SFH 7770 E6
Ambient Light and Proximity Sensor

1 Abstract

This application note describes technical details and provides some application guidelines for the combined ambient light and proximity sensor SFH 7770 E6.
Compared to its predecessors (SFH 7770 E4 and E5), ***the SFH 7770 E6 features a mores sensitive proximity sensor*** (up to five times improved sensitivity) ***and improved ambient light sensor*** (sensitive down to 0.03 lx).

The document starts with a general introduction to the device, followed by a brief overview on the features (Sec. 2) and operating modes (Sec. 3) of the sensor. The integration and operation of the sensor in an I²C bus environment is described in Sec. 4, whereas Sec. 5 covers the interrupt capabilities of the SFH 7770 E6.
Sec. 6 provides a functional description of the sensor. Optical design guidelines and application relevant information are given in Sec. 7 followed by the guidelines for the electrical design in Sec. 8. Finally Sec. 9 presents a sample software code.

More general information about ambient light sensing, technical data and I²C bus are available in the following documents:

- OSRAM OS general application note on ambient light sensing [1]
- SFH 7770 E6 datasheet [2]
- I²C bus specification [3]
- Driving a LED with an external driver (> 200 mA) application note [4]

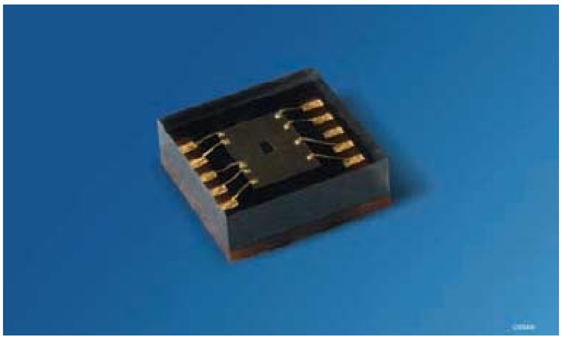

Fig. 1: Ambient Light and Proximity Sensor SFH 7770 E6.

2 Introduction

The SFH 7770 E6 is a compact device which is designed for simultaneous detection of ambient light and proximity of reflecting objects. Applications are mobile phones, PDAs, notebooks, cameras and other consumer products.

The device includes the following features:

- *Proximity Sensor (PS)*
  - Detection-range up to 200 mm
  - Gesture recognition possible
  - Outputs to drive up to three IR emitters
  - Optimized for 850 nm emitters
  - Immune to ambient light
- *Ambient Light Sensor (ALS)*
  - 0.03lx - 65000lx- High linearity
  - Spectral sensitivity well matched to the human eye
- *I²C interface*
  - 100 kHz / 400 kHz and 3.4 MHz mode
  - 3 programmable measurement modes (STAND-BY, TRIGGERED, FREE-RUNNING)
- Current consumption < 5 µA in STAND-BY mode
- Small package size, 2.8 x 2.8 x 0.9 mm³

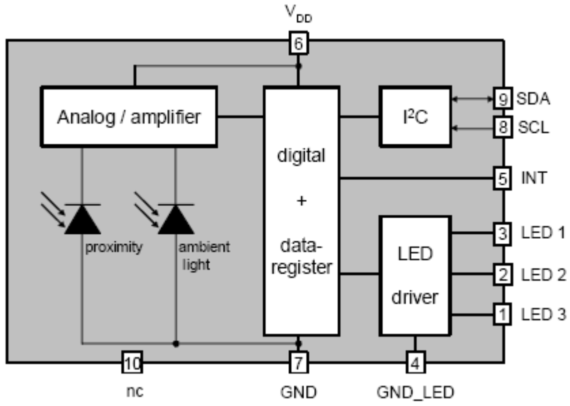

Fig. 2: Circuitry of SFH 7770 E6.

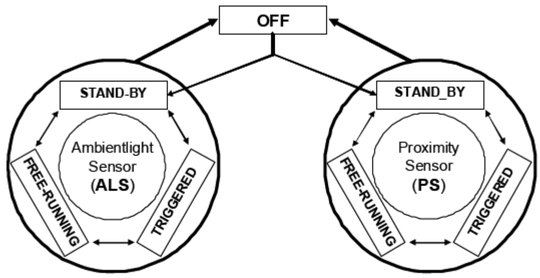

Fig. 3: ALS and PS operating modes.

Sensor Overview

The SFH 7770 E6 comprises photodiodes for both ambient light detection and proximity sensing (See Fig. 2). The photodiodes are integrated into a single ASIC which also performs the A/D conversion of the detected signals and provides an interface for communication with a host device via the I²C bus.
From functional points of view, the SFH 7770 E6 offers the following features and properties, which are briefly introduced in this section:

- Proximity sensor
- Ambient light sensor
- Operating modes : TRIGGERED, STANDBY, FREE-RUNNING
- LED drivers
- I²C communication
- Pinning

Pin #	Pin label	Description
1	LED 3	Cathode of LED 3
2	LED 2	Cathode of LED 2
3	LED 1	Cathode of LED 1
4	GND_LED	Separate LED ground
5	INT	Interrupt pin
6	V_{DD}	Supply voltage
7	GND	V_{DD} Ground
8	SCL	I²C bus clock line
9	SDA	I²C bus data line
10	N.C.	Not connected

Tab. 1: SFH 7770 E6 pinning.

For proximity sensing, up to three LEDs can be driven simultaneously (see Fig 2). This allows the operation of three independent proximity sensing channels. At maximum refresh rate, all three PS values are updated every 10 ms, which makes the sensor ready for reliable gesture recognition. All proximity channels deliver continuous signals with 8bit resolution. Therefore the PS may not only be used for pure proximity detection, but also for distance measurements.
The ambient light sensor is working independently from the proximity sensor. The ambient light readings are updated with a rate down to 100 ms, depending on the programmable settings (default: 500 ms).

3 Operating Modes

The SFH 7770 E6 can be operated in three different modes (see Fig. 3), in which the proximity resp. the ambient light function can be used independently from each other. The three basic modes are:

Mode	Bit Rate
Standard mode (Sm)	≤ 100 kbit/s
Fast mode (Fm)	≤ 400 kbit/s
High speed mode (Hs)	≤ 3.4 Mbit/s

Tab. 2: The I²C-bus protocol speed mode compatibility of the SFH 7770 E6.

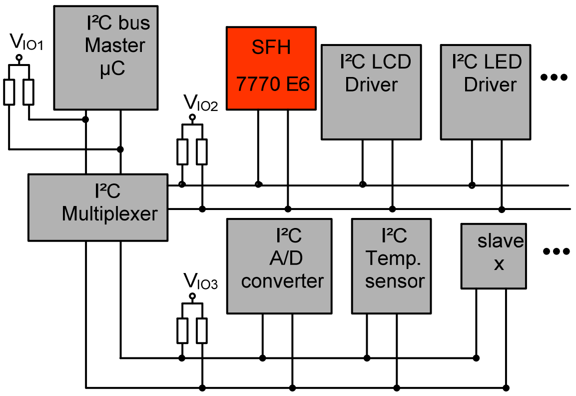

Fig. 4: I2C-bus network.

free-running: The sensor measures continuously and writes the results into the relevant registers, ready to be read via the I²C-bus interface. Optionally the interrupt alert function with the user-defined threshold levels (PS and/or ALS) will be executed if such an event takes place.
triggered: The measurements are initiated via I²C-bus instruction. Data are available after processing is finished (10 ms total delay time for PS, 100 ms for ALS).
stand-by: The initial state after power-up. The SFH 7770 E6 is in low power mode (I_{DD} < 5 µA), no operations are carried out, but the device is ready to respond to I²C-bus commands.
additionally, there is the off-state:
off: The SFH 7770 E6 is inactive, supply current is below 2 µA. The SDA, SCL and INT pins are in Z-state (high impedance). All register entries are reset to the default values.

The transition time between the modes, t_{trans}, is < 10 ms. The delay time between standby

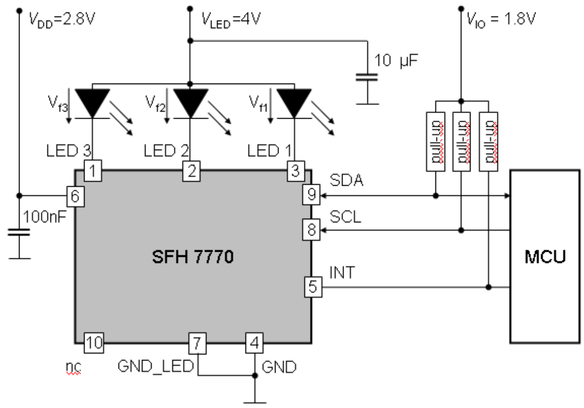

Fig. 5: Application diagram for basic operation of SFH 7770 E6.

and start of measurement is < 10 ms. The voltage V_{DD} to switch the SFH 7770 E6 into the off-state is < 1.4 V. To reach the stand-by mode at least 2.0 V are required.

4 I²C – Bus Communication

The address of the SFH 7770 E6 is 0x38.

The SFH 7770 E6 is a digital ambient light and proximity sensor. The communication is performed via a 2-wire I²C bus interface, so the device can be integrated into a typical multi master / multi slave I²C bus environment. A typical I²C bus network consists of a master and different I²C bus slave devices as depicted in Fig. 4.
Tab. 2 presents the different bus protocol speeds the SFH 7770 E6 is able to handle.

4.1 I²C - Bus Environment

The SFH 7770 E6 is a digital ambient light and proximity sensor. The communication is performed via a 2-wire I²C bus interface, so the device can be integrated into a typical multi-master / multi-slave I²C bus environment. A typical I²C bus network consists of a master and different I²C bus slave devices. For a more detailed discussion on the topic of I²C-bus please refer to [3].

1. Activate Ambient Light Sensor

S	SFH7770 Address (0x38)	W	A	ALS Control Register (0x80)	A	Activate Free Running Mode (0x03)	A	P

2. Activate Proximity Sensor

S	SFH7770 Address (0x38)	W	A	PS_1 Control Register (0x81)	A	Activate Free Running Mode (0x03)	A	P
S	SFH7770 Address (0x38)	W	A	Set I_LED_1 Register (0x82)	A	Set LED 1 Current to 200mA (0x1E)	A	P

3. Wait

4. Read Out PS Data

S	SFH7770 Address (0x38)	W	A	PS Data 1 Register (0x8F)	A	P
S	SFH7770 Address (0x38)	R	A	PS 1 Data	NA	P

5.1 Read Out ALS Data (LSB)

S	SFH7770 Address (0x38)	W	A	ALS Data Register LSB - (0x8C)	A	P
S	SFH7770 Address (0x38)	R	A	ALS Data (LSB)	NA	P

5.2 Read Out ALS Data (MSB)

S	SFH7770 Address (0x38)	W	A	ALS Data Register MSB - (0x8D)	A	P
S	SFH7770 Address (0x38)	R	A	ALS Data (MSB)	NA	P

Communication from Master to SFH 7770

Communication from SFH 7770 to Master

W: Master Writes
R: Master Reads
A: Acknowledge
NA: Not Acknowledge
S: Start Condition
P: Stop Condition

Fig. 6: I²C-bus communication for the example described below.

The minimum requirements to drive the SFH 7770 E6 are shown in the simplified network (Fig. 5). In this case a microcontroller unit acts as the I²C bus master. The I²C bus lines SDA (Serial Data Line) and SCL (Serial Clock Line) need to be referenced via pull–up resistors to the digital voltage level V_{IO}. Typical pull-up resistors are 560 Ω to 1kΩ.
The built-in I²C-bus interface is compatible with all common I²C-bus modes (see Tab. 3). The logic voltage (V_{IO}) of the SFH 7770 E6 ranges from 1.6 V – 2.0 V (according to I²C-bus specification [2]).

4.2 I²C - Bus Communication

By embedding the SFH 7770 E6 in an I²C-bus network and after applying V_{DD} = 2.5 V, the communication can start as follows (Fig. 6 illustrates this I²C-bus conversation):

1. Activation of the ALS:
The default mode of the sensor is STAND-BY and the SFH 7770 E6 needs to be activated by the master (e.g. microcontroller).

Each I²C bus communication begins with a start command "S" of the Master (SDA line

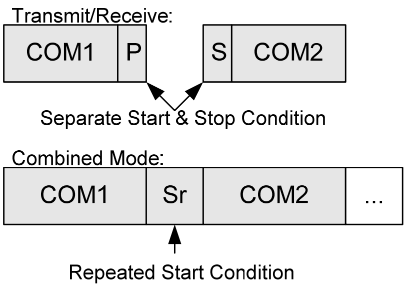

Fig. 7: Combined mode structure.

is changing from “1” to “0” during SCL line stays “1”) followed by the address of the slave (SFH 7770 E6 address is 0x38). After the 7bit slave address the read (1) and write (0) R/W bit of the master will follow. The R/W bit controls the communication direction between the master and the addressed slave. The slave is responding the proper communication with an acknowledge command. Acknowledge “A” (or not acknowledge “NA”) is performed from the receiver by pulling the SDA line down (or leave in “1” state).
For the activation of the sensor the master needs to write an activation command (0x03) into the corresponding control register for the ALS (0x80). Each command needs to be acknowledged by the slave. After activation the master ends the communication with a STOP command “P” (SDA line is changing from LOW to HIGH during SCL line stays HIGH). In this example the measurement interval time is kept at the default value (500 ms).

2. Activation of the PS:
For the activation of the PS sensor the master needs to write the activation command (0x03) into the corresponding control register (0x81). By writing 0x1E into the I_LED register (0x82) the LED current is set to 200 mA. The measurement interval is left at the default value (100 ms). After activation the master ends the communication with a STOP command.

3. Wait time:
After activation, the sensor will change from STAND-BY to FREE-RUNNING mode. After a delay of 100 ms for ALS / 10 ms for PS the first measurement value is available and can be read via the I²C-bus.

4. PS value: reading data
The PS value is accessible via the output register (0x8F). After reading the 8-bit word, the communication can be ended by the master with a not acknowledge “NA” and the stop command “P”. The PS output reading of the SFH 7770 E6 can then be converted from hexadecimal to decimal.

5. ALS value: reading data (LSB and MSB)
The sensor’s 16bit ALS measurement value is composed of 2 bytes (LSB & MSB). The bytes are accessible via the two output registers (0x8C, 0x8D). After addressing the LSB (least significant byte) resp. the MSB (most significant byte) output register, the communication direction has got to be changed from the slave to the master by repeating the address and the R/W byte with a changed R/W bit. After reading LSB and MSB, the communication is ended by the master with a not acknowledge “NA” and the stop condition “P”. The conversion of the ALS output data of the SFH 7770 E6 from hexadecimal to decimal can easily be calculated:

ALS_DATA_LSB = F0 (1111 0000)
ALS_DATA_MSB = 83 (1000 0011)

Final result (hexadecimal): 83 F0 counts
Final result (decimal): 33776 counts, which correspond to around 30.4 klx (based on a conversion factor of typ. 0.9 lux/count).

After finishing the measurement, the SFH 7770 E6 mode may be changed to STAND-BY via the control register.

Combined mode
To ensure interference free communication the I²C-bus combined mode should be used. Instead of performing two independent read or write commands (COM 1 & COM 2) the commands can be combined by a repeated

start condition “Sr” (Fig. 7 illustrates the combined mode structure).
The start and repeated start commands (“Sr”) are the same: the SDA line is changing from “1” to “0” during SCL line “1”. The “Sr” condition is placed behind “A” or “NA”. The combined mode is not limited to 2 read/write commands, so the addressing of the sensor and reading/writing of multiple register values can be performed within one block.

Block read mode
The Block read mode of the SFH 7770 E6 can be used to read all output registers in cyclic manner.

After addressing and reading an output register (e.g. LSB) in normal mode, the master is not placing the stop condition, but sends an acknowledge and continues to read the output registers. The SFH 7770 E6 will automatically increase the register address and the content of the next sensor output register can be read following the register addresses:

80→81→…→98→99→80→81→...

For register addresses and content see Sec. 8.3 and Tab. 3.

The block read mode can be ended by placing a not acknowledge (NA) with the subsequent stop condition from the master.

4.3 Registers

The SFH 7770 E6 has 23 different registers (see Tab. 3). The following pages will describe the registers and their structure resp. content.

I²C Addr.	Type	Name	Description
0x20	RW	INT_ACCESS	Integration time access
0x26	RW	ALS_INT_TIME	ALS integration time
0x27	RW	PS_INT_TIME	PS integration time (burst length)
0x80	RW	ALS CONTROL	SW reset , ALS operation mode control
0x81	RW	PS CONTROL	PS operation mode control
0x82	RW	I_LED 1 and LED 2	Setting LED 1 and LED 2 pulse current
0x83	RW	I_LED 3	Setting LED 3 pulse current
0x84	RW	ALS & PS TRIG	Forced mode ALS and PS measurement triggering
0x85	RW	PS INTERVAL	PS measurement rate in stand-alone mode
0x86	RW	ALS INTERVAL	ALS measurement rate in stand-alone mode
0x8A	R	PART_ID	Part number and revision IDs
0x8B	R	MAN_ID	Manufacturer ID
0x8C	R	ALS_DATA_LSB	ALS measurement data, least significant bits
0x8D	R	ALS_DATA_MSB	ALS measurement data, most significant bits
0x8E	R	ALS_PS STATUS	Status of meas. data (ALS and PS)
0x8F	R	PS_DATA_1	PS_LED_1 measurement data
0x90	R	PS_DATA_2	PS_LED_2 measurement data
0x91	R	PS_DATA_3	PS_LED_3 measurement data
0x92	RW	INT_SET	Interrupt settings
0x93	RW	PS_THR LED_1	PS_LED_1 interrupt threshold level
0x94	RW	PS_THR LED_2	PS_LED_2 interrupt threshold level
0x95	RW	PS_THR LED_3	PS_LED_3 interrupt threshold level
0x96	RW	ALS UP_THR LSB	ALS interrupt upper threshold level, least significant bits
0x97	RW	ALS UP_THR MSB	ALS interrupt upper threshold level, most significant bits
0x98	RW	ALS LO_THR LSB	ALS interrupt lower threshold level, least significant bits
0x99	RW	ALS LO_THR MSB	ALS interrupt lower threshold level, most significant bits

Tab. 3: SFH 7770 E6 control and data registers.

INTEGRATION TIME ACCESS: Allows access to reg. 0x26, 0x27 (ALS_INT_TIME, PS_INT_TIME)

Note: After setting bit '0' there must be a stop condition to confirm writing. It is recommended to set the bit '0' back to '0' after the changes in the integration registers 0x26 and 0x27 have been made.

R/W-Register 0x81								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	not used							**Integration Time Access:**
default	**XXXXXXX**							0 Not Accessible
								1 Accessible

ALS INTEGRATION TIME: Ambient light measurement integration time:

The ALS integration time is responsible for setting the ALS sensitivity range and the lx/count value. An increase of the ALS integration time by a factor of 10 increases also the ALS sensitivity level by a factor of 10. The default setting of 100 ms results in a range from approximately 0.3 lx to 6553 lx with a resolution of 0.1 lx/count.

0x26 is only accessible if the access-bit in register 0x20 is set to '1'. It is recommended to set this access bit back to '0' after changes have been made. When reading or writing in block-read/-write mode, it is recommended to start at register 0x26 and stop at 0x27, as there are other registers accessible which are not intended to be accessible by the user. Afterwards set 0x20 back to '0'.

R/W-Register 0x26								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	not used					**ALS INTEGRATION TIME (typ. RANGE, RESOLUTION)**		
default	**XXXXX**					**000 100 ms (0.3 lx – 6.5 klx, 0.1 lx/ct)**		
						001 200 ms (0.15 lx – 3.2 klx, 0.05 lx/ct)		
						010 500 ms (0.06 lx – 1.3 klx, 0.02 lx/ct)		
						011 1000 ms (0.03 lx – 650 lx, 0.01 lx/ct)		
						100 10 ms (3 lx – 65 klx, 1 lx/ct)		
						101 20 ms (1.5 lx – 32 klx, 0.5 lx/ct)		
						110 50 ms (0.6 lx – 13 klx, 0.2 lx/ct)		
						111 50 ms (0.6 lx – 13 klx, 0.2 lx/ct)		

PS INTEGRATION TIME (BURST LENGTH): Proximity measurement integration time

An increase in PS integration time results in an increased PS signal level. E.g. an increase in PS integration time by a factor of 10 increases the PS counts by around 50 counts (due to pseudo-logarithmic relationship).

0x27 is only accessible if access-bit in register 0x20 is set to '1'. It is recommended to set this access bit back to '0' after changes have been made. When reading or writing in block-read/-write mode, it is recommended to start at register 0x26 and stop at 0x27, as there are other registers accessible which are not intended to be accessible by the user. Afterwards set 0x20 back to '0'.

R/W-Register 0x27								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	not used					**PS INTEGRATION TIME**		
default	**XXXXX**					**100 750 us**		
						000 100 us		
						001 200 us		
						010 300 us		
						011 500 us		
						100 750 us		
						101 1000 us		
						110 1500 us		
						111 2500 us		

ALS CONTROL: Software reset and control of ambient light sensor

SW reset (bit #2 „1") sets all registers to default (same as POWER-UP). Afterwards it is automatically set back to „0" by the SFH 7770 E6.

R/W-Register 0x80								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
	not used					**complete SW reset**	**mode of ambient light sensor**	
default	**00000**					**0**	**00 STAND-BY**	
						1 SW reset	00 STAND-BY	
							01 STAND-BY	
							10 TRIGGERED (by MCU)	
							11 FREE-RUNNING (internally triggered)	

PS CONTROL: Control of proximity sensor

R/W-Register 0x81								
Bit	7	6	5	4	3	2	1	0
	not used						mode of Proximity Sensor	
default	XXXXXX						**00 STAND-BY**	
							00 STAND-BY	
							01 STAND-BY	
							10 TRIGGERED by MCU	
							11 FREE-RUNNING (internally triggered)	

I_LED_1 and I_LED_2: Activation of LED and Emitter (LED 1 and LED 2) current setting

The register allows the activation of the LEDs. The following combinations of active LEDs are available: LED 1, LED 1 + LED 2, LED 1 + LED 3 and LED 1 + LED 2 + LED 3. In addition the LED pulse currents for LED 1 and LED 2 can be set.

R/W-Register 0x82								
Bit	7	6	5	4	3	2	1	0
	activation of LEDs		setting LED_2 pulse current			setting LED_1 pulse current		
Default	**00**		**011 50 mA**			**011 50 mA**		
	00 LED 1 active		000 5 mA			000 5 mA		
	01 LED 1 and 2 active		001 10 mA			001 10 mA		
	10 LED 1 and 3 active		010 20 mA			010 20 mA		
	11 all LEDs active		011 50 mA			011 50 mA		
			100 100 mA			100 100 mA		
			101 150 mA			101 150 mA		
			110 200 mA			110 200 mA		

I_LED_3: Emitter (LED 3) current setting

R/W-Register 0x83								
Bit	7	6	5	4	3	2	1	0
	Not used					setting LED_3 pulse current		
Default	XXXXX					**011 50 mA**		
						000 5 mA		
						001 10 mA		
						010 20 mA		
						011 50 mA		
						100 100 mA		
						101 150 mA		
						110 200 mA		

ALS & PS TRIG: MCU-triggered measurement (for ambient light sensor and proximity sensor)

If „1“ is set a new measurement will start after I^2C stop command from MCU. As soon as the measurement is finished the corresponding bit of the register will automatically be set to „0“ by the SFH 7770 E6.

R/W-Register 0x84								
Bit	7	6	5	4	3	2	1	0
	not used						trigger ambient light	trigger proximity
default	XXXXXX						1	1

PS INTERVAL: Proximity measurement: time interval setting (repetition time) for FREE-RUNNING mode

R/W-Register 0x85								
Bit	7	6	5	4	3	2	1	0
	not used				time-interval			
default	XXXX				**0101 100 ms**			
					0000 10 ms			
					0001 20 ms			
					0010 30 ms			
					0011 50 ms			
					0100 70 ms			
					0101 100 ms			
					0110 200 ms			
					0111 500 ms			
					1000 1000 ms			
					1001 2000 ms			

ALS INTERVAL: Ambient light measurement: time interval setting (repetition time) for FREE-RUNNING mode

R/W-Register 0x86								
Bit	7	6	5	4	3	2	1	0
	not used					time-interval		
default	XXXXX					**010 500 ms**		
						000 100 ms		
						001 200 ms		
						010 500 ms		
						011 1000 ms		
						100 2000 ms		

PART_ID: Part number and revision Identification

R-Register 0x8A								
Bit	7	6	5	4	3	2	1	0
	Part number ID				Revision ID			
	1001				0100			

MAN_ID: Manufacturer Identification

R-Register 0x8B								
Bit	7	6	5	4	3	2	1	0
	Manufacturer Identification							
	0000				0011			

ALS_DATA_LSB: Ambient light measurement data (0x8C: LSB)

The result of the ambient light sensor is a 16bit word with MSB and LSB. It is stored in two registers. The binary data can be converted directly to decimal „lx" values (max. 65535lx).

R-Register 0x8C								
Bit	7	6	5	4	3	2	1	0
	LSB data							
default	00000000							

ALS_DATA_MSB: Ambient light measurement data (0x8D: MSB)

R-Register 0x8D								
Bit	7	6	5	4	3	2	1	0
	MSB data							
default	00000000							

ALS_PS STATUS: Status of measurement data for ambient light sensor (ALS) and proximity sensor (PS)

After the measurement data is available in the register (0x8E), the corresponding statusbit (bit #6 for ALS; bit #0, #2 and #4 for PS) is set to „1“. After data has been read by the MCU the statusbit is automatically reset to “0” by the SFH 7770 E6.

Bit #7 is set „1“, if the measured ALS value is outside the threshold level settings (register 0x96... 0x99). Bit #1, #3, #5 is set to “1” if the measured PS value is above the threshold level (register 0x93). The status of register 0x8E will always be updated if new measurement data is available.

R-Register 0x8E								
Bit	7	6	5	4	3	2	1	0
	ALS Threshold	ALS data	PS LED 3 threshold	PS LED 3 data	PS LED 2 threshold	PS LED 2 data	PS LED 1 threshold	PS LED 1 data
default	00000000							

PS_DATA_1: Proximity measurement data from LED_1 (8bit, logarithmic scale)

R-Register 0x8F								
Bit	7	6	5	4	3	2	1	0
	data							
default	00000000							

PS_DATA_2: Proximity measurement data from LED_2 (8bit, logarithmic scale)

R-Register 0x90								
Bit	7	6	5	4	3	2	1	0
	data							
default	00000000							

PS_DATA_3: Proximity measurement data from LED_3 (8bit, logarithmic scale)

R-Register 0x91								
Bit	7	6	5	4	3	2	1	0
	data							
default	00000000							

INT_SET: Interrupt register / INT output.

In bit #6 and #5 the trigger source for the last interrupt event is stated. Data from status register (0x8E) are used. In latched mode (set by bit #3) this remains unchanged until the interrupt register has been read by the MCU. Afterwards the bits are reset automatically to “0” by the SFH 7770 E6. In unlatched mode it is updated after every measurement. The output polarity of the interrupt function can be changed by bit #2. The interrupt can be triggered by an ambient light sensor event and / or by a proximity sensor event (bit #1 and bit #0). Z-state means the output is in high-impedance state.

R/W-Register 0x92								
Bit	7	6	5	4	3	2	1	0
	not used	Interrupt trigger source		not used	Output mode	Output polarity	Interrupt mode (triggered by..)	
R/W	not used	R only		not used	R/W	R/W	R/W	
default	X	00		X	1	0	00	
		00 ALS			0 latched	0 active L	00 Z state	
		01 PS_LED_1			1 not latched	1 active H	01 only PS	
		10 PS_LED_2					10 only ALS	
		11 PS_LED_3					11 PS and ALS	

PS_THR LED_1: Threshold level for proximity sensor

RW-Register 0x93								
Bit	7	6	5	4	3	2	1	0
	data							
default	11111111							

PS_THR LED_2: Threshold level for proximity sensor

RW-Register 0x94								
Bit	7	6	5	4	3	2	1	0
	data							
default	11111111							

PS_THR LED_3: Threshold level for proximity sensor

RW-Register 0x95								
Bit	7	6	5	4	3	2	1	0
	data							
default	11111111							

ALS UP_THR LSB: Upper threshold level for ambient light sensor (LSB)

RW-Register 0x96								
Bit	7	6	5	4	3	2	1	0
	LSB data (upper threshold)							
default	11111111							

ALS UP_THR MSB: Upper threshold level for ambient light sensor (MSB)

RW-Register 0x97								
Bit	7	6	5	4	3	2	1	0
	MSB data (upper threshold)							
default	11111111							

ALS_LO_THR LSB: Lower threshold level for ambient light sensor (LSB)

RW-Register 0x98								
Bit	7	6	5	4	3	2	1	0
	LSB data (upper threshold)							
default	11111111							

ALS_LO_THR MSB: Lower threshold level for ambient light sensor (MSB)

RW-Register 0x99								
Bit	7	6	5	4	3	2	1	0
	LSB data (upper threshold)							
default	11111111							

5 Interrupt Alert

The SFH 7770 E6 provides an interrupt pin, which can be configured completely by the user. The register 0x92 allows configuring the interrupt as active low or active high. Additionally, the interrupt function can be configured to operate in latched or normal mode. In normal mode the interrupt event/signal is updated after every measurement, whereas in the latched mode it is guaranteed that even short peaks are detected (e.g. the interrupt is held as long as the microcontroller reads out the interrupt register).

Interrupt Event Definition	
proximity sensor	PS data > PS threshold
ambient light sensor	ALS data > ALS upper threshold
	ALS data < ALS lower threshold

Tab. 4: Interrupt event definition.

The interrupt can be set for a PS (PS threshold) and/or ALS (upper and lower ALS threshold) event. For the exact interrupt event definition please refer to Tab. 4. This is especially valuable as it allows the SFH 7770 E6 to operate as stand alone device in the *free-running mode,* independent from the main microcontroller. This functionality relieves the microcontroller from active involvement in the PS / ALS monitoring resp. measurement cycle and reduces significantly the I^2C-bus traffic, thus reducing the overall power consumption of the system. Only if the user-defined thresholds are violated, the interrupt signal will inform the microcontroller and the predefined actions can be executed (e.g. after read-out of the interrupt and PS / ALS data registers to get the actual data - if desired).

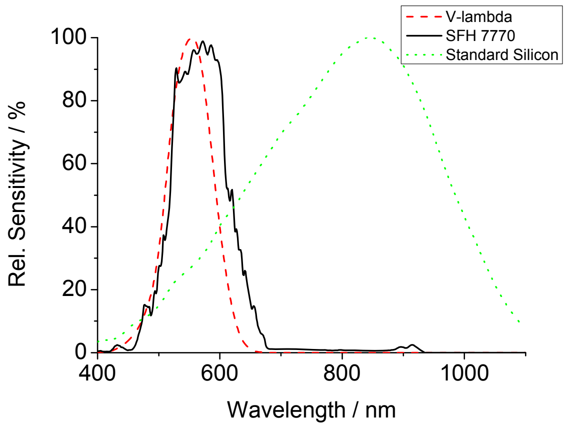

Fig. 8: Spectral sensitivity of the SFH 7770 E6, compared to the sensitivity of the human eye and of a standard silicon detector.

t_{int}	ALS Range	ALS Resolution
10 ms	3.0 lx - 65535 lx	1.00 lx/count
20 ms	1.5 lx - 32767 lx	0.50 lx/count
50 ms	0.60 lx - 13106 lx	0.20 lx/count
100 ms	0.30 lx - 6553 lx	0.10 lx/count
200 ms	0.15 lx - 3277 lx	0.05 lx/count
500 ms	0.06 lx - 1311 lx	0.02 lx/count
1000 ms	0.03 lx - 655 lx	0.01 lx/count

Tab. 5: ALS integration time settings t_{int} and their relation to ALS range and resolution (default: t_{int} = 100 ms).

6 Functional Description

The SFH 7770 E6 comprises an ambient light sensor (ALS) and a proximity sensor (PS) within one single device. In this section the operation and properties of the ALS and PS parts are described.

6.1 Ambient Light Sensor (ALS)

The SFH 7770 E6 has an on-chip photodiode which detects the level of the ambient light. The analog photodiode signal is processed by the IC. Finally, the (digital) value is stored in a register and can be read via the I²C bus. The digital ALS count can then be converted into an illumination value, i.e. ‘lux’, via the simple relationship according to Tab. 5.

6.1.1 Spectral Sensitivity of the ALS

The ambient light sensor is intended to provide illumination data, e.g. to control and adjust the display brightness. To support this functionality the SFH 7770 E6 provides a convenient user interface.

The ambient light sensor delivers output values in the range from 0 to 65535 (16 bit).

Low output values correspond to a low illumination of the sensor, while high values indicate high illumination. The range of the ambient light sensor sensitivity can be set by the user and covers more than 4 ½ decades. Two threshold levels for the ambient light sensor can be set via the I^2C-bus, a lower and an upper threshold. In the case of exceeding these thresholds an interrupt is generated automatically, allowing e.g. the microcontroller to act accordingly (see Sec. 4.3 and Sec. 5 for the relevant registers and settings).

6.1.2 Spectral Sensitivity of the ALS

The human eye’s wavelength range of significant sensitivity is between 400 nm and 700 nm with its peak at around 555 nm (often called V-lambda characteristic).
The spectral sensitivity of the SFH 7770 E6 aims to mimic the sensitivity of the human eye as close as possible and provides a real improvement compared to standard silicon-photodetectors (see Fig. 8).

6.1.3 Sensitivity Range of the ALS

The range of the ALS can be programmed by the user via the ALS integration time (register 0x26). Per default (integration time = 100 ms) the range covers 0.3 lx to 6.5 klx with a resolution of typ. 0.1 lx/count. A doubling of the integration time changes the sensitivity range by a factor of two. Please

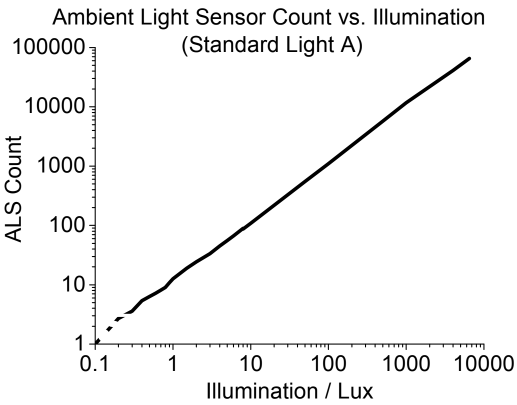

Fig. 9: Ambient light sensor count vs. illumination (integration time = 100 ms).

Cover Transmission (visible light)	corresponding ALS range outside Cover	corr. ALS resolution outside Cover
100 %	0.3 lx – 6553 lx	0.1 lx
50 %	0.6 lx– 13000 lx	0.2 lx
20 %	1.5 lx– 32500 lx	0.5 lx
10 %	3.0 lx– 65535 lx	1.0 lx

Tab. 6: Impact of cover glass transmission on ALS range and resolution (based on an integration time setting of 100 ms, resulting in a conversion factor of typ. 0.1 lx/count of the sensor).

refer to Tab. 5 for details. Also note: Prior to accessing the ALS integration time register (0x26) the integration time access register (0x20) has to be set accordingly.

6.1.4 Output and Linearity of the ALS

The sensors output count is linear vs. the illumination level E_V over a wide range (see Fig. 9). This conversion between the ALS count and the illumination is typ. 0.1 lx/count (standard light A) for the default ALS integration time of 100 ms (please refer to Tab. 5 for different ALS integration time settings and their relation to the ALS resolution). For an exact absolute calibration it is recommended that the user performs a measurement within the application for each device. Deviation from the linearity is usually within ± 5 % (normalized to 100 lx).
Tab. 6 presents the relationship of cover glass transmission on the measured ALS range.

The sensor output is a 16 bit serial I²C bus output. The maximum output signal is 65535 counts (=65535lx). The final 16 bit data is spread over two 8-bit registers: the ALS data register 0x8C includes the 8 least significant bits, while register 0x8D includes the 8 most significant bits.
In the following, an example for converting the ALS reading into a lux values is given:

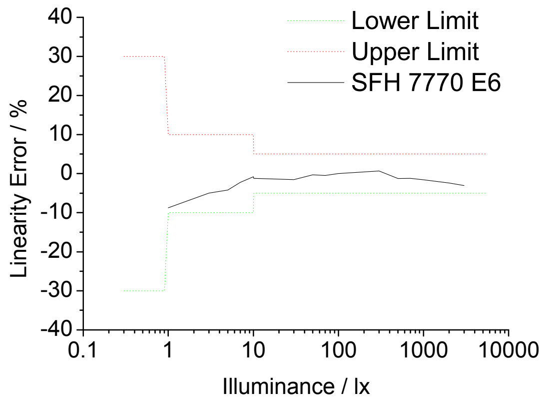

Fig. 10: SFH 7770 E6 deviation of the linear output characteristic referenced to 100 lx vs. illuminance and with an ALS integration time setting of 100 ms.

Register 0x8C value = 1001 0001 (binary) and 0x91 (hex), respectively.

Register 0x8D value = 1000 0010 (binary) and 0x82 (hex), respectively.

The actual illuminance value is calculated by combining the registers as below:

1000 0010 1001 0001 (binary)

= 0x8291 (hex)

= 33425 (decimal)

This value needs to be multiplied with the resolution, set in reg. 0x26 (e.g. 0.1 lx/ct per default), to obtain the illumination value of 3.425 klx. This value still has to be corrected if the sensor is located behind a dark cover glass according to e.g. Tab. 6 (impact of visible light attenuating cover glass).

The specified limits for the linearity error are presented in Fig. 10, together with a sample curve.

6.1.5 Flicker Rejection and Timing

Flicker rejection describes the attenuation of light source fluctuations to the sensor output. Light source fluctuations are mainly caused by AC line voltage. The line frequency (50 Hz / 60 Hz or higher) is transferred to a 100 Hz / 120 Hz optical signal. Flicker rejection means the attenuation of the optical input noise level compared to the sensor output noise level. The SFH 7770 E6 is able to suppress the optical input noise by averaging the input signal over multiples of 50 ms. The flicker rejection is typically 13dB for the SFH 7770 E6, i.e. only 5% of the optical input 'noise' will be transferred to the sensor output.

Please note: To achieve flicker-free measurements (e.g. 50 / 60 Hz driven light bulbs) integration times with a multiple of 50 ms are recommended (i.e. 50 ms, 100 ms aso.). By choosing 10 ms or 20 ms OSRAM recommends averaging several measurements to achieve reliable flicker-free ALS values.

6.1.6 Sample Programs for ALS

The SFH 7770 E6 is operated completely via the I²C bus. All read and write commands must be given by an I²C *master* device (e.g. microcontroller unit). The SFH 7770 E6 always acts as an I²C *slave* device with I²C address 0x38.

Tab. 7 shows a simple program for reading out the ALS values when SFH 7770 E6 is operated in stand-alone mode. The setting of the SFH 7770 E6 is performed by the yellow marked steps. It is sufficient to execute those only once after power-up of the SFH 7770 E6. The other steps may be executed repeatedly.

Table 8 contains an enhanced version for ALS operation. To set the sensitivity range of the ALS the steps 2 and 3 are necessary. Step 2 allows access to the integration time (ALS range) register (access set in register 0x20) and step 2 sets the desired ALS range in register 0x26.
The ALS is operated in TRIGGERED mode, i.e. each measurement has to be initiated by an I²C master device. The upper/lower threshold levels are set to 1024 lx and 128 lx, respectively. The operation mode of the interrupt pin is also set (step 9):

a) The interrupt acts in NON-LATCHED mode
b) The interrupt bit and the logic level at the interrupt pin is "0" when the ALS signal is within the threshold range (128 lx - 1024 lx)
c) Only the ALS can trigger interrupt (not PS)

Once a measurement is triggered by the I²C master device, the status of the measurement can be read from register 0x8E (step 10). If new ALS data are available, the bit #6 of the register contains "1". Once the data have been read out (steps 11, 12), the bit is set back to "0" automatically. If the ALS value is out of the bounds as defined in registers 0x8C and 0x8D, one bit in register 0x92 is set to "1" (step 13). This allows detecting whether the ALS interrupt conditions are fulfilled by just reading a single bit, rather than reading out the complete illumination value.

Step No	Register	R/W	Value	Comment
1	0x80	W	0x04	Reset device
2	0x80	W	0x03	Set ALS to FREE-RUNNING mode
3	0x86	W	0x02	Set repetition time to 500ms
4	---	---	---	Wait >500ms
5	0x8C	R		Read LSB of ALS
6	0x8D	R		Read MSB of ALS
7	---	---	---	Wait >500ms
8	0x8C	R		Read LSB of ALS
9	0x8D	R		Read MSB of ALS

Tab. 7: Simple program for ALS operation.

6.2 Proximity Sensor (PS)

The SFH 7770 E6 is capable of detecting the proximity of objects via reflection of pulsed IR light (preferably at 850 nm). Up to three LEDs may be operated and read out independently by pulsing the LEDs subsequently (multiplexing, see Fig. 11). This allows the receiver unit of the proximity sensor to separate signals coming from the different LEDs. For each of the 3 signals, threshold levels for an interrupt alert can be set via the I²C bus command (see Tab. 3 for the relevant register).

6.2.1 PS Functionality

The chip contains drivers for up to 3 external LEDs. Each of the 3 LED drivers is capable of sinking an LED current from 5 mA up to 200 mA. The device is actively regulating the pulsed LED current. No additional resistors for current limitation are required.
The reflected light is detected by an on-chip photodiode. The photodiode signal is processed by analog amplifiers and digital logic. Finally, the signal level is stored in an 8-bit register and can be read out via the I²C bus. The output indicates the presence of an object which reflects the IR light and can assume values between 0 and 254 (pseudo-logarithmic scale). Hence, no post-processing of the reading is necessary. In order to reduce the noise of the proximity

Step No	Register	R/W	Value	Comment
1	0x80	W	0x04	Reset device
2	0x20	W	0x01	Allow access to Integration Time register
3	0x26	W	0x04	Set ALS range to 3 lx … 65 klx
4	0x80	W	0x02	Set ALS to TRIGGERED mode
5	0x96	W	0x00	Set upper threshold level to 1024 lx (LSB)
6	0x97	W	0x04	Set upper threshold level to 1024 lx (MSB)
7	0x98	W	0x80	Set upper threshold level to 128 lx (LSB)
8	0x99	W	0x00	Set upper threshold level to 128 lx (MSB)
9	0x92	W	0x0A	Set INTERRUPT output pin
10	0x84	W	0x02	Trigger ALS measurement
11	---	---	---	Wait 20ms
12	0x8E	R		Read status of ALS data: if bit6 is "0" goto step 9.
13	0x8C	R		Read LSB of ALS
14	0x8D	R		Read MSB of ALS
15	0x92	R		Check interrupt bit (bit 7)

Tab. 8: Enhanced program for ALS operation.

signal, several data points may be averaged by the master device, which is operating the SFH 7770 E6.
To achieve a high ambient light suppression, the SFH 7770 E6 uses a 667 kHz LED burst for 750 µs per LED for a single measurement (default setting). Fig. 11 illustrates the burst signal during a complete measurement cycle (all three LEDs are operated). Measurement repetition time in the free running mode can be selected between 10 ms and 2000 ms. For maximum detection distance an emitter (LED) with a wavelength of around 850 nm is recommended. Other relevant paramters, which influence the maximum detection distance, are, among others: LED type, PS

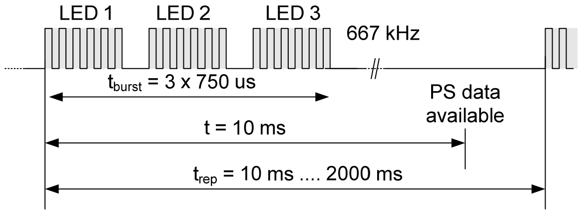

Fig. 11: LED current and timing during one proximity measurement cycle (LED integration time setting to 750 us).

Cover Transmission (at 850 nm)	corresponding detection distance (approximation)
100 % (no glass)	100 %
90 % (clear glass)	90 %
80 %	80 %
70 %	70 %

Tab. 9: Impact of cover glass (IR-) transmission on PS detection range.

integration time t_{burst}, LED current and the size / reflection properties of the target.

6.2.2 PS Signal and Detection Range

The strength of the reflected proximity signal and hence the output reading of the PS depends on the current setting of the LEDs. Large detection ranges are obtained for LED currents 100 mA, 150 mA or 200 mA.
Fig. 12 and 13 present the proximity values vs. distance for a 100 x 100 mm^2 Kodak White (90% reflectivity) target. The emitter (SFH 4650) was placed 5 mm away from the SFH 7770 E6.
As indicated by Fig. 12 and 13, the above setup allows a maximum detection range of about 20 cm (by using e.g. 200 mA LED current with SFH 4650 and a PS integration time of 1000 µs). Larger detection distances might require the use of e.g. stacked LEDs or LEDs with narrow and intense radiation characteristics (e.g. lensed LEDs like SFH 4059).

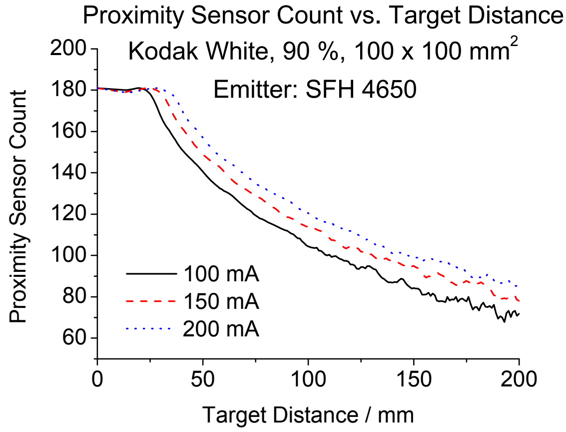

Fig. 12: Proximity sensor signal count vs. target distance and LED drive current (PS integration time t_{burst} = 750 us) – one LED SFH 4650 on.

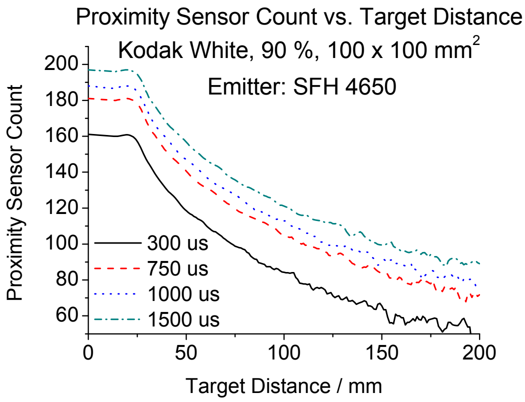

Fig. 13: Proximity sensor signal count vs. PS integration time t_{burst} (LED currnet = 100 mA) – one LED SFH 4650 on.

If used as a pure proximity switch it is recommended to set the threshold level not below 80 counts to avoid interference with noise.
As a rule of thumb, 30 counts result in almost a quadrupling in irradiance (PS signal level) whereas 10 counts represent roughly a factor of 1.55 in analog signal level. As a general rule OSRAM recommends for a robust design the setting of the threshold levels to be up to around 10 times above any noise level. The factor 10

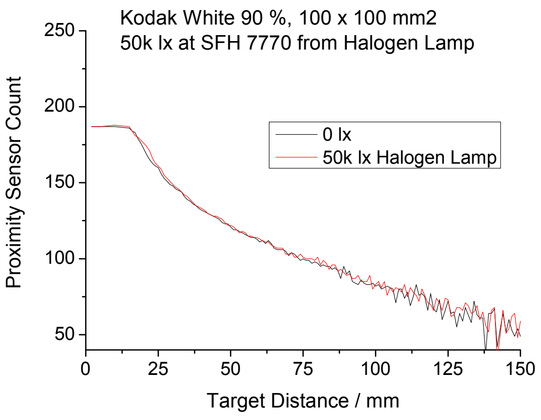

Fig. 14: Proximity signal in different ambient light conditions. Even in a high brightness environment (50k lx on SFH 7770 E6) the sensor shows no significant changes.

corresponds to around 60 counts in PS signal due to the pseudo-logarithmic relationship.

The digital proximity count signal is correlated to the detected irradiance E_e. There is an approximate logarithmic relationship between the digital PS signal the analog signal level (irradiance):

$$E_e \approx \left(10^{0.017 \cdot counts + 0.11 - 370.4 \frac{1}{s} \cdot t_{burst}} \right) \frac{\mu W}{cm^2} \quad \text{Eq. (1)}$$

In general IR absorbing cover glasses reduce the maximum detection distance. Tab. 9 presents an approximate relationship between detection distance and cover glass IR absorption.

6.2.3 Ambient Light Suppression of the PS-Signal

Due to its design the SFH 7770 E6 features an excellent immunity of the proximity measurement against even ultra-high ambient light levels. Fig. 14 demonstrates this outstanding feature. Even in environments of 50 klx the proximity signal is completely unaffected by even illumination with a halogen lamp which contains a high level of IR radiation.

6.2.4 Sample Programs for PS

The SFH 7770 E6 is operated completely via the I²C bus. All read and write commands must be given by an I²C MASTER device (e.g. microcontroller unit). The SFH 7770 E6 always acts as an I²C SLAVE device with I²C address 0x38.

Tab. 10 shows a simple program for one single PS channel (and thus one external LED). The SFH 7770 E6 is operated in stand-alone mode. The setting of the device is performed by the yellow marked steps. It is sufficient to execute these steps only one single time after the power-up of the SFH 7770 E6. The other steps may be executed repeatedly.

Tab. 11 contains a more complex program for PS operation with three LEDs and increased integration time for increased detection distance (access integration time register 0x27 via the access register 0x20). The PS is operated in TRIGGERED mode, i.e. each measurement has to be initiated by an I²C master device. The threshold levels for all 3 channels are set to 90 counts. The operation mode of the interrupt pin is also set (step 8):

a) Interrupt pin is triggered by LED1
b) Interrupt acts in NON-LATCHED mode
c) The interrupt bit and the logic level at the interrupt pin is “0” when the PS signal is below the threshold
d) Only PS can trigger interrupt (not ALS)

Once a measurement is triggered, the status of the measurement can be read from register 0x8E (step 11). If new PS data are available, the bits 0, 2, 4 for channels 1, 2, 3, respectively of the register are set to “1”.

Step No	Register	R/W	Value	Comment
1	0x80	W	0x04	Reset device
2	0x81	W	0x03	Set PS to FREE-RUNNING mode
3	0x85	W	0x05	Set repetition time to 100ms
4	0x82	W	0x06	Operate LED1, I_LED=200mA
5	---	---	---	Wait >100ms
6	0x8F	R		Read PS data (LED1)
7	---	---	---	Wait >100ms
8	0x8F	R		Read PS data (LED1)
9	…			

Tab. 10: Simple program for PS operation with a single LED (LED1).

Once the data have been read out (steps 12, 13, 14), the bits are set back to "0" automatically.
If the PS value of a channel is above its limit (defined in registers 0x93, 0x94 and 0x95), the corresponding bit (bit 1,3 and 5 in register 0x92) is set to "1" (step 15). This allows the user to detect PS events for each channel by reading single bits, rather than reading out the complete set of PS values.

6.3 Current Consumption

The following equations give an idea on the total power consumption of the SFH 7770 E6 during operation.

By operating the PS in the free-running mode, the current consumption (including LED current, I_{LED}) can be approximated by the following Eq. (depending on the measurement interval time t_{rep_PS} and the PS integration time t_{burst}):

$$I_{AVG\_PS} = 0.5 \cdot t_{burst} \frac{(I_{LED} + 100mA)}{t_{rep\_PS}} \qquad \text{Eq. (2)}$$

The current consumption during operation of the ALS depends on the ALS integration

Step No	Register	R/W	Value	Comment
1	0x80	W	0x04	Reset device
2	0x20	W	0x01	Allow access to integration time register
3	0x27	W	0x05	Set PS integration time to 1000 us
2	0x81	W	0x02	Set PS to TRIGGERED mode
3	0x82	W	0xF6	All LEDs active, I_LED1&2=200mA
4	0x83	W	0x06	I_LED3=200mA
5	0x93	W	0x5A	Set LED1 threshold level to 90 counts
6	0x94	W	0x5A	Set LED2 threshold level to 90 counts
7	0x95	W	0x5A	Set LED3 threshold level to 90 counts
8	0x92	W	0x2D	Set INTERRUPT output pin (trigger source LED1)
9	0x84	W	0x01	Trigger PS measurement
10	---	---	---	Wait 2ms
11	0x8E	R		Read status of PS data: if bits 0,2,4 are "0" goto step 9.
12	0x8F	R		Read PS value (LED1)
13	0x90	R		Read PS value (LED2)
14	0x91	R		Read PS value (LED3)
15	0x92	R		Check PS interrupt bits (bits)

Tab 11: Enhanced program for PS operation.

time t_{int} as well as the repetition time t_{rep_ALS} can be approximated by:

$$I_{AVG\_ALS} = 1mA \cdot \frac{t_{int}}{t_{rep\_ALS}} \qquad \text{Eq. (3)}$$

Example for total PS current consumption:
I_{LED} = 100 mA, t_{burst} = 300 us and t_{rep} = 100 ms => I_{AVG_PS} = 300 µA.

This compares to a stand-by current consumption of less than 5 µA (typ. 2-3 µA).

7 Design Guidelines

In many applications, the SFH 7770 E6 will be operated together with one or more LEDs. This section contains recommendations for the optical design of the whole ensemble, e.g. relative positions of SFH 7770 E6 and the LED(s), optical separators and optical crosstalk features. Optimal setups for gesture recognition (like hand movements) are discussed as well.

7.1 Component Arrangement

The arrangement of the SFH 7770 E6 and the LEDs depends on the desired application. If only the presence and/or the distance of an object (e.g. hand, ear, hair) has to be detected, one LED is sufficient. If the movement of an object in 2 or 3 dimensions has to be detected, all 3 channels of the proximity sensor will be used.
In the following sections some example configurations for the placement of the LEDs and the SFH 7770 E6 will be given. In any case it is recommended to check the functionality under the customer's application conditions (eg. cover glass model, placement distance, and height above the sensor as well as aperture openings aso.)

7.1.1 Setup for Proximity Detection

In general, the emitter should be placed close to the SFH 7770 E6. This ensures the highest level of the reflected signal, when the object is close to the detector. Alternatively, the LED current can be kept low, which leads to less power consumption. On the maximum detection *distance*, the component spacing only has minor influence. The effect of different emitter-detector spacings is shown in Fig. 16 (setup according to Fig. 15).

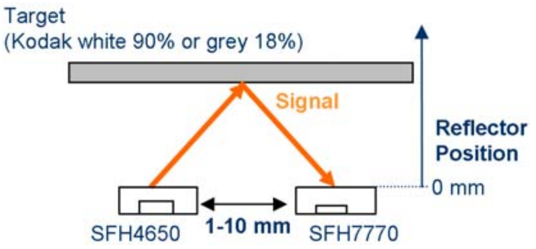

Fig. 15: *Setup for SFH 7770 E6 and one LED SFH4650.*

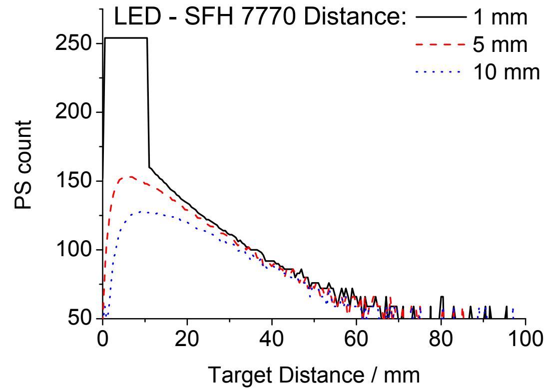

Fig. 16: SFH 7770 E6 proximity signal vs. reflector distance vs. LED SFH 4650 - SFH 7770 spacing (at low LED currents).

If the signal is within the operating range of the SFH 7770 E6, the reading of the proximity sensor decreases in an almost linear way with the distance of the object/target. In this case the PS reading can be used to estimate the distance quantitatively. Please note that the intensity of the reflected signal depends strongly on the reflectivity of the object (e.g. color of the hair). This has to be taken into account during the evaluation of the signal.

7.1.2 Setups for Gesture Recognition

Detecting an object's movement in 3 dimensions can be done with a rectangular arrangement of SFH 7770 E6 and 3 LEDs. Fig. 17 shows two possible arrangements.

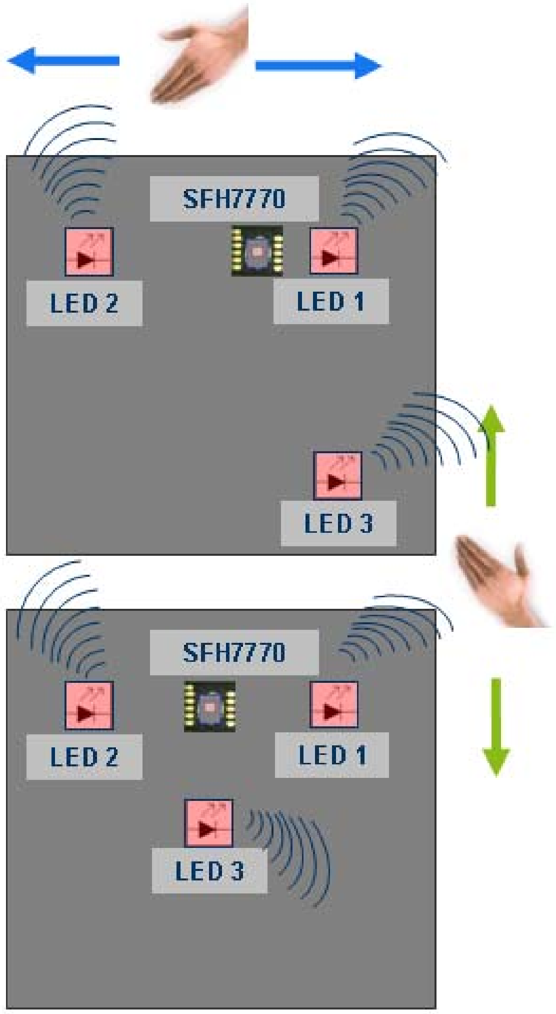

Fig. 17: Setups for 3D gesture recognition with SFH 7770 E6, comprising three LEDs. The top drawing shows the 'L' arrangement whereas the bottom presents the 'T' setup.

The setup shown in the upper part of Fig. 17 has LED1 placed close to the SFH 7770 E6, while LED2 and 3 are placed in some distance in a rectangular "L" type fashion. LED1 is intended to measure the distance of the moving object, as discussed in the previous section. Observing the signals of LED1 and 2 simultaneously gives information on left/right movements of the object, while LED 1 and 3 measure forward/backward movements. This setup is preferably used when the components have to be arranged around a rectangular shaped touchscreen or the display of a mobile phone.

If the SFH 7770 E6 and the LEDs can be arranged more freely, it is recommended to use the T-shaped setup according to Fig. 17. The components are arranged in a "T" shaped form, with all LEDs placed in equal distance from the SFH 7770 E6. When an object finds itself right above the SFH 7770 E6, all 3 proximity channels will have approximately the same signal height. The "L" shaped setup in Fig. 17 will always show a dominant signal from LED1, since it is placed close to the SFH 7770 E6. The "T" shaped setup allows easy interpretation of the relative signals of the 3 LEDs. As an example, the ratio of the proximity signals from LED 2 and 1 gives direct information on the left/right position of the object.

In order to draw such conclusions it is assumed that all LEDs have the same radiation characteristics and optical power which is not the case in reality. To cover the variations of the LEDs it may be necessary to perform a calibration of each ensemble individually.

7.1.3 Example: Gesture Detection

In this section, a sample setup for gesture recognition is presented, together with a flowchart for distinguishing different movements of a reflecting object.

For the hardware setup one of the configurations from the previous section is used, see Fig. 18. The SFH 7770 E6 is surrounded by 3 LEDs of type SFH 4059 at a distance of 1.5 cm. In order to eliminate optical crosstalk, a separator frame of height ~ 3 mm is inserted between the LEDs and the SFH 7770 E6.

The setup is intended to detect horizontal hand movements. The open hand is expected to be moved left/right and back/forth across the arrangement in a distance of a few centimeters above the SFH 7770 E6. The movement direction is

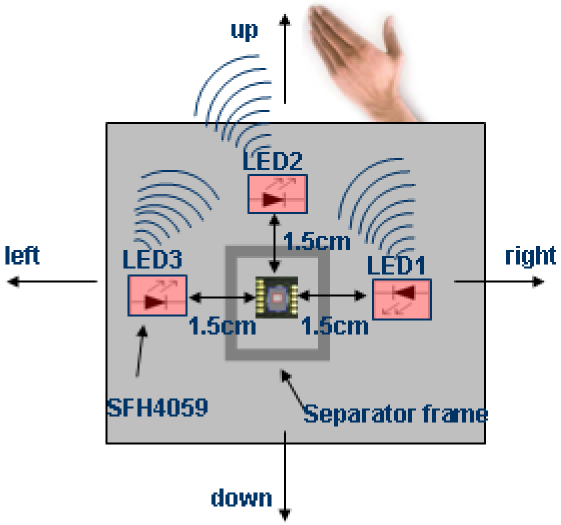

Fig. 18: Sample setup for gesture recognition with SFH 7770 E6, each comprising 3 LEDs.

detected by evaluating the relative signal height of LEDs 1, 2 and 3.

A flowchart how to derive the direction of the hand movement is presented in Fig. 19. The operating principle of the flow is as follows:

1) *Readout of all PS channels every 10ms.*
A repeat rate of 10ms is the fastest possible measurement rate for the proximity sensor. The actual measurement of all three proximity channels occurs within 1 ms to 2 ms (depending on the PS integration time setting, i.e. after all three LED bursts are finished). The PS data are then available and accessible via the I²C bus. The total minimum time delay is here 1 ms.

2) *Check if a least one PS signal (channel 1, 2 or 3) is above a given threshold.*
In order to eliminate unwanted response of the sensor to noise, the signal value must be above a fixed threshold level. A suitable level has to be determined for each application separately. Typical levels are ~80-100 output counts of the proximity sensor.

3) *Determine which channel has the maximum and minimum signal values*

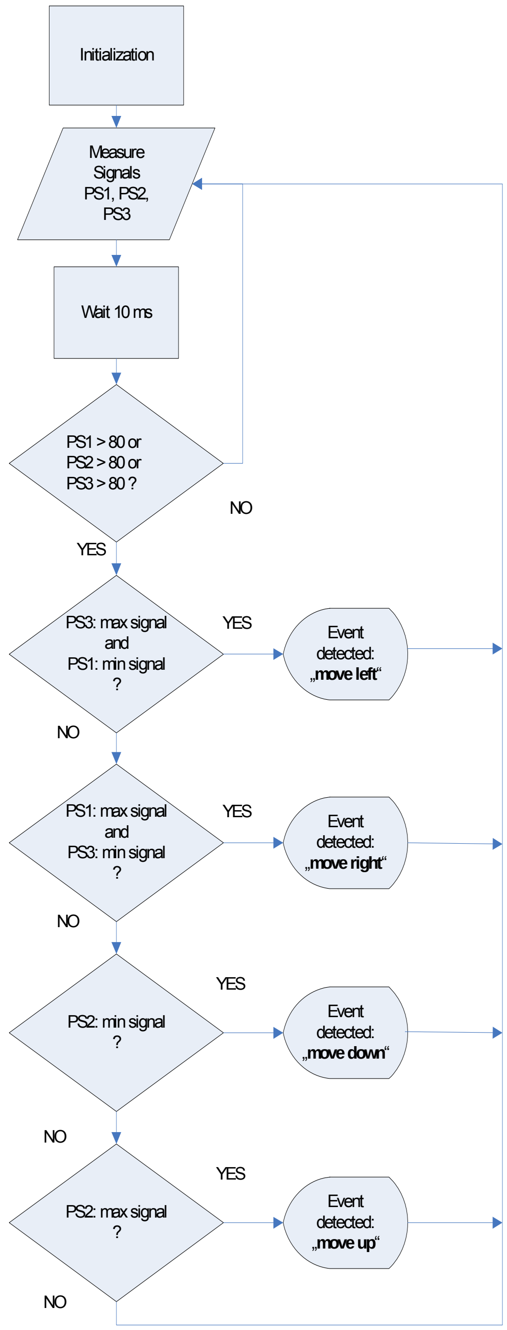

Fig. 19: Gesture recognition sample flow for SFH 7770 E6 and 3 LEDs.

The minimum and maximum evaluation is only done if at least one of the considered proximity signals is above the threshold level. Otherwise, a new proximity measurement is initiated.

Emitter Type	Image	Half Angle	Dimensions (LxWxH)
SFH4059		+/-10°	3.20 mm x 1.60 mm x 1.85 mm
SFH4650		+/-15°	3.10 mm x 2.25 mm x 1.60 mm
SFH4253		+/- 60°	3.20 mm x 2.80 mm x 1.90 mm

Tab. 12: Selection of 850 nm IR emitters.

4) *Decide from maximum and minimum reading which event has taken place*
For each of the four events, different (and exclusive) criteria have to be fulfilled. For example, if channel one and three deliver the maximum and minimum signal, respectively, the event is interpreted as a desired movement to the right (direction from SFH 7770 E6 to the LED1).

7.2 LED Types

The proximity sensor SFH 7770 E6 is designed for the use of 850 nm infrared LEDs. The typical radiation characteristics depend on the type of the LED.
If collimation optics (like lens, reflector) are included, usually the emission half angle of the LED is small. An LED without collimation optics shows a nearly Lambertian (cosine shaped) angular characteristic, with an emission half angle of ~±60°.
The dependence of the PS reading on the emitter type is shown in Fig. 21 (setup according to Fig. 20). A stripe shaped reflector with 15 mm width is swept over SFH 7770 E6 and one LED. The reflector shape is supposed to imitate a finger which is swept across the devices. Please note the pseudo-logarithmic scaling. 10 counts represent around a factor of 1.55.
The sweep was done at a (vertical) distance of 25mm from the SFH 7770 E6 / LED. At reflector position "0" the reflector is centered above the SFH 7770 E6.

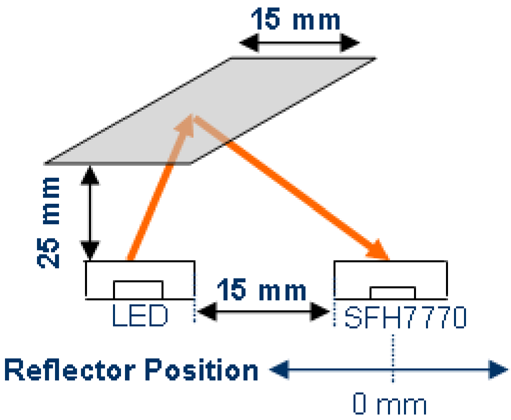

Fig. 20: stripe-shaped reflector is swept across the SFH 7770 E6 and the LED.

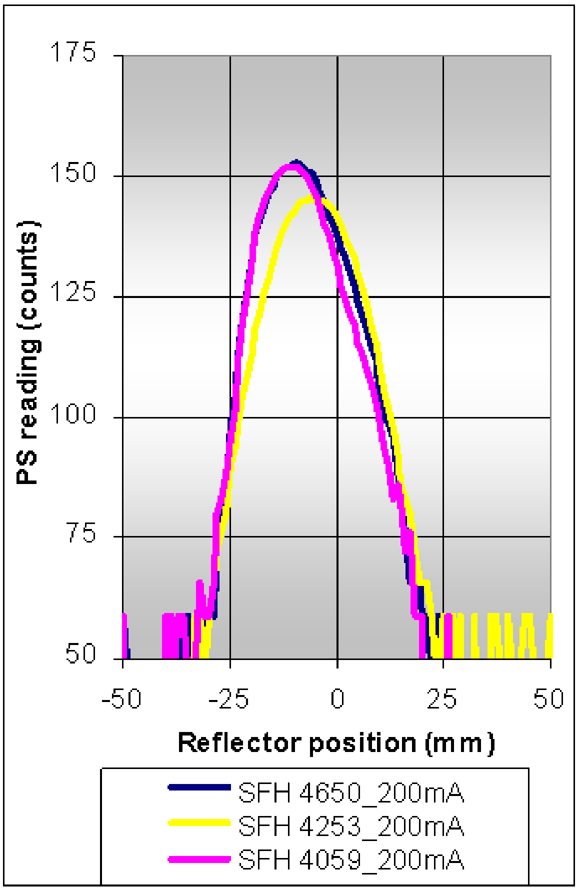

Fig. 21: SFH 7770 E6 proximity sensor reading during the reflector sweep for 3 LED types.

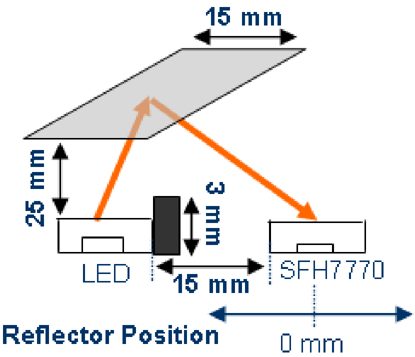

Fig. 22: Setup for the sweep of stripe-reflectors with aperture.

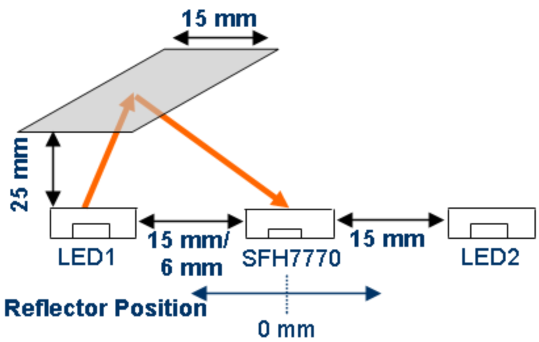

Fig. 24: Setup for the sweep of stripe-reflectors.

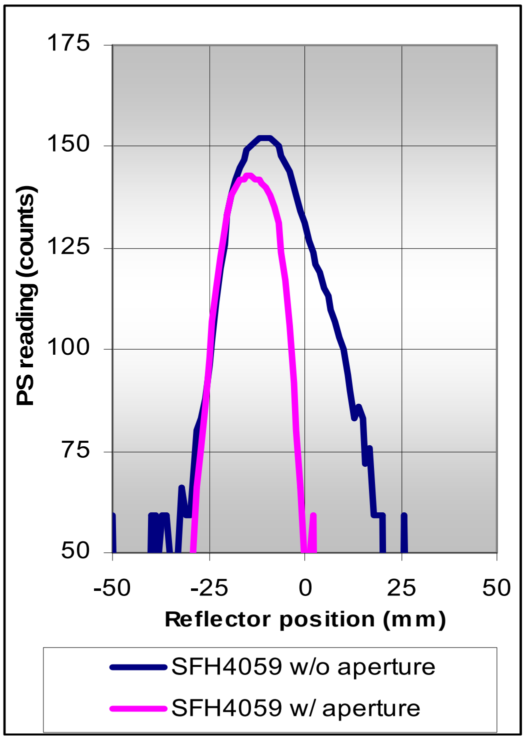

Fig. 23: SFH 7770 E6 proximity sensor reading at sweeping a stripe reflector with Emitter SFH4059 with aperture (according to setup in Fig. 24).

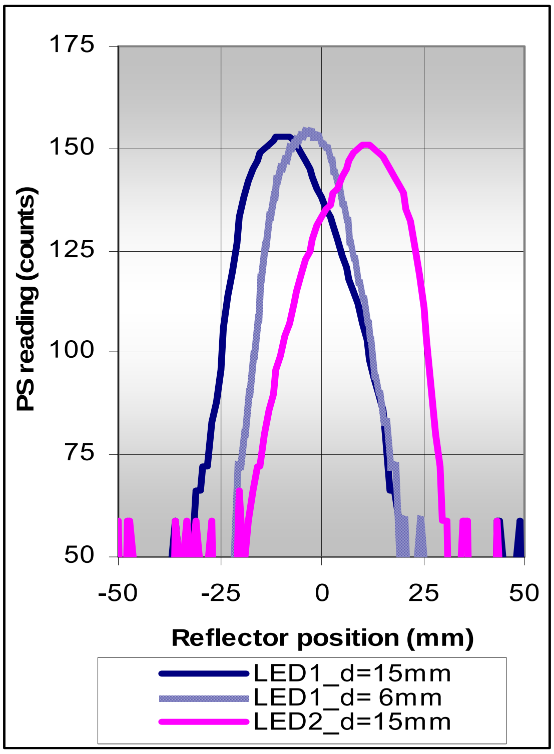

Fig. 25: SFH 7770 E6 proximity sensor reading at sweeping a stripe reflector (Emitter 4650) at various spacings according to setup in Fig. 22.

The sweep has been performed for 3 different emitters: SFH 4650, SFH 4059 and SFH 4253. The emitters and the corresponding emission angles are shown in Tab. 12. For more information on IR emitters please visit the OSRAM OS website at http://www.osram-os.com/osram_os/EN/.

Fig. 21 also indicates that emitters with a narrow emission angle and high intensity are resulting in a higher reflected peak signal. The reflector position where the reflected signal has its maximum is shifted as well.

The choice of the emitter type depends on the requirements of the application. If a high signal at a comparably small spatial region is desired, an emitter with narrow emission angle will be the best choice. If detection contrast is not an issue and a wide spatial region has to be covered, a wide angle emitter will fit best.

7.3 Separators and LED Spacing

A stronger spatial confinement of the radiation characteristic and hence the detection region can be achieved by the use of apertures positioned above the sensor. Any restriction of the LED beam leads to a better contrast at gesture recognition applications. On the other hand, any aperture reduces the intensity of the reflected signal.

The effect is demonstrated in Fig. 23. The same reflector sweep has been performed as in Fig. 21. The aperture is created by placing an optical absorber with 3 mm height next to the LED.
The distance between SFH 7770 E6 and the LED also influences the relative signals for multi-LED operation and hence the contrast at gesture recognition applications.

In Fig. 24 resp. 25 two LEDs (SFH 4650) are operated with SFH 7770 E6. The LEDs are placed on opposite sides of the SFH 7770 E6 and a stripe reflector is moved horizontally. The difference between the signals from LED1 and 2 allows making a good estimate for the position of the reflector. When the LEDs are spaced at 15 mm from the SFH 7770 E6, the detection contrast is sufficient for many applications. When the LEDs are moved closer to the SFH 7770 E6 (to 6 mm in Fig. 25), the LED beams are overlapping more and the contrast between the two signals gets lower. At the same time, the signal height increases. Thus, a tradeoff between signal height and contrast has to be performed.

7.4 Optical Crosstalk

When proximity sensing is performed with up to three LEDs, it is desirable that only light from a reflecting object reaches the SFH 7770 E6. Depending on the optical setup, additional and unintended light paths from the LED to the detector may exist, which is referred to as ‘(optical) crosstalk’. In this section, several sources of crosstalk are discussed, together with measures for suppression of the effect.

Usually, the LED and the SFH 7770 E6 are operated behind a cover glass, which has a reflectivity of typically 4% for each surface. If there is an air gap between LED/SFH 7770 E6 and the glass, a considerable portion of light gets reflected directly to the detector by the glass surfaces, see Fig. 26. In extreme cases, the signal reflected via the cover glass exceeds the signal of interest. The result is a decreased operating range for the detection distance: the signal of interest clearly has to exceed the noise floor generated by the crosstalk, in addition to the SFH 7770 E6 inherent noise.

The crosstalk can be reduced by introducing a separator between IRED and the detector, see also Fig. 26. A careful design of separator (width, height) can minimize range reduction by eliminating the crosstalk. The remaining crosstalk may be suppressed by offset subtraction.

Fig. 27 shows additional sources of crosstalk:

a) Since the reflection of the IR light occurs not only at the bottom surface of the cover, but also on the top side, a separator design is recommended, which is blocking both reflections. The separator material should be absorbing and preferably diffusely reflecting. This also leads to the attenuation of multiple reflections within the cover.

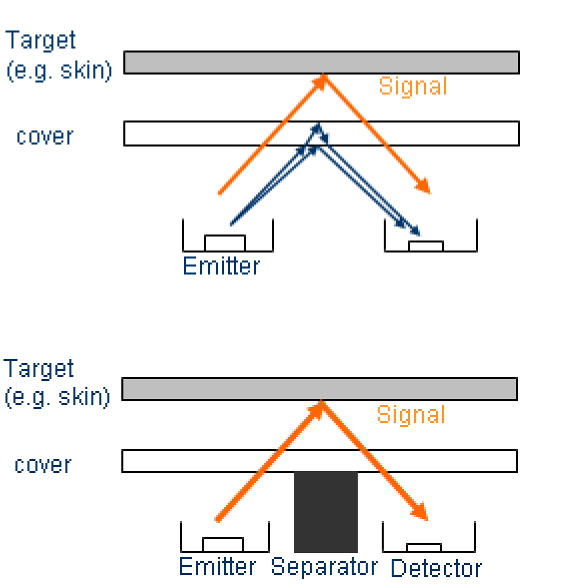

Fig. 26: Top: Optical crosstalk via cover glass. Bottom: Crosstalk elimination by introducing a separator.

b) If an air gap between separator and cover glass exists, additional light paths between emitter and detector may be created. Especially, when the surface of the separator is reflective, light from the emitter may reach the detector via multiple reflections (see also Fig. 27). In order to avoid this, the surface of the separator should be diffusely reflective. Additionally, it should be placed as close to the cover glass as possible.

c) If there is a gap between the separator and the PCB upon which the LED and the SFH 7770 E6 are mounted, some reflection may occur also at the bottom of the separator. The height of the air gap should be minimized as well.

For further technical support please contact OSRAM OS sales at http://www.osram-os.com

7.5 Zero-Distance Detection

In some cases the detection of objects is required, which are in direct touch with the

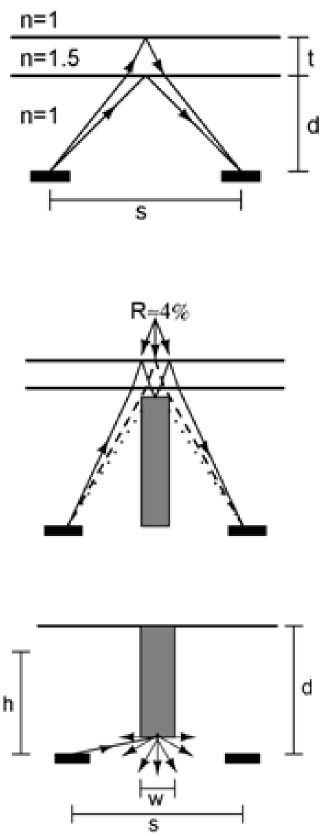

Fig. 27: Top: Optical crosstalk via cover glass. Middle: Crosstalk via cover glass & separators top surface. Bottom: Crosstalk via bottom side of a separator.

cover glass ('zero distance detection', see Fig. 28). For example, this 'object' can be a finger, hair or the human ear. In principle, the SFH 7770 E6 is capable of detecting the presence of objects in direct contact with the cover. It is important to design the separator between LED and SFH 7770 E6 carefully. If it is too wide, it will block the signal reflected by the object. If it is too narrow, the crosstalk signal will increase and thus mask the signal of interest.
The light path for reflection from the zero-distance object is very similar to the reflection at the top side of the cover glass. Nevertheless, zero distance detection is made possible by the following effects:

a) The reflecting object (e.g. hand) is not placed exactly above the separator.

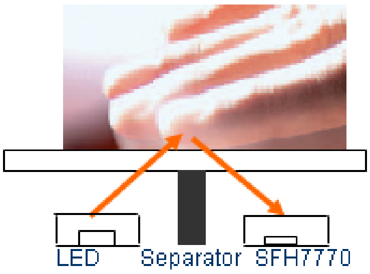

Fig. 28: Zero-distance detection.

b) Although a finger is placed on the cover, the light scattering takes place within the skin, and not on its surface.

7.6 Placement of the SFH 7770 E6

The photosensitive area of both ALS and PS detectors is located within a square of 0.5 mm x 0.5 mm in center of the package. Fig. 29 indicates the position of the sensitive area.
At designing apertures and the field of view of the sensor, only this sensitive area has to be taken into account. The same holds for placing the part behind a light guide.

When placing the sensor behind a cover window opening, the recommended cover opening aperture α is up to e.g. > +/- 60° (see Fig. 30), depending on the radiation characteristics of the LED.

Optical properties of a light guide or the limited aperture based on a cover window opening will shadow the sensor within certain angular ranges. The overall directional characteristic is getting changed and the horizontal detection range will be reduced.

8 Electrical Design Guidelines

This section contains guidelines for electrical circuitry and operation of SFH 7770 E6.

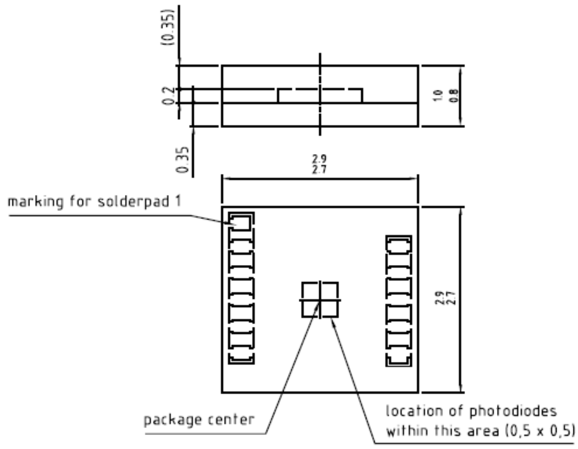

Fig. 29: Position and size of the sensitive area within the SFH 7770 E6.

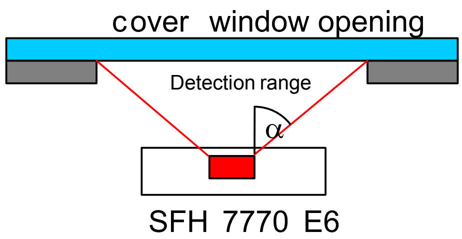

Fig. 30: Cover window opening.

8.1 General Circuitry and Pinning

The SFH 7770 E6 is a 10-pin SMT device. The pin assignment is shown in Fig. 31 and 32, together with some external components which are recommended for operation. The 10 pins may be classified as follows:

a) *Supply voltage* (V_{DD}, GND, pins 6 and 4): For safe operation of the device, the supply voltage must be in the range 2.3V …3.1V. A bypass capacitor of >100nF close to the device is recommended.

b) *LED connectors* (LED1, 2 & 3, LED_GND, Pins 1-3, 7): Each LED pin is set up as open drain output, where the cathode of an external 850 nm IRED can be connected to. The anodes of all LEDs are connected to an external voltage V_{LED}, which should be buffered, i.e. connected directly to the mobile phone battery. The LED_GND connector is separated from the GND pin.

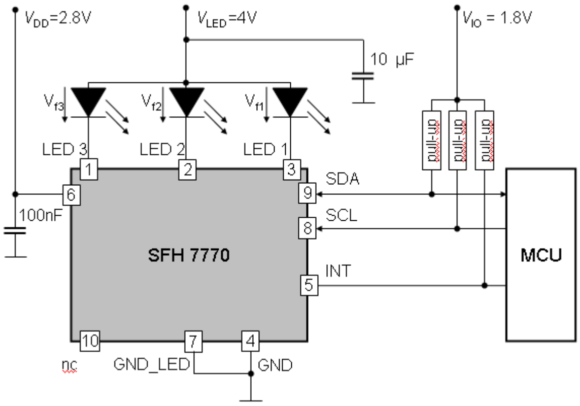

Fig. 31: Application diagram for basic operation of SFH 7770 E6.

Please note that the voltage difference between GND and LED_GND must not exceed 500 mV for proper operation of the device.

c) *I²C bus pins* (SCL, SDA, pins 8 & 9): The I²C bus communication with a microcontroller unit is performed via these pins. Pull-up resistors to a voltage level of 1.6V … 2.0V are required.

d) *Interrupt pin* (INT, pin 5): The INT pin is set up as an open drain logic output, which should be connected to an external logic voltage via a pull-up resistor. The resulting logic signal can be evaluated by external circuitry, like a microcontroller unit.

e) *Pin 10:* not connected. It is recommended to connect pin 10 to GND.

The pin assignment of SFH 7770 E6 is shown in Fig. 32 for top/bottom view. At top view, pin 1 is clearly marked with an extra metal spot. The size of the soldering pads of the device can also be seen from Fig. 24.

8.2 Supply Voltages

The SFH 7770 E6 is suitable for a V_{DD} supply voltage of 2.3 V to 3.1 V.
To achieve maximum sensitivity concerning the proximity functionality it is mandatory to have a stable (battery-like) power supply.

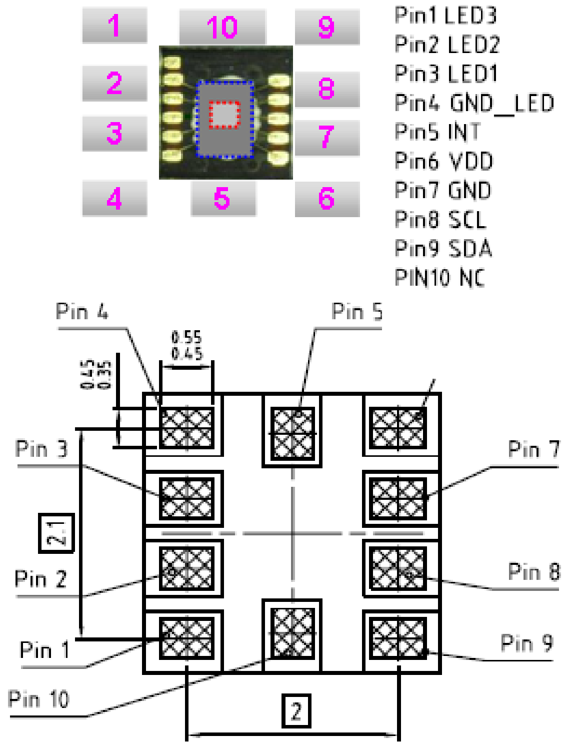

Fig. 32: Pin assignment of SFH 7770 E6.

The recommendation therefore is to connect V_{LED} directly to the battery. This ensures the necessary LED current during the burst operation (up to 200 mA peak, depending on the actual settings of the proximity sensors LED current). It is further recommended to use capacitors as close to the component as possible. This ensures minimum voltage drops at the supply pins of the SFH 7770 E6 and provides the necessary peak burst current. Typ. values are 100 nF || 4.7 µF at the V_{LED} side (for up to 200 mA burst current) and 100 nF || 1.0 µF for the V_{DD} circuit (ASIC supply). The 4.7 µF capacitor can be reduced if the LED burst circuit is reduced to lower levels, e.g. 50 mA.

8.3 LED Connection and Operation

The SFH 7770 E6 is suitable to handle up to three LEDs. During driving the LEDs, the parasitic inductances and capacitances are very important, especially at LED currents > 20 mA.

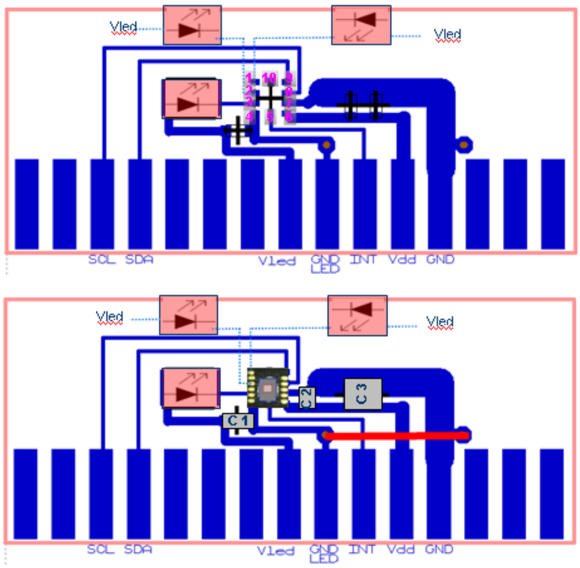

Fig. 33: Sample demo board pcb design (top) and component placement (bottom) for operating the SFH 7770 E6 with up to three LEDs.

It is recommended to use a low-inductance layout for all LED connections. The parasitic LED wire inductances must be kept low (e.g. < 40 nH for 200 mA LED current, < 400 nH for 20 mA LED current). Special care has to be taken if the distance between the LED and the SFH 7770 E6 is in the range of several centimeters, as required for e.g. gesture recognition applications. In this case, a PCB design with low inductance paths has to be found for the connecting paths between SFH 7770 E6 and the LED. Furthermore, a bypass capacitor between V_{LED} and GND_LED with >1.0 µF is strongly recommended.

Overall it is possible to use the SFH 7770 E6 and trigger an external circuit with 5 mA 'LED' drive circuit to trigger the LED with the desired drive current. This is highly recommended if the distance between LED and SFH 7774 is beyond the above limits. It also allows driving the LED with currents beyond 200 mA to maximize the detection distance.

Please note that during the 'LED-off'-state an off-current of 0.5 mA is driven by the SFH 7770 E6.

8.4 PCB Design

As mentioned in the previous sections, the use of a low inductance wiring is advisable, especially for LED connections. A pcb with at least 2 layers is recommended, where one layer is used as a ground plane. The GND and the LED_GND pins of SFH 7770 E6 should be connected to the ground plane close to the device.

Fig. 33 shows a demo pb design for SFH 7770 E6 and one LED. The connections to two further LEDs are marked as dotted lines. As mentioned above, the design should include an additional ground plane (not shown). The V V_{LED} line is buffered with a single 10 µF capacitor (C1), while the VDD line is blocked by capacitors o (C2 and C3).

In addition, the board layout has to be designed in a way that the capacitance on SCL and SDA line fulfills the I²C bus specification [3]. The maximum clock frequency correlates with the bus frequency. The high-speed mode (Hs) of 3.4 MHz requires a maximum bus line capacitance for SDA and SCL of 100 pF. For the Hs mode with 1.7 MHz the bus capacitance can be increased up to 400 pF.

8.5 Pull-Up Resistors

Pull-up resistors are used to reference the digital data lines SDA & SCL to the V_{IO} level. The maximum useable pull-up resistor depends on the overall bus capacitance (wires, connections and pins) due to the specified I²C bus rise time. Sink current needs to be calculated for each application. Details and instructions are provided at [3].

9 Sample Software Code

Below are simple C-codes which can be used to operate the SFH 7770 E6 in connection with a microcontroller (e.g. PIC18F46J50 from Microchip). The program consists of the commented main micro C-code for the microcontroller, using the two subroutines

`I2C_w_3`: 3 write statements
`I2C_w_2_r_1`: 2 write and 1 read statement.

The main program can be implemented into a repeating loop to get the actual PS resp. ALS data or operate in interrupt mode.

9.1 Operating the ALS

9.1.1 C-code in main program:

```
sfh_address = 0x38;                       // address of SFH 7770 E6
I2C_w_3 (sfh_address*2, 0x80, 0x03);      // initialize ALS of the SFH 7770 E6
I2C_w_2_r_1 (sfh_address*2, 0x8C);        // read low byte of ALS, register 0x8C
lux  =  Content;
I2C_w_2_r_1 (sfh_address*2, 0x8D);        // read high byte of ALS, register 0x8D
lux  =  (lux + Content* 256);             // combining low+high byte to decimal value
```

9.1.2 I2C_w_3 subroutine

```
void I2C_w_3     (unsigned char addw, unsigned char com, unsigned char daw)
{
   unsigned char var;
   OpenI2C (MASTER, SLEW_ON);     // ConFig.s I2C bus module, 100 kHz transfer
   SSP1ADD  = 0x27;               // setting I2C 100 kHz frequency with f osc = 16 MHz
   StartI2C ();                   // Generates I2C bus start condition
   IdleI2C ();                    // Loop till I2C bus is idle
   var = WriteI2C(addw);          // Microchips' Write command to write device address
   if (var ==  0) write_s++;      // var =  0: no bus error
   if (var == -1) write_c++;      // var = -1: slave did not acknowledge write
   if (var == -2) write_ac++;     // var=-2:write collision (bus not ready to tx)
   if (var  <  0) goto stop;      // stop further transmission if error occurred

   var = WriteI2C(com);           // write device register address
   if (var ==  0) write_s++;      // counting of good transmissions
   if (var == -1) write_c++;      // counting of no acknowledge errors
   if (var == -2) write_ac++;     // counting of write collision errors
   if (var  <  0) goto stop;

   var = WriteI2C(daw);           // write register content
   if (var ==  0) write_s++;
   if (var == -1) write_c++;
   if (var == -2) write_ac++;

   stop:
   StopI2C ();                    // generates I2C bus stop condition
   CloseI2C ();                   // master I2C module disabled
}
```

9.1.3 Subroutine I2C_w_2_r_1

```
void I2C_w_2_r_1 (unsigned char addr, unsigned char com)
{
   unsigned char var;
   OpenI2C (MASTER, SLEW_ON);
   SSPADD   = 0x27;
   StartI2C ();
   IdleI2C ();
   var = WriteI2C(addr);
   if (var ==  0) read_s++;
```

```
    if (var == -1) read_c++;
    if (var == -2) read_ac++;
    if (var  <  0) goto stop;

    var = WriteI2C(com);
    if (var ==  0) read_s++;
    if (var == -1) read_c++;
    if (var == -2) read_ac++;
    if (var  <  0) goto stop;

    RestartI2C ();                      // generates I2C bus restart condition
    IdleI2C ();
    var = WriteI2C(addr+1);
    if (var ==  0) read_s++;
    if (var == -1) read_c++;
    if (var == -2) read_ac++;
    if (var  <  0) goto stop;
    Content = 0;
    Content = ReadI2C ();
    SSPCON2bits.ACKDT = 1;              // No master Acknowledge to terminate sequence
    SSPCON2bits.ACKEN = 1;              // sending No Acknowledge bit
    PIR1bits.SSPIF = 0;
    while (SSPCON2bits.ACKEN == 1);
    PIR1bits.SSPIF = 0;                 // waiting till NA causes interrupt
    stop:
    StopI2C ();
    CloseI2C ();
}
```

9.2 Operating the PS

Below is a small C-code for the main program to operate the proximity sensor of the SFH 7770 E6. The two subroutines, I2C_w_3 and I2C_w2_r1 are the same as above.

C-code for main program:

```
sfh_address = 0x38;                   // address of SFH 7770 E6
I2C_w_3 (sfh_address*2, 0x81, 0x03);  // initialize PS of the SFH 7770 E6
I2C_w_3 (sfh_address*2, 0x82, 0x1E);  // set PS LED current to 200 mA
I2C_w_2_r_1 (sfh_address*2, 0x8F);    // read data byte of PS, register 0x8F
PS  =  Content;
```

9.3 Operating the ALS and PS in Interrupt Mode

The small C-code below operates the SFH 7770 E6 in the interrupt mode. The ALS and PS are in free-running mode. The interrupt event can occur through an ALS or PS event. The limits for ALS `(LB_LL, HB_LL, LB_HL, HB_HL)` and PS `(Prox_Limit)` are set within the program. After the interrupt has triggered the microcontroller the relevant sensor is determined and the ALS or PS value is read out.

C-code for main program:

```
// ALS:

I2C_w_3         (0x38*2, 0x80, 0x03);         // ALS free running mode
I2C_w_3         (0x38*2, 0x86, 0x00);         // new data every 100 ms
I2C_w_3         (0x38*2, 0x98, LB_LL);        // setting low byte of low ALS limit
I2C_w_3         (0x38*2, 0x99, HB_LL);        // setting high byte of low ALS limit
I2C_w_3         (0x38*2, 0x96, LB_HL);        // setting low byte of high ALS limit
I2C_w_3         (0x38*2, 0x97, HB_HL);        // setting high byte of high ALS limit

// Prox:

I2C_w_3         (0x38*2, 0x81, 0x03);         // Prox free running mode
I2C_w_3         (0x38*2, 0x82, 0x1E);         // IR LED with 200 mA
I2C_w_3         (0x38*2, 0x85, 0x00);         // new data every 10 ms
```

```
I2C_w_3        (0x38*2, 0x93, Prox_Limit);     // setting byte for high prox limit

I2C_w_3        (0x38*2, 0x92, 0x03);           // interrupt triggered by PS and ALS,
latched and ground when active

// Interrupt routine:                          // called when interrupt happened

I2C_w_2_r_1 (0x38*2, 0x8E);
// reading Status Register, Function returns register value as variable Content
if ( (Content & 0x80) == 0x80)
// &=bitwise AND,check whether ALS triggered interrupt
{
       I2C_w_2_r_1 (0x38*2, 0x8C);             // read low byte of ADC, register 0xC
       Content1 = Content;
       I2C_w_2_r_1 (0x38*2, 0x8D);             // read high byte of ADC, register 0xD
       Lux = Content * 256 + Content1;
}

Else                                           // Interrupt must be caused by prox
sensor
{
       I2C_w_2_r_1 (0x38*2, 0x8F);             // read Prox data register 0x8F
       Prox = Content;                         // Value in uW/cm^2 =10power(Content/51)
}

// end of interrupt routine
```

9.4 Implementation into a Mobile Phone Environment

Below are two example flowcharts, describing how the SFH 7770 E6 can be implemented into a microcontroller based mobile phone environment. The interrupt function allows for low-power stand-alone operation of the device.
The first flowchart illustrates a possible operation of the ambient light sensor, the second flowchart relates to the operation of the proximity sensor.

9.4.1 Operation of the ALS

Fig. 34 illustrates a flowchart for a microcontroller based ambient light sensing example. The interrupt (set to active low) alerts the microcontroller only in case the actual ambient light value is outside of the defined ALS window. Using the interrupt functionality and operating the SFH 7770 E6 in the free-running mode helps to minimize traffic on the I^2C-bus as well as to relieve the microcontroller from unnecessary work load.

This arrangement helps to save valuable battery power.

By adapting dynamically new thresholds (with hysteresis) relative to the actual ALS value (after an interrupt event took place) it is possible to define very fine steps for adapting the display brightness (quasi-continuous).
By inverting the interrupt polarity (register 0x92) the interrupt alert function can be inverted from outside the ALS window to inside the ALS window (only in non-latched mode). Like stated above, it is recommended to use a hysteresis by defining the thresholds in order to avoid flickering of the interrupt event.

9.4.2 Operation of the PS

Fig. 35 illustrates the flowchart for a microcontroller based proximity sensing example. Operating the SFH 7770 E6 in the stand alone mode plus using the interrupt functionality helps to save battery power.

The interrupt (set to active low) alerts the microcontroller only in case an object passes a certain distance threshold (towards the sensor, e.g. in a mobile phone). This allows the mobile phone to turn-off the display e.g. during a call to save battery power.
A new threshold (with hysteresis) and the inverting of the interrupt logic of the SFH 7770 E6 - after an event has taken place - allow to adapt the sensor to detect the motion in the opposite direction (only for non-latched interrupt mode). By adapting dynamically new thresholds it is recommended to set a certain hysteresis level to avoid flickering of the interrupt event.

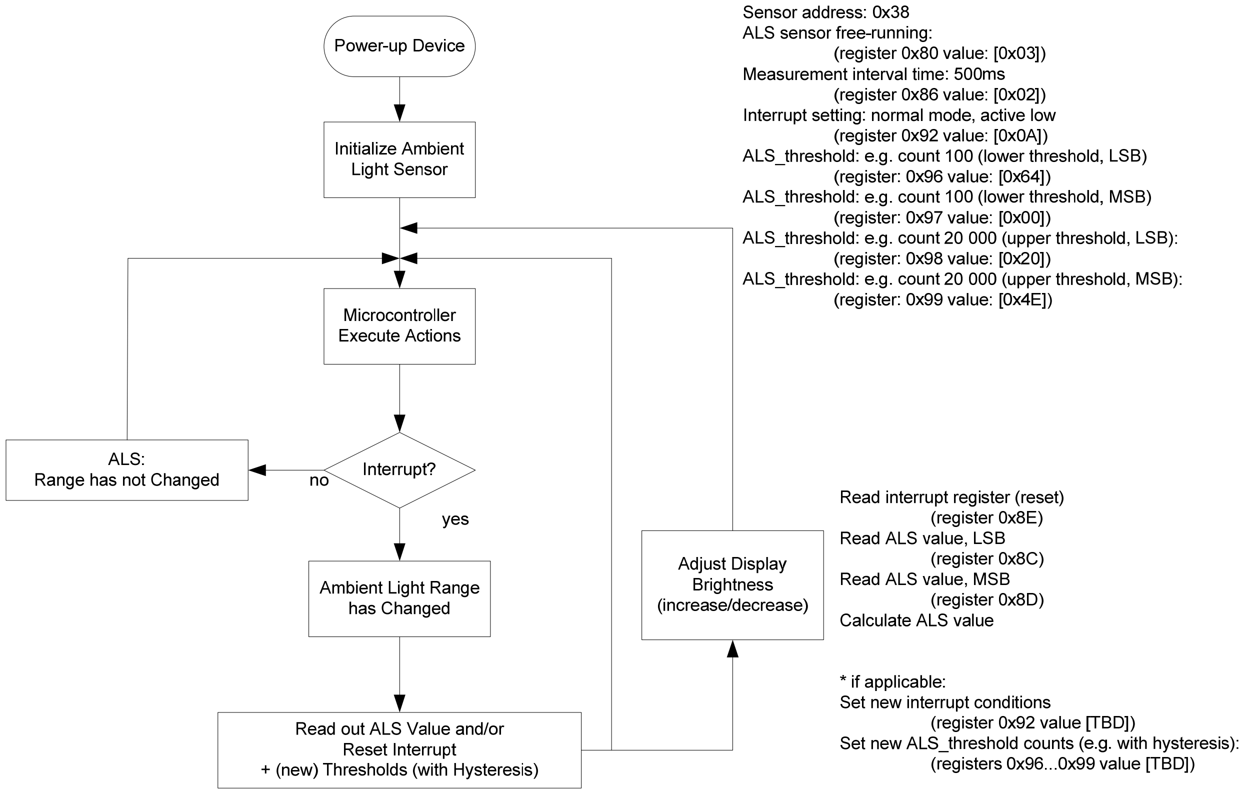

*Fig. 34: Flowchart for a microcontroller based **ambient light sensing** example.*

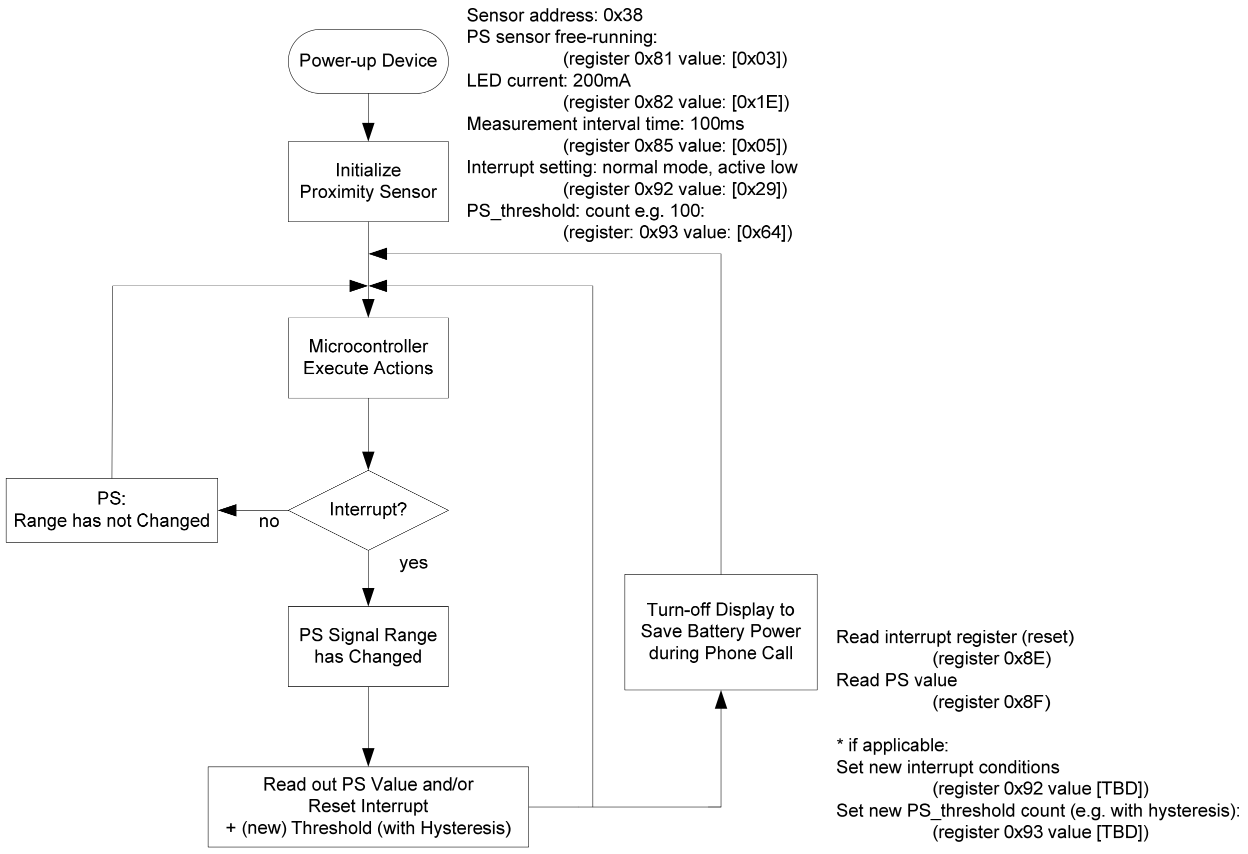

*Fig. 35: Flowchart for a microcontroller based **proximity sensing** example.*

13. Literature

[1] Application notes can be downloaded from **http://www.osram-os.com/osram_os/EN**
[2] SFH 7770 E6 Datasheet can be downloaded from **http://www.osram-os.com/osram_os/EN**
[3] "UM10204 I2C-bus specification and user manual" from NXP Rev. 03 – 19 June 2007
[1] Application notes can be downloaded from **http://www.osram-os.com/osram_os/EN**

Author: Dr. Bernhard Stojetz, Dr. Hubert Halbritter