# Optical Sensor
## Product Data Sheet
### LTR-559ALS-01

**Spec No.: DS86-2013-0003**

Effective Date: 07/09/2014

Revision: B

**Optical Sensor**
**LTR-559ALS-01**

## 1. Description

The LTR-559ALS-01 is an integrated low voltage $I^2C$ digital light sensor [ALS] and proximity sensor [PS] with built-in emitter, in a single miniature chipled lead-free surface mount package. This sensor converts light intensity to a digital output signal capable of direct $I^2C$ interface. It provides a linear response over a wide dynamic range from 0.01 lux to 64k lux and is well suited to applications under high ambient brightness. With built-in proximity sensor (emitter and detector), LTR-559ALS-01 offers the feature to detect object at a user configurable distance.

The sensor supports an interrupt feature that removes the need to poll the sensor for a reading which improves system efficiency. The sensor also supports several features that help to minimize the occurrence of false triggering. This CMOS design and factory-set one time trimming capability ensure minimal sensor-to-sensor variations for ease of manufacturability to the end customers.

## 2. Features

- $I^2C$ interface (Fast Mode @ 400kbit/s)

- Ultra-small ChipLED package

- Built-in temperature compensation circuit

- Low active power consumption with standby mode

- Supply voltage range from 2.4V to 3.6V capable of 1.7V logic voltage

- Operating temperature range from -30 ℃ to +70 ℃

- RoHS and Halogen free compliant

- Light Sensor

  ➢ Close to human eye spectral response

  ➢ Immunity to IR / UV Light Source

  ➢ Automatically rejects 50 / 60 Hz lightings flicker

  ➢ 6 dynamic range from 0.01 lux to 64k lux

  ➢ 16-bit effective resolution

- Proximity Sensor

  ➢ Built-in LED driver, emitter and detector

  ➢ Programmable LED drive settings

  ➢ 11-bit effective resolution

  ➢ High ambient light suppression

# 3. Applications

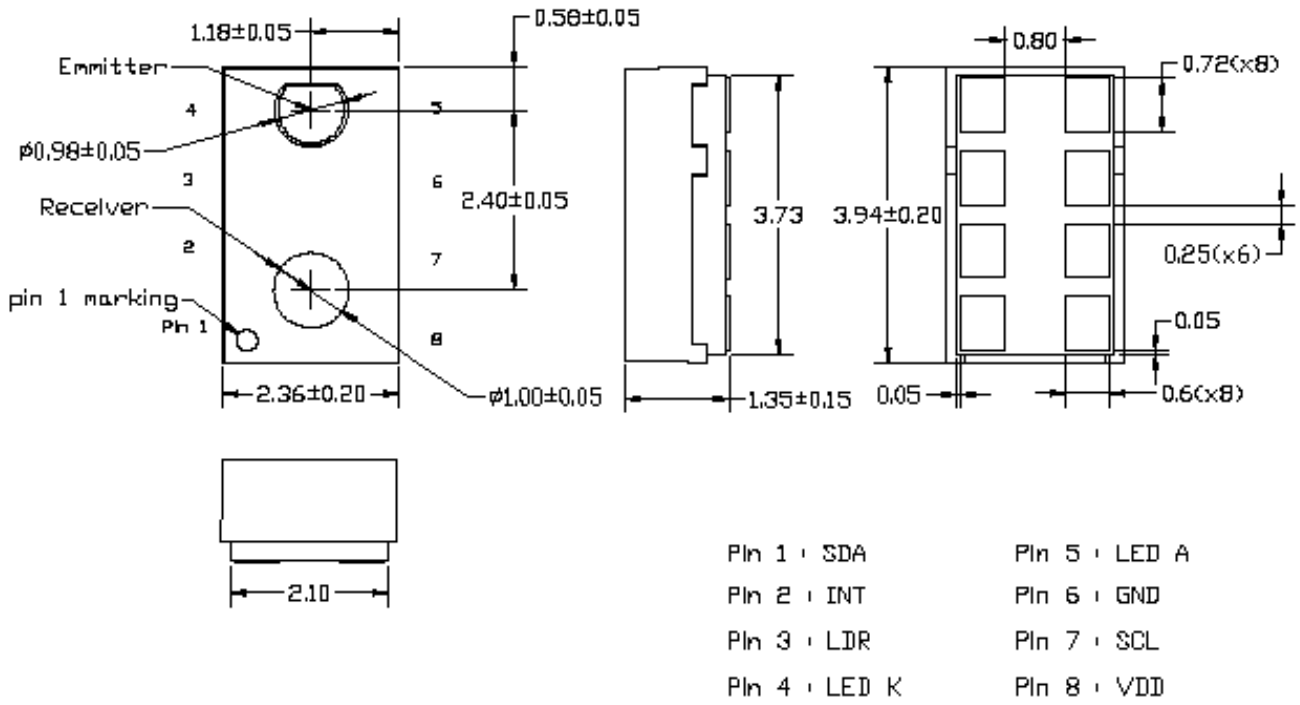To control display backlight in

- Mobile Devices: Mobile phone, PDA

- Computing Devices: Notebook PC, Desktop Monitor

- Consumer Devices: LCD/PDP TV backlight systems, Cameras, Personal Navigation Device, Digital Photo Frame

- Dashboard

# 4. Ordering Information

| Part Number | Packaging Type | Package | Quantity |
|---|---|---|---|
| LTR-559ALS-01 | Tape and Reel | 8-pin chipled package | 8000 |

**Optical Sensor**
**LTR-559ALS-01**

## 5. Outline Dimensions



Pin 1 : SDA       Pin 5 : LED A
Pin 2 : INT       Pin 6 : GND
Pin 3 : LDR       Pin 7 : SCL
Pin 4 : LED K     Pin 8 : VDD

1. All dimensions are in millimeters

## 6. Functional Block Diagram
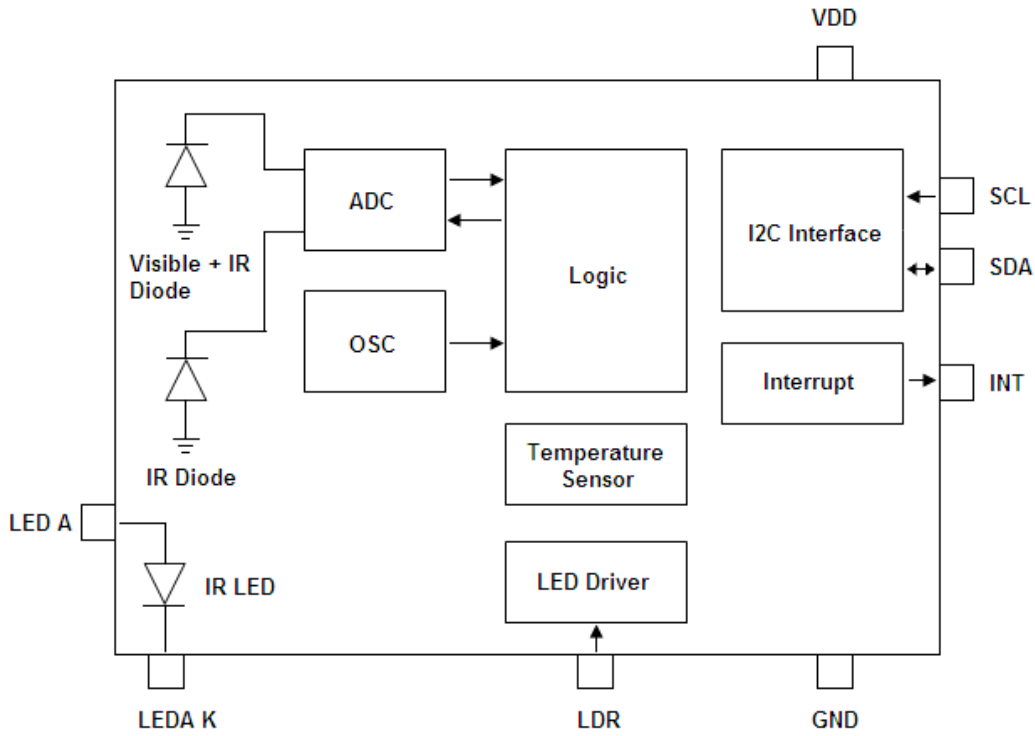


## 7. Application Circuit
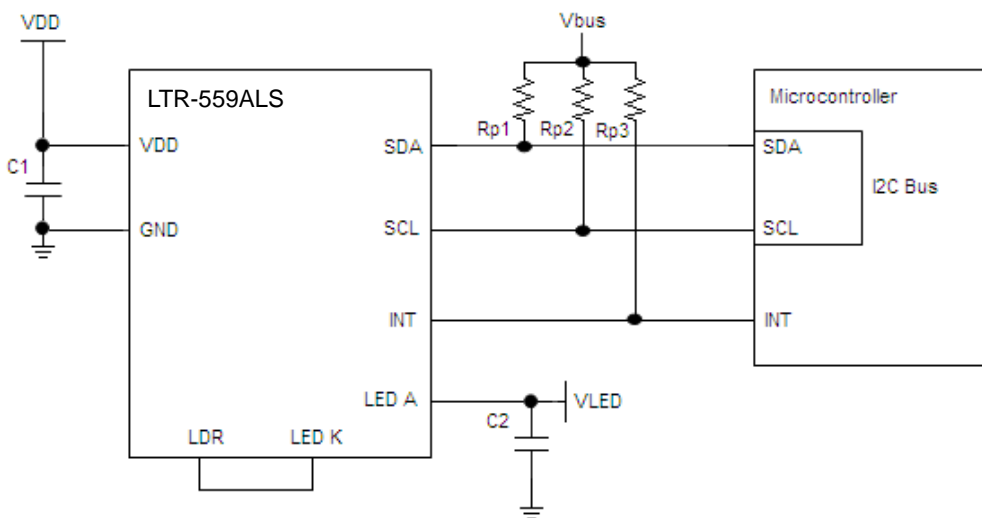
### I/O Pins Configuration Table

| Pin | I/O Type | Symbol | Description |
|---|---|---|---|
| 1 | I/O | SDA | I$^2$C serial data. This pin is an open drain input / output. |
| 2 | O | INT | Level Interrupt Pin. This pin is an open drain output. |
| 3 | I | LDR | LED Driver for proximity emitter. This pin is an open drain input. |
| 4 | O | LED K | LED Cathode. Connect to LDR pin if using internal LED driver circuit. |
| 5 | I | LED A | LED Anode. Connect to VDD or VBAT on PCB |
| 6 | | GND | Ground |
| 7 | I | SCL | I$^2$C serial clock. This pin is an open drain input. |
| 8 | | VDD | Power Supply Voltage |
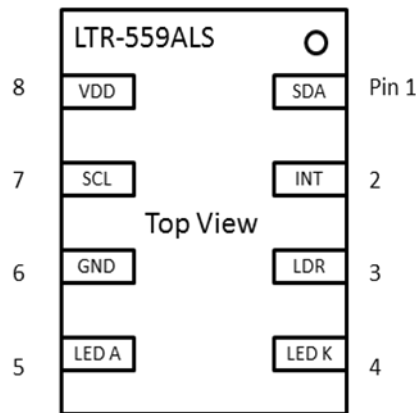


### Recommended Application Circuit Components

| Component | Recommended Value |
|---|---|
| Rp1, Rp2, Rp3 [1] | 1 kΩ to 10 kΩ |
| C1, C2 | 1uF ±20%, X7R Ceramic |

[1] Selection of pull-up resistors value is dependent on bus capacitance values. For more details, please refer to I$^2$C Specifications: http://www.nxp.com/documents/user_manual/UM10204.pdf

## 8. Ratings and Specifications

### Absolute Maximum Ratings at Ta = 25°C

| Parameter | Symbol | Rating | Unit |
|---|---|---|---|
| Supply Voltage | VDD | 3.8 | V |
| Digital Voltage Range | SCL, SDA, INT | -0.5 to 3.8 | V |
| Digital Output Current | SCL, SDA, INT | -1 to 20 | mA |
| Storage Temperature | $T_{stg}$ | -40 to 100 | °C |

Note: Exceeding these ratings could cause damage to the sensor. All voltages are with respect to ground. Currents are positive into, negative out of the specified terminal.

### Recommended Operating Conditions

| Description | Symbol | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| Supply Voltage | VDD | 2.4 | | 3.6 | V |
| LED Supply Voltage | VLED | 2.5 | | 4.35 | V |
| Interface Bus Power Supply Voltage | $V_{IO}$ | 1.7 | | 3.6 | V |
| Operating Temperature | $T_{ope}$ | -30 | | 70 | °C |

### Electrical & Optical Specifications

All specifications are at VDD = 3.0V, $T_{ope}$ = 25°C, unless otherwise noted.

| Parameter | Min. | Typ. | Max. | Unit | Condition |
|---|---|---|---|---|---|
| Active Supply Current | | | 250 | uA | Active Mode, $T_{ope}$ = 25°C |
| Standby Current | | | 5 | uA | Standby / Sleep Mode |
| Initial Startup Time | 60 | 100 | 1000 | ms | (Note 1) |
| Wakeup Time from Standby | | | 10 | ms | (Note 1) |

## Light Sensor

| Parameter | Min. | Typ. | Max. | Unit | Condition |
|---|---|---|---|---|---|
| Full Scale ADC Count | | | 65535 | count | |
| Dark ADC Count | 0 | | 6 | count | Ch0, Lux = 0 |
| | 0 | | 6 | count | Ch1, Lux = 0 |
| ADC Count (Gain=96) | 3200 | 4900 | 6600 | count | Ch0<br>White LED 200 Lux<br>Integration Time : 50ms<br>Measurement Time : 100ms |
| | 1200 | 1800 | 2400 | count | Ch1<br>White LED 200 Lux<br>Integration Time : 50ms<br>Measurement Time : 100ms |
| ALS Ratio | 0.15 | 0.26 | 0.35 | | Ratio = Ch1/(Ch1 + Ch0)<br>Lux = 200 (White LED) |

## Proximity Sensor

| Parameter | Min. | Typ. | Max. | Unit | Condition |
|---|---|---|---|---|---|
| Full Scale ADC Count | | | 2047 | count | |
| Peak Sensitivity | 840 | 855 | 870 | nm | |
| Detection Distance | 20 | | | nm | 100mA, 4 pulses,<br>18% Gray card |
| Ambient Light Suppression | | | 50k | Lux | Direct sunlight |
| LED Pulse Count | 1 | | 15 | Pulses | |
| LED Pulse Frequency | 30k | | 100k | Hz | Increment of 10k Hz |
| LED Duty Cycle | 25 | | 100 | % | Increment of 25% |
| LED Peak Current | | 5 | | mA | LED Peak Current = 000 |
| | | 10 | | mA | LED Peak Current = 001 |
| | | 20 | | mA | LED Peak Current = 010 |
| | | 50 | | mA | LED Peak Current = 011 |
| | | 100 | | mA | LED Peak Current = 1XX |
| Optical Rise / Fall Time | 100 | | | ns | |

**Note 1: Startup Sequence**

```
           ┌─────────────────────────────────┐
           │      Supply VDD to Sensor        │
           │    (Sensor in Standby Mode)      │
           └─────────────────────────────────┘
                          │
              Wait 100 ms (min) - initial startup time
                          │
                          ▼
           ┌─────────────────────────────────┐
           │      I2C Command (Write)         │
           │   To enable sensor to Active Mode │
           └─────────────────────────────────┘
                          │
              Wait 10 ms (max) - wakeup time from standby
                          │
                          ▼
           ┌─────────────────────────────────┐
           │  Sensor is Active and starts measurement │
           └─────────────────────────────────┘
```
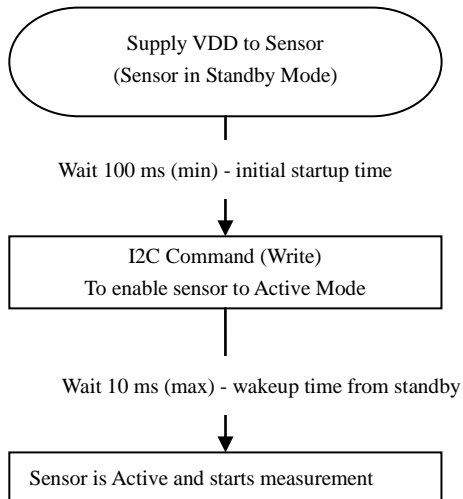
## LUX Formula

Refer to Appendix A for the lux formula

## AC Electrical Characteristics

All specifications are at VBus = 1.8V, $T_{ope}$ = 25°C, unless otherwise noted.

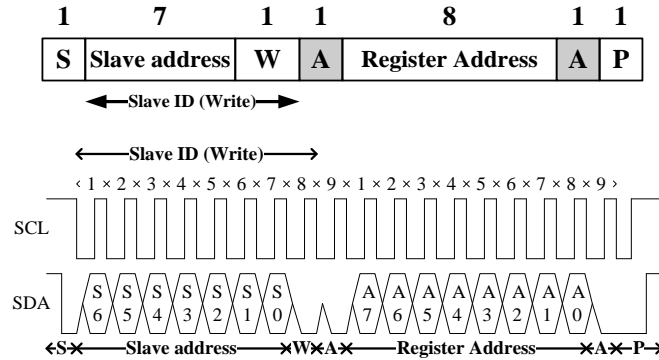| Parameter | Symbol | Min. | Max. | Unit |
|---|---|---|---|---|
| SCL clock frequency | $f_{SCL}$ | 1 | 400 | kHz |
| Bus free time between a STOP and START condition | $t_{BUF}$ | 1.3 | | us |
| Hold time (repeated) START condition. After this period, the first clock pulse is generated | $t_{HD;STA}$ | 0.6 | | us |
| LOW period of the SCL clock | $t_{LOW}$ | 1.3 | | us |
| HIGH period of the SCL clock | $t_{HIGH}$ | 0.6 | | us |
| Set-up time for a repeated START condition | $t_{SU;STA}$ | 0.6 | | us |
| Set-up time for STOP condition | $t_{SU;STO}$ | 0.6 | | us |
| Rise time of both SDA and SCL signals | $t_r$ | 30 | 300 | ns |
| Fall time of both SDA and SCL signals | $t_f$ | 30 | 300 | ns |
| Data hold time | $t_{HD;DAT}$ | 0.3 | 0.9 | us |
| Data setup time | $t_{SU;DAT}$ | 100 | | ns |
| Pulse width of spikes which must be suppressed by the input filter | $t_{SP}$ | 0 | 50 | ns |
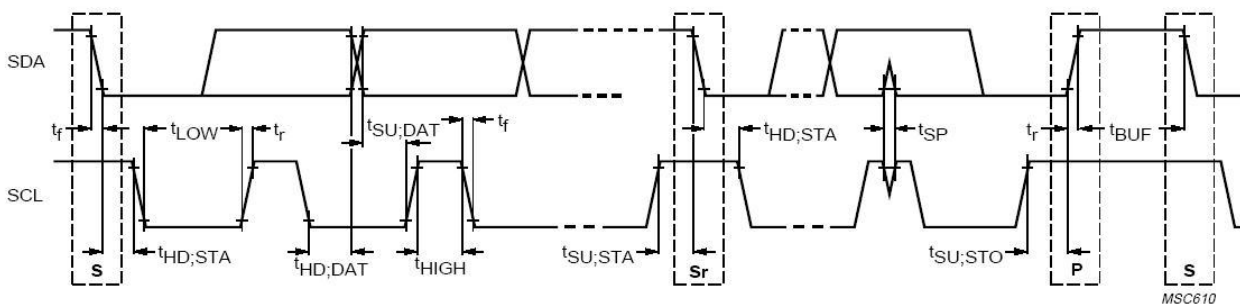


**Definition of timing for I$^2$C bus**

# 9. Principles of Operation

## I$^2$C Protocols

· **I$^2$C Write Protocol (type 1):**

| 1 | 7 | 1 | 1 | 8 | 1 | 1 |
|---|---|---|---|---|---|---|
| S | Slave address | W | A | Register Address | A | P |

Slave ID (Write)

Slave ID (Write)

‹ 1 × 2 × 3 × 4 × 5 × 6 × 7 × 8 × 9 × 1 × 2 × 3 × 4 × 5 × 6 × 7 × 8 × 9 ›

SCL

SDA

S6 S5 S4 S3 S2 S1 S0   A7 A6 A5 A4 A3 A2 A1 A0

‹S×——Slave address——×W×A×——Register Address——×A‹P→

· **I$^2$C Write Protocol (type 2):**

| 1 | 7 | 1 | 1 | 8 | 1 | 8 | 1 | 1 |
|---|---|---|---|---|---|---|---|---|
| S | Slave address | W | A | Register Address | A | Register Command | A | P |

Slave ID (Write)

Slave ID (Write)

‹ 1 × 2 × 3 × 4 × 5 × 6 × 7 × 8 × 9 × 1 × 2 × 3 × 4 × 5 × 6 × 7 × 8 × 9 × 1 × 2 × 3 × 4 × 5 × 6 × 7 × 8 × 9 ›

SCL

SDA

S6 S5 S4 S3 S2 S1 S0   A7 A6 A5 A4 A3 A2 A1 A0   A7 A6 A5 A4 A3 A2 A1 A0

‹S×——Slave address——×W×A×——Register Address——×A×——Register Command——×A‹P→

· **I$^2$C Read Protocol:**

| 1 | 7 | 1 | 1 | 8 | 1 | 1 |
|---|---|---|---|---|---|---|
| S | Slave address | R | A | Register Command | N | P |

Slave ID (Read)

Slave ID (Read)

‹ 1 × 2 × 3 × 4 × 5 × 6 × 7 × 8 × 9 × 1 × 2 × 3 × 4 × 5 × 6 × 7 × 8 × 9 ›

SCL

SDA

S6 S5 S4 S3 S2 S1 S0   C7 C6 C5 C4 C3 C2 C1 C0

‹S×——Slave address——×R×A×——Register Command——×N‹P→

· I²C Read (Combined format) Protocol:
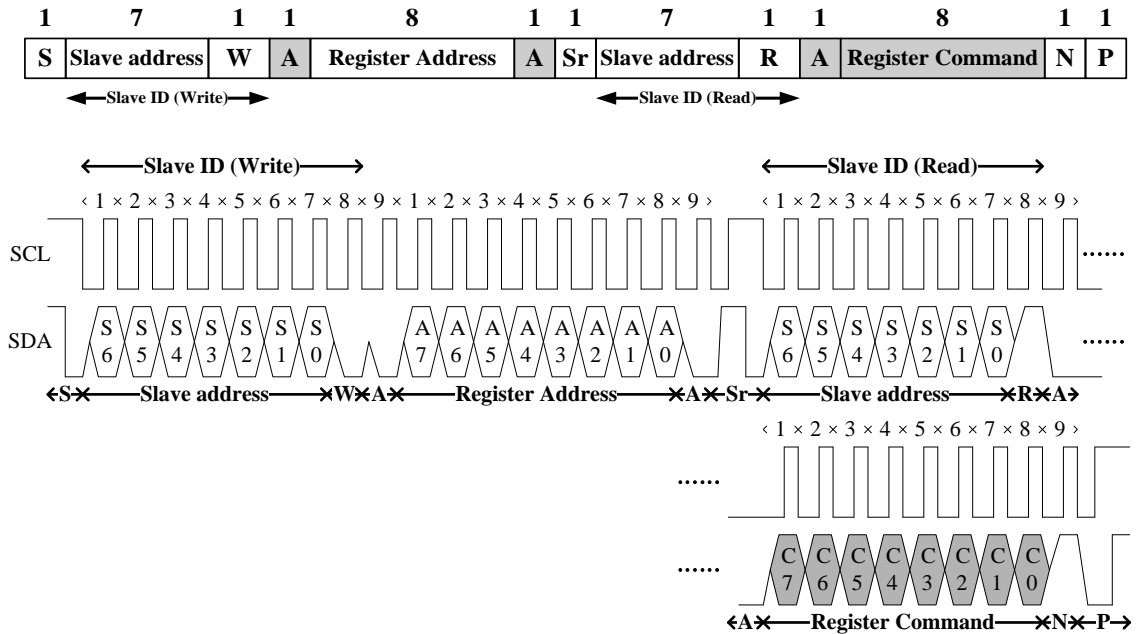


| | | |
|---|---|---|
| **A** | Acknowledge (0 for an ACK) | |
| **S** | Start condition | |
| **P** | Stop condition | |
| **W** | Write (0 for writing) | |
| | Slave-to-master | |

| | |
|---|---|
| **N** | Non-Acknowledge(1 for an NACK) |
| **Sr** | Repeated Start condition |
| **R** | Read (1 for read) |
| | Master-to-Slave |

## I²C Slave Address

The 7 bits slave address for this sensor is 0x23H. A read/write bit should be appended to the slave address by the master device to properly communicate with the sensor.

| I²C Slave Address | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Command Type** | **(0x29H)** | | | | | | | **W/R** | **value** |
| | **Bit7** | **Bit6** | **Bit5** | **Bit4** | **Bit3** | **Bit2** | **Bit1** | **Bit0** | |
| **Write** | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0x46H |
| **Read** | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0x47H |

## Register Set

| Addr | R / W | Register Name | Description | Reset Value |
|---|---|---|---|---|
| 0x80 | R/W | ALS_CONTR | ALS operation mode control SW reset | 0x00 |
| 0x81 | R/W | PS_CONTR | PS operation mode control | 0x00 |
| 0x82 | R/W | PS_LED | PS LED setting | 0x7F |
| 0x83 | R/W | PS_N_PULSES | PS number of pulses | 0x01 |
| 0x84 | R/W | PS_MEAS_RATE | PS measurement rate in active mode | 0x02 |
| 0x85 | R/W | ALS_MEAS_RATE | ALS measurement rate in active mode | 0x03 |
| 0x86 | R | PART_ID | Part Number ID and Revision ID | 0x92 |
| 0x87 | R | MANUFAC_ID | Manufacturer ID | 0x05 |
| 0x88 | R | ALS_DATA_CH1_0 | ALS measurement CH1 data, lower byte | 0x00 |
| 0x89 | R | ALS_DATA_CH1_1 | ALS measurement CH1 data, upper byte | 0x00 |
| 0x8A | R | ALS_DATA_CH0_0 | ALS measurement CH0 data, lower byte | 0x00 |
| 0x8B | R | ALS_DATA_CH0_1 | ALS measurement CH0 data, upper byte | 0x00 |
| 0x8C | R | ALS_PS_STATUS | ALS and PS new data status | 0x00 |
| 0x8D | R | PS_DATA_0 | PS measurement data, lower byte | 0x00 |
| 0x8E | R | PS_DATA_1 | PS measurement data, upper byte | 0x00 |
| 0x8F | R/W | INTERRUPT | Interrupt settings | 0x00 |
| 0x90 | R/W | PS_THRES_UP_0 | PS interrupt upper threshold, lower byte | 0xFF |
| 0x91 | R/W | PS_THRES_UP_1 | PS interrupt upper threshold, upper byte | 0x07 |
| 0x92 | R/W | PS_THRES_LOW_0 | PS interrupt lower threshold, lower byte | 0x00 |
| 0x93 | R/W | PS_THRES_LOW_1 | PS interrupt lower threshold, upper byte | 0x00 |
| 0x94 | R/W | PS_OFFSET_1 | PS offset, upper byte | 0x00 |
| 0x95 | R/W | PS_OFFSET_0 | PS offset, lower byte | 0x00 |
| 0x97 | R/W | ALS_THRES_UP_0 | ALS interrupt upper threshold, lower byte | 0xFF |
| 0x98 | R/W | ALS_THRES_UP_1 | ALS interrupt upper threshold, upper byte | 0xFF |
| 0x99 | R/W | ALS_THRES_LOW_0 | ALS interrupt lower threshold, lower byte | 0x00 |
| 0x9A | R/W | ALS_THRES_LOW_1 | ALS interrupt lower threshold, upper byte | 0x00 |
| 0x9E | R/W | INTERRUPT PERSIST | ALS / PS Interrupt persist setting | 0x00 |

*Notes:*

1)  *When reading ALS data registers, read sequence should always be from lower address to higher address (E.g. For ALS data, Ch1 data should be read first followed by Ch0 data. Read sequence should be 0x88, 0x89, 0x8A, 0x8B. When 0x8B is read, all four ALS data registers will be populated with new set of data).*

2)  *When setting of INTERRUPT register (addr 0x8F) is necessary, it should be done before the device is in Active mode.*

**ALS_CONTR Register (0x80)**

The ALS_CONTR register controls the ALS Gain setting, ALS operation modes and software (SW) reset for the sensor. The ALS sensor can be set to either standby mode or active mode. At either of these modes, the I$^2$C circuitry is always active. The default mode after power up is standby mode. During standby mode, there is no ALS measurement performed but I$^2$C communication is allowed to enable read/write to all the registers.

| 0x80 | ALS_CONTR (default = 0x00) | | | | | | | |
|------|------|------|------|------|------|------|------|------|
|      | **B7** | **B6** | **B5** | **B4** | **B3** | **B2** | **B1** | **B0** |
|      | *Reserved* | | | *ALS Gain* | | | *SW Reset* | *ALS Mode* |

| Field | Bits | Default | Type | Description | |
|-------|------|---------|------|----|----|
| Reserved | 7:5 | 000 | -- | -- | MUST write to 000 (default) |
| ALS Gain | 4:2 | 000 | RW | 000 | Gain 1X ➔ *1 lux to 64k lux* (default) |
|          |     |     |    | 001 | Gain 2X ➔ *0.5 lux to 32k lux* |
|          |     |     |    | 010 | Gain 4X ➔ *0.25 lux to 16k lux* |
|          |     |     |    | 011 | Gain 8X ➔ *0.125 lux to 8k lux* |
|          |     |     |    | 100 | Reserved |
|          |     |     |    | 101 | Reserved |
|          |     |     |    | 110 | Gain 48X ➔ *0.02 lux to 1.3k lux* |
|          |     |     |    | 111 | Gain 96X ➔ *0.01 lux to 600 lux* |
| SW reset | 1 | 0 | RW | 0 | Initial start-up procedure is NOT started (default) |
|          |   |   |    | 1 | Initial start-up procedure is started, bit has default value of 0 after start-up |
| ALS mode | 0 | 0 | RW | 0 | Stand-by mode (default) |
|          |   |   |    | 1 | Active mode |

**PS_CONTR Register (0x81)**

The PS_CONTR register controls the PS operation modes. The PS sensor can be set to either standby mode or active mode. At either of these modes, the $I^2C$ circuitry is always active. The default mode after power up is standby mode. During standby mode, there is no PS measurement performed but $I^2C$ communication is allowed to enable read/write to all the registers.

PS Saturation Indicator Enable bit is used for enabling the saturation indicator in Bit 7 of PS_DATA register (0x8E).

| 0x81 | PS_CONTR (default = 0x00) | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | **B7** | **B6** | **B5** | **B4** | **B3** | **B2** | **B1** | **B0** |
| | *Reserved* | | *PS Saturation Indicator Enable* | *Reserved* | | | *PS Mode* | |

| Field | Bits | Default | Type | Description | |
|---|---|---|---|---|---|
| Reserved | 7:6 | 00 | -- | -- | MUST write to 00 |
| PS Saturation Indicator Enable | 5 | 0 | RW | 0 | Saturation indicator disable |
| | | | | 1 | Saturation indicator enable |
| Reserved | 4:2 | 000 | -- | -- | Must be 0 |
| PS Mode | 1:0 | 00 | RW | 00 | Stand-by mode (default) |
| | | | | 01 | |
| | | | | 10 | Active mode |
| | | | | 11 | |

**PS_LED Register (0x82)**

The PS_LED register controls the LED pulse modulation frequency, LED current duty cycle and LED peak current.

| 0x82 | PS_LED (default = 0x7F) | | | | | | |
|---|---|---|---|---|---|---|---|
| | **B7** | **B6** | **B5** | **B4** | **B3** | **B2** | **B1** | **B0** |
| | *LED Pulse Frequency* | | | *LED Duty Cycle* | | *LED Peak Current* | | |

| Field | Bits | Default | Type | Description | |
|---|---|---|---|---|---|
| LED Pulse Modulation Frequency | 7:5 | 011 | RW | 000 | LED pulse period = 30kHz |
| | | | | 001 | LED pulse period = 40kHz |
| | | | | 010 | LED pulse period = 50kHz |
| | | | | 011 | LED pulse period = 60kHz (default) |
| | | | | 100 | LED pulse period = 70kHz |
| | | | | 101 | LED pulse period = 80kHz |
| | | | | 110 | LED pulse period = 90kHz |
| | | | | 111 | LED pulse period = 100kHz |
| LED Duty Cycle | 4:3 | 11 | RW | 00 | DUTY = 25% |
| | | | | 01 | DUTY = 50% |
| | | | | 10 | DUTY = 75% |
| | | | | 11 | DUTY = 100% (default) |
| LED Current | 2:0 | 111 | RW | 000 | LED pulsed current level = 5mA |
| | | | | 001 | LED pulsed current level = 10mA |
| | | | | 010 | LED pulsed current level = 20mA |
| | | | | 011 | LED pulsed current level = 50mA |
| | | | | 100 | LED pulsed current level = 100mA (default) |
| | | | | 101 | |
| | | | | 110 | |
| | | | | 111 | |

Part No. : LTR-559ALS-01
BNS-OD-FC002/A4

**PS_N_Pulses Register (0x83)**

The PS_N_Pulses register controls the number of LED pulses to be emitted.

| 0x83 | PS_N_Pulses (default = 0x01) | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | **B7** | **B6** | **B5** | **B4** | **B3** | **B2** | **B1** | **B0** |
| | *Reserved* | | | | *LED Pulse Count* | | | |

| Field | Bits | Default | Type | Description | |
|---|---|---|---|---|---|
| Reserved | 7:4 | 0000 | -- | -- | 0000 (default) |
| PS number of LED pulses | 3:0 | 0001 | RW | 0000 | RESERVED |
| | | | | 0001 | Number of pulses = 1 (default) |
| | | | | 0010 | Number of pulses = 2 |
| | | | | 0011 | Number of pulses = 3 |
| | | | | 0100 | Number of pulses = 4 |
| | | | | 0101 | Number of pulses = 5 |
| | | | | 0110 | Number of pulses = 6 |
| | | | | 0111 | Number of pulses = 7 |
| | | | | 1000 | Number of pulses = 8 |
| | | | | 1001 | Number of pulses = 9 |
| | | | | 1010 | Number of pulses = 10 |
| | | | | 1011 | Number of pulses = 11 |
| | | | | 1100 | Number of pulses = 12 |
| | | | | 1101 | Number of pulses = 13 |
| | | | | 1110 | Number of pulses = 14 |
| | | | | 1111 | Number of pulses = 15 |

**PS_MEAS_RATE Register (0x84)**

The PS_MEAS_RATE register controls the timing of the periodic measurements of the PS in active mode. PS Measurement Repeat Rate is the interval between PS_DATA registers update.

| 0x84 | PS_MEAS_RATE (default = 0x02) | | | | | | | |
|------|-----|-----|-----|-----|-----|-----|-----|-----|
|  | **B7** | **B6** | **B5** | **B4** | **B3** | **B2** | **B1** | **B0** |
|  | *Reserved* | | | | *PS Measurement Repeat Rate* | | | |

| Field | Bits | Default | Type | Description | |
|-------|------|---------|------|-------------|---|
| Reserved | 7:4 | 0000 | -- | -- | 0000 (default) |
| PS measurement rate | 3:0 | 0010 | RW | 0000 | 50ms |
| | | | | 0001 | 70ms |
| | | | | 0010 | 100ms (default) |
| | | | | 0011 | 200ms |
| | | | | 0100 | 500ms |
| | | | | 0101 | 1000ms |
| | | | | 0110 | 2000ms |
| | | | | 0111 | |
| | | | | 1XXX | 10ms |

**ALS_MEAS_RATE Register (0x85)**

The ALS_MEAS_RATE register controls the integration time and timing of the periodic measurement of the ALS in active mode. ALS Measurement Repeat Rate is the interval between ALS_DATA registers update. ALS Integration Time is the measurement time for each ALS cycle.

ALS Measurement Repeat Rate must be set to be equal or larger than the ALS Integration Time. If ALS Measurement Repeat Rate is set to be smaller than ALS Integration Time, it will automatically be reset to be equal to ALS Integration Time by the IC internally.

| 0x85 | ALS_MEAS_RATE (default = 0x03) | | | | | | |
|------|------|------|------|------|------|------|------|
|  | **B7** | **B6** | **B5** | **B4** | **B3** | **B2** | **B1** | **B0** |
|  | *Reserved* | | *ALS Integration Time* | | | *ALS Measurement Repeat Rate* | | |

| Field | Bits | Default | Type | Description | |
|-------|------|---------|------|-------------|---|
| Reserved | 7:6 | 00 | -- | -- | -- |
| ALS integration time | 5:3 | 000 | RW | 000 | 100ms (default) |
|  |  |  |  | 001 | 50ms |
|  |  |  |  | 010 | 200ms |
|  |  |  |  | 011 | 400ms |
|  |  |  |  | 100 | 150ms |
|  |  |  |  | 101 | 250ms |
|  |  |  |  | 110 | 300ms |
|  |  |  |  | 111 | 350ms |
| ALS measurement rate | 2:0 | 011 | RW | 000 | 50ms |
|  |  |  |  | 001 | 100ms |
|  |  |  |  | 010 | 200ms |
|  |  |  |  | 011 | 500ms (default) |
|  |  |  |  | 100 | 1000ms |
|  |  |  |  | 101 | 2000ms |
|  |  |  |  | 110 |  |
|  |  |  |  | 111 |  |

## PART_ID Register (0x86) (Read Only)

The PART_ID register defines the part number and revision identification of the sensor.

| 0x86 | PART_ID (default = 0x92) | | | | | | | |
|------|------|------|------|------|------|------|------|------|
| | **B7** | **B6** | **B5** | **B4** | **B3** | **B2** | **B1** | **B0** |
| | *Part Number ID* | | | | *Revision ID* | | | |

| Field | Bits | Default | Type | Description |
|-------|------|---------|------|-------------|
| Part Number ID | 7:4 | 1001 | R | Part ID 0x09H |
| Revision ID | 3:0 | 0010 | R | Revision ID 0x02H |

## MANUFAC_ID Register (0x87) (Read Only)

The MANUFAC_ID register defines the manufacturer identification of the sensor.

| 0x87 | MANUFAC_ID (default = 0x05) | | | | | | | |
|------|------|------|------|------|------|------|------|------|
| | **B7** | **B6** | **B5** | **B4** | **B3** | **B2** | **B1** | **B0** |
| | *Manufacturer ID* | | | | | | | |

| Field | Bits | Default | Type | Description |
|-------|------|---------|------|-------------|
| Manufacturer ID | 7:0 | 00000101 | R | Manufacturer ID (0x05H) |

**ALS_DATA_CH1 Register (0x88 / 0x89) (Read Only)**

The ALS_DATA registers should be read as a group, with the lower address read back first (i.e. read 0x88 first, then read 0x89). These two registers should also be read before reading channel-0 data (from registers 0x8A, 0x8B).

When the I$^2$C read operation starts, all four ALS data registers are locked until the I$^2$C read operation of register 0x8B is completed. This will ensure that the data in the registers is from the same measurement even if an additional integration cycle ends during the read operation. New measurement data is stored into temporary registers and the ALS_DATA registers are updated as soon as there is no on-going I$^2$C read operation.

The ALS ADC channel-1 data is expressed as a 16-bit data spread over two registers. The ALS_DATA_CH1_0 and ALS_DATA_CH1_1 registers provide the lower and upper byte respectively.

| 0x88 | ALS_DATA_CH1_0 (default = 0x00) | | | | | | | |
|------|------|------|------|------|------|------|------|------|
|      | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|      | *ALS Data Ch1 Low* | | | | | | | |

| 0x89 | ALS_DATA_CH1_1 (default = 0x00) | | | | | | | |
|------|------|------|------|------|------|------|------|------|
|      | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|      | *ALS Data Ch1 High* | | | | | | | |

| Field | Address | Bits | Default | Type | Description |
|-------|---------|------|---------|------|-------------|
| ALS Data Ch1 Low | 0x88 | 7:0 | 00000000 | R | ALS ADC channel 1 lower byte data |
| ALS Data Ch1 High | 0x89 | 7:0 | 00000000 | R | ALS ADC channel 1 upper byte data |

**ALS_DATA_CH0 Register (0x8A / 0x8B) (Read Only)**

These two registers should be read after reading channel-1 data (from registers 0x88, 0x89). Lower address register should be read first (i.e. read 0x8A first, then read 0x8B). See ALS_DATA_CH1 register information above. The ALS ADC channel-0 data is expressed as a 16-bit data spread over two registers. The ALS_DATA_CH0_0 and ALS_DATA_CH0_1 registers provide the lower and upper byte respectively.

| 0x8A | ALS_DATA_CH0_0 (default = 0x00) | | | | | | | |
|------|------|------|------|------|------|------|------|------|
|  | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|  | ALS Data Ch0 Low | | | | | | | |

| 0x8B | ALS_DATA_CH0_1 (default = 0x00) | | | | | | | |
|------|------|------|------|------|------|------|------|------|
|  | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|  | ALS Data Ch0 High | | | | | | | |

| Field | Address | Bits | Default | Type | Description |
|-------|---------|------|---------|------|-------------|
| ALS Data Ch0 Low | 0x8A | 7:0 | 00000000 | R | ALS ADC channel 0 lower byte data |
| ALS Data Ch0 High | 0x8B | 7:0 | 00000000 | R | ALS ADC channel 0 upper byte data |

**Optical Sensor**
**LTR-559ALS-01**

**ALS_ PS_STATUS Register (0x8C) (Read Only)**

The ALS_PS_STATUS register stores the information about ALS data status.  New data means data has not been read yet. When the measurement is completed and data is written to the data register, the data status bit will be set to logic 1. When the data register is read, the data status bit will be set to logic 0.

Interrupt status determines if the ALS interrupt criteria are met. It will check if the ALS measurement data is outside of the range defined by the upper and lower threshold limits.

ALS Data Valid bit indicates the validity of the ALS data in Ch0 and Ch1.

| 0x8C | ALS_PS_STATUS (default = 0x00) | | | | | | |
|---|---|---|---|---|---|---|---|
| | **B7** | **B6** | **B5** | **B4** | **B3** | **B2** | **B1** | **B0** |
| | *ALS Data Valid* | *ALS Gain* | | | *ALS Interrupt Status* | *ALS Data Status* | *PS Interrupt Status* | *PS Data Status* |

| Field | Bits | Default | Type | Description | |
|---|---|---|---|---|---|
| ALS Data Valid | 7 | 0 | R | 0 | ALS Data is Valid (default) |
| | | | | 1 | ALS Data is Invalid |
| ALS Data Gain Range | 6:4 | 000 | R | 000 | ALS measured data in Gain 1X (default) |
| | | | | 001 | ALS measured data in Gain 2X |
| | | | | 010 | ALS measured data in Gain 4X |
| | | | | 011 | ALS measured data in Gain 8X |
| | | | | 100 | Invalid |
| | | | | 101 | Invalid |
| | | | | 110 | ALS measured data in Gain 48X |
| | | | | 111 | ALS measured data in Gain 96X |
| ALS interrupt status | 3 | 0 | R | 0 | Interrupt signal INACTIVE (default) |
| | | | | | Interrupt signal ACTIVE |
| ALS data status | 2 | 0 | R | 0 | OLD data (data already read), (default) |
| | | | | 1 | NEW data (first time data is being read) |
| PS Interrupt Status | 1 | 0 | R | 0 | Interrupt signal INACTIVE (default) |
| | | | | 1 | Interrupt signal ACTIVE |
| PS Data Status | 0 | 0 | R | 0 | OLD data (data already read), (default) |
| | | | | 1 | NEW data (first time data is being read) |

Part No. : LTR-559ALS-01
BNS-OD-FC002/A4

## PS_DATA_0 Register (0x8D / 0x8E) (Read Only)

The PS ADC channel data are expressed as a 11-bit data spread over two registers. The PS_DATA_0 and PS_DATA_1 registers provide the lower and upper byte respectively. When the $I^2C$ read operation starts, both the registers are locked until the $I^2C$ read operation is completed.   This will ensure that the data in the registers is from the same measurement even if an additional integration cycle ends during the read operation.   New measurement data is stored into temporary registers and the PS_DATA registers are updated as soon as there is no on-going $I^2C$ read operation.

PS Saturation Flag is used for monitoring the internal IC saturation. It will be flagged when the IC has reached saturation and not able to perform any further PS measurement. The PS Saturation Indicator Enable bit in PS_CONTR Register (0x81) has to be enabled in order to use this feature. If it is not enable, the flag will always be indicated as 0.

| 0x8D | PS_DATA_0 (default = 0x00) | | | | | | | |
|------|----|----|----|----|----|----|----|----|
|  | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|  | *PS Data Low* | | | | | | | |

| 0x8E | PS_DATA_1 (default = 0x00) | | | | | | | |
|------|----|----|----|----|----|----|----|----|
|  | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|  | *PS Saturation Flag* | *Reserved* | | | | *PS Data High* | | |

| Field | Address | Bits | Default | Type | | Description |
|-------|---------|------|---------|------|----|-------------|
| PS Data, Low | 0x8D | 7:0 | 00000000 | R | -- | PS ADC lower byte data |
| PS Saturation Flag | 0x8E | 7 | 0 | R | 0 | PS not saturated |
|  |  |  |  |  | 1 | PS saturated |
| Reserved | 0x8E | 6:3 | 00000 | -- | -- | -- |
| PS Data, High | 0x8E | 2:0 | 000 | R | -- | PS ADC upper byte data |

**INTERRUTP Register (0x8F) (Read Only)**

The INTERRUPT register controls the operation of the interrupt pin and functions. When the Interrupt Mode is set to 00, the INT output pin 2 is inactive / disabled and will not trigger any interrupt. However at this condition, the ALS_PS_STATUS register will still be updated.

Note that when this register is to be set with values other than its default values, it should be set before device is in Active mode.

| 0x8F | INTERRUPT (default = 0x00) | | | | | | | |
|------|------|------|------|------|------|------|------|------|
|  | **B7** | **B6** | **B5** | **B4** | **B3** | **B2** | **B1** | **B0** |
|  | *Reserved* | | | | | *Interrupt Polarity* | *Interrupt Mode* | |

| Field | Bits | Default | Type | Description | |
|-------|------|---------|------|-------------|---|
| RESERVED | 7:3 | 00000 | -- | -- | Must write as 00000 |
| Interrupt Polarity | 2 | 0 | RW | 0 | INT pin is considered active when it is a logic 0 (default) |
|  |  |  |  | 1 | INT pin is considered active when it is a logic 1 |
| Interrupt Mode | 1:0 | 0 | RW | 00 | Interrupt pin is INACTIVE / high impedance state (default) |
|  |  |  |  | 01 | Only PS measurement can trigger interrupt |
|  |  |  |  | 10 | Only ALS measurement can trigger interrupt |
|  |  |  |  | 11 | Both ALS and PS measurement can trigger interrupt |

**PS_THRES Register (0x90 / 0x91 / 0x92 / 0x93)**

The PS_THRES_UP and PS_THRES_LOW registers determines the upper and lower limit of the interrupt threshold value respectively. These two values form a range and the interrupt function compares if the measurement value in PS_DATA registers is inside or outside the range. The interrupt function is active if the measurement data is outside the range defined by the upper and lower limits. The data format for PS_THRES must be the same as PS_DATA registers. These registers must be applied dynamically to have hysteresis interruption.

| 0x90 | PS_THRES_UP_0 (default = 0xFF) | | | | | | | |
|------|------|------|------|------|------|------|------|------|
|      | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|      | PS Upper Threshold Low | | | | | | | |

| 0x91 | PS_THRES_UP_1 (default = 0x07) | | | | | | | |
|------|------|------|------|------|------|------|------|------|
|      | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|      | Reserved | | | | PS Upper Threshold High | | | |

| 0x92 | PS_THRES_LOW_0 (default = 0x00) | | | | | | | |
|------|------|------|------|------|------|------|------|------|
|      | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|      | PS Lower Threshold Low | | | | | | | |

| 0x93 | PS_THRES_LOW_1 (default = 0x00) | | | | | | | |
|------|------|------|------|------|------|------|------|------|
|      | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|      | Reserved | | | | PS Lower Threshold High | | | |

Part No. : LTR-559ALS-01
BNS-OD-FC002/A4

| Field | Address | Bits | Default | Type | Description |
|-------|---------|------|---------|------|-------------|
| PS Upper Threshold Low | 0x90 | 7:0 | 11111111 | RW | PS upper threshold lower byte data |
| Reserved | 0x91 | 7:3 | 00000 | -- | Reserved |
| PS Upper Threshold High | 0x91 | 2:0 | 111 | RW | PS upper threshold upper byte data |
| PS lower threshold, Low | 0x92 | 7:0 | 00000000 | RW | PS lower threshold lower byte data |
| Reserved | 0x93 | 7:3 | 00000 | -- | Reserved |
| PS lower threshold High | 0x93 | 2:0 | 000 | RW | PS lower threshold upper byte data |

**PS_OFFSET Register (0x94 / 0x95)**

The PS_OFFSET register defines the offset compensation value for proximity offsets caused by device variations, optical crosstalk and other environment factors. This value will be used and cancelled from the original PS raw data such that the data in PS_DATA register (0x8D and 0x8E) are the compensated value.

| 0x94 | PS_OFFSET_1 (default = 0x00) | | | | | | | |
|------|------|------|------|------|------|------|------|------|
|      | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|      | *Reserved* | | | | | | *PS Offset, High byte* | |

| 0x95 | PS_OFFSET_0 (default = 0x00) | | | | | | | |
|------|------|------|------|------|------|------|------|------|
|      | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|      | *PS Offset, Low byte* | | | | | | | |

**ALS_THRES Register (0x97 / 0x98 / 0x99 / 0x9A)**

The ALS_THRES_UP and ALS_THRES_LOW registers determines the upper and lower limit of the interrupt threshold value respectively. These two values form a range and the interrupt function compares if the measurement value in ALS_DATA registers is inside or outside the range. The interrupt function is active if the measurement data is outside the range defined by the upper and lower limits. The data format for ALS_THRES must be the same as ALS_DATA registers.

| 0x97 | ALS_THRES_UP_0 (default = 0xFF) | | | | | | | |
|------|------|------|------|------|------|------|------|------|
|  | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|  | *ALS Upper Threshold Low* | | | | | | | |

| 0x98 | ALS_THRES_UP_1 (default = 0xFF) | | | | | | | |
|------|------|------|------|------|------|------|------|------|
|  | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|  | *ALS Upper Threshold High* | | | | | | | |

| 0x99 | ALS_THRES_LOW_0 (default = 0x00) | | | | | | | |
|------|------|------|------|------|------|------|------|------|
|  | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|  | *ALS Lower Threshold Low* | | | | | | | |

| 0x9A | ALS_THRES_LOW_1 (default = 0x00) | | | | | | | |
|------|------|------|------|------|------|------|------|------|
|  | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|  | *ALS Lower Threshold High* | | | | | | | |

| Field | Address | Bits | Default | Type | Description |
|---|---|---|---|---|---|
| ALS upper threshold Low | 0x97 | 7:0 | 11111111 | RW | ALS upper threshold lower byte data |
| ALS upper threshold High | 0x98 | 7:0 | 11111111 | RW | ALS upper threshold upper byte data |
| ALS lower threshold Low | 0x99 | 7:0 | 00000000 | RW | ALS lower threshold lower byte data |
| ALS lower threshold High | 0x9A | 7:0 | 00000000 | RW | ALS lower threshold upper byte data |

**INTERRUPT PERSIST Register (0x9E)**

The INTERRUPT PERSIST register controls the N number of times the measurement data is outside the range defined by the upper and lower threshold limits before asserting the interrupt.

| 0x9E | INTERRUPT PERSIST (default = 0x00) | | | | | | | |
|------|------|------|------|------|------|------|------|------|
| | **B7** | **B6** | **B5** | **B4** | **B3** | **B2** | **B1** | **B0** |
| | *PS Persist* | | | | *ALS Persist* | | | |

| Field | Bits | Default | Type | Description | |
|-------|------|---------|------|------|------|
| PS Persist | 7:4 | 0000 | RW | 0000 | Every PS value out of threshold range (default) |
| | | | | 0001 | 2 consecutive PS values out of threshold range |
| | | | | …. | …. |
| | | | | 1111 | 16 consecutive PS values out of threshold range |
| ALS Persist | 3:0 | 0000 | RW | 0000 | Every ALS value out of threshold range (default) |
| | | | | 0001 | 2 consecutive ALS values out of threshold range |
| | | | | --- | --- |
| | | | | 1111 | 16 consecutive ALS values out of threshold range |

## 10. Device Operation (using Interrupt)

Below flow diagram illustrates the LTR-559ALS operation involving the use of Thresholds and Interrupts.
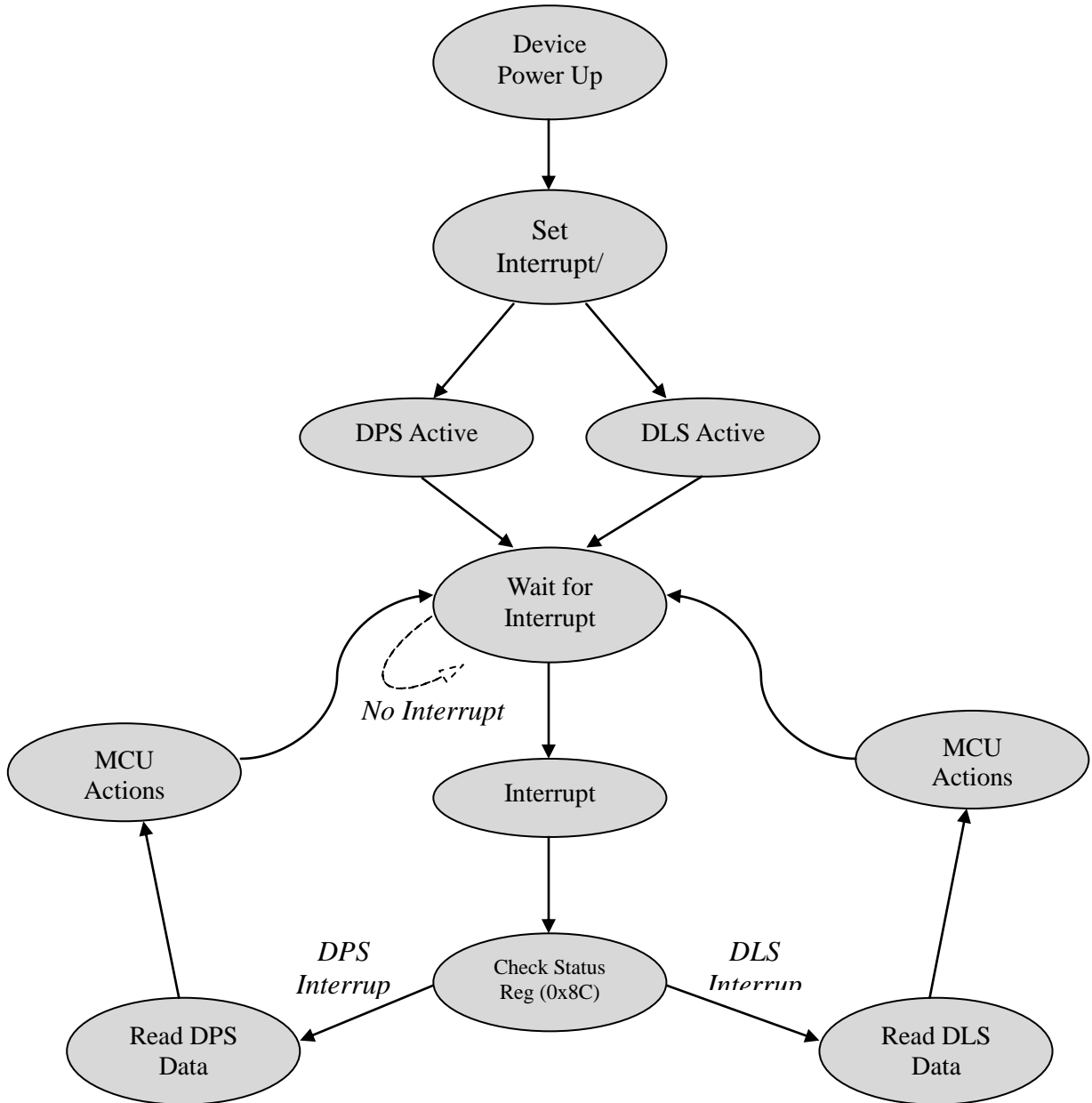
# 11. Pseudo Codes Examples

### Control Registers

// The Control Registers define the operating modes and gain settings of the ALS and PS of LTR-559.
// Default settings are 0x00 for both registers (both in Standby mode).

Slave_Addr = 0x23                      // Slave address of LTR-559 device

**// Enable ALS**
Register_Addr = 0x80              // ALS_CONTR register
Command = 0x01                   // For Gain X1
                                        // For Gain X2, Command = 0x05
                                        // For Gain X4, Command = 0x09
                                        // For Gain X8, Command = 0x0D
                                        // For Gain X48, Command = 0x19
                                        // For Gain X96, Command = 0x1D
WriteByte(Slave_Addr, Register_Addr, Command)

**// Enable PS**
Register_Addr = 0x81              // PS_CONTR register
Command = 0x03

WriteByte(Slave_Addr, Register_Addr, Command)

### PS LED Registers

// The PS LED Registers define the LED pulse modulation frequency, duty cycle and peak current.
// Default setting is 0x7F (60kHz, 100%, 100mA).

Slave_Addr = 0x23                      // Slave address of LTR-559 device

**// Set LED Pulse Freq 30kHz (duty cycle 100%, peak curr 100mA)**
Register_Addr = 0x82              // PS_LED register
Command = 0x1F                   // Pulse Freq = 30kHz, (duty cyc 100%, peak curr 100mA)
                                        // For Pulse Freq = 40kHz, (100%, 100mA), Command = 0x3F
                                        // For Pulse Freq = 50kHz, (100%, 100mA), Command = 0x5F
                                        // For Pulse Freq = 60kHz, (100%, 100mA), Command = 0x7F
                                        // For Pulse Freq = 70kHz, (100%, 100mA), Command = 0x9F
                                        // For Pulse Freq = 80kHz, (100%, 100mA), Command = 0xBF
                                        // For Pulse Freq = 90kHz, (100%, 100mA), Command = 0xDF
                                        // For Pulse Freq = 100kHz, (100%, 100mA), Command = 0xFF
WriteByte(Slave_Addr, Register_Addr, Command)

**// Set LED Duty Cycle 25% (pulse freq 60kHz, peak curr 100mA)**
Register_Addr = 0x82              // PS_LED register
Command = 0x67                   // Duty Cycle = 25%, (pulse freq 60kHz, peak curr 100mA)
                                          // For Duty Cycle = 50%, (60kHz, 100mA), Command = 0x6F
                                          // For Duty Cycle = 75%, (60kHz, 100mA), Command = 0x77
                                          // For Duty Cycle = 100%, (60kHz, 100mA), Command = 0x7F
WriteByte(Slave_Addr, Register_Addr, Command)

Part No. : LTR-559ALS-01
BNS-OD-FC002/A4

**// Set LED Peak Current 5mA (pulse freq 60kHz, duty cycle 100%)**
Register_Addr = 0x82                                    // PS_LED register
Command = 0x78                                          // Peak Current = 5mA, (pulse freq 60kHz, duty cyc 100%)
                                                        // For Peak Current = 10mA, (60kHz, 100%), Command = 0x79
                                                        // For Peak Current = 20mA, (60kHz, 100%), Command = 0x7A
                                                        // For Peak Current = 50mA, (60kHz, 100%), Command = 0x7B

WriteByte(Slave_Addr, Register_Addr, Command)


## PS Measurement Rate

// The PS_MEAS_RATE register controls the PS measurement rate.
// Default setting of the register is 0x02 (repeat rate 100ms)

Slave_Addr = 0x23                                       // Slave address of LTR-559 device

**// Set PS Repeat Rate 50ms**
Register_Addr = 0x84                                    // PS_MEAS_RATE register
Command = 0x00                                          // Meas rate = 50ms
                                                        // For Meas rate = 500ms, Command = 0x04

WriteByte(Slave_Addr, Register_Addr, Command)


## ALS Measurement Rate

// The ALS_MEAS_RATE register controls the ALS integration time and measurement rate.
// Default setting of the register is 0x03 (integration time 100ms, repeat rate 500ms)

Slave_Addr = 0x23                                       // Slave address of LTR-559 device

**// Set ALS Integration Time 200ms, Repeat Rate 200ms**
Register_Addr = 0x85                                    // ALS_MEAS_RATE register
Command = 0x12                                          // Int time = 200ms, Meas rate = 200ms
                                                        // For Int time = 400ms, Meas rate = 500ms, Command = 0x1B

WriteByte(Slave_Addr, Register_Addr, Command)


## ALS Data Registers (Read Only)

// The ALS Data Registers contain the ADC output data for the respective channel.
// These registers should be read as a group, with the lower address being read first.

Slave_Addr = 0x23                                       // Slave address of LTR-559 device

**// Read back ALS_DATA_CH1**
Register_Addr = 0x88                                    // ALS_DATA_CH1 low byte address
ReadByte(Slave_Addr, Register_Addr, Data0)
Register_Addr = 0x89                                    // ALS_DATA_CH1 high byte address
ReadByte(Slave_Addr, Register_Addr, Data1)

**// Read back ALS_DATA_CH0**
Register_Addr = 0x8A                                    // ALS_DATA_CH0 low byte address
ReadByte(Slave_Addr, Register_Addr, Data2)
Register_Addr = 0x8B                                    // ALS_DATA_CH0 high byte address
ReadByte(Slave_Addr, Register_Addr, Data3)

ALS_CH1_ADC_Data = (Data1 << 8) | Data0                // Combining lower and upper bytes to give 16-bit Ch1 data
ALS_CH0_ADC_Data = (Data3 << 8) | Data2                // Combining lower and upper bytes to give 16-bit Ch0 data

## ALS / PS Status Register (Read only)

// The ALS_PS_STATUS Register contains the information on Interrupt, ALS and PS data availability status.
// This register is read only.

Slave_Addr = 0x23           // Slave address of LTR-559 device

**// Read back Register**
Register_Addr = 0x8C       // ALS_PS_STATUS register address
ReadByte(Slave_Addr, Register_Addr, Data)

Interrupt_Status = Data & 0x0A    // Interrupt_Status = 8(decimal) ➔ ALS Interrupt
                                   // Interrupt_Status = 2(decimal) ➔ PS Interrupt
                                   // Interrupt_Status = 10(decimal) ➔ Both Interrupt

NewData_Status = Data & 0x05    // NewData_Status = 4(decimal) ➔ ALS New Data
                                   // NewData_Status = 1(decimal) ➔ PS New Data
                                   // NewData_Status = 5(decimal) ➔ Both New Data

ALS_Data_Valid = Data & 0x80    // ALS_Data_Valid = 0x00 ➔ ALS New Data is valid (usable)
                                   // ALS_Data_Valid = 0x80 ➔ ALS New Data is invalid, discard and
                                   // wait for new ALS data

## PS Data Registers (Read only)

// The PS Data Registers contain the ADC output data.
// These registers should be read as a group, with the lower address being read first.

Slave_Addr = 0x23           // Slave address of LTR-559 device

**// Read back PS_DATA registers**
Register_Addr = 0x8D       // PS_DATA low byte address
ReadByte(Slave_Addr, Register_Addr, Data0)
Register_Addr = 0x8E       // PS_DATA high byte address
ReadByte(Slave_Addr, Register_Addr, Data1)

PS_ADC_Data = (Data1 << 8) | Data0    // Combining lower and upper bytes to give 16-bit PS data

## Interrupt Registers

// The Interrupt register controls the operation of the interrupt pins and function.
// The default value for this register is 0x08 (Interrupt inactive)
// Note that when this register is to be set with values other than its default values,
// it should be set before device is in active mode.

Slave_Addr = 0x23           // Slave address of LTR-559 device

**// Set Interrupt Polarity for Active Low, both ALS and PS trigger**
Register_Addr = 0x8F       // Interrupt Register address
Command = 0x03             // Interrupt is Active Low and both ALS and PS can trigger
                                   // For Active High Interrupt, both trigger, Command = 0x07
                                   // For Active High Interrupt, ONLY ALS trigger, Command = 0x06
                                   // For Active High Interrupt, ONLY PS trigger, Command = 0x05

WriteByte(Slave_Addr, Register_Addr, Command)

## ALS Threshold Registers

// The ALS_THRES_UP and ALS_THRES_LOW registers determines the upper and lower limit of the interrupt threshold value.
// Following example illustrates the setting of the ALS threshold window of decimal values of 200 (lower threshold)
// and 1000 (upper threshold)

```
Slave_Addr = 0x23                          // Slave address of LTR-559 device

// Upper Threshold Setting (decimal 1000)
ALS_Upp_Threshold_Reg_0 = 0x97             // ALS Upper Threshold Low Byte Register address
ALS_Upp_Threshold_Reg_1 = 0x98            // ALS Upper Threshold High Byte Register address
Data1 = 1000 >> 8                          // To convert decimal 1000 into two eight bytes register values
Data0 = 1000 & 0xFF
WriteByte(Slave_Addr, ALS_Upp_Threshold_Reg_0, Data0)
WriteByte(Slave_Addr, ALS_Upp_Threshold_Reg_1, Data1)

// Lower Threshold Setting (decimal 200)
ALS_Low_Threshold_Reg_0 = 0x99             // ALS Lower Threshold Low Byte Register address
ALS_Low_Threshold_Reg_1 = 0x9A            // ALS Lower Threshold High Byte Register address
Data1 = 200 >> 8                           // To convert decimal 200 into two eight bytes register values
Data0 = 200 & 0xFF
WriteByte(Slave_Addr, ALS_Low_Threshold_Reg_0, Data0)
WriteByte(Slave_Addr, ALS_Low_Threshold_Reg_1, Data1)
```

## PS Threshold Registers

// The PS_THRES_UP and PS_THRES_LOW registers determines the upper and lower limit of the interrupt threshold value.
// Following example illustrates the setting of the PS dynamic threshold with hysteresis interruption for
// decimal value 1000 (for NEAR detection) and 500 (for FAR detection)

```
Slave_Addr = 0x23                          // Slave address of LTR-559 device

//For NEAR detection (decimal 1000)
PS_Upp_Threshold_Reg_0 = 0x90             // PS Upper Threshold Low Byte Register address
PS_Upp_Threshold_Reg_1 = 0x91             // PS Upper Threshold High Byte Register address
Data1 = 1000 >> 8                          // To convert decimal 1000 into two eight bytes register values
Data0 = 1000 & 0xFF
WriteByte(Slave_Addr, PS_Upp_Threshold_Reg_0, Data0)
WriteByte(Slave_Addr, PS_Upp_Threshold_Reg_1, Data1)

PS_Low_Threshold_Reg_0 = 0x92             // PS Lower Threshold Low Byte Register address
PS_Low_Threshold_Reg_1 = 0x93             // PS Lower Threshold High Byte Register address
Data1 = 0 >> 8                             // To convert decimal 0 into two eight bytes register values
Data0 = 0 & 0xFF
WriteByte(Slave_Addr, PS_Low_Threshold_Reg_0, Data0)
WriteByte(Slave_Addr, PS_Low_Threshold_Reg_1, Data1)

//For FAR detection (decimal 500)
PS_Upp_Threshold_Reg_0 = 0x90             // PS Upper Threshold Low Byte Register address
PS_Upp_Threshold_Reg_1 = 0x91             // PS Upper Threshold High Byte Register address
Data1 = 2047 >> 8                          // To convert decimal 2047 into two eight bytes register values
Data0 = 2047 & 0xFF
WriteByte(Slave_Addr, PS_Upp_Threshold_Reg_0, Data0)
WriteByte(Slave_Addr, PS_Upp_Threshold_Reg_1, Data1)

PS_Low_Threshold_Reg_0 = 0x92             // PS Lower Threshold Low Byte Register address
PS_Low_Threshold_Reg_1 = 0x93             // PS Lower Threshold High Byte Register address
Data1 = 500 >> 8                           // To convert decimal 500 into two eight bytes register values
Data0 = 500 & 0xFF
WriteByte(Slave_Addr, PS_Low_Threshold_Reg_0, Data0)
WriteByte(Slave_Addr, PS_Low_Threshold_Reg_1, Data1)
```
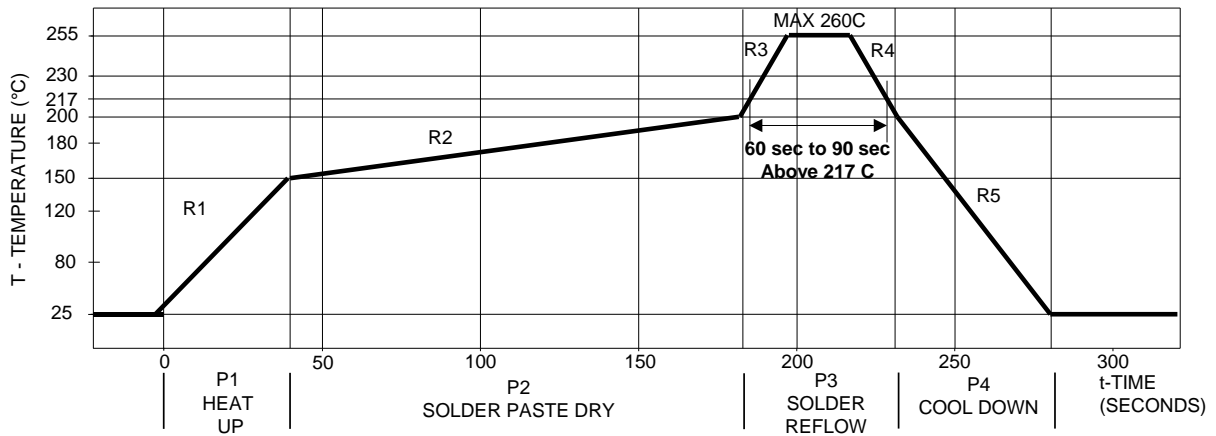
## 12. Recommended Leadfree Reflow Profile



| Process Zone | Symbol | ΔT | Maximum ΔT/Δtime or Duration |
|---|---|---|---|
| Heat Up | P1, R1 | 25°C to 150°C | 3°C/s |
| Solder Paste Dry | P2, R2 | 150°C to 200°C | 100s to 180s |
| Solder Reflow | P3, R3 | 200°C to 260°C | 3°C/s |
| | P3, R4 | 260°C to 200°C | -6°C/s |
| Cool Down | P4, R5 | 200°C to 25°C | -6°C/s |
| Time maintained above liquidus point , 217°C | | > 217°C | 60s to 90s |
| Peak Temperature | | 260°C | - |
| Time within 5°C of actual Peak Temperature | | > 255°C | 20s |
| Time 25°C to Peak Temperature | | 25°C to 260°C | 8mins |

It is recommended to perform reflow soldering no more than twice.

## 13. Moisture Proof Packaging

All LTR-559ALS-01 are shipped in moisture proof package. Once opened, moisture absorption begins. This part is compliant to JEDEC J-STD-033A Level 3.

**Time from Unsealing to Soldering**

After removal from the moisture barrier bag, the parts should be stored at the recommended storage conditions and soldered within seven days. When the moisture barrier bag is opened and the parts are exposed to the recommended storage conditions for more than seven days, the parts must be baked before reflow to prevent damage to the parts.

**Recommended Storage Conditions**

| Storage Temperature | 10°C to 30°C |
|---|---|
| Relative Humidity | Below 60% RH |

**Baking Conditions**

| Package | Temperature | Time |
|---|---|---|
| In Reels | 60°C | 48 hours |
| In Bulk | 100°C | 4 hours |

Baking should only be done once.

Part No. : LTR-559ALS-01
BNS-OD-FC002/A4

## 14. Recommended Land Pattern and Metal Stencil Aperture
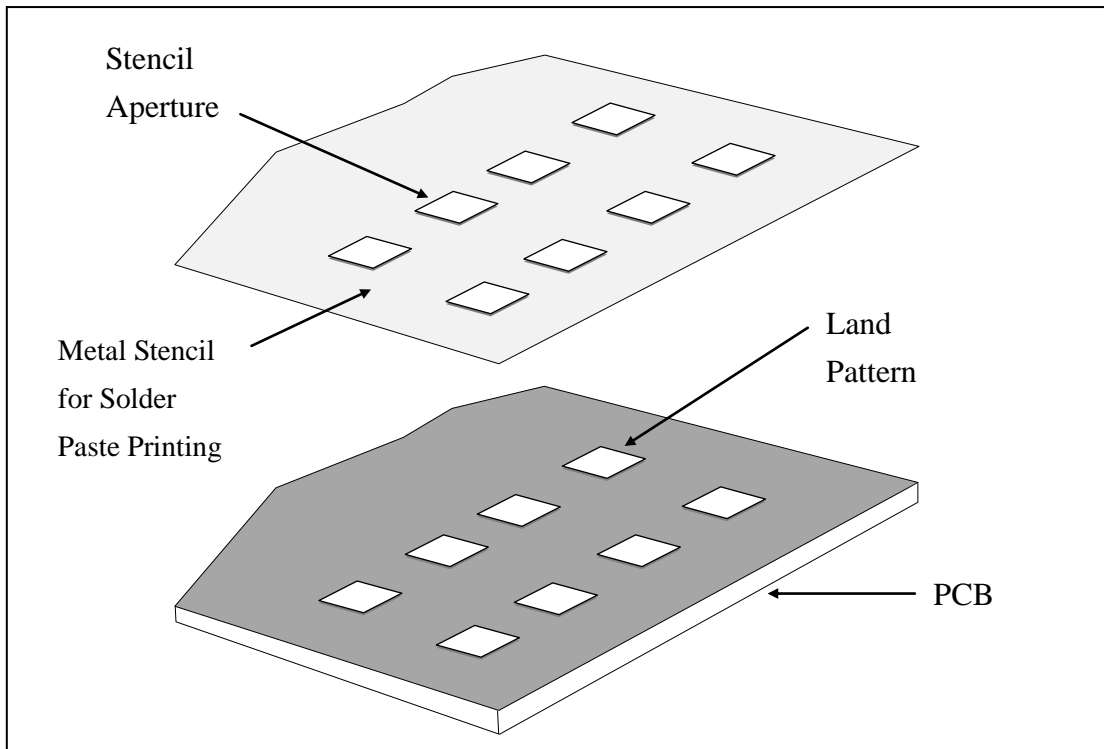


**Recommended Land Pattern**
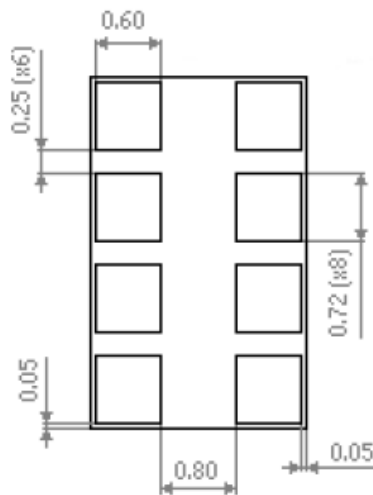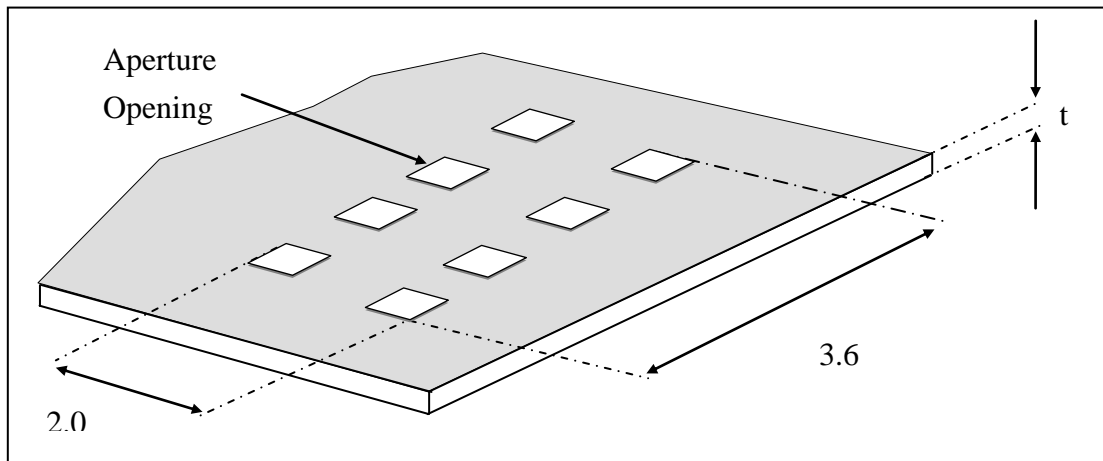


Note: All dimensions are in millimeters

**Recommended Metal Stencil Aperture**

It is recommended that the metal stencil used for solder paste printing has a thickness (t) of 0.11mm (0.004 inches / 4 mils) or 0.127mm (0.005 inches / 5 mils).

The stencil aperture opening is recommended to be 0.72mm x 0.60mm which has the same dimension as the land pattern.   This is to ensure adequate printed solder paste volume and yet no shorting.
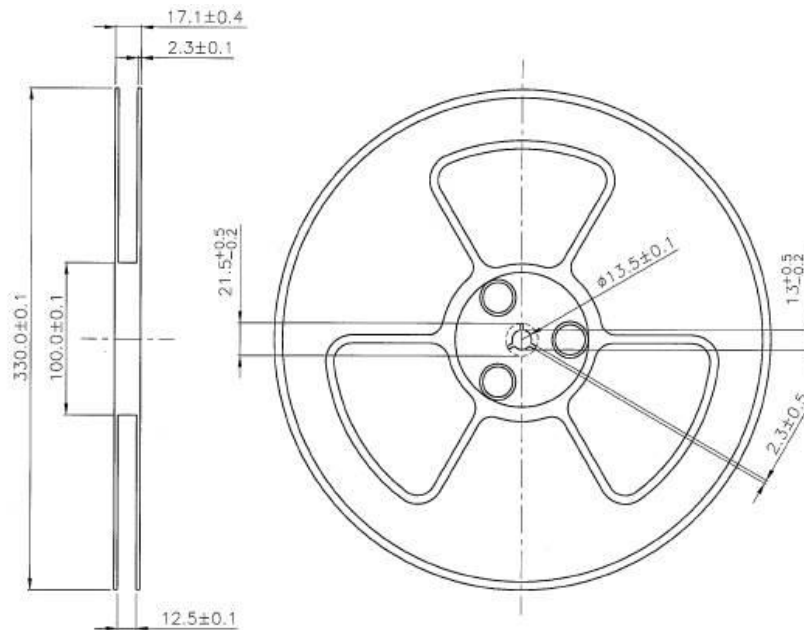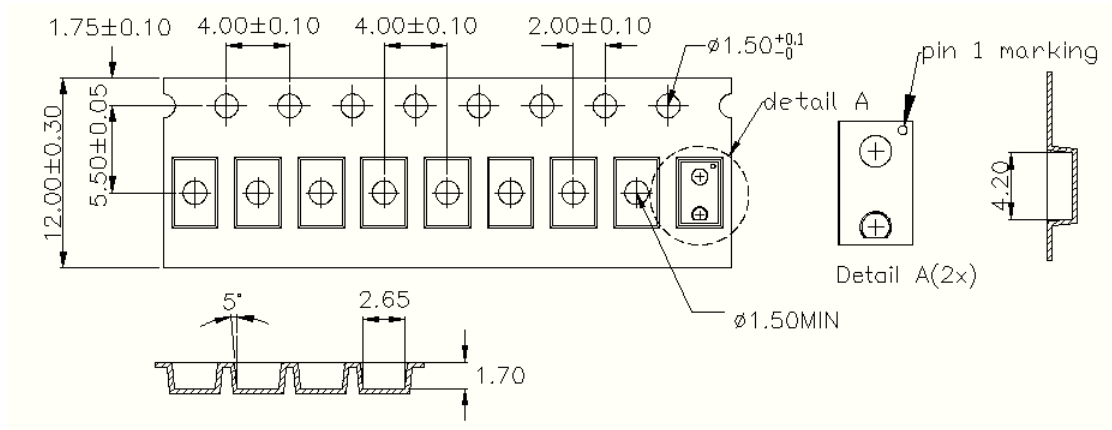
Aperture
Opening

t

3.6

2.0

Note:

1. All dimensions are in millimeters

## 15. Package Dimension for Tape and Reel





Notes:

1. All dimensions are in millimeters (inches)
2. Empty component pockets sealed with top cover tape
3. 13 inch reel - 8000 pieces per reel
4. In accordance with ANSI/EIA 481-1-A-1994 specifications