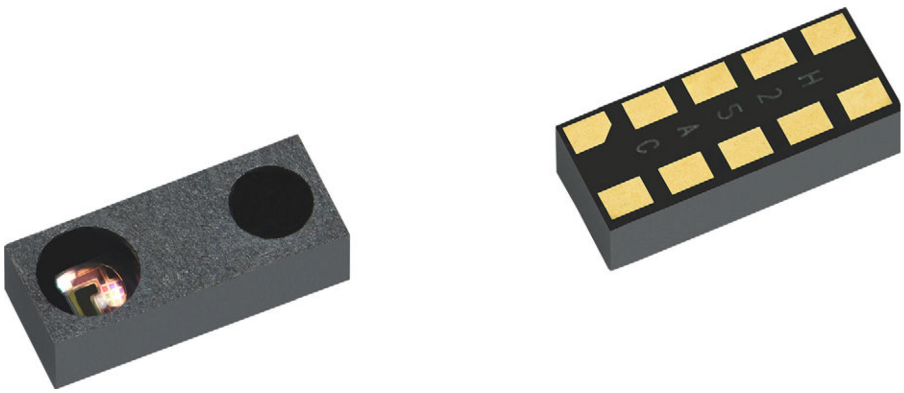
VISHAY SEMICONDUCTORS

www.vishay.com

Optical Sensors | Application Note

Designing the VCNL36758 Into an Application

By Jerry Liu



ABSTRACT

This application note provides an introduction to the functionality of the VCNL36758 sensor, application circuits, and mechanical design considerations.

CONTACT

consumersensors@vishay.com

Please visit our website at www.vishay.com/optoelectronics/ for more information.

APPLICATION NOTE

Designing the VCNL36758 Into an Application

CONTENTS

Designing the VCNL36758 Into an Application

1. INTRODUCTION

The VCNL36758 integrates a high sensitivity proximity sensor (PS) and an IR LED into one single package. It incorporates a photodiode, amplifiers, and analog / digital circuits into a single chip by the CMOS process. The VCNL36758 has been developed for long range operation up to 60 cm. The sensor is intended for longer distance applications, such as:

- Presence detection to activate displays in printers, copiers, tablets, and home appliances
- Collision detection in robots and toys
- Proximity sensing and lighting control in offices, corridors, and public buildings
- Parking space availability in lots and garages
- Proximity detection in lavatory appliances

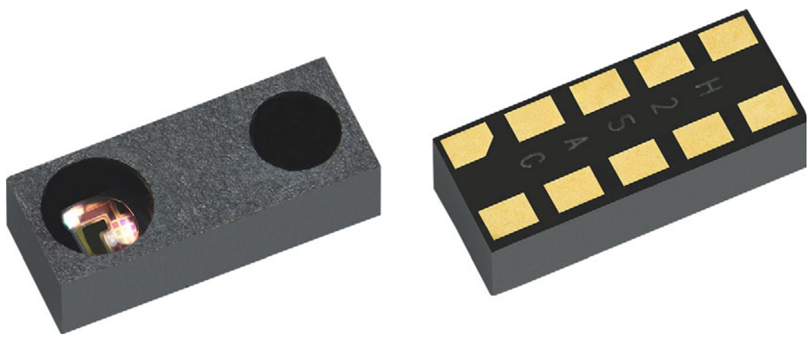


Fig. 1 - VCNL36758

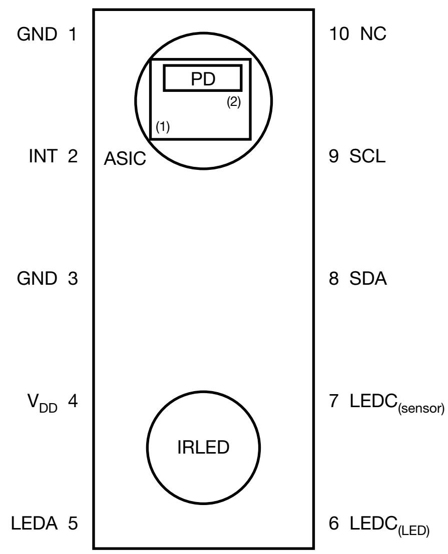
KEY BENEFITS OF USING THE VCNL36758	
BENEFITS	**DESCRIPTION**
Small package for tight space requirements	5.0 x 2.0 x 1.5 (L x W x H in mm) package
Superior proximity detection	The proximity sensor can detect up to a 60 cm distance
	The proximity sensor supports sunlight cancellation up to 100 klux

This application note describes the functionality, application circuits, register settings, and mechanical design considerations for the sensor.

Designing the VCNL36758 Into an Application

2. PIN DESCRIPTION AND BLOCK DIAGRAM

2.1 Pin Description



Notes
(1) ASIC - Application-specific integrated circuit
(2) PD - Photodiode

Fig. 2 - Pin Assignment (top view) of the VCNL36758

TABLE 1 - PIN DESCRIPTION

PIN NUMBER	PIN NAME	TYPE	DESCRIPTION
1	GND	I	Ground
2	INT	O (open drain)	Interrupt
3	GND	I	Ground
4	V_{DD}	I	Supply voltage
5	LEDA	I	LED anode
6	$LEDC_{(LED)}$	I	LED cathode
7	$LEDC_{(sensor)}$	I	Sensor cathode
8	SDA	I / O (open drain)	I^2C serial data
9	SCL	I (open drain)	I^2C serial clock
10	NC	-	No connection

Designing the VCNL36758 Into an Application

2.2 Block Diagram

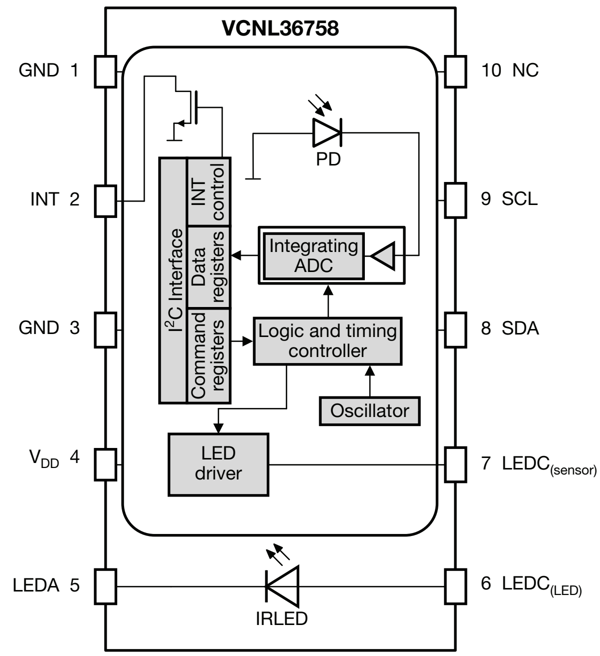


Fig. 3 - Block Diagram of the VCNL36758

TABLE 2 - BLOCK DIAGRAM DESCRIPTION

COMPONENT	DESCRIPTION
Command registers	Command registers are the memory storage for writing and reading I^2C commands
I^2C interface	The I^2C interface is a communication interface with active low open drain circuitry
Interrupt control	Interrupt control is a circuit block with active low open drain output
PD	PD is a photodiode that converts the reflected infrared signal from the object into photocurrent; it is then fed to the 12-bit ADC
12-bit ADC	The 12-bit analog to digital converter converts the analog signal to the digital signal; the input signal is then amplified
Logic and timing controller	The logic and timing controller controls the timing for the proximity measurement
LED driver	The LED driver is a circuitry block that limits the driving current based on the selected setting
Oscillator	The oscillator generates the clock signal to synchronize all of the device functionalities

Designing the VCNL36758 Into an Application

3. APPLICATION CIRCUIT

3.1 Application Circuit With a Single Device

Fig. 4 shows application circuit examples with a single device. When pins 8 and 9 are connected to the clock and data signal from the microcontroller, as shown in Fig. 4, they will then be configured as an SCL pin and SDA pin, respectively. The 7-bit slave address option of 0x60 will be automatically selected.

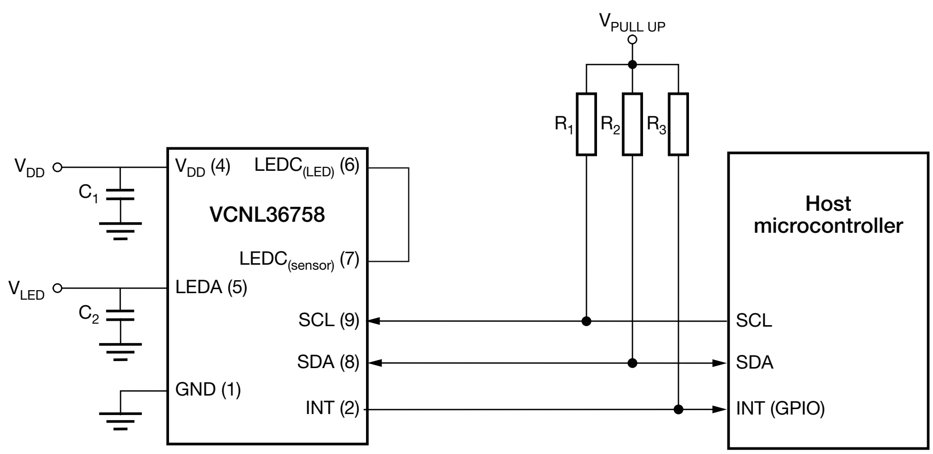


Fig. 4 - Application Circuit Example for VCNL36758 - Slave Address 0x60

Table 4 shows the required values and the explanation for the individual application circuit parameters.

TABLE 4 - APPLICATION CIRCUIT PARAMETERS

CIRCUIT PARAMETER	VALUE	DESCRIPTION
V_{DD}	1.7 V to 3.6 V	A stable power supply such as a low dropout regulator or a switching regulator is required; the power supply isolation can be further improved with a decoupling capacitor C_1
V_{LED}	2.9 V to 4.8 V	A stable power supply such as a low dropout regulator or a switching regulator that can supply an adequate amount of power (max. IRLED pulse driving current of 240 mA) is required; the power supply isolation can be further improved with a decoupling capacitor C_2; the minimum voltage can support the selected driving current of the IR LED
$V_{PULL\ UP}$	1.7 V to 3.6 V	A stable power supply such as a low dropout regulator or a switching regulator is required; a voltage level shifter is required if the I^2C bus voltage from the microcontroller is higher than 3.6 V
C_1 to C_2	100 nF to 1 µF	Decoupling capacitors are recommended to reduce the noise in the supply voltage
R_1 to R_2	2.2 kΩ to 4.7 kΩ	Pull-up resistors within the range of 2.2 kΩ to 4.7 kΩ are recommended; any increase in bus capacitance or resistance will increase the logic-high transition time
R_3	4.7 kΩ to 22 kΩ	Pull-up resistor within the range of 4.7 kΩ to 22 kΩ is recommended

Designing the VCNL36758 Into an Application

4. REGISTER DESCRIPTION

TABLE 5 - REGISTER DESCRIPTION OVERVIEW					
REGISTER GROUP	BIT NAME	REGISTER NAME	COMMAND CODE	DESCRIPTION	ACCESS
Basic initialization	PS_CAL	PS_CONF1_L	0x00	Enable / disable the internal calibration	Write and read
	PS_ON			Set this bit = "1" to enable bias circuit	
	PS_SD	PS_CONF2_L	0x03	PD shutdown setting	
	PS_MODE	PS_CONF3_L	0x04	Set the mode of the sensor to either auto or active force mode	
	PS_TRIG			Set the bit to 1 to trigger an individual active force mode measurement	
	PS_OFFSET			Enable / disable the internal crosstalk cancellation	
Emitter settings (LED)	PS_IT	PS_CONF2_H	0x03	Set the integration time for one measurement pulse	
	PS_MPS			Set the number of infrared measurement pulses	
	PS_CURRENT	PS_CONF3_H	0x04	Set the LED driving current	
Detector settings	PS_HG	PS_CONF2_H	0x03	Set the digital gain in the ADC	
Measurement period / rate	PS_PERIOD	PS_CONF2_L	0x03	Set the measurement period	
Offset count cancellation	-	PS_CANC	0x07	Set the offset count cancellation value	
Sunlight cancellation	PS_SC_EN	PS_CONF3_H	0x04	Enable / disable the sunlight cancellation	
	PS_SC_LEVEL			Sunlight cancellation level setting	
Interrupt	PS_PERS	PS_CONF2_L	0x03	Set the interrupt persistence number	
	PS_INT			Set the interrupt mode setting	
	PS_SMART_PERS			Enable / disable the smart persistence	
	PS_SSINT	PS_CONF2_H		Set the interrupt mode setting	
	PS_SP_INT	PS_CONF3_L	0x04	Enable / disable the sunlight protection mode interrupt	
	-	PS_THDL	0x05	Set the low threshold interrupt value	
	-	PS_THDH	0x06	Set the high threshold interrupt value	
	PS_SPFLAG	INT_FLAG	0xF5	Sunlight protection mode interrupt event flag	
	PS_IF_CLOSE			High threshold crossing interrupt event flag	Read only
	PS_IF_AWAY			Low threshold crossing interrupt event flag	
Readout registers	-	PS_DATA	0xF4	Proximity output data	
	VCNL36758_ID_L	VCNL36758_ID	0xF6	Slave address 0x60: ID = 0x58	
	VCNL36758_ID_H			ID = 0x01	

Designing the VCNL36758 Into an Application

4.1 Basic Initialization

The sensor can be initialized with the PS_CAL, PS_ON, PS_MODE, PS_TRIG, PS_SD, and PS_OFFSET bits, which are found in three different registers: PS_CONF1_L, PS_CONF2_H, and PS_CONF3_L.

TABLE 6 - BASIC INITIALIZATION

REGISTER GROUP	BIT NAME	REGISTER NAME	COMMAND CODE	FUNCTION	BIT	VALUE	DESCRIPTION
Basic initialization	PS_CAL	PS_CONF1_L	0x00	Enable / disable the internal calibration	7	0x0 (0b0)	Disable (default)
						0x1 (0b1)	Enable
	PS_ON			Set this bit = "1" to enable bias circuit	1	0x0 (0b0)	Turn off the sensor (shutdown) (default)
						0x1 (0b1)	Turn on the sensor
	PS_SD	PS_CONF2_L	0x03	PS shutdown setting	0	0x0 (0b0)	Turn on the sensor
						0x1 (0b1)	Shutdown (default)
	PS_MODE	PS_CONF3_L	0x04	Set the mode of the sensor to either auto or active force mode	7	0x0 (0b0)	Auto mode (default)
						0x1 (0b1)	Active force mode
	PS_TRIG			Set the bit to 1 to trigger an individual active force mode measurement	5	0x0 (0b0)	Off (default)
						0x1 (0b1)	Trigger
	PS_OFFSET			Enable / disable the internal crosstalk cancellation	3	0x0 (0b0)	Disable (default)
						0x1 (0b1)	Enable

PS_CAL - Set the sensor to factory calibration values, when set to "1", ensuring original accuracy and stability. This bit will automatically return to "0".

PS_ON - The sensor can be turned on by setting this bit to 1 and turned off by setting it to 0. The sensor will mostly be in the idle state and only be in the active state during the measurement phase.

PS_MODE - Set this bit to 0 to activate the auto mode and 1 to activate the active force mode. Auto mode means the measurement will be triggered continuously with a rate defined by the measurement period. This measurement period depends on the PS_PERIOD setting. Active force mode means the measurement will have to be triggered manually by setting the PS_TRIG bit to 1.

PS_TRIG - This bit must be set to 1 when using active force mode to trigger a measurement. Otherwise, the bit must be set to 0.

PS_OFFSET - This bit can be set to 1 during initialization to perform internal crosstalk cancellation. The internal crosstalk is the crosstalk between the infrared signal from the LED and the photodiode in the open air (without a window cover). The signal from the internal crosstalk is measured in the final test and is stored internally. If this bit is enabled, the count from the crosstalk will be deducted. This crosstalk is usually between 5 to 20 counts.

PS_SD - The sensor can be turned on by setting this bit to 0 and turned off by setting it to 1.

Fig. 5 and Fig. 6 show the basic initialization of the two available modes. The basic initialization steps are usually useful when first testing with the proximity sensor. The proximity measurement will be based on the default values. In practice, more sensor parameters to change the strength of the LED infrared signal, the gain of the analog-digital converter (ADC), the measurement period, and the interrupt should be set before starting the measurement.

Designing the VCNL36758 Into an Application

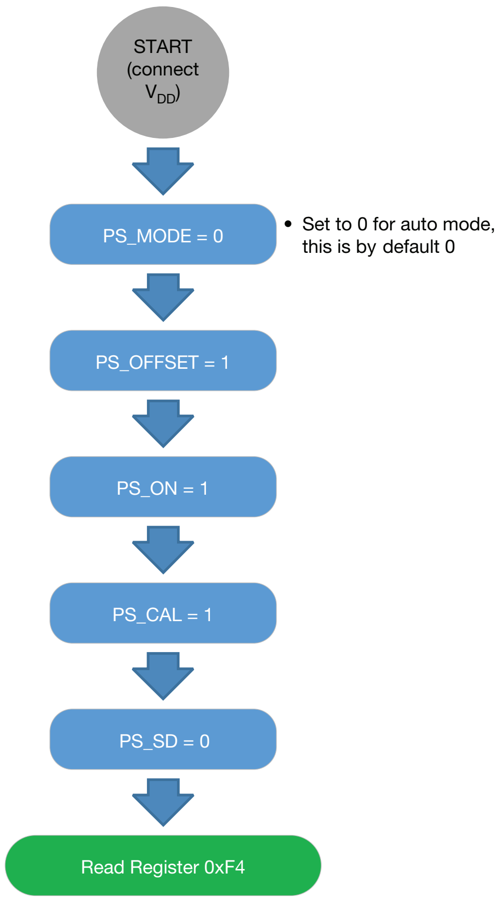


Fig. 5 - Basic Initialization Example Steps for Auto Mode

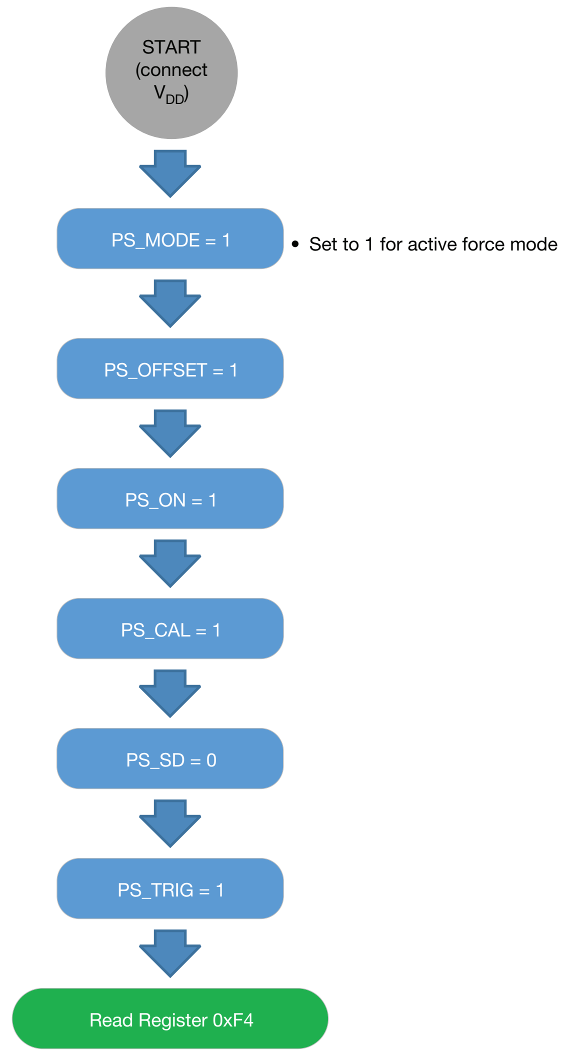


Fig. 6 - Basic Initialization Example Steps for Active Force Mode

Designing the VCNL36758 Into an Application

4.2 Emitter Settings (IRLED)

TABLE 8 - EMITTER SETTINGS (IRLED)

REGISTER GROUP	BIT NAME	REGISTER NAME	COMMAND CODE	FUNCTION	BIT	VALUE	DESCRIPTION
Emitter settings (IRLED)	PS_IT	PS_CONF2_H	0x03	Set the integration time for one measurement pulse, the pulse length "T" is 50 μs	15 : 14	0x0 (0b00)	1T (default)
						0x1 (0b01)	2T
						0x2 (0b10)	4T
						0x3 (0b11)	8T
	PS_MPS			Set the number of infrared measurement pulses	13 : 12	0x0 (0b00)	1 pulse (default)
						0x1 (0b01)	2 pulses
						0x2 (0b10)	4 pulses
						0x3 (0b11)	8 pulses
	PS_CURRENT	PS_CONF3_H	0x04	Set the IRLED driving current	10 : 8	0x0 (0b000)	70 mA (default)
						0x1 (0b001)	95 mA
						0x2 (0b010)	110 mA
						0x3 (0b011)	130 mA
						0x4 (0b100)	170 mA
						0x5 (0b101)	200 mA
						0x6 (0b110)	220 mA
						0x7 (0b111)	240 mA

PS_CURRENT - The magnitude of the driving current of the IRLED pulse.

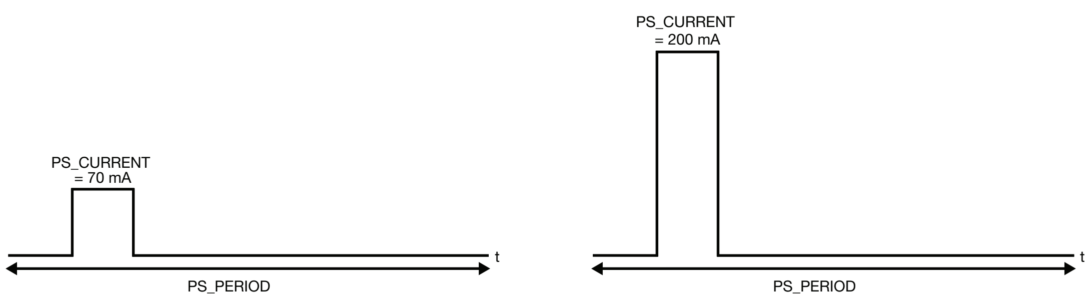


Fig. 7 - The Behavior of the Pulse Magnitude With Different PS_CURRENT

Designing the VCNL36758 Into an Application

PS_IT - The length of the IRLED pulse width with multiple values of T. The length of T is 50 μs.

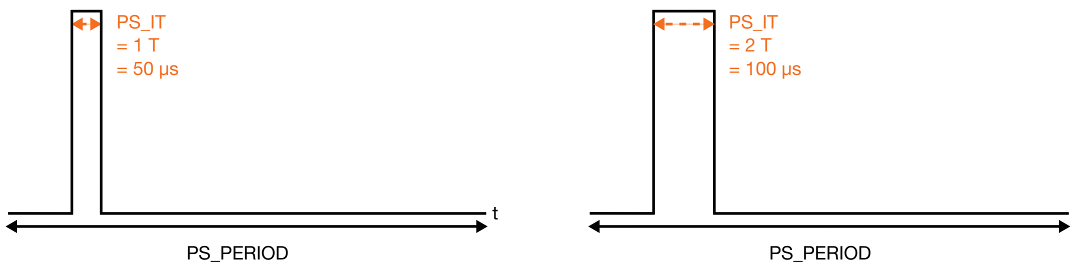


Fig. 8 - The Behavior of the Pulse Length With Different PS_IT

PS_MPS - The number of IRLED pulses within one measurement period.

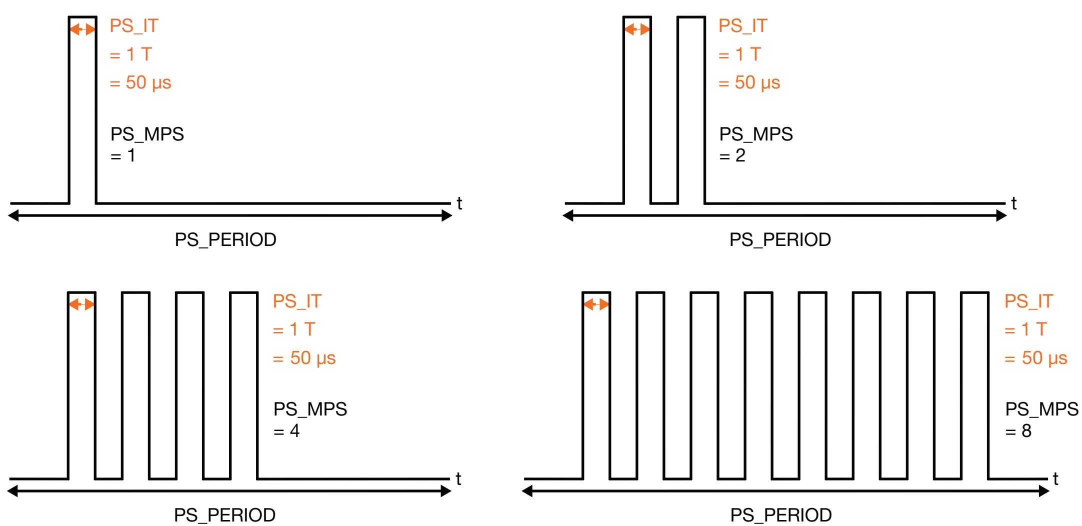


Fig. 9 - The Number of Pulses' Behavior With Different PS_MPS

Designing the VCNL36758 Into an Application

4.3 Detector Settings

TABLE 9 - DETECTOR SETTINGS							
REGISTER GROUP	BIT NAME	REGISTER NAME	COMMAND CODE	FUNCTION	BIT	VALUE	DESCRIPTION
Detector settings	PS_HG	PS_CONF2_H	0x03	Set the digital gain in the ADC	10	0x0 (0b0)	x 1 gain (default)
						0x1 (0b1)	x 2 gain

The detector of the proximity sensor consists of the photodiode and the ADC. The detector-related parameters PS_HG in register PS_CONF2_H double the magnitude of proximity counts given the same reflected infrared signal magnitude from the object arrived at the detector.

PS_HG - The counts can be doubled when the bit is set to 1.

4.4 Measurement Period / Rate

TABLE 10 - MEASUREMENT PERIOD / RATE							
REGISTER GROUP	BIT NAME	REGISTER NAME	COMMAND CODE	FUNCTION	BIT	VALUE	DESCRIPTION
Measurement period / rate	PS_PERIOD	PS_CONF2_L	0x03	Set the measurement period	7 : 6	0x0 (0b00)	10 ms (default)
						0x1 (0b01)	20 ms
						0x2 (0b10)	40 ms
						0x3 (0b11)	80 ms

PS_PERIOD defines the period at which the infrared signal pulses are executed to perform proximity measurements. In most applications, the measurement period between 80 ms is enough to find a compromise between acceptable performance and low power consumption.

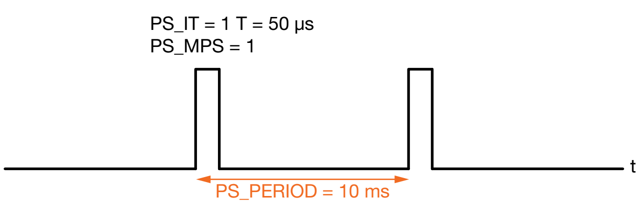


Fig. 10 - LEDA Pin for the Measurement Period of 10 ms

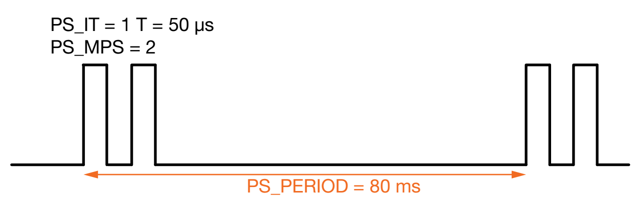


Fig. 11 - LEDA Pin for the Measurement Period of 80 ms

Designing the VCNL36758 Into an Application

4.5 Offset Count Cancellation

TABLE 11 - OFFSET COUNT CANCELLATION							
REGISTER GROUP	BIT NAME	REGISTER NAME	COMMAND CODE	FUNCTION	BIT	VALUE	DESCRIPTION
Offset count cancellation	PS_CANC_L	PS_CANC	0x07	Set the offset count cancellation value	7 : 0	0 to 4095	Low byte
	PS_CANC_H				11 : 8		High byte

Crosstalk between the infrared signal from the IRLED and the photodiode happens when introducing the window cover. This is because part of the light will be reflected back to the sensor due to fresnel reflection. Therefore, it is recommended to perform offset count cancellation by writing the PS_CANC register with the offset counts. Fig. 12 shows the offset count calibration by measuring the offset counts due to the window cover, internal crosstalk, and noise. These offset counts can then be written to the register PS_CANC, as shown in Fig. 13. By using this approach, the change in counts is directly influenced by the object and the effect of the part to part tolerance can be reduced.

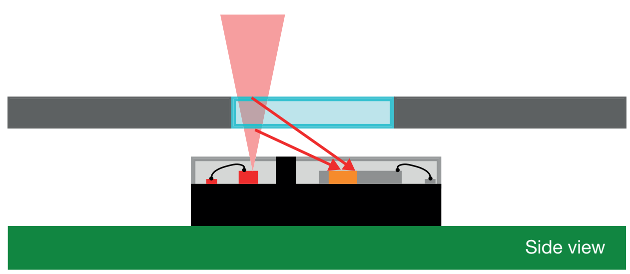


Fig. 12 - Offset Count Calibration

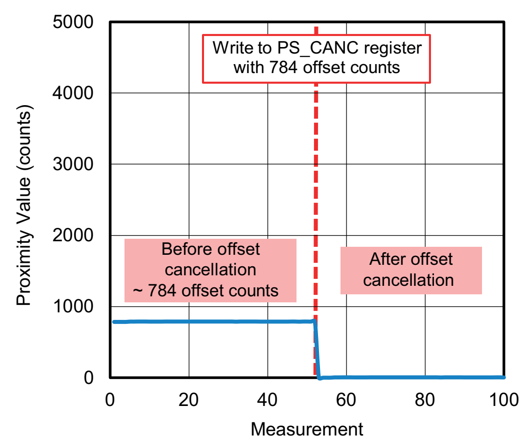


Fig. 13 - Offset Count Cancellation

Designing the VCNL36758 Into an Application

4.6 Sunlight Cancellation

TABLE 12 - SUNLIGHT CANCELLATION

REGISTER GROUP	BIT NAME	REGISTER NAME	COMMAND CODE	FUNCTION	BIT	VALUE	DESCRIPTION
Sunlight cancellation	PS_SC_EN	PS_CONF3_H	0x04	Enable / disable the sunlight cancellation	15	0x0 (0b0)	Disable (default)
						0x1 (0b1)	Enable
	PS_SC_LEVEL			Sunlight cancellation level setting	14 : 13	0x0 (0b00)	LEVEL1 (default)
						0x1 (0b01)	LEVEL2
						0x2 (0b10)	LEVEL3
						0x3 (0b11)	LEVEL4

DC ambient light sources in the wavelength region between 800 nm and 1200 nm, such as sunlight and halogen lights, cause disturbances in the photodiode and the ADC circuitry. Therefore, the sensor needs to perform cancellation to reduce the noise from these disturbances. The sunlight cancellation PS_SC bit can be enabled to allow the sensor to measure the photocurrent contribution from the DC ambient light sources before driving the infrared signal pulse during the proximity measurement. After the proximity measurement, the sensor deducts the DC noise photocurrent from the total photocurrent. As a result, only the photocurrent due to the reflected signal from the object is converted by the ADC circuitry. This sunlight cancellation mechanism is depicted in Fig. 14.

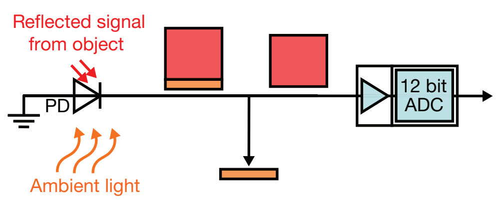


Fig. 14 - Sunlight Cancellation by Using the Active Sunlight Cancellation Current in the VCNL36758

The sensor can perform the cancellation of sunlight up to 100 klux. The sensor then goes into the sunlight protection mode beyond 100 klux of sunlight. Sunlight protection mode is a mode where the photodiode is in saturation due to high sunlight illuminance. Therefore, the sensor can no longer detect the object and will only output 0 in the PS_DATA register. This can be observed in Fig. 15. If the sunlight protection mode interrupt PS_SP_INT is enabled, the sensor will pull the interrupt line low each time the sensor goes into sunlight protection mode. As a result, the sunlight protection mode interrupt flag PS_SPFLAG changes from 0 to 1. This interrupt flag can be cleared by reading the INT_FLAG register. Therefore, the application should ignore the PS_DATA when it remains 0 and PS_SPFLAG consistently changes to 1 after clearing the INT_FLAG register.

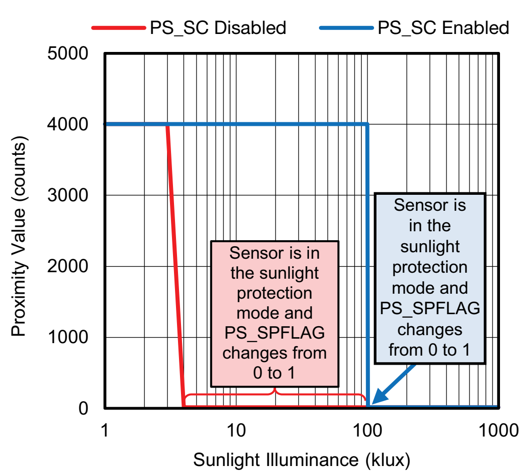


Fig. 15 - The Overall Behavior of the Sunlight Cancellation in the Proximity Sensor With Increasing Sunlight Illuminance

APPLICATION NOTE

When PS_SC_EN is set to 1, PS_SC_LEVEL can be used to select different sunlight cancellation levels. The table below shows the capability of each sunlight cancellation setting.

TABLE 13 - THE MAXIMUM SUNLIGHT LUX LOAD FOR SUN_LEVEL	
Disable	4 klux
LEVEL1	12 klux
LEVEL2	24 klux
LEVEL3	42 klux
LEVEL4	100 klux

If the PS_SC bit is disabled, the sensor goes into the sunlight protection mode when the illuminance of the sunlight is beyond 4 klux. This can be observed in Table 13.

4.7 Interrupt

TABLE 14 - INTERRUPT

REGISTER GROUP	BIT NAME	REGISTER NAME	COMMAND CODE	FUNCTION	BIT	VALUE	DESCRIPTION
Interrupt	PS_PERS	PS_CONF2_L	0x03	Set the interrupt persistence number	5 : 4	0x0 (0b00)	1 time (default)
						0x1 (0b01)	2 times
						0x2 (0b10)	3 times
						0x3 (0b11)	4 times
	PS_INT			Set the interrupt mode setting	3 : 2	0x0 (0b00)	Interrupt disable (default)
						0x1 (0b01)	Logic high / low mode
						0x2 (0b10)	First high
						0x3 (0b11)	Interrupt disable
	PS_SMART_ PERS			Enable / disable the smart persistence	1	0x0 (0b0)	Disable (default)
						0x1 (0b1)	Enable
	PS_SSINT	PS_CONF2_H		Trigger by each high / low threshold event	8	0x0 (0b0)	Disable (default)
						0x1 (0b1)	Enable
	PS_SP_INT	PS_CONF3_L	0x04	Enable / disable the sunlight protection mode interrupt setting	2	0x0 (0b0)	Disable (default)
						0x1 (0b1)	Enable
	PS_THDL_L	PS_THDL	0x05	Set the low threshold interrupt value	7 : 0	0 to 4095	Low byte
	PS_THDL_H				11 : 8		High byte
	PS_THDH_L	PS_THDH	0x06	Set the high threshold interrupt value	7 : 0	0 to 4095	Low byte
	PS_THDH_H				11 : 8		High byte
	PS_SPFLAG	INT_FLAG	0xF5	Read the sunlight protection mode interrupt event flag	12	0x0 (0b0)	No sunlight protection mode interrupt
						0x1 (0b1)	Sunlight protection mode interrupt event flag
	PS_IF_ CLOSE			Read the high threshold crossing interrupt event flag	9	0x0 (0b0)	No high threshold crossing interrupt event flag
						0x1 (0b1)	High threshold crossing interrupt event flag
	PS_IF_ AWAY			Read the low threshold crossing interrupt event flag	8	0x0 (0b0)	No low threshold crossing interrupt event flag
						0x1 (0b1)	Low threshold crossing interrupt event flag

Designing the VCNL36758 Into an Application

The interrupt pin allows the proximity sensor to autonomously send an interrupt signal to the microcontroller when the sensor reading exceeds the high threshold or falls below the low threshold. There are three modes of interrupt in the VCNL36758:

- Logic high / low mode
- Trigger by each high / low threshold event
- First high

Besides that, the proximity sensor also provides persistence and smart persistence features, which prevent the occurrence of false detection. The interrupt will only be triggered after a defined consecutive threshold crossing event.

Logic High / Low Mode

In this mode, both PS_INT = 1 and PS_SSINT = 0 must be set:

- The interrupt line is pulled low when the proximity counts cross the high threshold, as indicated by the blue line in Fig. 16
- The interrupt line is pulled high when the proximity counts cross the low threshold, as indicated by the purple line in Fig. 16
- If the count is equal to the threshold, it cannot be triggered, as shown by the orange circles in Fig. 16
- Consecutive high threshold events cannot be triggered until the proximity count crosses the low threshold first, and vice versa, as shown by the green circles in Fig. 16
- Logic mode allows the proximity sensor to autonomously send an interrupt signal to the microcontroller directly without the need to read the INT_FLAG register

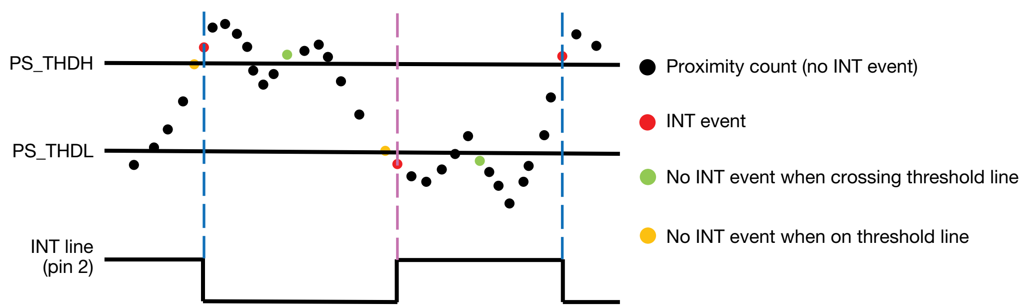


Fig. 16 - The Behavior of the INT Line (pin 2) Using Logic High / Low Mode With PS_PERS = 1, PS_INT = 1, and PS_SSINT = 0

Designing the VCNL36758 Into an Application

First High

In this mode, both PS_INT = 2 and PS_SSINT = 0 must be set:

- The interrupt line is pulled low when the proximity counts cross the high or low threshold, as indicated by the red circle in Fig. 17
- Consecutive events can be triggered until the proximity count crosses the threshold first, as shown by the green circles in Fig. 17
- If the count is equal to the threshold, it cannot be triggered, as shown by the orange circles in Fig. 17
- The first interrupt event can be either a high or low threshold crossing event
- The interrupt flags PS_IF_CLOSE and PS_IF_AWAY change from 0 to 1 when the proximity counts cross the high and low thresholds, respectively
- The interrupt flag PS_IF_CLOSE or PS_IF_AWAY can be cleared by reading the INT_FLAG register, as indicated by the blue line in Fig. 17

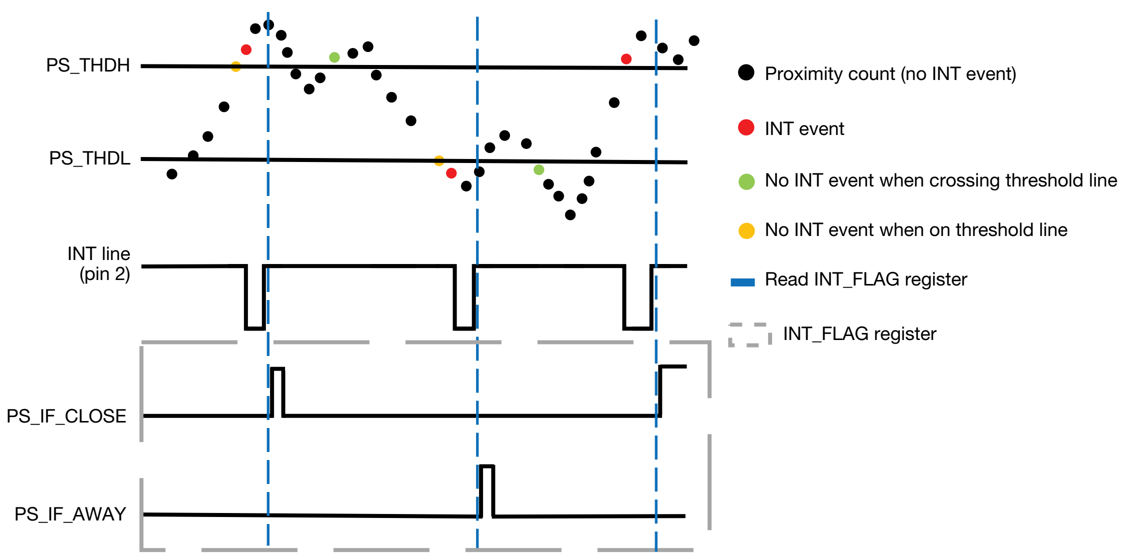


Fig. 17 - The Behavior of the INT Line (pin 2) and INT_FLAG Register Using the First High With PS_PERS = 1, PS_INT = 2, and PS_SSINT = 0

Trigger by Each High / Low Threshold Event

In this mode, both PS_INT = 2 and PS_SSINT = 1 must be set:

- The interrupt line is pulled low when the proximity counts cross the high or low threshold, as indicated by the red circle in Fig. 18
- If the count is equal to the threshold, it cannot be triggered, as shown by the orange circles in Fig. 18
- The first interrupt event can be either a high or low threshold crossing event
- The interrupt flags PS_IF_CLOSE and PS_IF_AWAY change from 0 to 1 when the proximity counts cross the high and low thresholds, respectively
- The interrupt flag PS_IF_CLOSE or PS_IF_AWAY can be cleared by reading the INT_FLAG register, as indicated by the blue line in Fig. 18

Designing the VCNL36758 Into an Application

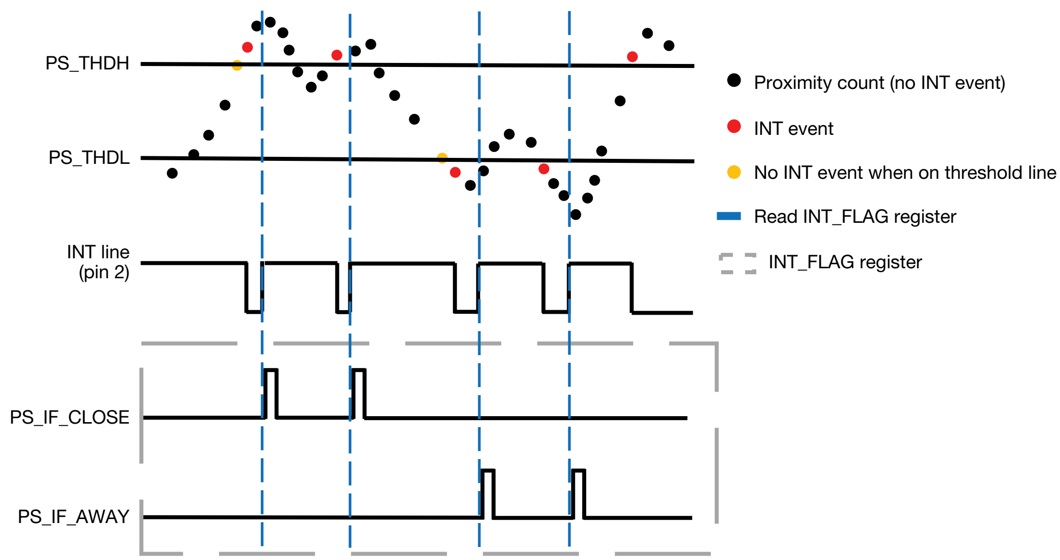


Fig. 18 - The Behavior of the INT Line (pin 2) and INT_FLAG Register Using the Trigger by Each High / Low Threshold Event With PS_PERS = 1, PS_INT = 2, and PS_SSINT = 1

Persistence and Smart Persistence Mode

PS_PERS determines the number of consecutive threshold-crossing events required to trigger an interrupt. PS_PERS can be set between 2 and 4; for instance, when PS_PERS is set to 4, after the initial threshold-crossing event, three additional measurements are taken to confirm the persistence of the crossing, thereby reducing false detections. The interrupt is triggered only after the number of consecutive threshold-crossing events specified by the PS_PERS setting has been reached. This applies to all interrupt modes. Fig. 19 illustrates the behavior of the INT line (pin 2), with the measurement period defined by the red line formula, where the period equals PS_PERIOD (ms). Fig. 19 also shows how the INT line performs under different PS_PERS settings. For example, when PS_PERS is set to 3, the interrupt is pulled low after the second threshold-crossing measurement following the initial threshold-crossing event. The same logic applies to other PS_PERS settings.

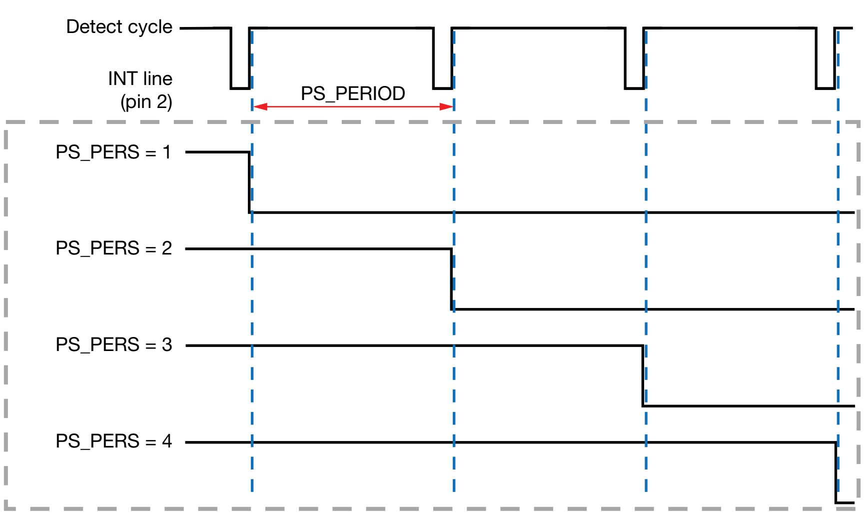


Fig. 19 - The Behavior of the INT Line (pin 2), Disabled PS_SMART_PERS, and all PS_PERS

Designing the VCNL36758 Into an Application

When smart persistence (PS_SMART_PERS) is enabled, after the first threshold event occurs, three consecutive rapid measurements are taken. After the third measurement, the system resumes its regular measurement period. The intervals between these measurements are depicted by the green line in Fig. 20, and the interval is about 2 ms. Fig. 20 shows the behavior of the INT line (pin 2), where the measurement period follows the red line, with the period equal to PS_PERIOD (ms). Fig. 20 also illustrates how the INT line performs under different PS_PERS settings. For example, when PS_PERS is set to 3, the interrupt is pulled low after the third consecutive rapid measurement that crosses the threshold. The same logic applies to other PS_PERS settings.

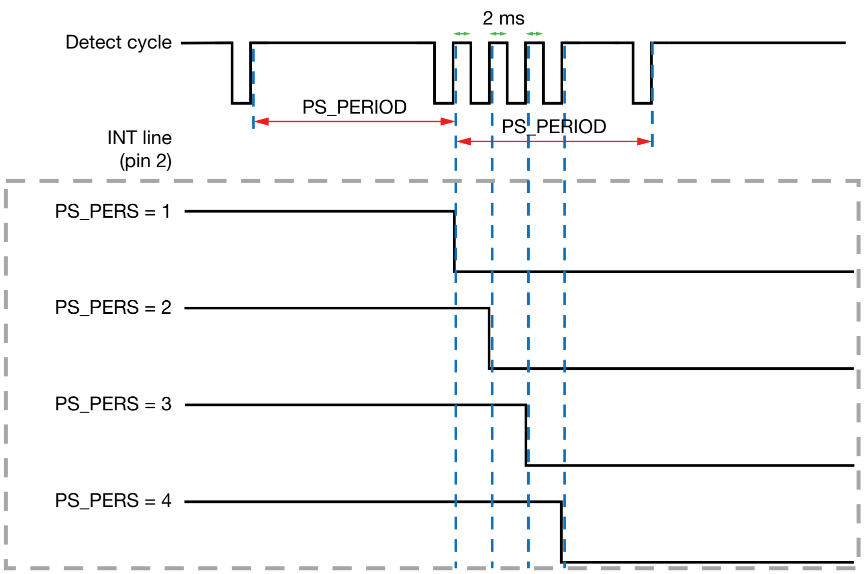


Fig. 20 - The Behavior of the INT Line (pin 2), Enabled PS_SMART_PERS, and all PS_PERS

PS INT Response Time

Disabled PS_SMART_PERS:

$$\text{Response time (ms)} = \text{PS\_PERS} \times (\text{PS\_PERIOD})$$

Enabled PS_SMART_PERS:

$$\text{Response time (ms)} = (\text{PS\_PERS - 1}) \times (2 \times \text{PS\_IT}) + (\text{PS\_PERIOD})$$

Sunlight Protection Mode

The sensor enters the sunlight protection mode when the photodiode is in saturation due to very high sunlight illuminance. Therefore, the sensor can no longer detect the object and will only output 0 in the PS_DATA register. The application can detect this situation by enabling PS_SP_INT and reading the PS_SPFLAG when the interrupt line has been pulled low and PS_DATA suddenly goes into 0 to confirm this situation. This is explained in section 4.6.

Designing the VCNL36758 Into an Application

4.8 Readout Registers

TABLE 15 - READOUT REGISTERS

REGISTER GROUP	BIT NAME	REGISTER NAME	COMMAND CODE	FUNCTION	BIT	VALUE	DESCRIPTION
Readout registers	PS_DATA_L	PS_DATA	0xF4	Read the proximity output data	7 : 0	0 to 4095	Low byte
	PS_DATA_H				11 : 8		High byte
	VCNL36758_ID_L	VCNL36758_ID	0xF6	Read the device ID	7 : 0	0x58 (0b01011000)	Device with a slave address of 0x60
	VCNL36758_ID_H				15 : 8	0x01 (0b00000001)	Should be kept default

The VCNL36758 has two readout registers, which are PS_DATA and VCNL36758_ID. PS_DATA is the 12-bit register in which the proximity counts data will be stored after each measurement. The register can be read via the I^2C communication from a microcontroller. There are two methods of reading the register PS_DATA:

- Data polling - continuously reading the register PS_DATA
- Interrupt - read the register PS_DATA after the INT line (pin 2) has been triggered

Depending on the application's requirements, the interrupt method is recommended to reduce power consumption.

On the other hand, VCNL36758_ID is the register in which the device ID is stored. This allows the sensor to be identified with a specific slave address depending on the connection of pins 8 and 9 of the sensor with the microcontroller. Therefore, the register VCNL36758_ID can be a good first register to read when first communicating with the sensor to test the I^2C communication and to check the slave address.

Designing the VCNL36758 Into an Application

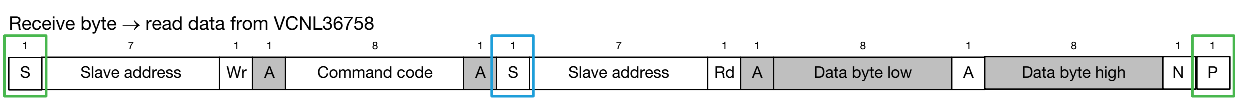
5. I²C AND TIMING

5.1 I²C Write and Read Protocol

Communication with the VCNL36758 can be performed via I²C. The I²C write and read protocol when communicating with the proximity sensor is shown in Fig. 21.

Fig. 21 - I²C Write and Read Protocol

It is imperative that only the restart condition for the I²C read is implemented instead of the stop and restart condition. For example, Fig. 22 shows a I²C command when reading the register VCNL36758_ID for the VCNL36758 with a slave address of 0x60. Here, the restart condition, indicated by the blue box, has been implemented and the low data byte of 0x58 matches the device ID stated in the datasheet for the VCNL36758 with the slave address of 0x60.



S | 110 0000 | 0 | 0 | 1111 0110 | 0 | S | 110 0000 | 1 | 0 | 0101 1000 | 0 | 0000 0001 | 1 | P

S | 0x60 | 0 | 0 | 0xF6 | 0 | S | 0x60 | 1 | 0 | 0x58 | 0 | 0x01 | 1 | P

Fig. 22 - The Correct I²C Read When Reading the Register VCNL36758_ID With a Restart Condition

5.2 Timing

5.2.1 Proximity Measurement Timing

In auto mode, one complete proximity measurement depends on the PS_PERIOD setting. However, the sensor is only active within a short time.

On the other hand, one complete measurement in the active force mode requires adding both the active measurement period and PS_PERIOD.

One active measurement period for both depends on the register settings PS_IT, PS_MPS, and DC_KILL, as shown below:

$$t_{measurement,\ active} = 1.2 \times (DC\_KILL + (2 \times PS\_IT \times PS\_MPS)) \quad (2)$$

Factor 1.2 is included in the equation above to consider the part to part tolerance of 20 %. On the other hand, the term DC_KILL is the process in which the sensor measures the DC noise in the background before the actual measurement pulse from the LED is emitted. This DC noise signal will then be deducted from the total reflected signal. DC_KILL takes 1 ms to complete. This is shown in Fig. 23.

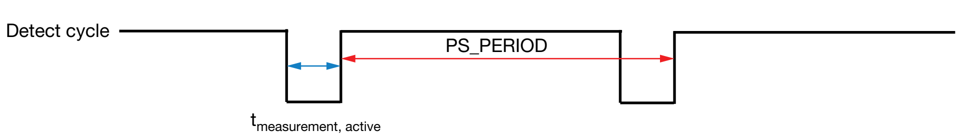


Fig. 23 - The Measurement Time

Designing the VCNL36758 Into an Application

Example - Active Measurement Time PS_MPS = 8

For example, given PS_IT = 1 T = 50 μs, PS_MPS = 8, the total active measurement time can be calculated as follows:

$$t_{measurement,\ active} = 1.2 \text{ x } (1 \text{ ms} + (2 \text{ x } 1 \text{ x } 50 \text{ μs x } 8)) = 2.16 \text{ ms}$$

5.2.2 Timing Specification

Fig. 24 and Fig. 25 show the timing specification for the auto and active force mode, respectively.

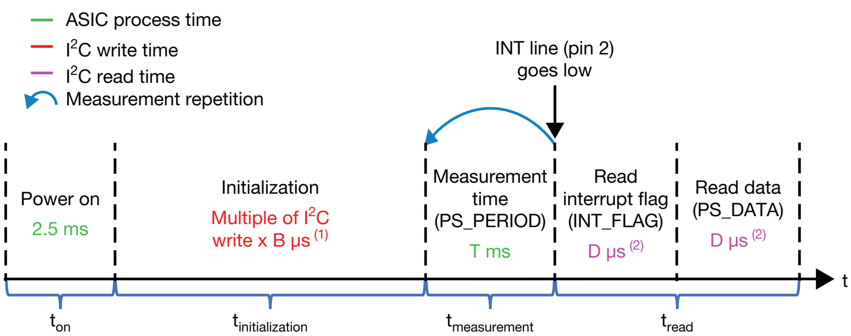


Fig. 24 - Timing Specification for the Auto Mode

Notes

(1) B μs - The parameter B refers to the time taken for a complete write I²C protocol. This depends on the selected I²C mode. Please refer to Table 16

(2) D μs - The parameter D refers to the time taken for a complete read I²C protocol. This depends on the selected I²C mode. Please refer to Table 16

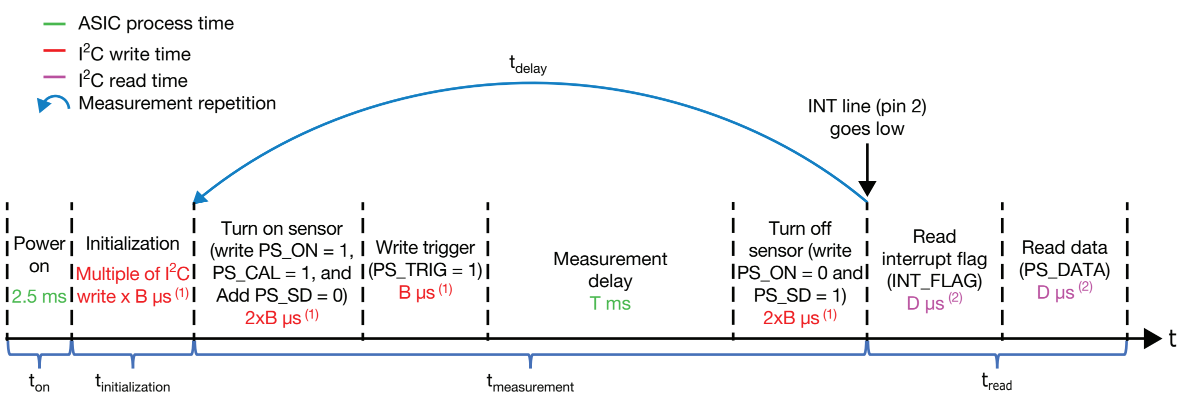


Fig. 25 - Timing Specification for the Active Force Mode

Notes

(3) B μs - The parameter B refers to the time taken for a complete write I²C protocol. This depends on the selected I²C mode. Please refer to Table 16

(4) D μs - The parameter D refers to the time taken for a complete read I²C protocol. This depends on the selected I²C mode. Please refer to Table 16

A measurement delay equal to the active measurement time defined by Equation 2 plus PS_PERIOD is required for the measurement to be completed after the trigger of the active force mode. The delay depends on the register settings of PS_IT, PS_MPS, and PS_PERIOD. The required delay is:

$$\text{Measurement delay} = t_{measurement,\ active} = 1.2 \times (DC\_KILL + (2 \times PS\_IT \times PS\_MPS)) \quad (3)$$

Designing the VCNL36758 Into an Application

Factor 1.2 in the equation is needed to consider the 20 % part to part tolerance of the measurement.

The individual I2C write and read time per byte depends on the I2C mode, as shown in Table 16:

TABLE 16 - I2C WRITE AND READ PROTOCOL TIME PER BYTE			
I2C MODE	CLOCK FREQUENCY (kHz) (1)	I2C WRITE PROTOCOL TIME (μs) (2)	I2C READ PROTOCOL TIME (μs) (2)
Standard	100	400	500
Fast	400	100	125

Notes

(1) Maximum limit

(2) Typical value with tolerance; it could vary depending on the master

Example - Auto Mode With Standard I2C Mode

Table 17 shows an example of the timing specification for the auto mode when initializing five registers, as well as when turning on the sensor. The sensor needs approximately 4.9 ms for the total initialization phase. The exact timing depends on the I2C clock configuration from the master and the number of registers written. On the other hand, the sensor needs one measurement period to complete a measurement. In the case of when a PS_PERIOD of 10 ms has been selected, the measurement time is 10 ms. In many cases, the application uses the interrupt method to read the sensor and reduce power consumption. In this case, once the INT line (pin 2) has been pulled low, the interrupt service routine in the microcontroller will be executed and the microcontroller will then read the register INT_FLAG to clear the interrupt flag. The register PS_DATA will then be read. Both reading processes will each need 0.5 ms for the standard I2C mode. Therefore, the total measurement time is 11 ms.

TABLE 17 -TIMING SPECIFICATION EXAMPLE FOR AUTO MODE AND STANDARD I2C MODE (1)			
PARAMETER	REGISTER	REMARKS	TIME (ms)
t_{on}	-	The power on reset after the V_{DD} pin is connected to the power supply	2.5
$t_{initialization}$	PS_CONF1_L/H	Write the sensor settings	0.4
	PS_CONF2_L/H		0.4
	PS_CONF3_L/H		0.4
	PS_THDL	Write the low threshold setting	0.4
	PS_THDH	Write the high threshold setting	0.4
	PS_CONF1_L/H	Turn on the sensor by writing PS_ON = 1 and PS_CAL = 1	0.4
$t_{initialization\_total}$	-	-	4.9
$t_{measurement}$	-	Actual measurement time depends on the selected PS_PERIOD; here is an example of when PS_PERIOD = 10 ms	10
t_{read}	INT_FLAG	Read the interrupt flag	0.5
	PS_DATA	Read the proximity data	0.5
$t_{measurement\_total}$	-	-	11

Notes

(1) Standard I2C mode comes with a 100 kHz clock frequency. One can approximate the time taken for a complete write and read I2C protocol to be 400 μs and 500 μs per byte, respectively. Please refer to Table 16

Designing the VCNL36758 Into an Application

6. PROXIMITY MEASUREMENT

6.1 Software Flow Chart for the Proximity Measurement

Fig. 26 shows a typical software flow chart for the proximity measurement using the auto mode and enabled interrupt. The measurement starts with the initialization code. Please note that the provided sensor initialization code is just an example. In practice, the designer should change the initialization code based on the application's requirements. After the initialization, the sensor should perform the offset count cancellation by reading the proximity count without the object. This offset count should then be written to the offset cancellation register PS_CANC. After writing the cancellation register, the sensor should perform proximity measurement in a loop. Next, the microcontroller should wait for the interrupt line to be pulled low via an interrupt service routine. When the interrupt has been pulled low, the interrupt flag register and the proximity data will be read.

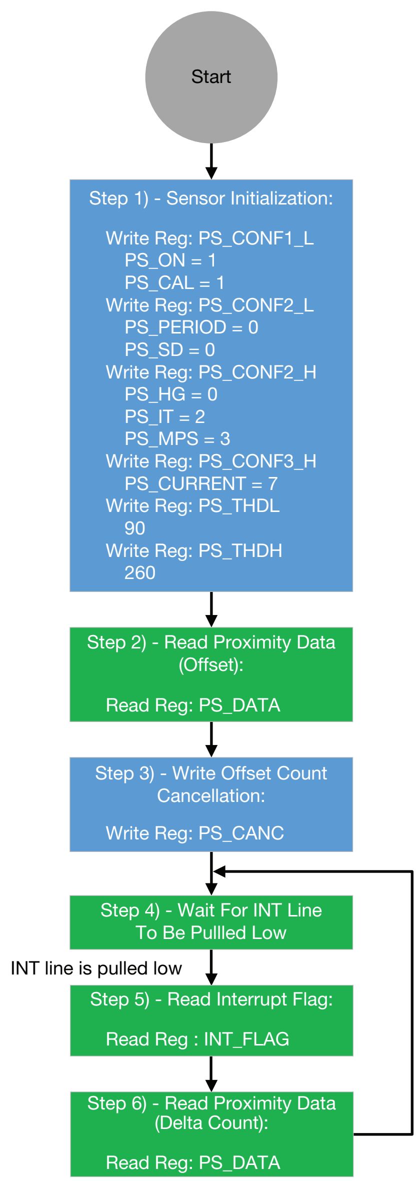


Fig. 26 - Typical Software Flow Chart for the Auto Mode

Designing the VCNL36758 Into an Application

6.2 Proximity Value vs. Distance Curve

The behavior of the proximity sensor can be characterized by the proximity value against the distance. In some applications, the sensor will not be placed underneath a window cover. The designer could then use the proximity value against the distance for the case without the window cover as a reference. On the other hand, most applications will have the requirement of placing the sensor underneath a window cover. Here, the proximity value against the distance for the case with the window cover can be used as a reference. The designer can expect a decrease in the dynamic range due to the increase of crosstalk from the window cover.

6.2.1 Proximity Value vs. Distance Curve Without the Window Cover

Fig. 27 show the typical proximity counts output against distance for three different driving currents and two integration times. Here, the measured reference object medium is the Black Card. This card has approximately 60 % reflectivity at 940 nm. The sensor can detect up to 60 cm with a minimum of 230 delta counts when measuring with the Black Card and when higher sensor settings of PS_IT = 4T and PS_CURRENT = 240 mA have been selected.

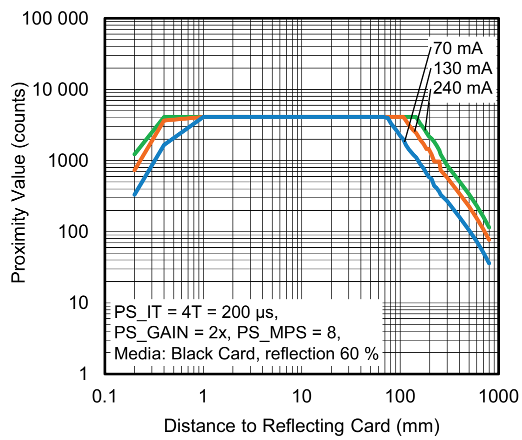


Fig. 27 - Proximity Value vs. Distance

While the Black Card is used as a reference, in practice, the proximity values change with the different objects' reflectivity at the same distance. Therefore, the designer should consider this effect during the design phase.

Designing the VCNL36758 Into an Application

6.3 Temperature Behavior

When designing the VCNL36758 for the end application, the temperature behavior of the proximity counts should also be considered. Fig. 28 shows the average relative proximity counts against the ambient temperature of the VCNL36758.

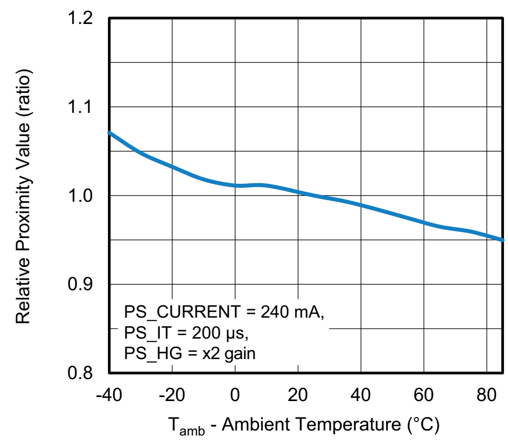


Fig. 28 - Average Relative Proximity Counts vs. Ambient Temperature of VCNL36758

The sensor has a negative temperature coefficient as the temperature increases from room temperature to 85 °C. The designer should consider this behavior and design the end application with enough headroom for the proximity counts, fluctuation with temperature.

Another good design approach to reduce the effect of the proximity counts, fluctuation due to temperature is regular calibration without an object. One can observe that the change in proximity counts with temperature happens slower than the change in proximity counts with changing distance of an object. By observing the increase or decrease of offset counts, the system can recognize that as the change of temperature and proximity correction could be implemented.

Designing the VCNL36758 Into an Application

7. AVERAGE CURRENT CONSUMPTION

Fig. 29 shows the current consumption components through the pins.

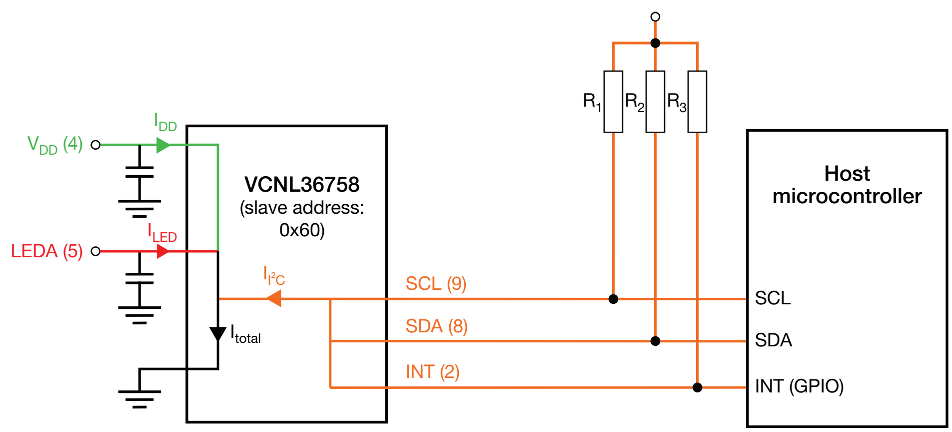


Fig. 29 - Current Consumption Components Through the Pins

The general total current consumption can be calculated as follows:

$$I_{total} = I_{DD} + I_{LED} + I_{I^2C} \quad (4)$$

The main current components are from I_{DD} and I_{LED}. The current contribution from I_{I^2C} for I^2C communication is negligible if the interrupt is activated. However, if the polling mode is used, the current contribution from these pins should be considered in the calculation.

Fig. 30 shows the current consumption profile of the VCNL36758.

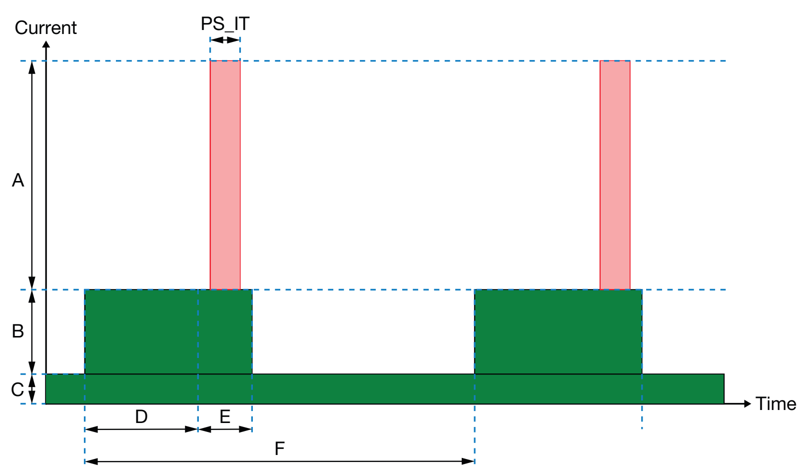


Fig. 30 - Current Consumption Profile of the VCNL36758

Designing the VCNL36758 Into an Application

Table 18 describes the parameters of the current consumption profile in Fig. 30.

TABLE 18 - AVERAGE CURRENT CONSUMPTION PARAMETER			
PARAMETER	**PARAMETER NAME**	**VALUE**	**DESCRIPTION**
A	I_{LED}	70 mA to 240 mA	The magnitude of I_{LED} depends on the selected PS_CURRENT setting; this current is drawn by the pin LEDA
B	I_{DD} (Active)	200 µA	I_{DD} (active) is the supply current consumed during the active state by the pin V_{DD}
C	I_{DD} (Idle)	175 µA	I_{DD} (idle) is the supply current consumed during the idle state by the pin V_{DD}
D	DC_KILL	1000 µs	DC_KILL is the period in which the sunlight and DC noise cancellation happens
E	Proximity measurement pulse	2 x PS_IT x PS_MPS	Proximity measurement is the duration in which the sensor sends the LED pulse; the values depend on PS_IT and PS_MPS
D + E [(1)]	Active measurement period	DC_KILL + (2 x PS_IT x PS_MPS)	The active measurement period is the duration during which the sensor performs the complete proximity measurement, which consists of DC_KILL and proximity measurement pulse
F	Measurement period	10 ms to 80 ms	The measurement period in which the measurement is repeated, which depends on the selected PS_PERIOD setting

Note

[(1)] The active measurement period could have ± 20 % part to part tolerance, which is not considered here

When the sensor is switched on, it will always consume 175 µA. This is indicated by the parameter “C” in Fig. 30 over the whole time as long as the sensor is switched on. During the active measurement period, as indicated by the parameter “D” and “E”, the sensor consumes an additional current of 200 µA on top of 175 µA. Besides that, during the active measurement period, the sensor will emit a short current pulse PS_CURRENT with a duration of PS_IT. The number of pulses within the duration “E” will also be multiplied by the parameter PS_MPS.

The average current consumption over the measurement period PS_PERIOD in the VCNL36758 with interrupt mode can be calculated as follows:

$$I_{average} = I_{DD}(idle) + \frac{((DC\_KILL + (2 \times PS\_IT \times PS\_MPS)) \times I_{DD}(active)) + (PS\_IT \times PS\_CURRENT \times PS\_MPS)}{PS\_PERIOD} \quad (5)$$

Example - Average current Consumption Calculation

For example, given PS_IT = 1T = 50 µs, PS_MPS = 1, PS_PERIOD = 10 ms, and PS_CURRENT = 70 mA, the average current consumption with negligible I_{I^2C} can be calculated as follows:

$$I_{average} = 175\ \mu A + \frac{((1000\ \mu s + (2 \times 1 \times 50\ \mu s \times 1)) \times 200\ \mu A) + (1 \times 50\ \mu s \times 70\ mA \times 1)}{10\ ms} = 547\ \mu A$$

Designing the VCNL36758 Into an Application

8. MECHANICAL AND OPTICAL DESIGN

As light waves propagate through space, they naturally spread out. This phenomenon also occurs with the infrared light emitted by the LED, which broadens as it moves farther from the sensor. Consequently, the diameter of the window cover must be carefully adjusted to meet the minimum size requirements, ensuring that the opaque housing does not obstruct the infrared signal. This minimum diameter is influenced by the distance between the top of the cover and the top of the sensor, the emission angle of the LED, and the photodiode's field of view.

Typically, the window diameter needs to increase as the distance from the sensor increases. The LED in the sensor emits light with an angle of ± 15°. On the other hand, the photodiode in the sensor has an angle of half sensitivity of ± 35° in the X-axis direction and ± 30° in the Y-axis direction. These angles are illustrated in Fig. 31 and Fig. 32.

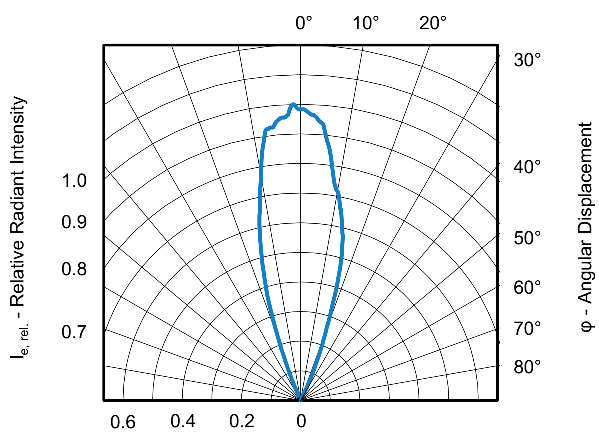


Fig. 31 - Relative Radiant Intensity vs. Angular Displacement of the LED in the VCNL36758

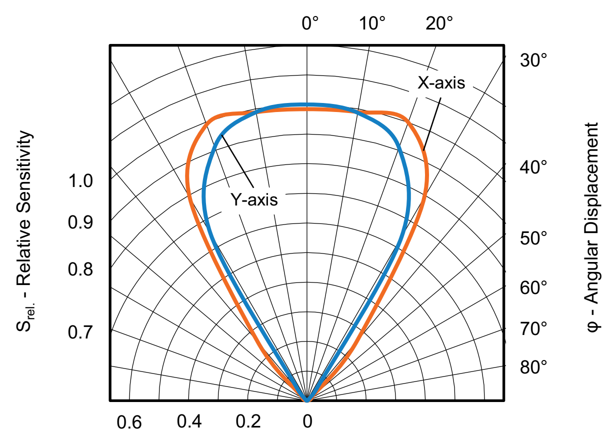


Fig. 32 - Relative Radiant Intensity vs. Angular Displacement of the Photodiode in the VCNL36758

Designing the VCNL36758 Into an Application

Window Design

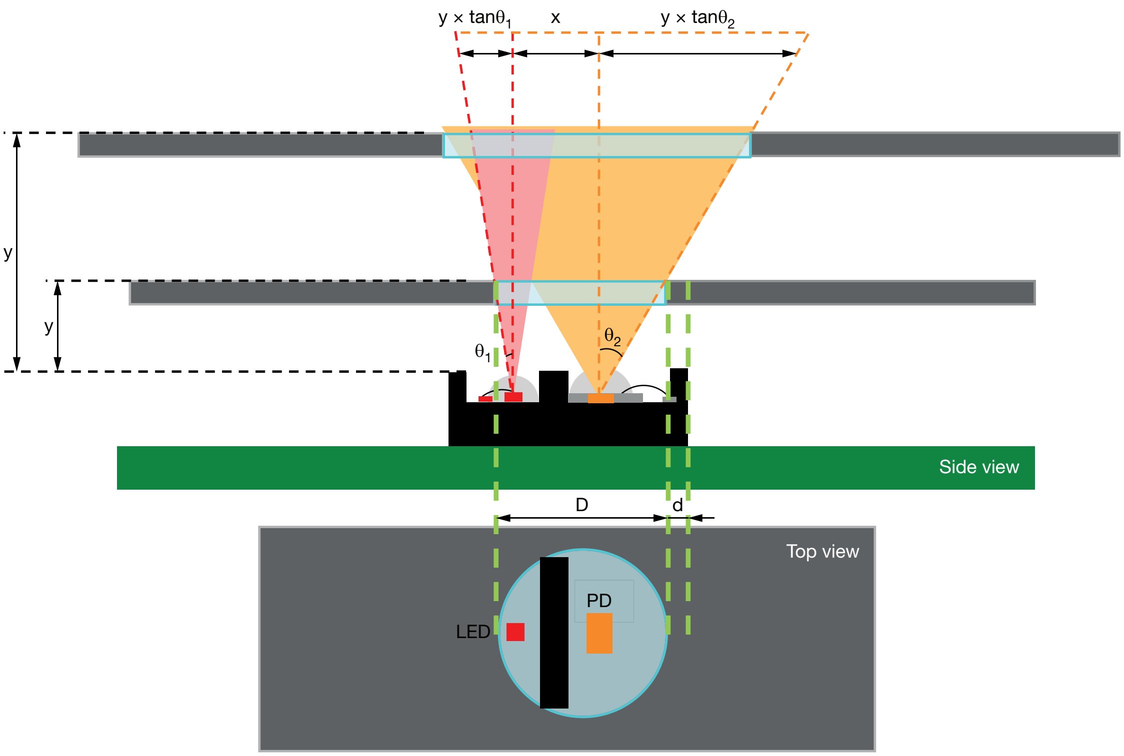


Fig. 33 - The Calculation of the Window Cover Diameter With an Air Gap

Explanation of Variables

- x: this represents the distance between the center of the LED and the center of the photodiode (PD). In Fig. 33, this is the horizontal distance between the red and orange blocks; the value of x is given as 3.1 mm, which includes the mechanical placement tolerance of the LED and photodiode within the sensor package
- y: this is the distance from the top of the window cover to the top of the sensor; it represents the vertical distance between the black line at the top (which marks the position of the window cover) and the top of the sensor; this is an important parameter, as it determines the necessary window cover size to avoid blocking the infrared signal
- θ1: this angle represents the emission angle of the LED; it is the angle between the red dashed line (indicating the direction of the emitted signal) and the vertical axis; Fig. 33 suggests that θ1 should ideally be 15°, but a larger angle (e.g., 20°) may be used to account for mechanical tolerance and ensure proper signal transmission
- θ2: this angle represents the field of view of the photodiode (PD); it is the angle between the orange dashed line (indicating the field of view) and the vertical axis; Fig. 33 shows this as an angle that allows the photodiode to capture the reflected infrared signal from the object. The recommended value is θ2 = 40°, though the photodiode's total field of view is ± 35°
- d: represents the distance between the edge of the package near the photodiode (PD) and the window cover; this distance assists with alignment during assembly

Calculation of Minimum Window Cover Diameter D

The minimum window cover diameter D can be calculated using the following formula:

$$D = y \times \tan\theta_2 + x + y \times \tan\theta_1$$

This formula considers both the emission angle θ1 and the field of view θ2 to determine the necessary window cover diameter, ensuring that the infrared signal is not blocked during either emission or reception.

Designing the VCNL36758 Into an Application

Calculation of Minimum Distance d:

The distance between the photodiode (PD) and the edge of the package can be calculated using the following formula:

$$d = 1 - y \times \tan\theta_2$$

When y exceeds 1.19 mm, d becomes negative, indicating that the edge of the window will be on the right side of the package. Conversely, if d is positive, the edge of the window will be on the left side of the package.

TABLE 19 - MINIMUM WINDOW COVER DIAMETER

y (mm)	D (mm)	d (mm)
0.5	3.70	0.58
1.0	4.30	0.16
1.5	4.90	-0.26
2.0	5.51	-0.68
2.5	6.11	-1.10

Rubber Design

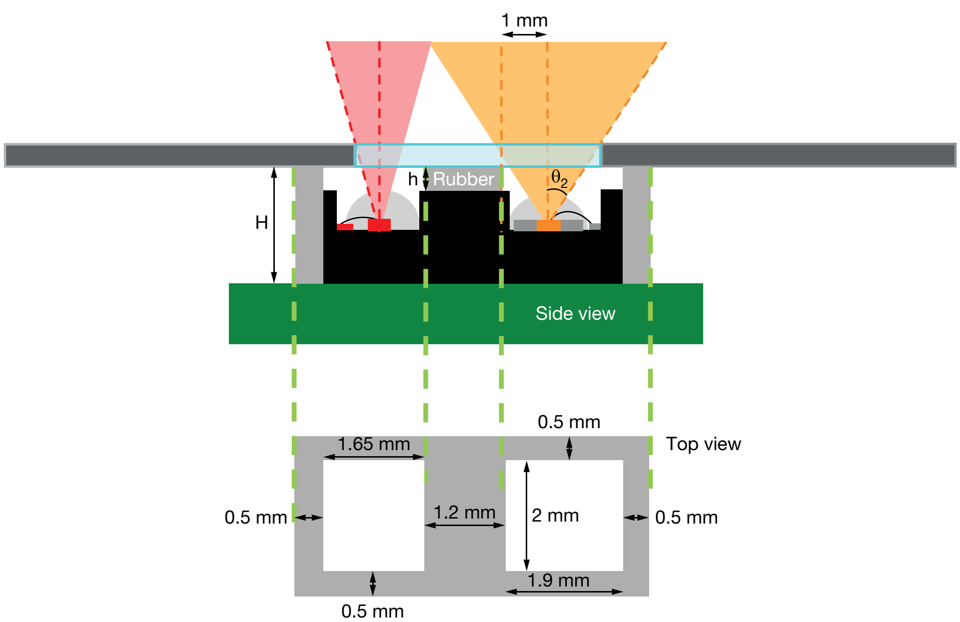


Fig. 34 - The Calculation of the Rubber Height in Contact With the Lens

Calculation of Maximum Height H

The dimensions (length and width) of the rubber are marked in Fig. 34. During assembly, the inner edge of the rubber should be positioned adjacent to the VCNL36758 package. The maximum height at the center is calculated using the formula:

$$1 \geq h \times \tan\theta_2$$

where h is the distance between the center of the package and the lens when they are in direct contact. Since h impacts the sensor's light reception angle, it is recommended that the maximum height h be 1.19 mm.

To determine the overall height H of the rubber, the height of the VCNL36758 package must be added. Therefore, the recommended maximum rubber height is 2.69 mm.

Designing the VCNL36758 Into an Application

Additional Considerations

The placement of the window cover directly on top of the sensor is usually preferred, as this avoids the sensor's peak sensitivity area and helps reduce crosstalk. However, achieving this during assembly can be challenging due to mechanical tolerance, which often results in an air gap. To resolve this issue, a rubber must be included and tightly positioned between the package and the lens to prevent significant crosstalk caused by the lens, which can reduce the sensor's dynamic range. If an air gap is present, it should ideally be less than 1 mm to minimize its impact on the sensor's performance.