User's Guide

SBAU310–February 2018

OPT3101 Distance Sensor System Calibration

This user's guide details the calibration procedure for the OPT3101 device to get accurate distance measurement.

Contents

1 Introduction 2
2 Initialization 2
3 Reference Documents 2
4 System Calibration 2

List of Figures

1 OPT3101 Phase Data Path 2
2 OPT3101 System Calibration Steps 3
3 Frequency Calibration Block Diagram 3
4 Illumination Crosstalk Calibration Procedure 7
5 Phase Offset Calibration Procedure 9
6 Distance Error With Ambient Current: SFH2701 as Photodiode 11
7 Distance Error With Ambient Current: SFH213FA as Photodiode 11
8 PWL Based Correction for Ambient Dependent Phase 12

List of Tables

1 GP2 Configuration Registers 4
2 Frequency Calibration Registers 4
3 Digital Frequency Trim Registers 5
4 Internal Crosstalk Correction Registers 5
5 Illumination Crosstalk Measurement Register Settings 6
6 Illumination Crosstalk Correction Registers 7
7 Crosstalk Temperature Correction Registers 8
8 Phase Offset Correction Registers 9
9 Phase Temperature Coefficient Registers for On-Chip Temperature Sensor 10
10 Phase Temperature Coefficient Registers for External Temperature Sensor 11
11 Ambient Dependent Phase Correction Registers 12

1 Introduction

OPT3101 is a fully integrated Time of Flight (ToF) based distance sensor AFE. Figure 1 shows the data path on the device. The OPT3101 performs the following correction on the chip to get accurate distance measurements.

- Crosstalk
- Phase offset
- Phase correction with ambient
- Phase correction with temperature
- Frequency
- Square-wave nonlinearity

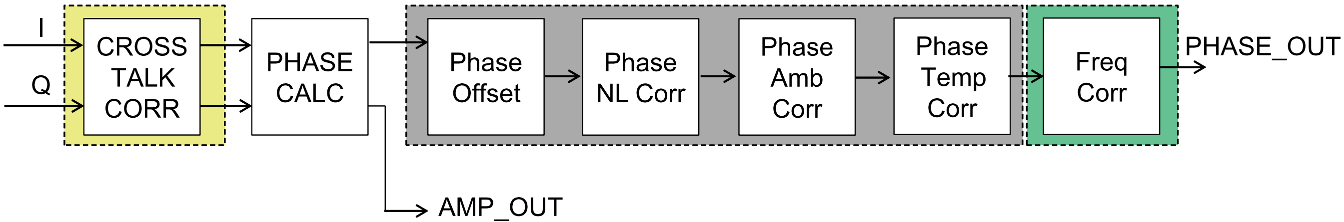

Figure 1. OPT3101 Phase Data Path

2 Initialization

After device power up and supplies are stable, apply the device reset by giving an active low pulse duration > 30 µs on the $\overline{RST\_MS}$ pin.

Write the following registers to set the device running in required condition:

- Write all of the required register values to set the device to operate at the desired sample rate (NUM_SUB_FRAMES, NUM_AVG_SUB_FRAMES).
- Select the required maximum ambient current support by writing IAMB_MAX_SEL

  All of the calibration steps should be performed in non-HDR mode. This is default power-up state after reset is applied (EN_ADAPTIVE_HDR = 0).
- Enable on-chip temperature conversion EN_TEMP_CONV = 1.
- Write I^2C host settings to read the external temperature sensor if it is present in the system. To view the register settings, see the *I^2C Master Register Settings* table in *OPT3101 ToF based Long Range Proximity and Distance Sensor AFE*.
- Enable Timing Generator TG_EN = 1.

3 Reference Documents

- *OPT3101 ToF based Long Range Proximity and Distance Sensor AFE*
- *Introduction to Time-of-Flight Optical Proximity Sensor System Design*

4 System Calibration

Figure 2 shows the list of steps to be carried for OPT3101 system calibration. This calibration routine must be done one time per board. Some of the correction coefficients, such as phase correction with temperature and phase correction with ambient, must be done once per system design using a batch of 5 to 10 systems (the same coefficients can be used for all the systems).

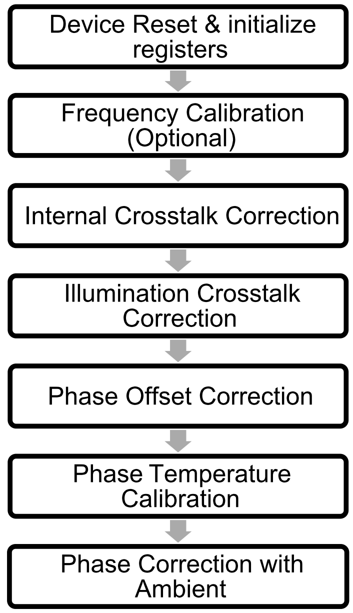

Figure 2. OPT3101 System Calibration Steps

4.1 Frequency Calibration

4.1.1 Dynamic Frequency Calibration using External Reference Clock

In a ToF based distance measurement system, accurate estimation of the frequency is critical in determining the absolute accuracy. The OPT3101 device uses on-chip high frequency oscillator for illumination. To correct for the frequency variations of the on-chip oscillator, the OPT3101 has built-in frequency calibration using an external low frequency reference clock. Figure 3 shows the block diagram of the frequency calibration on the chip.

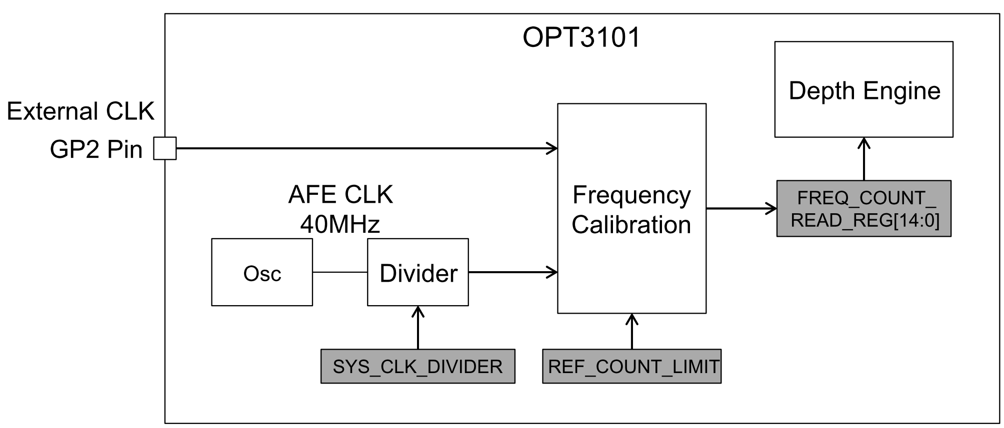

Figure 3. Frequency Calibration Block Diagram

The OPT3101 has a divider in the system clock path to support a wide range of external clock frequencies (from 32.768 kHz to 40 MHz for frequency calibration). Lower frequency reference clocks should be used to avoid high power consumption from switching. Frequencies closer to modulation clock frequency of 10 MHz should not be used for reference clock to avoid electrical crosstalk issues. This divider should be programmed to get the divided system clock frequency f_{SYS_DIV} closer to the external reference frequency (f_{EXT}). The value of SYS_CLK_DIV to be programmed can be calculated from Equation 1. With this register setting the the system clock frequency used for frequency calibration is given by Equation 2.

$$SYS\_CLK\_DIV = round\left(\log_2\left(\frac{40e^6}{f_{EXT}}\right)\right) \quad (1)$$

$$f_{SYS\_DIV} = \frac{40MHz}{2^{SYS\_CLK\_DIV}} \quad (2)$$

The reference count value should be set according to Equation 3:

$$REF\_COUNT\_LIMIT = 2^{14} \times \frac{f_{SYS\_DIV}}{f_{EXT}} \quad (3)$$

For example, if external clock f_{EXT} = 32768 Hz, divide the system clock by 2^{10} (SYS_CLK_DIV = 10 calculated using Equation 1) to get it closer to f_{EXT}.

$$f_{SYS\_DIV} = \frac{40MHz}{2^{SYS\_CLK\_DIV}} = 39.0625kHz \quad (4)$$

$$REF\_COUNT\_LIMIT = 2^{14} \times \frac{f_{SYS\_DIV}}{f_{EXT}} = 19531 \quad (5)$$

The external reference clock should be provided on GP2 pin. Configure the device according to the settings listed in Table 1 to accept external clock on GP2 pin.

Table 1. GP2 Configuration Registers

Register	Address	Value	Description
GPIO2_IBUF_EN	78h[16]	1	Enable input buffer of GP2 pin. External reference clock should be connected to this pin for frequency calibration.
GPIO2_OBUF_EN	78h[15]	1	Enable output buffer of GP2 pin.

Table 2 lists the register settings to enable the frequency calibration. Under this condition, the device measures the internal on-chip oscillator frequency with respect to the reference clock. The result of this measurement is stored in the FREQ_COUNT_READ_REG register. This value will be used to multiply the phase in depth engine to provide an accurate phase output.

Table 2. Frequency Calibration Registers

Register	Address	Value	Description
ENABLE_AUTO_FREQ_COUNT	0Fh[21]	1	Determines which value to be used for frequency correction. 0 – Trimmed value 1 – Measured value from frequency calibration
ENABLE_FLOOP	0Fh[22]	1	Enables the frequency calibration block.
EN_FREQ_CORR	0Fh[23]	1	Enable frequency correction for the phase output
REF_COUNT_LIMIT	0Fh[14-0]	Equation 3	This sets the limit of system clock (f_{SYS_DIV}) counts at which the the reference clock counter is stopped.
SYS_CLK_DIV	0Fh[20-17]	Equation 1	Programs system clock divider for frequency calibration. This should be adjusted to get it closer to the external reference frequency.
ENABLE_CONT_FCALIB	10h[15]	1	Enables continuous frequency calibration. 0 – Frequency is measured only when START_FREQ_CALIB = 1 1 – Frequency is continuously measured.
FREQ_COUNT_READ_REG	10h[14-0]	Read Register	Read register which holds the value of frequency correction when frequency calibration is enabled.
START_FREQ_CALIB	0Fh[16]		starts the freq_calib. This register write is required only when ENABLE_CONT_FCALIB = 0.

4.1.2 On Chip Frequency Trim Correction

This section is applicable for systems not having external reference clock for frequency calibration detailed in Section 4.1.1. The device oscillator will be trimmed for frequency within ±3%. Further phase will be corrected digitally using a trimmed frequency correction coefficient. The correction trim value can be read through FREQ_COUNT_REG. Since this trim is measured at room temperature, the phase at other temperatures could be inaccurate depending on the oscillator frequency variation. A 100°C temperature change will introduce an error of up to 3.6% in the oscillator frequency and phase measurement. For applications requiring high accuracy, it is recommended to use dynamic frequency calibration.

Table 3. Digital Frequency Trim Registers

Register	Address	Value	Description
ENABLE_AUTO_FREQ_COUNT	0Fh[21]	0	Determines which value to be used for frequency correction. 0 – Trimmed value 1 – Measured value from frequency calibration
FREQ_COUNT_REG	011h[14:0]	Read Register	Digital Frequency Correction Trim Value.
EN_FREQ_CORR	0Fh[23]	1	Enable frequency correction for the phase output

4.2 Crosstalk Calibration

Any digital, illumination driver switching at the modulation clock frequency (10 MHz) creates electrical crosstalk. Additionally, depending on the system, there could be some optical crosstalk. Crosstalk is categorized as internal crosstalk or illumination crosstalk.

4.2.1 Internal Crosstalk

Electrical crosstalk due to any digital switching at the modulation clock frequency (10 MHz), except from the illumination driver, is treated as internal crosstalk. This crosstalk can be estimated and corrected dynamically at any point of time in the system. This crosstalk can be corrected automatically using internal crosstalk calibration engine inside the device. No additional device setup is required for measuring this crosstalk. Setting INT_XTALK_CALIB register bit to 1 initiates the crosstalk measurement. Result of this crosstalk measurement can be readout from registers IPHASE_XTALK, QPHASE_XTALK, with IQ_READ_DATA_SEL = 0. To get accurate estimate of the crosstalk, a digital filter is also present in the crosstalk measurement path. Based on the accuracy requirement, the filter time constant $\tau = 2^{XTALK\_FILT\_TIME\_CONST}$ should be set. At least 5τ frames should be allowed for settling of crosstalk measurement. For accuracy, at least 2^{10} subframes should be allowed for filter averaging. XTALK_FILT_TIME_CONST can be calculated by using Equation 6.

$$XTALK\_FILT\_TIME\_CONST = \log_2\left(\frac{2^{10}}{1+NUM\_AVG\_SUB\_FRAMES}\right) \quad (6)$$

Table 4. Internal Crosstalk Correction Registers

Register	Address	Value	Description
XTALK_FILT_TIME_CONST	2Eh[23-20]	Equation 6	Time constant for crosstalk filtering. Time constant $\tau = 2^{XTALK\_FILT\_TIME\_CONST}$ frames. At least 3τ should be allowed for settling of crosstalk measurement.
USE_XTALK_FILT_INT	2Eh[5]	1	Select filter or direct sampling for internal crosstalk measurement. 0 – Direct sampling, 1 – Filter
USE_XTALK_REG_INT	2Eh[6]	0	Select register value or internally calibrated value for internal crosstalk. 0 – Calibration value, 1 – Register value

Table 4. Internal Crosstalk Correction Registers (continued)

Register	Address	Value	Description
INT_XTALK_CALIB	2Eh[4]	1 → 0	The device initializes the internal electrical crosstalk measurement upon setting this bit. INT_XTALK_CALIB = 1 Wait for 5τ frames ($\tau = 2^{XTALK\_FILT\_TIME\_CONST}$). Number of frames should be at least 5τ for the first time after power up. For subsequent calibration cycles during runtime, number of frames of calibration can be made smaller, as only crosstalk changes need to be updated. INT_XTALK_CALIB = 0

Internal crosstalk correction can be skipped and directly illumination crosstalk measurement can be performed. In this scenario, illumination crosstalk measurement will capture the sum of the internal and illumination crosstalk.

4.2.2 Illumination Crosstalk

All of the crosstalk resulting from the illumination driver switching (electrical and optical) is treated as illumination crosstalk. Estimation of this crosstalk requires masking the photodiode from receiving any modulated optical signal. Figure 4 shows the procedure for this crosstalk calibration. Setting the ILLUM_XTALK_CALIB register bit to 1 initiates the crosstalk measurement. The result of this crosstalk measurement is stored in internal device registers, which can be readout from registers IPHASE_XTALK, QPHASE_XTALK, with IQ_READ_DATA_SEL = 1. Crosstalk correction from these internal registers can be applied by writing the USE_XTALK_REG_ ILLUM bit to 0. For later use, these register values can be stored in external memory. After every device reset these stored crosstalk values should be loaded into the registers IPHASE_XTALK_REG_HDR<i>_TX<j>, QPHASE_XTALK_REG_HDR<i>_TX<j>.

The crosstalk read register (IPHASE_XTALK, QPHASE_XTALK) is signed 24 bit while the load registers (IPHASE_XTALK_REG_HDR<i>_TX<j>, QPHASE_XTALK_REG_HDR<i>_TX<j>) are signed 16 bit. The read register should be appropriately scaled down to signed 16 bit and the scale factor (ILLUM_XTALK_REG_SCALE) should be programmed along with signed 16-bit register. The OPT3101 device supports three illumination channels. Each channel has a different illumination crosstalk because of different current paths and optical crosstalk. Each channel requires crosstalk measurement for two currents ILLUM_DAC_L_TX<j> and ILLUM_DAC_H_TX<j> if used in HDR mode. This crosstalk depends on the illumination current. For any change in illumination driver current setting (ILLUM_DAC), this crosstalk should be re-estimated. This crosstalk measurement also uses similar digital filter as internal crosstalk measurement to reduce noise and improve accuracy and the settings of this filter are same as internal crosstalk.

Table 5. Illumination Crosstalk Measurement Register Settings

Register	Address	Value	Description
USE_XTALK_FILT_ILLUM	2Eh[7]	1	Select filter or direct sampling for Illumination crosstalk measurement. 0 – Direct sampling, 1 – Filter
USE_XTALK_REG_ ILLUM	2Eh[8]	0	Select register value or internally calibrated value for illumination crosstalk. 0 – internal calibration value, 1 – Register value
ILLUM_XTALK_CALIB	2Eh[12]	1 → 0	The device initializes the Illumination crosstalk measurement upon setting this bit. This measurement should be done with the photodiode masked such that no modulated light is received. ILLUM_XTALK_CALIB = 1 Wait for 5τ frames ($\tau = 2^{XTALK\_FILT\_TIME\_CONST}$). ILLUM_XTALK_CALIB = 0

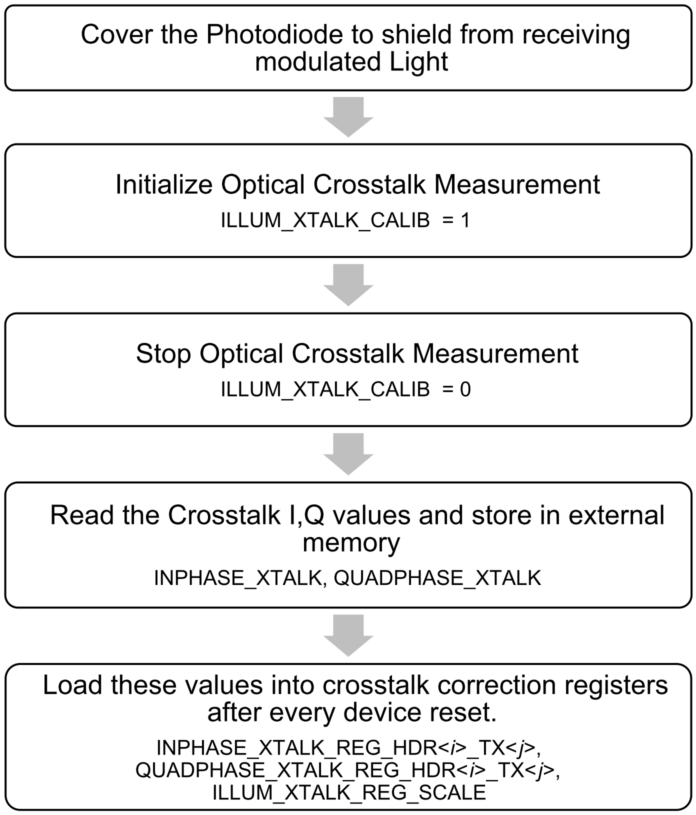

Figure 4. Illumination Crosstalk Calibration Procedure

Table 6. Illumination Crosstalk Correction Registers

Register	Address	Value	Description
USE_XTALK_REG_ ILLUM	2Eh[8]	1	Select register value or internally calibrated value for illumination crosstalk. 0 – internal calibration value, 1 – Register value
ILLUM_XTALK_REG_SCALE	2E[19-17]		Scale factor Illumination crosstalk register (IPHASE_XTALK_REG_HDR<i>_TX<i>, QPHASE_XTALK_REG_HDR<i>_TX<j>, i={0,1}; j={0,1,2}). Scale = $2^{ILLUM\_XTALK\_REG\_SCALE}$
IPHASE_XTALK_REG_HDR<i>_TX<j>			Register for Illumination crosstalk in-phase component. TX<j> indicates illumination channel HDR<i> indicates illumination DAC current. HDR0: ILLUM_DAC_L_TX<j>, HDR1: ILLUM_DAC_H_TX<j>
QPHASE_XTALK_REG_HDR <i>_TX<j>			Register for Illumination crosstalk quad-phase component.

4.2.3 Scale Value for Crosstalk Measurement

The crosstalk measurement uses a default gain of 2^6 (FORCE_SCALE_VALUE = 0) in the signal path to improve the measurement accuracy. A system with crosstalk greater than 32768 × 1.64676 / 2^6 = 843 requires adjusting this scale so that crosstalk measurement does not saturate. Typically expected crosstalk is < 200 codes with highest LED current.

First measure crosstalk with the highest LED current to be used (mask photodiode and measure the amplitude: AMP_OUT register). From the measured crosstalk amplitude, calculate the FORCE_SCALE_VAL using Equation 7.

$$FORCE\_SCALE\_VAL = ceil\left(\log 2\left(ceil\left(crosstalk\ amplitude / 843\right)\right)\right) \quad (7)$$

Use this calculated FORCE_SCALE_VAL in the default initialization settings. This is only one time activity for a system design. Across different boards it will not change. Mostly this step is never required in a well designed board where the crosstalk is much less than 843 and the default value of FORCE_SCALE_VAL can be used.

4.2.4 Crosstalk Temperature Correction

Illumination crosstalk can vary with the temperature. To correct for this variation, OPT3101 has on-chip correction in the vector domain (see Equation 8).

$$(I + jQ)_{out} = (I + jQ) - (I_{xtalk\_corr} + jQ_{xtalk\_corr}) \quad (8)$$

$$(I_{xtalk\_corr} + jQ_{xtalk\_corr}) = \frac{(TMAIN - TMAIN\_CALIB) \times (TEMP\_COEFF\_XTALK\_IPHASE + j.\ TEMP\_COEFF\_XTALK\_QPHASE)}{2^{(-5 + SCALE\_TEMP\_COEFF\_XTALK)}} \quad (9)$$

To measure temperature coefficient of crosstalk, vary the temperature and measure the illumination crosstalk as discussed in Section 4.2.2. Find the slope of in-phase IPHASE_XTALK and quadrature phase crosstalk QPHASE_XTALK component versus temperature (TMAIN in 12-bit format). Multiply the measured slope with $2^{(-5\ +\ SCALE\_TEMP\_COEFF\_XTALK)}$ to obtain the coefficient to be written to the register. Choose the SCALE_TEMP_COEFF_XTALK such that all the coefficient values fit within 8-bit signed number (–128 to 127).

Table 7. Crosstalk Temperature Correction Registers

Register	Address	Description
SCALE_TEMP_COEFF_XTALK	3Ah[19:17]	Scaling factor for crosstalk temperature correction.
TEMP_COEFF_XTALK_IPHASE_HDR<i>_TX<j>		In-phase crosstalk temperature coefficient. TX<j> indicates illumination channel HDR<i> indicates illumination DAC current. HDR0: ILLUM_DAC_L_TX<j>, HDR1: ILLUM_DAC_H_TX<j>
TEMP_COEFF_XTALK_QPHASE_HDR<i>_TX<j>		Quadrature-phase crosstalk temperature coefficient

4.3 Phase Offset Calibration

Phase offset measurement should be carried out after the crosstalk correction. For this measurement, set the target object at a known distance such that the received signal amplitude is high (AMP_OUT>10000). Perform phase measurement. Compute the phase offset by subtracting the measured phase and ideal expected phase using Equation 10. Also read the temperature of the device from on-chip temperature sensor TMAIN, external temperature sensor TILLUM (if present) and ambient output AMB_DATA. Store phase offset and temperature and ambient at which this phase offset calibration is carried out. These values should be loaded into PHASE_OFFSET_HDR<i>_TX<j>, TMAIN_CALIB_HDR<i>_TX<j>, TILLUM_CALIB_HDR<i>_TX<j> and AMB_CALIB register after every device reset. Setting EN_PHASE_CORR = 1 applies the phase offset correction.

$$\varphi_{offset} = (\varphi_{meas} - \varphi_{object}) \times \frac{1}{K_{\varphi}} \quad (10)$$

$$\varphi_{object} = \frac{d_{object}}{\left(c / 2f_{MOD}\right)} \times 2^{16}$$

where

- f_{MOD} = 10 MHz
- K_{φ} = freq_count / 2^{14}
- c = 299792458 m/s (11)

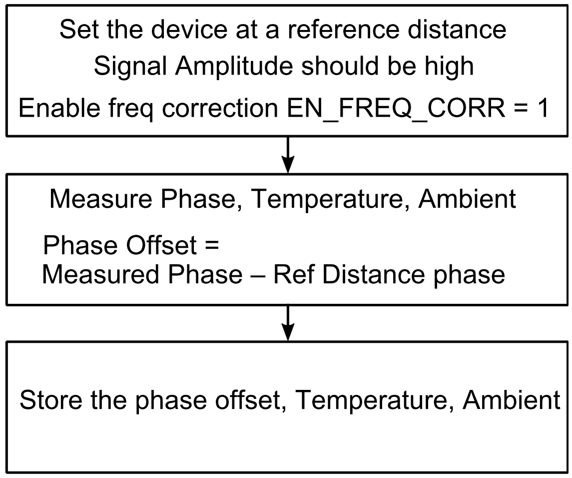

Figure 5. Phase Offset Calibration Procedure

In a system with dynamic frequency calibration using external reference clock, freq_count can be read from register FREQ_COUNT_READ_REG . In a system using on chip trim (without an external clock reference), freq_count can be obtained from register FREQ_COUNT_REG.

Table 8. Phase Offset Correction Registers

Register	Address	Description
EN_PHASE_CORR	43h [0]	Enables phase offset correction
PHASE_OFFSET_HDR0_TX0	42h[15-0]	Phase offset for TX0 Illumination Channel with current of ILLUM_DAC_L_TX0
PHASE_OFFSET_HDR1_TX0	51h[15-0]	Phase offset for TX0 Illumination Channel with current of ILLUM_DAC_H_TX0
PHASE_OFFSET_HDR0_TX1	52h[15-0]	Phase offset for TX1 Illumination Channel with current of ILLUM_DAC_L_TX1
PHASE_OFFSET_HDR1_TX1	53h[15-0]	Phase offset for TX1 Illumination Channel with current of ILLUM_DAC_H_TX1
PHASE_OFFSET_HDR0_TX2	54h[15-0]	Phase offset for TX2 Illumination Channel with current of ILLUM_DAC_L_TX2
PHASE_OFFSET_HDR1_TX2	55h[15-0]	Phase offset for TX2 Illumination Channel with current of ILLUM_DAC_H_TX2

4.4 Phase Temperature Correction

In any time of flight based distance measurement system, delay variation due to temperature will cause measurement errors. This should be corrected so that measured phase is not affected by the temperature. Phase versus temperature is very linear for system with OPT3101 over the temperature range of –40°C to +100°C, which can be corrected using linear correction.

4.4.1 Measuring Temperature Coefficient using on Chip Temperature Sensor

OPT3101 has built-in temperature sensor that gives the die temperature, which can be used for phase correction with temperature. Load Illumination crosstalk values into the register, perform internal crosstalk correction and Set EN_TEMP_CORR = 0 before measuring the phase temperature coefficient. Sweep device temperature and measure die temperature TMAIN and PHASE_OUT, make sure that Amplitude is high for the measurement. During the measurement OPT3101 system should be kept away from the blowing air of the heating device. Measure the slope of phase (PHASE_OUT) vs temperature (TMAIN) and multiply this slope by $2^{(10 - SCALE\_PHASE\_TEMP\_COEFF)}$ to obtain phase temperature coefficient as given in Equation 12. Value of SCALE_PHASE_TEMP_COEFF should be chosen such that the TEMP_COEFF_* is maximum and < 2047 to reduce quantization errors. Preferred value of SCALE_PHASE_TEMP_COEFF is two for the TI EVM. Dynamic frequency calibration discussed in Section 4.1.1 should be used during this measurement and frequency correction should also be enabled (EN_FREQ_CORR = 1).

$$TEMP\_COEFF\_MEAS = \frac{d(PHASE)}{d(TMAIN)} \times 2^{(10-SCALE\_PHASE\_TEMP\_COEFF)} \quad (12)$$

TMAIN is the raw register readout from the on chip temperature sensor in 12-bit format. Measure and save TEMP_COEFF_MEAS for all the illumination DAC currents to be used in the system. TEMP_COEFF_MEAS is constant for a given system design. Perform this measurement on a batch of 3 to 5 boards and use the average of measured coefficients for all the boards.

Measured temperature coefficient should be divided by the respective device's phase scaling factor, $k_{\varphi,tcalib}$ = (freq_count) / 16384, to obtain the temperature coefficient to be written into the device register. For a system with dynamic frequency calibration k_{φ} measured during phase calibration (at TMAIN_CALIB temperature) should be used.

$$TEMP\_COEFF = \frac{TEMP\_COEFF\_MEAS}{k_{\varphi,tcalib}} = TEMP\_COEFF\_MEAS \times \frac{16384}{freq\_count} \quad (13)$$

In a system with dynamic frequency calibration, freq_count can be read from FREQ_COUNT_READ_REG register during phase offset calibration. In a system using on chip trim (without an external clock reference), freq_count can be obtained from register FREQ_COUNT_REG. The OPT3101 will internally correct for the phase temperature variation as per Equation 14.

$$PHASE\_OUT = Phase - (TMAIN - TMAIN\_CALIB) \times TEMP\_COEFF \times \frac{2^{SCALE\_PHASE\_TEMP\_COEFF}}{2^{10}} \quad (14)$$

Separate TEMP_COEFF and TMAIN_CALIB registers for each illumination channel and each current of the HDR mode are available. TMAIN_CALIB is the temperature read from TMAIN during phase offset calibration. Separate TEMP_COEFF and TMAIN_CALIB registers for each illumination channel and each current of the HDR mode are available. TEMP_COEFF varies with the Illumination current and should be measured for each illumination current to be used in the system.

Table 9. Phase Temperature Coefficient Registers for On-Chip Temperature Sensor

Parameter	Address	Description
EN_TEMP_CORR	43h[1]	Enable temperature correction
SCALE_PHASE_TEMP_COEFF	43h[8:6]	Adjust scale factor for temperature coefficient
TMAIN_CALIB_HDR<i>_TX<j>		Calibration temperature of on-chip temperature sensor for TX<j> illumination channel with current of ILLUM_DAC_L/H_TX<j>
TEMP_COEFF_MAIN_HDR<i>_TX<j>		Phase temperature coefficient for on-chip temperature for TX<j> illumination channel with current of ILLUM_DAC_L/H_TX<j>

TMAIN is in a 12-bit format, from which the actual temperature in °C can be calculated using Equation 15

$$T = TMAIN / 8 - 256 \quad (15)$$

4.4.2 Measuring Temperature Coefficient using External Temperature Sensor

For this mode, the device should be configured to read the external temperature sensor, which will be stored in TILLUM device register. Follow the procedure similar to Section 4.4.1 to compute the temperature coefficient. Temperature coefficient measured using Equation 16 should be further multiplied by 16384/freq_count value.

$$TEMP\_COEFF = \frac{d(PHASE)}{d(TILLUM)} \times 2^{10-SCALE\_PHASE\_TEMP\_COEFF} \quad (16)$$

$$PHASE\_OUT = Phase - (TILLUM - TILLUM\_CALIB) \times TEMP\_COEFF \times \frac{2^{SCALE\_PHASE\_TEMP\_COEFF}}{2^{10}} \quad (17)$$

Table 10. Phase Temperature Coefficient Registers for External Temperature Sensor

PARAMETER	DESCRIPTION
TILLUM_CALIB_HDR<i>_TX<j>	Calibration temperature of external temperature sensor or TX<j> illumination channel with current of ILLUM_DAC_L/H_TX<j>.
TEMP_COEFF_ILLUM_HDR<i>_TX<j>	Phase temperature coefficient for illumination using external temperature sensor for TX<j> illumination channel with current of ILLUM_DAC_L/H_TX<j>

Temperature from external temperature sensor in °C can be calculated using Equation 18

$$T = TILLUM / 16 - 128 \quad (18)$$

4.5 *Phase Correction with Ambient*

With ambient light, photo diode DC bias current changes. This will cause photo diode AC signal phase change, phase (delay) reduces with increase in DC bias current. This variation is strongly dependent on the photo diode bandwidth. High bandwidth photo diodes will have less phase variation with ambient compared to low bandwidth photodiodes. Figure 6 shows the distance error with high bandwidth photodiode SFH2701 and Figure 7 shows the distance error with low bandwidth photodiode SFH213FA. For the same amount ambient current, SFH213FA has approximately 5x more phase variation compared to SFH2701. For the SFH2701, phase versus ambient is linear and can be corrected using first order polynomial. For the SFH213FA, phase versus ambient is non-linear and requires PWL correction.

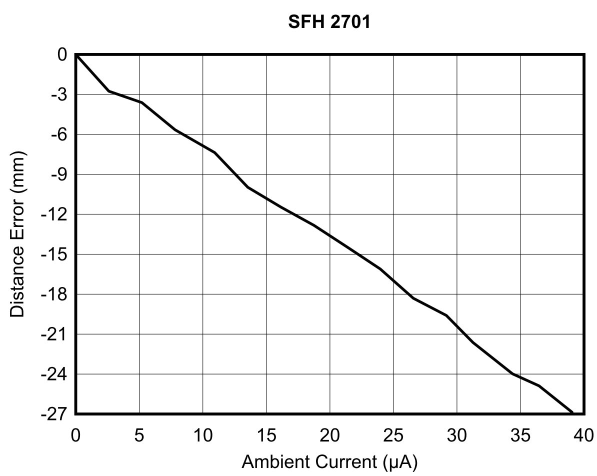

Figure 6. Distance Error With Ambient Current: SFH2701 as Photodiode

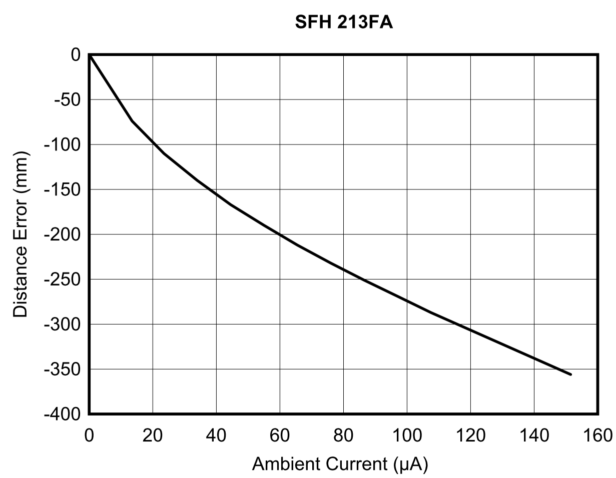

Figure 7. Distance Error With Ambient Current: SFH213FA as Photodiode

To correct for this error, four-segment PWL phase correction with ambient is implemented using on-chip ambient ADC output (AMB_DATA). Equation 20 give the phase correction with ambient and corresponding register settings are listed in Table 7 .

$$
\begin{aligned}
PHASE\_CORR_{AMB} &= AMB \times C_0 && : AMB < X_0 \\
&= C_0X_0 + (AMB - X_0) \times C_1 && : X_0 < AMB < X_1 \\
&= C_0X_0 + C_1(X_1 - X_0) + (AMB - X_1) \times C_2 && : X_1 < AMB < X_2 \\
&= C_0X_0 + C_1(X_1 - X_0) + C_2(X_2 - X_1) + (AMB - X_2) \times C_3 && : AMB > X_2
\end{aligned}
$$

where

- AMB = AMB_DATA - AMB_CALIB

- AMB_CALIB = AMB_DATA measured during phase offset calibration (19)

$$PHASE\_OUT = Phase - \frac{PHASE\_CORR_{AMB}}{2^{(2+SCALE\_AMB\_PHASE\_CORR\_COEFF)}} \quad (20)$$

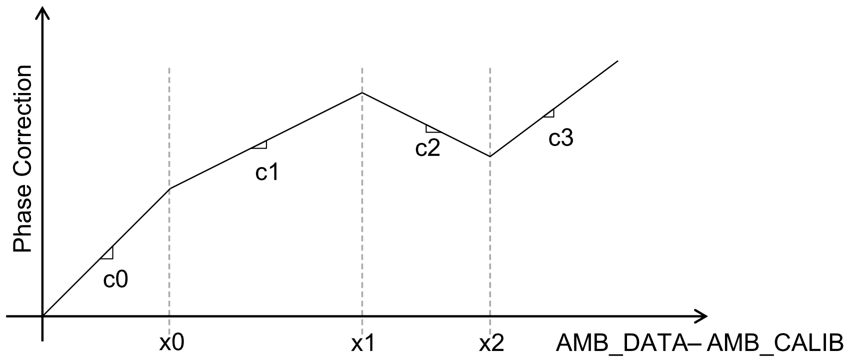

Figure 8. PWL Based Correction for Ambient Dependent Phase

Table 11. Ambient Dependent Phase Correction Registers

Register		Address	Description
AMB_PHASE_CORR_PWL_X0	X0	B8h[9:0]	First knee point of PWL phase correction with ambient.
AMB_PHASE_CORR_PWL_X1	X1	B9h[19:10]	Second knee point of PWL phase correction with ambient.
AMB_PHASE_CORR_PWL_X2	X2	B9h[9:0]	Third knee point of PWL phase correction with ambient.
AMB_PHASE_CORR_PWL_COEFF0	C0	0Ch[23:16]	Slope of first segment for PWL phase correction with ambient.
AMB_PHASE_CORR_PWL_COEFF1	C1	B4h[7:0]	Slope of second segment for PWL phase correction with ambient.
AMB_PHASE_CORR_PWL_COEFF2	C2	B4h[15:8]	Slope of third segment for PWL phase correction with ambient.
AMB_PHASE_CORR_PWL_COEFF3	C3	B4h[23:16]	Slope of fourth segment for PWL phase correction with ambient.
SCALE_AMB_PHASE_CORR_COEFF		B5h[2:0]	Scaling factor for ambient based PWL phase correction.
AMB_DATA		Ah[11:2]	Ambient ADC output.
AMB_CALIB		Bh[23:14]	Ambient calibration value. Write AMB_DATA measured during the phase offset calibration.

To measure these coefficients, vary the ambient light and capture the phase and ambient ADC output (AMB_DATA). Set the target object such that amplitude is > 10000 during this measurement to improve the measurement accuracy. From this data determine how many segment are required for the best fit and find the slopes of Phase vs AMB_DATA-AMB_CALIB for each segment. AMB_CALIB is the ambient ADC output, measured during phase offset calibration. There is a scaling factor of $2^{(2\ +\ SCALE\_AMB\_PHASE\_CORR\_COEFF)}$ to improve the accuracy of this correction. Multiply the obtained slope from phase vs ambient with this scaling factor to get the coefficient values to be written to the registers. These coefficients further need to be multiplied with frequency ratio = 16384 / freq_count. The ambient coefficient scaling factor SCALE_AMB_PHASE_CORR_COEFF should be selected such that all the coefficient values fit within 8-bit signed number (–128 to 127).

Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Date	Revision	Notes
February 2018	*	Initial Release.

IMPORTANT NOTICE FOR TI DESIGN INFORMATION AND RESOURCES

Texas Instruments Incorporated ('TI") technical, application or other design advice, services or information, including, but not limited to, reference designs and materials relating to evaluation modules, (collectively, "TI Resources") are intended to assist designers who are developing applications that incorporate TI products; by downloading, accessing or using any particular TI Resource in any way, you (individually or, if you are acting on behalf of a company, your company) agree to use it solely for this purpose and subject to the terms of this Notice.

TI's provision of TI Resources does not expand or otherwise alter TI's applicable published warranties or warranty disclaimers for TI products, and no additional obligations or liabilities arise from TI providing such TI Resources. TI reserves the right to make corrections, enhancements, improvements and other changes to its TI Resources.

You understand and agree that you remain responsible for using your independent analysis, evaluation and judgment in designing your applications and that you have full and exclusive responsibility to assure the safety of your applications and compliance of your applications (and of all TI products used in or for your applications) with all applicable regulations, laws and other applicable requirements. You represent that, with respect to your applications, you have all the necessary expertise to create and implement safeguards that (1) anticipate dangerous consequences of failures, (2) monitor failures and their consequences, and (3) lessen the likelihood of failures that might cause harm and take appropriate actions. You agree that prior to using or distributing any applications that include TI products, you will thoroughly test such applications and the functionality of such TI products as used in such applications. TI has not conducted any testing other than that specifically described in the published documentation for a particular TI Resource.

You are authorized to use, copy and modify any individual TI Resource only in connection with the development of applications that include the TI product(s) identified in such TI Resource. NO OTHER LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE TO ANY OTHER TI INTELLECTUAL PROPERTY RIGHT, AND NO LICENSE TO ANY TECHNOLOGY OR INTELLECTUAL PROPERTY RIGHT OF TI OR ANY THIRD PARTY IS GRANTED HEREIN, including but not limited to any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information regarding or referencing third-party products or services does not constitute a license to use such products or services, or a warranty or endorsement thereof. Use of TI Resources may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

TI RESOURCES ARE PROVIDED "AS IS" AND WITH ALL FAULTS. TI DISCLAIMS ALL OTHER WARRANTIES OR REPRESENTATIONS, EXPRESS OR IMPLIED, REGARDING TI RESOURCES OR USE THEREOF, INCLUDING BUT NOT LIMITED TO ACCURACY OR COMPLETENESS, TITLE, ANY EPIDEMIC FAILURE WARRANTY AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF ANY THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

TI SHALL NOT BE LIABLE FOR AND SHALL NOT DEFEND OR INDEMNIFY YOU AGAINST ANY CLAIM, INCLUDING BUT NOT LIMITED TO ANY INFRINGEMENT CLAIM THAT RELATES TO OR IS BASED ON ANY COMBINATION OF PRODUCTS EVEN IF DESCRIBED IN TI RESOURCES OR OTHERWISE. IN NO EVENT SHALL TI BE LIABLE FOR ANY ACTUAL, DIRECT, SPECIAL, COLLATERAL, INDIRECT, PUNITIVE, INCIDENTAL, CONSEQUENTIAL OR EXEMPLARY DAMAGES IN CONNECTION WITH OR ARISING OUT OF TI RESOURCES OR USE THEREOF, AND REGARDLESS OF WHETHER TI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

You agree to fully indemnify TI and its representatives against any damages, costs, losses, and/or liabilities arising out of your non-compliance with the terms and provisions of this Notice.

This Notice applies to TI Resources. Additional terms apply to the use and purchase of certain types of materials, TI products and services. These include; without limitation, TI's standard terms for semiconductor products http://www.ti.com/sc/docs/stdterms.htm), evaluation modules, and samples (http://www.ti.com/sc/docs/sampterms.htm).