Software Design Description (SDD)

Rev: A1

17 October 2013

ALS/IR Proximity Device Driver

For

ISL29028A sensor

VVDN Contact:

Bhupender Saharan

VVDN Technologies

+1 408 807 3951

bhupi@vvdntech.com

Revision History:

Date	Rev No.	Description	By
07 Oct 2013	A1	Software Design Description	VVDN

Table of Contents

Table of Figures

1 Introduction

This document describes software components of ISL29028A device driver for Android Jellybean (version 4.2) and Android Ice cream sandwich (version 4.0.4) which is to be developed by VVDN for Intersil Corporation.

This SDD is made for the reference of

1. Product managers and QAD at VVDN & Intersil to understand the software development phases.
2. Engineering Team at VVDN for System Architecture, Design and development of device driver.
3. System Integration and Verification teams at VVDN / Intersil for SW validation.

2 Scope of the document

This document describes the software design specification of ISL29028A device driver in detail.

3 Reference documents

S.NO	Description	Revision	Date
1	VVDN_ISLU_SNSR_SDD_A0.pdf	A1	05 Oct 2013
2			
3			

4 Overview of project

Intersil manufactures **ISL29028A**, an Ambient Light and Proximity sensor device. It is a low power, high sensitivity, Intelligent Interrupt capability and Sleep mode supported Ambient light sensor with an I2C (SM-Bus compatible) interface. Its advanced self-calibrated photodiode array emulates human eye response with excellent IR rejection. This device uses two independent ADCs for concurrently measuring ambient light and proximity in parallel. The scope of this project is to develop and test ISL29028A device driver.

5 Software Requirement

5.1.1 Operating system requirement

The sensor device driver should be supported on following operating systems

1. Android Jelly Bean (version 4.2)
2. Android Ice cream sandwich (version 4.0.4)

5.1.2 Application requirements

The sensor device driver should support the following operations

1. Configure device operation modes
   a. ALS
   b. IR
   c. Proximity

2. Configure the High and Low interrupt thresholds for ALS and Proximity
3. Configure interrupt persistency to 1/4/8/16
4. Configure the ALS/IR range
   a. Low Lux range [125 Lux]
   b. High Lux range [2000 Lux]

6 System Block Diagram

The Intersil ISL29028A sensor would be interfaced to a TI's OMAP4460 based Panda-board via an I2C and Interrupt line.

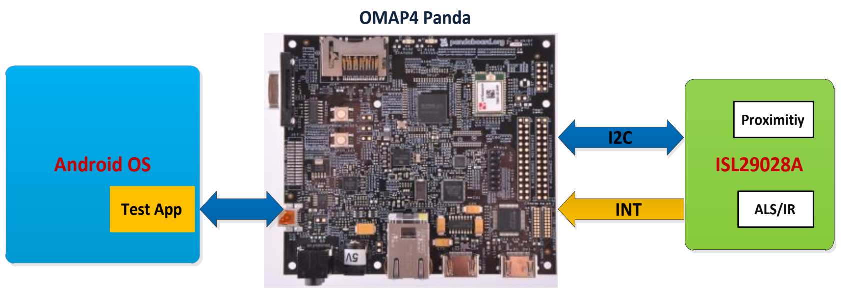

Figure 1 System block diagram

I2C interface would be used to

1. Configure the sensor device
2. Read sensor device (Light intensity or Proximity)

Sensor interrupt pin would be connected to an interruptible GPIO pin on Panda-board.

7 Software Architecture

The system software architecture diagram is shown below.

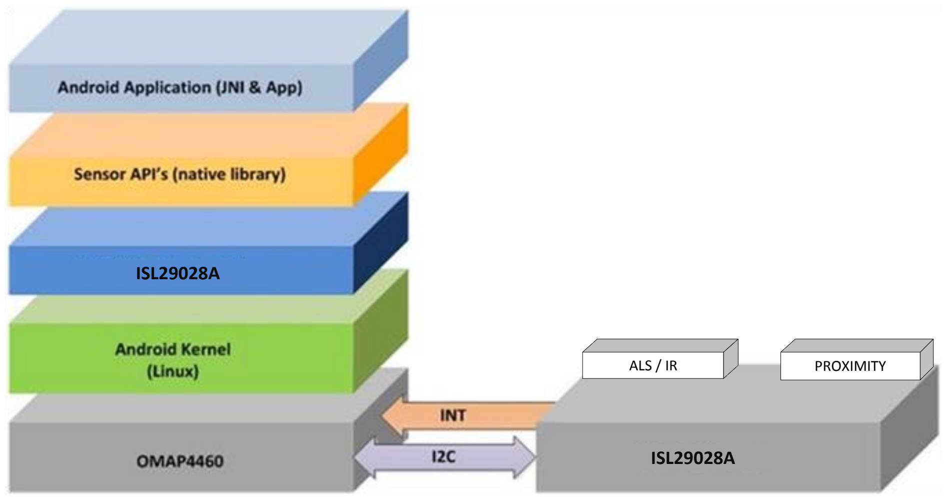

Figure 2 Software architecture diagram

8 Device driver use case model

The use case model for ISL29028A device driver is shown below. The functionalities provided by the device driver are as follows.

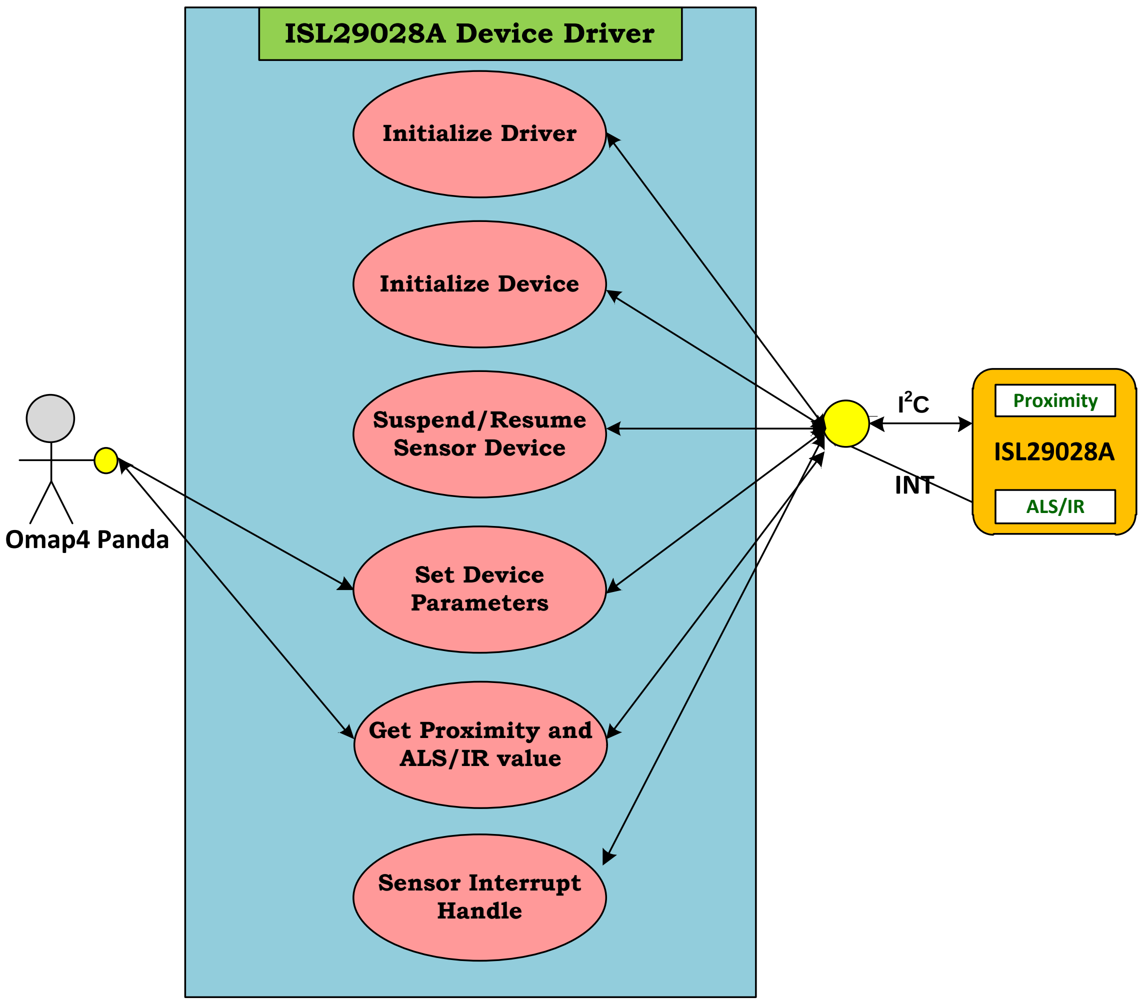

Figure 3 Sensor device driver use case diagram

8.1 Initialize the driver

The driver initialization code does the following

1. Registers the I2C device driver with the Linux I2C core framework.
2. Requests and set up the GPIO line (to be used as interrupt line) for handling the sensor interrupts.
3. Registers the Interrupt handlers for the IRQ number corresponding to the sensor interrupt line.

8.2 Initialize the device

Once the I2C core framework recognizes the ISL29028A (slave address 0x45h) on the I2C bus. It calls the probe function of the device driver (that supports/services the ISL29028A device) which verifies the ALS/Proximity sensor device, and writes the initial configuration parameters to the device. At this stage the sensor device is ready to operate in the default operating conditions.

8.3 Get Proximity and ALS/IR value

8.3.1 Proximity sensing mode

The Proximity sensing mode is enabled when ***PROX_EN = 1*** in CONFIGURATION register 0x01h. The Proximity value is read from the sensor device register PROX_DATA (Address 0x08h).

PROX_DATA is an 8-bit register; it stores the ADC conversion data for Proximity sensor. Reading PROX_DATA Register will give the 8-bit ADC value.

Add	Register	D7	D6	D 5	D 4	D3	D2	D1	D0
0x08h	PROX_DATA	0	0	0	0	0	0	0	0

8.3.2 ALS / IR sensing mode

The Ambient light intensity (ALS / IR) is read from the ISL29028A sensor device from the registers shown in the below table (Values are stored in 12-bit registers as MSB and LSB). The values are read when the user application sends an IOCTL request for reading ALS values.

REG Address	**Register name**	**DATA**
0x09h [7:0]	**ALSIR_DT1**	**8-bit Data**
0x0Ah [3:0]	**ALSIR_DT2**	**LSB Nibble**

Add	Register	D7	D6	D 5	D 4	D3	D2	D1	D0
0x09h	ALSIR_DT1	0	0	0	0	0	0	0	0

Add	Register	D15	D14	D13	D12	D11	D10	D9	D8
0x0Ah	ALSIR_DT1	X	x	X	x	0	0	0	0

|←-------------MSB Nibble-------------→|←-------------LSB Nibble --------------→|

x – Reserved / Unused

The Lower Nibble of Address 0x0Ah is used as the upper 4-bits of 12-bit ALS/IR data. The Upper 4-bits of device address 0x0Ah are unused or reserved.

Conversion between Lux and ADC code is:

$E_{calc} = \alpha_{range} \times OUT_{ADC}$;

ADC code is represented in Hex code.

The constant α in above expression is determined by the range bit ALS_RANGE (register 0x01-bit 1) and is independent of the light source type.

ALS_RANGE	α_{range} (Lux / Count)
0	0.0326
1	0.522

Table 1

Table 1shows two different scale factors:

1. Low range (ALS_RANGE = 0)
2. High range (ALS_RANGE = 1)

The following table shows the bit alignments of Register 0x01.

Add	Register	D7	D6	D5	D4	D3	D2	D1	D0
0x01h	CONFIG	PROX_EN	PROX_SLP			PROX_DR	ALS_EN	ALS_RANGE	ALSIR_MODE

8.4 Set device parameters

The programming of device parameters is taken care by IOCTL implementation of the driver. Each parameter is assigned a unique identifier number based on which driver decides what action is to be taken.

The general sequence diagram for programming a device parameter from a test application is shown below.

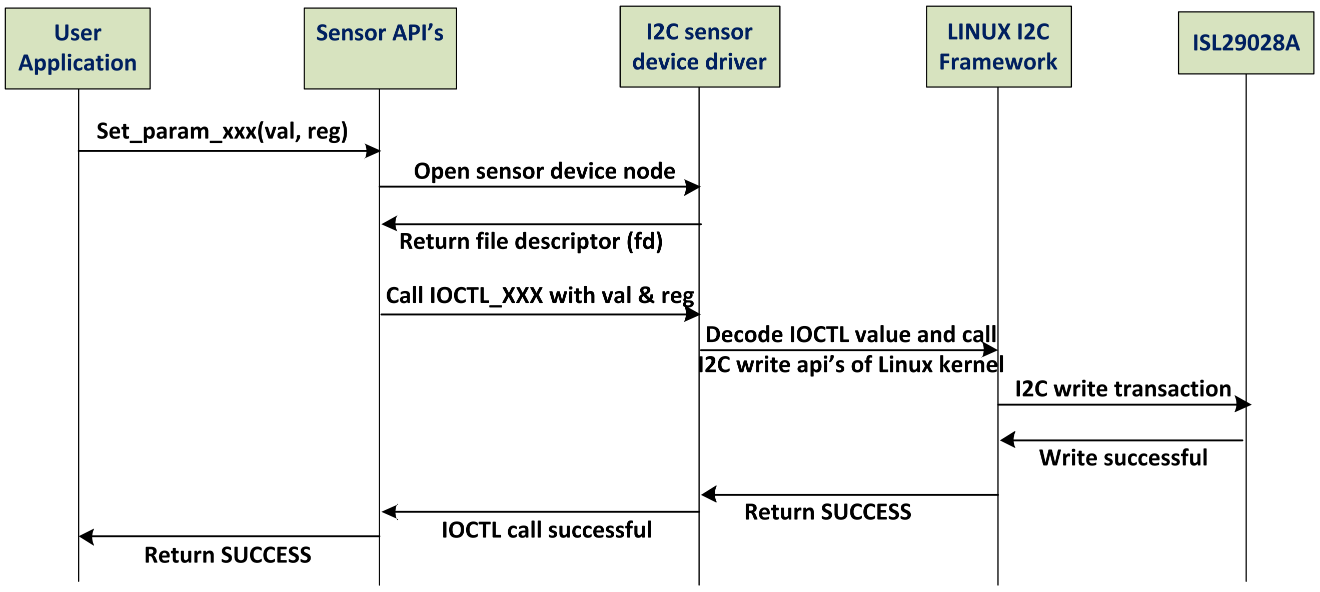

Figure 4 Sequence diagram for programming sensor device parameter

8.4.1 Device Operation mode

The device operation mode indicates the operating mode of the device. Sensor device supports the following operating modes.

- ALS / IR sensing mode
- Proximity sensing mode

The device operation mode indicates the device state and which light's component is being converted by ADC. The device operation mode is programmed by writing CONFIGURATION Register (0x01h) of sensor device.

a) ALS/IR Sensing mode

The ISL29028A is set for ambient light sensing when Register bit ***ALSIR_MODE = 0; ALR_EN = 1*** in register 0x01. When the bits are programmed for infrared (IR) sensing (***ALSIR_MODE = 1; ALS_EN = 1***), infrared light is converted into a current and digitized by the same ALS ADC. The result of an IR conversion is strongly related to the amount of IR energy incident on our sensor, but is unit less and is referred to in digital counts.

Add	Register	D7	D6	D5	D4	D3	D2	D1	D0
0x01h	CONFIG	PROX_EN	PROX_SLP			PROX_DR	ALS_EN	ALS_RANGE	ALSIR_MODE

b) Proximity sensing mode

The Proximity sensing mode is enabled when ***PROX_EN = 1*** in CONFIGURATION register 0x01. The Proximity value can be read from the Proximity sensor device register PROX_DATA (Address 0x08h).

User can put the sensor in Proximity mode by setting the PROX_EN bit.

Add	Register	D7	D6	D5	D4	D3	D2	D1	D0
0x01h	CONFIG	PROX_EN	PROX_SLP			PROX_DR	ALS_EN	ALS_RANGE	ALSIR_MODE

8.4.2 Range

The range of sensing defines the sensitivity range of the sensor device. There are two ranges for ALS/IR sensing.

8.4.2.1 ALS/IR Sensing Ranges

1. 125 Lux (Low Range)

   ALS_RANGE = 0;   # Low sensitivity mode

2. 2000 Lux (High range)

   ALS_RANGE = 1;   # High sensitivity mode

This parameter is programmed by writing to the **ALS sensing range** [D1] in CONFIGURATION register (0x01h) of sensor device.

Add	Register	D7	D6	D5	D4	D3	D2	D1	D0
0x01h	CONFIG	PROX_EN	PROX_SLP			PROX_DR	ALS_EN	ALS_RANGE	ALSIR_MODE

8.4.3 ADC resolution

The ADC conversion is done at 12-bit resolution. The ADC resolution affects the Lux per count of sensor device for different Lux ranges.

8.4.4 Interrupt threshold

The interrupt threshold value defines a window of Lux values. Whenever the current ADC measurement reaches above or below this threshold window, the sensor device raises an interrupt to the processor. There are separate thresholds for both Proximity and ALS/IR sensor.

8.4.4.1 Proximity Thresholds:

1. PROX_LT (0x03)
2. PROX_HT (0x04)

The Proximity threshold register is a 16-bit register. The sensor device address 0x03 and 0x04 together constitute 16-bit register and can be programmed separately. Proximity low Interrupt threshold can be programmed by writing address 0x03 of sensor. Proximity high Interrupt threshold can be programmed by writing address 0x04.

Add	Register	D7	D6	D 5	D 4	D3	D2	D1	D0
0x03h	PROX_LT	0	0	0	0	0	0	0	0
Add	Register	D15	D14	D 13	D 12	D11	D10	D9	D8
0x04h	PROX_HT	0	0	0	0	0	0	0	0

8.4.4.2 ALS/IR Thresholds

The ALS/IR threshold consists of separate 12-bit low threshold and 12-bit high threshold.

1. ALSIR_LT [ALSIR_TH2 (3:0) + ALSIR_TH1 (7:0)]
2. ALSIR_HT [ALSIR_TH3 (7:0) + ALSIR_TH2 (7:4)]

These parameters are programmed by writing to the following registers

Add	Register	D7	D6	D 5	D 4	D3	D2	D1	D0
0x05h	ALSIR_TH1	0	0	0	0	0	0	0	0

|←-----------------------Lower thresho4ld byte of ALSIR_LT-------------------→|

Add	Register	D3	D2	D 1	D 0	D11	D10	D9	D8
0x06h	ALSIR_TH2	1	1	1	1	0	0	0	0

|←-----Lower byte of ALSIR_HT-----→|←----Upper byte of ALSIR_LT-----→|

Add	Register	D7	D6	D 5	D 4	D3	D2	D1	D0
0x07h	ALSIR_TH3	1	1	1	1	1	1	1	1

|←----------------------Upper threshold byte of ALSIR_HT---------------------→|

The Low threshold consists of register ALSIR_TH2 [3:0] + ALSIR_TH1 [7:0] resisters making it 12-bit threshold register. The default value of lower threshold register is 0x00.

The High threshold consists of register ALSIR_TH2 [7:4] + ALSIR_TH2 [7:0] registers making it 12-bit threshold register. The default value of high threshold register is 0xfff.

8.4.5 Interrupt persistency

Interrupt persistency is used to avoid false interrupt cases due to noise or sudden spikes in the ambient light conditions. This parameter defines the 'N' number of consecutive interrupt threshold events after which the interrupt will be raised to processor and appropriate flags will be set.

The ISL29028A supports separate interrupt persistency (1/4/8/16) for Proximity and ALS/IR.

1. The Persistency for Proximity is programmed by writing bits [D6] and [D5] of address 0x02.
2. The Persistency for ALS/IR is programmed by writing bits [D2] and [D1] of address 0x02.

The following table shows the bits alignment of 0x02.

Add	Register	D7	D6	D 5	D 4	D3	D2	D1	D0
0x02h	PROX/ALS	PROX_FLAG	PROX_PRST		Unused	ALS_FLAG	ALS_PRST		INT_CTRL

Proximity persistency value is shown in following table.

Bits 6:5	**Action**
00	set PROX_FLAG if **1** conversion result trips the threshold value
01	set PROX_FLAG if **4** conversion results trip the threshold value
10	set PROX_FLAG if **8** conversion results trip the threshold value
11	set PROX_FLAG if **16** conversion results trip the threshold value

ALS/IR persistency value is shown in following table.

Bits 2:1	**Action**
00	set ALS_FLAG if **1** conversion result trips the threshold value
01	set ALS_FLAG if **4** conversion results trip the threshold value
10	set ALS_FLAG if **8** conversion results trip the threshold value
11	set ALS_FLAG if **16** conversion results trip the threshold value

PROX_FLAG will be set when Proximity interrupt occurs.

ALS_FLAG will be set when ALS interrupt occurs.

The interrupt control (INT_CTRL) bit is set corresponding to FLAG set.

```
if (PROX_FLAG || ALS_FLAG)
        INT_CTRL = 0;
else if (PROX_FLAG && ALS_FLAG)
        INT_CTRL = 1;
```

8.4.6 Proximity sleep time

This time shows the sleep time between proximity IR LED Pulses. Proximity sleep time value shows in below table.

Bits 6:4	**Action**
111	sleep time between prox IR LED pulses is 0.0 ms
110	sleep time between prox IR LED pulses is 12.5ms
101	sleep time between prox IR LED pulses is 50 ms
100	sleep time between prox IR LED pulses is 75 ms
011	sleep time between prox IR LED pulses is 100 ms
010	sleep time between prox IR LED pulses is 200 ms
001	sleep time between prox IR LED pulses is 400 ms
000	sleep time between prox IR LED pulses is 800 ms

8.5 Sensor device suspend/resume

The drivers suspend and resume routines does the following operation.

1. Put the sensor in standby mode when android OS sleeps
2. Wake the sensor from standby when android OS wakes up

Both these events are taken care in the device driver by use of driver's suspend and resume callback functions.

8.6 Handle sensor interrupts

The ALS sensor device has the following sources of interrupt

1. ALS value exceeds the lower or upper threshold
2. Proximity value exceeds the lower or upper interrupt
3. ADC completed the conversion

The interrupt handling scheme is explained in the below flowchart.

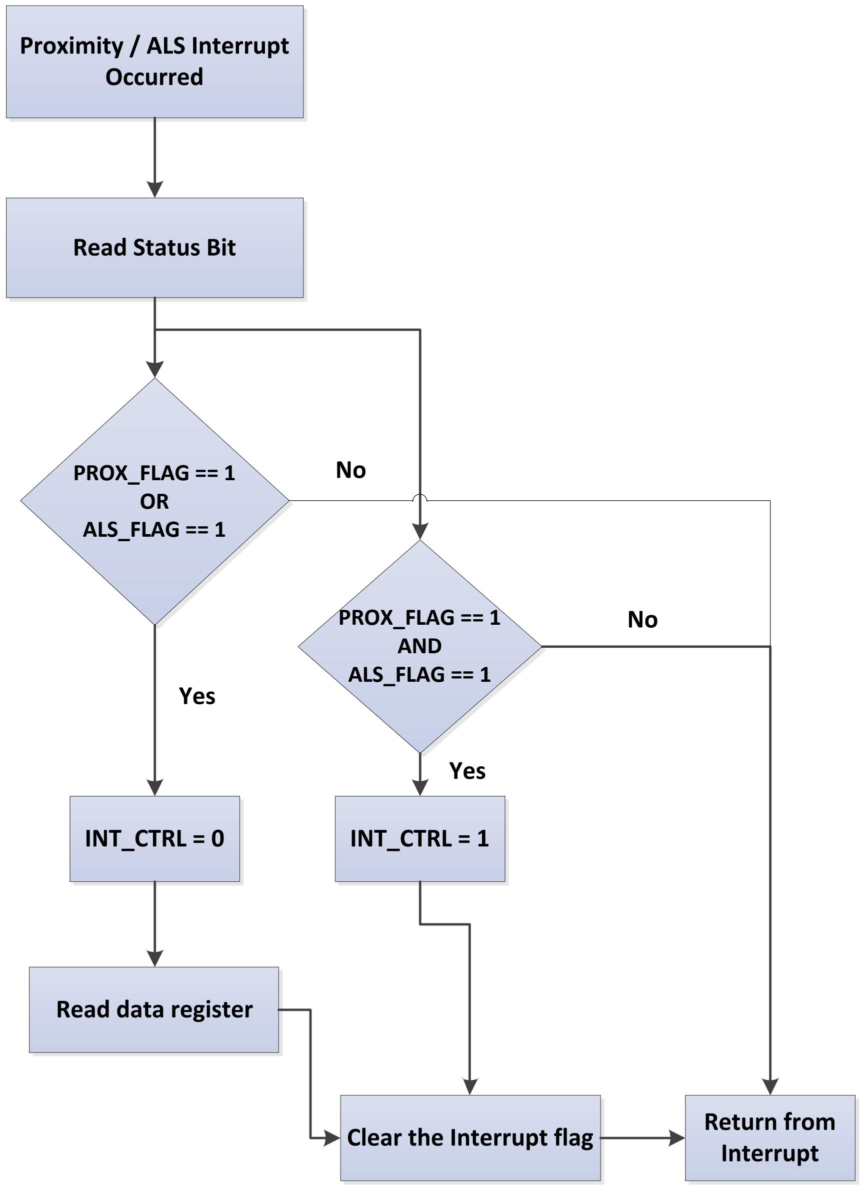

Figure 5 Sensor device interrupt handling

9 ALS sensor API's

Note: Detailed programmer API documentation for ISL29028A ALS sensor access would be provided along with the device driver and library code.