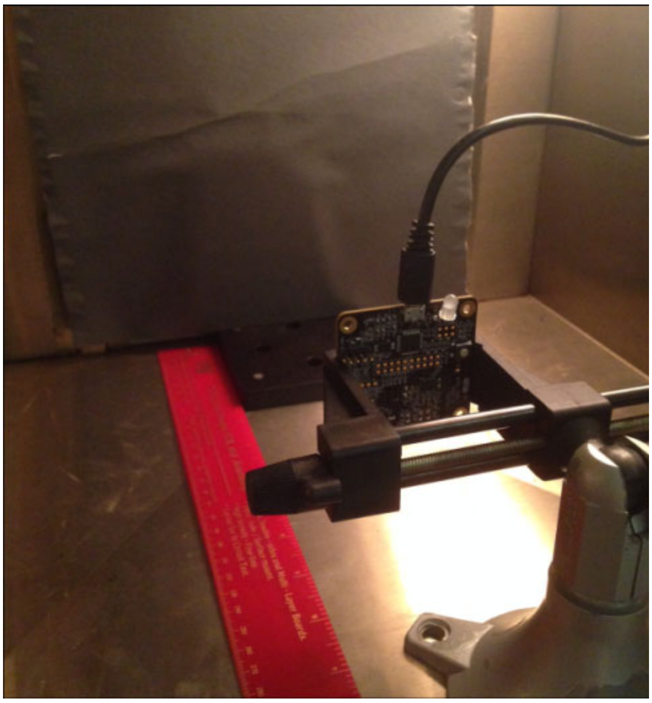


Figure 2. Example Oven Setup

Neither the target reflectivity or the distance from the target is critical. The distance should be recorded so that coefficients can be validated under the same conditions.

3.3 Temperature Sweep Procedure

The temperature should be varied over the entire intended operating range of the application recording the temperature code and phase register data. The exact temperature value is not important because the chip corrects measurements based on the ADC code. The sensor LSB is ~1.25°C.

The best technique is to mimic the end application conditions by ramping the temperature in 5°C steps over the intended operating range. Settling time should be 4 to 5 minutes, which allows the optical components and the board to reach full thermal equilibrium. A minimum of 100 data measurements should be made at each temperature so that noise can be averaged out. The more data points the better as the curve fitting process will average out measurement noise. The curve fitting software tool runs in seconds so large data files are not a problem.

3.4 Ambient Light Setup Overview

In Time of Flight (TOF) sensor applications, ambient light is defined as constant reflected (non-signal) light that strikes the photodiode. This light causes a DC current to flow from the photodiode through the transimpedance amplifier in the front end of the ISL29501 chip. This current flow is not a problem for the chip because the incoming signal is AC coupled. However, this current introduces a delay in the photodiode which must be calibrated out. To minimize this phase delay, all photodiode selections need to have a visible light filter. When performing a sweep, the intensity of the ambient light (an independent variable) is varied, and the sensor and phase register data is collected at a fixed distance. In the coefficient generator software a second order error function is generated from which the coefficients are calculated. These are subsequently programmed into the chip. After coefficients are programmed, the chip DSP subtracts the appropriate quantity in each distance measurement.

Note: Ambient light is reflected light. This is not sunlight directly in the field of view (FOV) of the PD. Sunlight that directly strikes the PD causes large non-linear changes in distance and magnitude and is not a supported condition.

3.4.1 Light Source and Setup

On the ISL29501-ST-EV1Z Sand Tiger reference design, the emitter and PD are paired to be sensitive to 850nm IR light. At this time Renesas does not have a recommendation for an artificial light source that mimics sunlight. For this reason Renesas recommends using an 850nm LED.

The board should be positioned facing a target at a fixed distance. A good distance is 0.5–1m. Ideally the light source would strike the target and not the PD directly. In practice the optical power to do this is prohibitively high. For this reason the ambient light source should directly strike the PD.

3.4.2 Ambient Light Sweep Procedure

The goal is to correct for phase error introduced by the sun. A visible light filter on the photodiode reduces this effect but sunlight contains IR that introduces error. Like in temperature correction the goal is to change a single variable, ambient light in this case, and record changes in the measured phase. The distance is fixed so observed changes in phase will be due to this light. From the data an error function is created, converted and written back into the chip. Once programmed the chip will subtract the error based on the ambient light (register 0xE3) from distance measurements in real time.

In ambient light data sweeps it is important that the temperature remains constant. To avoid self-heating of the board you can use low integration times (reg0x10 ≤ 0x06) to avoid this. A good technique is to run the part making measurements until the temperature reaches equilibrium, then start collecting data. Both temperature and ambient light calibrations are independent and have separate coefficients.

3.4.3 Choosing an Ambient Light Source

Because the goal is to correct for changes in sunlight, the obvious light source for testing is the sun. However, when light testing in sunlight, only the light intensity may vary. All other variables must be fixed. Adjusting for light intensity in sunlight can be a challenge, however, thus it is better to collect the data in a controlled laboratory environment. The problem is that most artificial light sources have a different spectral content than the sun. Another problem is that their spectral content changes as their intensity changes.

To get around these issues, the preferred light source is an SFH 4550 850nm LED. This is the same LED used in the ISL29501-ST-EV1Z Sand Tiger reference design. Several characteristics make the SFH4550 850nm LED the best choice: it is cheap and readily available, the spectral content varies very little with intensity, and the optical power has a linear relationship with the current passing through it. The downside is that the spectral content is very different than sunlight. To correct for this difference, a correlation factor was calculated and programmed into the coefficient tool.

3.4.4 Setting the Sweep Range

The minimum end should be zero current into the ambient source LED. The maximum end of the range depends on several factors. The emitter choice, the PD choice, and the visible light filter all affect the ambient light code. The tendency is to try to convert LUX values but it is better to determine your ambient light range experimentally.

To set the max end of your sweep find the maximum anticipated code in application conditions. A few measurements in true sunlight should reveal this number. Record the maximum measured value in register 0xE3 (ambient light sensor). It is better to end the sweep at a lower code than a higher code. The reason is that at very high ambient light levels the PD becomes very non-linear and a second order equation may not accurately describe the error function. On the ISL29501-ST-EV1Z Sand Tiger board, a sweep up to a maximum code of 0x18 is normally good.

3.4.5 Laboratory Setup

The lab setup is fairly simple. The ambient light source (LED) should be angled ~30 degrees off the center of the PD and positioned at a 1–2cm distance.

Figure 3 and Figure 4 show a SFH 4550 LED positioned to inject ambient light into the ISL29501-ST-EV1Z Sand Tiger photodiode. The clip leads are connected to a power supply so that current through the LED can be varied. The box is aimed at a target and distance measurements are made by recording the ambient light code and phase. The distance and the reflectivity of the target are not important. The target distance must remain constant so that changes in phase are the result of the ambient light change only. For best results, the return signal magnitude should be in the 1–10nA range. For emitter current programming, see the *ISL29501 Datasheet*.

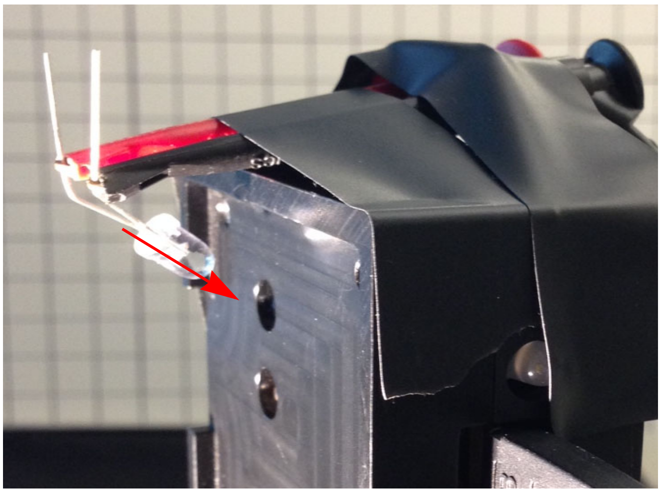


Figure 3. Ambient Light Setup Side View

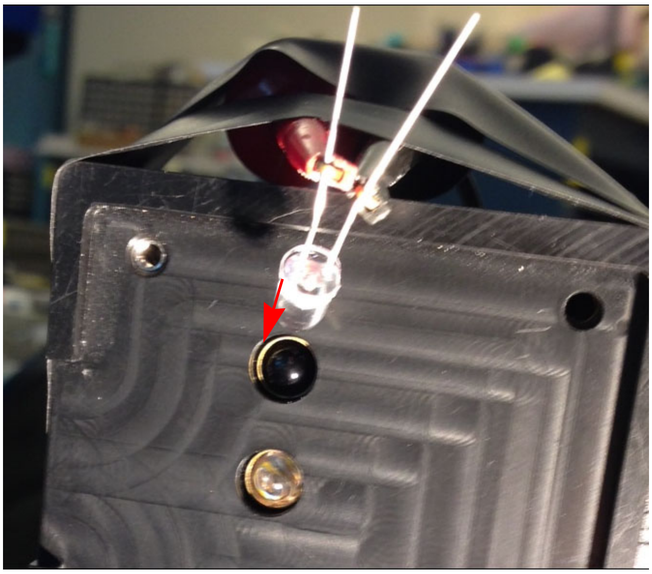


Figure 4. Ambient Light Setup Front View

4. Generating Coefficients

After data is collected, coefficients for a second order polynomial error curve need to be generated. To assist with this task, Renesas provides a tool Coeff_Gen.exe. The tool parses the data file, does a least squares curve fit and converts the coefficients to register values.

Coeff_Gen is a single executable file that runs under Windows. It requires .NET Framework 4.0, which is standard in Windows 8 and newer versions. For Windows 7 and older, .NET Framework 4.0 can be downloaded from Microsoft using this link: https://www.microsoft.com/en-us/download/details.aspx?id=17851.

4.1 Using Coeff_Gen

In general, red fill is used for fields that need attention. Yellow fields hold register values that will be programmed into the chip. Double-click the Coeff_Gen.exe file to start the tool. Figure 4 shows the initial state of the GUI. Additional fields will be added after the coefficient type is chosen.

The GUI can be divided into three sections:

- GUI Setup Conditions
- Ideal Coefficient Calculations
- Forced Exponent Adjustment

4.2 GUI Setup Conditions

The first action is to set the mode. The supported choices are Temperature and Ambient Light. A third option is Emitter Voltage, which is an alternate method of measuring temperature. It is not believed to be useful, however, and is not supported at this time. Click the pull-down labeled “Choose” and select the coefficient type. The next step is to set the coefficient reference. The GUI looks slightly different for Temperature and Ambient Light but in both cases a reference point must be set for the software. These fields are shown in Figure 5 and Figure 6.

4.2.1 Temperature

If Temperature is selected the temperature reference value must be entered in the GUI. It is defined as the temperature code in which distance calibration was executed. This entry gives the software and the chip (once programmed) a reference temperature in which correction will be zero (no correction). In the GUI this field is yellow which means that this is not only an input to the software but also a value that must be programmed into the chip. This reference value must be measured on each device and programmed into that specific chip. This effectively calibrates the temperature sensor. It is not necessary to collect data on every board. One set of coefficients will work for all boards with the same emitter and PD combination. For this scheme to work distance calibration must always be done at the same temperature.

4.2.2 Temperature Reference Adjustment

Temperature reference adjustment is easiest if distance calibration is always performed at the same temperature. If distance calibration cannot always be performed at the same temperature, two things need to be done:

- The temperature in degrees C must be accurately measured for each device at the time of distance calibration.
- The temperature reference measured at the time of distance calibration must be adjusted.

Note: dcal is shorthand notation for distance calibration in the following discussion.

Equation 1 is used to adjust the reference code (0x31):

(EQ. 1) $$\mathrm{TREFcodeadj} = \mathrm{TREFcodedcal} + 0.8 \times (\mathrm{TEMPdcal} - \mathrm{TEMPdcal\_ref})$$

- TREFcodeadj = the adjusted temp code for a particular device
- TREFcodedcal = the measured temp code at during distance calibration
- TEMPdcal = the temp in C during dcal for the test device
- TEMPdcal_ref = the temp in C during dcal for the ref device

The reference device is the one that was used to generate the temperature coefficients. This adjustment is made to the reference code to be programmed into register 0x31 based on the difference in the ambient temperature at dcal compared to the same for the reference device.

4.2.3 Ambient Light

The reference for ambient light is defined as the minimum data code from register 0xE3. Due to offsets in the ambient light sensing circuits the reference is usually 1 or 2. The GUI will search the data file and find this minimum code, which means that it should contain points where the ambient light is less than 100 lux. This is not a problem if the recommended 850nm LED is used as your light source in a lab. Florescent light has almost zero IR content so at 0 LED current, only chip offsets will be measured. If using other light sources the data file must contain data points where the PD was completely covered. This ambient light reference value is used by the software in calculations but is not written to the chip. Note: During distance calibration it is important that the ambient light is much less than 1 lsb of the chip sensor. Values of 100 lux or less are good.

4.2.4 Alternate Light Sources

The GUI contains three radio buttons for light source choices which are followed by a slope correction field (see Figure 6). The correction in this field compensates for the difference in spectral content of the chosen light source and real sunlight. For the **Sunlight** choice, the correction slope is 0. For the **850nm LED**, the correction is 5.86. For these two cases, the field is read only. Although not recommended, the user can select **Other** to use a light source other than the **Sunlight** or **850nM LED** options. The slope correction must then be entered in this field. To calculate slope correction, the user would need to characterize their light source and compare the performance of the ISL29501 in sunlight to their artificial light source. The slope of the difference in the two error functions is entered in this field.

4.2.5 Slope Correction

The slope correction factor compensates for the difference between sunlight and an alternate ambient light source. It is the difference between the error correction curve for sunlight and the error curve for a different (not sunlight) ambient light source. The calibration has been done for the 850nm LED and the correction slope is embedded in the software. If the user has a different light source or different optical components, the slope correction may not be accurate. To create a custom slope correction term, the user needs to create an ambient light versus phase error curve in sunlight and one with the new light source and subtract these curves. A linear curve fit of this different function is performed and its slope is entered in the **Corr Slope** field (**Other** must be selected).

4.3 Ideal Coefficient Calculations

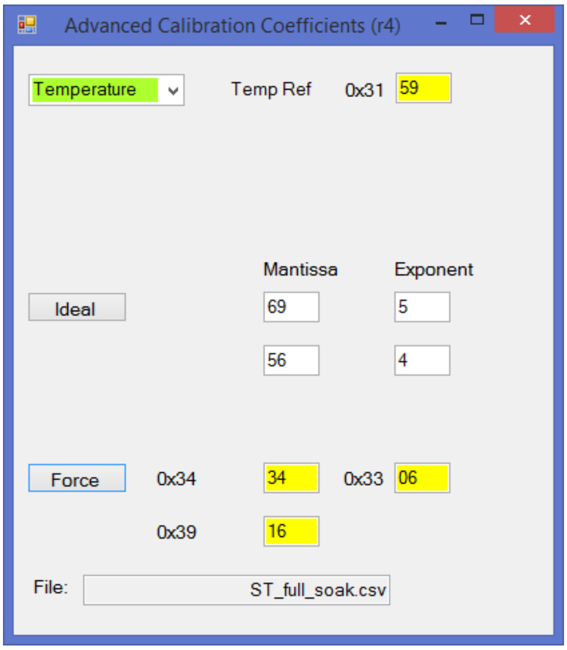
At this stage, 90% of the error correction process is complete. When the user presses the Ideal button, a file browser opens. The default starting directory is where the executable resides but the user can browse to the data file. This file must comply with the requirements in Data File Requirements. After the data file is chosen and **Open** is clicked, the program calculates the ideal coefficients for that file and they appear in the four fields. However, these are not the final values.

4.4 Ideal Coefficients Defined

Ideal coefficients are the result of the coefficient conversion to hexadecimal of the error curve data. Each coefficient has a mantissa and an exponent. The ISL29501 device has a single exponent that must cover all coefficients. In many cases, the exponent is not the same for all coefficients as in the example data in Figure 5 and Figure 6. To compensate for this difference, one additional step is required.

4.5 Force Exponent Adjustment

The exponent for all coefficients is programmed in register 0x33. In the GUI the user needs to enter a single value into the field labeled 0x33 and click the **Force** button. The software adjusts each mantissa to work with the chosen exponent. The best exponent choice is the highest numerical value of the four ideal (ambient and temperature) exponent values. When the Force step is completed, the final register values show in boxes with a yellow fill. There is no non-volatile memory in the ISL29501 so these values must be programmed into the chip during initialization to enable Advanced Calibration corrections. Figure 5 and Figure 6 show the result of a completed coefficient generation.



Advanced Calibration Coefficients (r4)

Temperature ▾ Temp Ref 0x31 59

Mantissa Exponent

Ideal 69 5

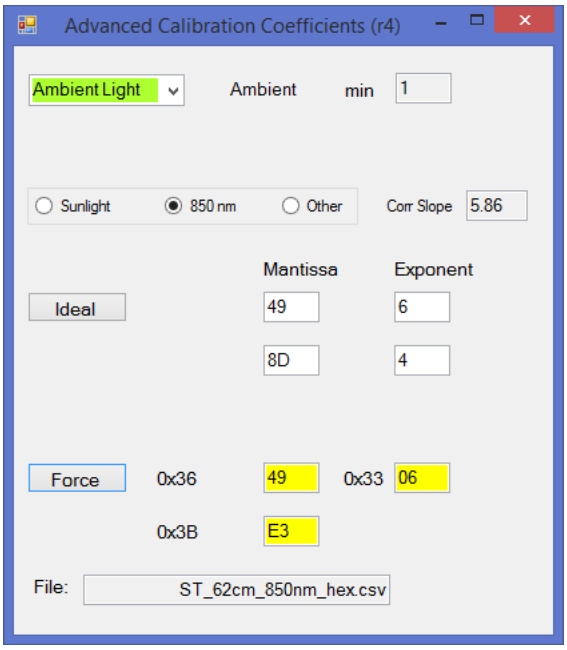
56 4

Force 0x34 34 0x33 06

0x39 16

File: ST_full_soak.csv

Figure 5. Completed Temperature Coefficients



Advanced Calibration Coefficients (r4)

Ambient Light ▾ Ambient min 1

○ Sunlight ● 850 nm ○ Other Corr Slope 5.86

Mantissa Exponent

Ideal 49 6

8D 4

Force 0x36 49 0x33 06

0x3B E3

File: ST_62cm_850nm_hex.csv

Figure 6. Completed Ambient Coefficients

4.6 Finalizing Coefficients

Figure 5 and Figure 6 show two examples of the GUI: one for temperature data and one for ambient light data. **Important:** In the GUI, two coefficients have an ideal exponent of 4, one has an exponent of 5, and one has an exponent of 6. The user must pick 6 as the exponent to force and eventually program in register 0x33. As explained earlier all fields with yellow backgrounds are chip register values and must be programmed during initialization. Any number of instances of the GUI can be run simultaneously, which is useful when comparing multiple devices or setup conditions. The File field shows the data file name so you can easily keep track of multiple sets of data. This field is read-only and cannot be used to enter a file name.

4.7 Measurements with Advanced Calibration Coefficients

After Advanced Calibration Coefficients are programmed in the chip the user must be careful with calibration. Distance calibration becomes tied to the temperature reference value. In other words, the user can run distance calibration again (part of standard calibration, see the *ISL29501 Datasheet*), but it must be done at the same temperature code that is entered in the temperature reference register 0x31. For this reason, it is easier to run distance calibration and record/save the value from register 0xE2 before programming advanced calibration coefficients.

After the advanced calibration registers are programmed, every distance measurement is dynamically corrected for changes in temperature and ambient light.

5. Revision History

Revision	Date	Description
1.00	Apr 11, 2022	Applied new template.
0.00	Mar 31, 2017	Initial release.

IMPORTANT NOTICE AND DISCLAIMER

RENESAS ELECTRONICS CORPORATION AND ITS SUBSIDIARIES (“RENESAS”) PROVIDES TECHNICAL SPECIFICATIONS AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for developers skilled in the art designing with Renesas products. You are solely responsible for (1) selecting the appropriate products for your application, (2) designing, validating, and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. Renesas grants you permission to use these resources only for development of an application that uses Renesas products. Other reproduction or use of these resources is strictly prohibited. No license is granted to any other Renesas intellectual property or to any third party intellectual property. Renesas disclaims responsibility for, and you will fully indemnify Renesas and its representatives against, any claims, damages, costs, losses, or liabilities arising out of your use of these resources. Renesas' products are provided only subject to Renesas' Terms and Conditions of Sale or other applicable terms agreed to in writing. No use of any Renesas resources expands or otherwise alters any applicable warranties or warranty disclaimers for these products.

(Rev.1.0 Mar 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/

Trademarks