MLX90632 FIR sensor

Datasheet

Features and Benefits

- Small size of 3x3mm

- Easy to integrate

- Factory calibrated

- External ambient and object temperature calculation

- Standard measurement resolution 0.02°C

- Medical measurement resolution 0.01°C

- Supply voltage of 3.3V, supply current 1mA (sleep current less than 2.5uA)

- I^2C compatible digital interface

- Software definable I^2C address with 1 LSB bit external address pin

- Field of View of 50°

- Default refresh rate 0.5s, configurable between 16ms and 2s

- Integrated post-calibration option

Application Examples

- High precision non-contact temperature measurements

- Body temperature measurement

- Non-contact thermometer for mobile and IoT application

- Temperature sensing element for residential, commercial and industrial building air conditioning

- Industrial temperature control of moving parts

- Home appliances with temperature control

- Healthcare

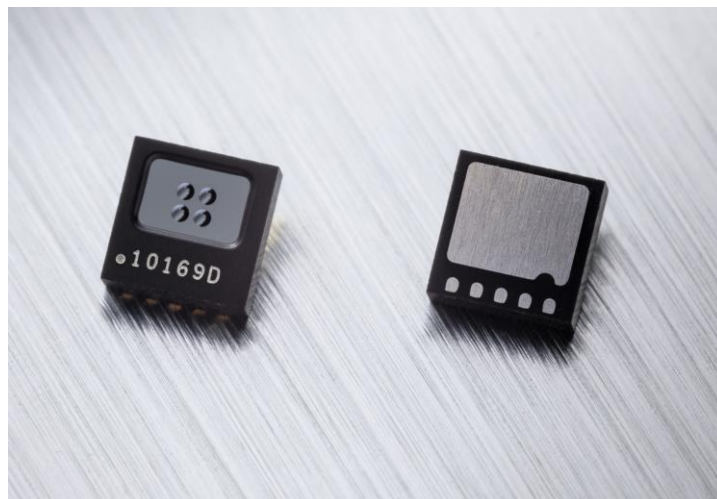- Livestock monitoring

- Drivers available at: https://github.com/melexis/mlx90632-library

Figure 1: Image of MLX90632

Melexis
INSPIRED ENGINEERING

MLX90632 FIR sensor

Datasheet

Description

The MLX90632 is a non-contact infrared temperature sensor in a small SMD SFN package. The device is factory calibrated with calibration constants stored in the EEPROM memory. The ambient and object temperature can be calculated externally based on these calibration constants and the measurement data.

A major strength of the MLX90632 is that the effect of temperature variations around the sensor package are reduced to a minimum. However, some extreme cases can still influence the measured temperature.
The accuracy of the thermometer can be influenced by temperature differences in the package induced by causes like (among others): hot electronics behind the sensor, heaters/coolers behind or beside the sensor or by a hot/cold object very close to the sensor that not only heats the sensing element in the thermometer but also the thermometer package. The MLX90632 contains specific circuits to minimize the effect of these influences. In the same way, localized thermal variations -like turbulence in the air- will not generate thermal noise in the output signal of the thermopile.

The MLX90632 is available in two different versions: standard and medical accuracy.
Both versions are calibrated in the ambient temperature range from -20 to 85˚C.
The difference between both versions is visible in accuracy and the object temperature range.
The medical version is factory calibrated with an accuracy of ±0.2˚C within the narrow object temperature range from 35 to 42˚C for medical applications. This version also allows Extended range operation. This measurement type option is implemented in order to give additional range to the medical devices. The object temperature range is limited from -20 to 100˚C. For more information see Section 11.2.
On the other hand, the standard version covers an object temperature range from -20 to 200˚C but offers an accuracy of ±1˚C.
It is very important for the application designer to understand that these accuracies are guaranteed and achievable when the sensor is in thermal equilibrium and under isothermal conditions (no temperature differences across the sensor package). To obtain this accuracy, it is also important that the full Field of View (FOV) of the sensor is filled by the object.

The typical supply voltage of the MLX90632 is 3.3V. For the I^{2}C communication with the master microcontroller, two versions of the sensor are available, working either at 3.3V or 1.8V I^{2}C reference voltage.
The communication to the chip is done by I^{2}C up to fast mode plus (FM+).
Through I^{2}C the external microcontroller has access to the following blocks:

- RAM memory used for measurement data, in this document mainly referred to as 'storage memory'

- EEPROM used to store the trimming values, calibration constants and device/measurement settings

Based on this data, the external microcontroller can calculate the object temperature and if needed the sensor temperature.

An optical filter (long-wave pass) that cuts off the visible and near infra-red radiant flux is integrated in the sensor to provide ambient light immunity. The wavelength pass band of this optical filter is from 2 till 14µm.

Contents

1. Ordering Information

Product	Temperature Code	Package	Option Code	Packing Form
MLX90632	S	LD	BCB-000	RE
MLX90632	S	LD	DCB-000	RE
MLX90632	S	LD	DCB-100	RE

Table 1 : Ordering codes for MLX90632

Legend:

Temperature Code:	S: from -20°C to 85°C sensor temperature
Package Code:	"LD" for SFN 3x3 package
Option Code:	XYZ-123 X: Accuracy • B: Standard accuracy • D: Medical accuracy Y: Pixel type • C: High stability version Z: Field Of View • B: 50 degrees 1: I^2C level • 0: 3V3 • 1: 1V8 2-3: • 00: Standard configuration • xx: Reserved
Packing Form:	"RE" for Reel
Ordering Example:	"MLX90632SLD-DCB-000-RE" For a FIR Sensor type in SFN 3x3 package with medical accuracy, Field Of View of 50 degrees and 3V3 I^2C level, delivered in Reel.

Table 2: Coding legend

2. **Glossary of Terms**

POR	**P**ower **O**n **R**eset
IR	**I**nfra**R**ed
I^{2}C	**I**nter-**I**ntegrated **C**ircuit
SDA	**S**erial **DA**ta – I^{2}C compatible communication pins
SCL	**S**erial **CL**ock – I^{2}C compatible communication pins
ACK / NACK	**A**cknowledge / **N**ot **A**cknowledge
SOC	**S**tart **Of** **C**onversion
EOC	**E**nd **Of** **C**onversion
FOV	**F**ield **Of** **V**iew
Ta	**A**mbient **T**emperature measured from the chip – (the package temperature)
To	**O**bject **T**emperature, 'seen' from IR sensor
SFN	**S**ingle **F**lat pack **N**o-lead
TBD	**T**o **B**e **D**efined
LSB	**L**east **S**ignificant **B**it
MSB	**M**ost **S**ignificant **B**it
EMC	**E**lectro-**M**agnetic **C**ompatibility
ESD	**E**lectro-**S**tatic **D**ischarge
HBM	**H**uman **B**ody **M**odel
CDM	**C**harged **D**evice **M**odel

Table 3: List of abbreviations

3. Absolute Maximum ratings

Parameter	Symbol	Min.	Typ.	Max.	Unit
Supply Voltage, (over voltage)	V_{DD}			5	V
Supply Voltage, (operating)	V_{DD}			3.6	V
Reverse Voltage	V_R			-1.5	V
Address-pin Voltage	V_{ADDR}			$V_{DD} + 0.6$	V
Operating Temperature Range,	T_A	-20		+85	°C
Storage Temperature Range,	T_S	-40		+105	°C
ESD Sensitivity					
- HBM (acc. AEC Q100 002)				2	kV
- CDM (acc. AEC Q100 011)				750	V
- Air discharge (acc. IEC61000-4-2)				+4	kV
- Contact discharge (acc. IEC61000-4-2)				+2	kV
DC current into SCL				10	µA
DC sink current, SDA pin		20			mA
DC clamp current, SDA pin				25	mA
DC clamp current, SCL pin				25	mA
EEPROM re-writes				10	

Table 4: Absolute maximum ratings

Exceeding the absolute maximum ratings may cause permanent damage. Exposure to absolute maximum-rated conditions for extended periods may affect device reliability.

4. Pin definitions and descriptions
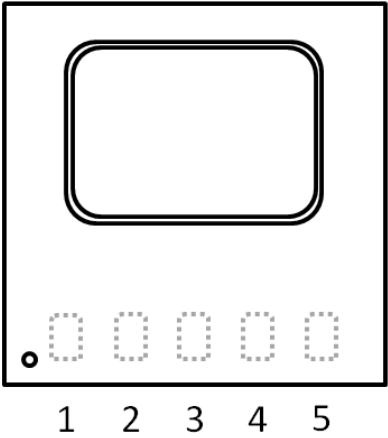
Figure 2: MLX90632 TOP view

Pin #	Name	Direction	Description
1	SDA	In/Out	I^2C Data line
2	VDD	POWER	Supply
3	GND	GND	Ground
4	SCL	In	I^2C Clock line
5	ADDR	In	LSB of I^2C address

Table 5: Pin definition

5. Electrical characteristics

All parameters are valid for T_A = 25 ˚C, V_{DD} = 3.3V (unless otherwise specified)

Parameter	Symbol	Test Conditions	Min	Typ	Max	Units
Supplies						
External supply	V_{DD}		3	3.3	3.6	V
Supply current	I_{DD}	No load	0.5	1	1.4	mA
Sleep current	I_{DDpr}	No load, erase/write EEPROM operations		1.5	2.5	uA
Power On Reset						
POR level	$V_{POR\ up}$	Power-up (full temp range)	1.3		2.4	V
POR level	$V_{POR\ down}$	Power-down (full temp range)	1.1		2.1	V
POR hysteresis	$V_{POR\ hys}$	Full temp range	200		500	mV
V_{DD} rise time (10% to 90% of specified supply voltage)	T_{POR}	Ensure POR signal			20	ms
Output valid (result in RAM)	Tvalid	After POR	64			ms
I²C compatible 2-wire interface						
I²C Voltage	V_{I2C}	I²C version = 1.8V I²C version = 3.3V	1.65 3	1.8 V_{DD}	1.95 3.6	V V
Input high voltage	V_{IH}	Over temperature and supply	0.7*V_{I2C}		V_{I2C}+0.5	V
Input low voltage	V_{IL}	Over temperature and supply	-0.5		0.3*V_{I2C}	V
Output low voltage	V_{OL}	Over temperature and supply	0		0.4	V
Address pin voltage ("1")	$V_{ADDR,HI}$		2	V_{DD}	V_{DD}+0.5	V
Address pin voltage ("0")	$V_{ADDR,LO}$		0		0.5	V
ADDR leakage	$I_{ADDR,\ leak}$				1	µA
SCL leakage	$I_{SCL,\ leak}$	V_{SCL}=3.6V, Ta=+85°C			1	µA
SDA leakage	$I_{SDA,\ leak}$	V_{SDA}=3.6V, Ta=+85°C			1	µA
SCL capacitance	C_{SCL}				10	pF
SDA capacitance	C_{SDA}				10	pF
Slave address	SA	Factory default, ADDR-pin grounded		3A		hex

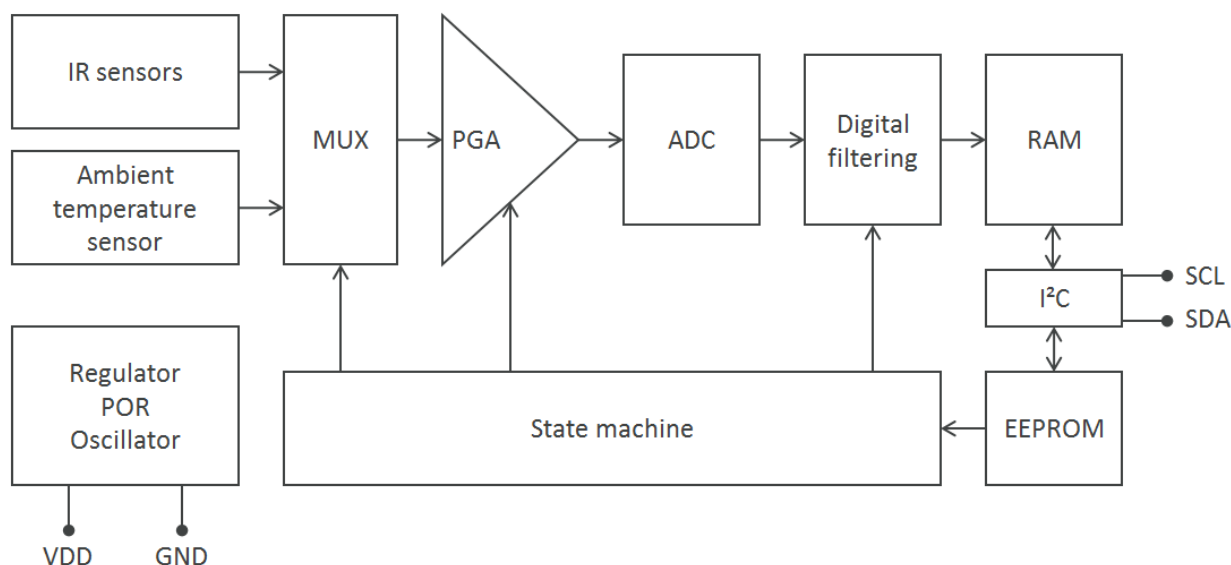*Table 6: Electrical characteristics*

6. Detailed General Description

6.1. Block diagram

6.2. Description

The MLX90632 is a far infrared, non-contact temperature sensor which is factory calibrated to a high accuracy. Internally, electrical and thermal precautions are taken to compensate for thermally harsh external conditions. The thermopile sensing element voltage signal is amplified and digitized. After digital filtering, the raw measurement result is stored in the RAM memory. Furthermore, the MLX90632 contains a sensor element to measure the temperature of the sensor itself. The raw information of this sensor is also stored in RAM after processing. All above functions are controlled by a state machine. The result of each measurement conversion is accessible via I^2C.

The communication to the chip is done by I^2C up to fast mode plus (FM+). The requirement of the standard is to run at frequencies up to 1MHz. Through I^2C the external unit can have access to the following blocks:

- The control registers of the internal state machines

- RAM (**96cells** x 16bit) for pixel and auxiliary measurement data, in this document mainly referred to as 'storage memory'.

- EEPROM (**256cells** x 16bit) used to store the trimming values, calibration constants and various device/measurement settings.

From the measurement data and the calibration data the external unit can calculate both the sensor temperature and the object temperature. The calculation allows the customer to adjust the calibration for his own application in case an optical window or obstructions are present.

7. Memory map

Some bits in the registers below are Melexis reserved. Those bits need to be read and masked, prior to writing operation.

Access	Address	Name	Description
		EEPROM	
Read-only	0x2400~0x2404	Melexis reserved	
Read-only	0x2405	ID0[15:0]	Chip ID
Read-only	0x2406	ID1[15:0]	Chip ID
Read-only	0x2407	ID2[15:0]	Chip ID
Read-only	0x2408	ID_CRC16	CRC
Read-only	0x2409	EE_PRODUCT_CODE	Sensor information
Read-only	-	Melexis reserved	
Read-only	0x240B	EE_VERSION	EEPROM version
Read-only	0x240C	EE_P_R [15:0]	P_R calibration constant (16-bit, Least Significant Word)
Read-only	0x240D	EE_P_R [31:16]	P_R calibration constant (16-bit, Most Significant Word)
Read-only	0x240E	EE_P_G [15:0]	P_G calibration constant (16-bit, Least Significant Word)
Read-only	0x240F	EE_P_G [31:16]	P_G calibration constant (16-bit, Most Significant Word)
Read-only	0x2410	EE_P_T [15:0]	P_T calibration constant (16-bit, Least Significant Word)
Read-only	0x2411	EE_P_T [31:16]	P_T calibration constant (16-bit, Most Significant Word)
Read-only	0x2412	EE_P_O [15:0]	P_O calibration constant (16-bit, Least Significant Word)
Read-only	0x2413	EE_P_O [31:16]	P_O calibration constant (16-bit, Most Significant Word)
Read-only	0x2414	EE_Aa [15:0]	Aa calibration constant (16-bit, Least Significant Word)
Read-only	0x2415	EE_Aa [31:16]	Aa calibration constant (16-bit, Most Significant Word)
Read-only	0x2416	EE_Ab [15:0]	Ab calibration constant (16-bit, Least Significant Word)
Read-only	0x2417	EE_Ab [31:16]	Ab calibration constant (16-bit, Most Significant Word)
Read-only	0x2418	EE_Ba [15:0]	Ba calibration constant (16-bit, Least Significant Word)
Read-only	0x2419	EE_Ba [31:16]	Ba calibration constant (16-bit, Most Significant Word)
Read-only	0x241A	EE_Bb [15:0]	Bb calibration constant (16-bit, Least Significant Word)
Read-only	0x241B	EE_Bb [31:16]	Bb calibration constant (16-bit, Most Significant Word)
Read-only	0x241C	EE_Ca [15:0]	Ca calibration constant (16-bit, Least Significant Word)
Read-only	0x241D	EE_Ca [31:16]	Ca calibration constant (16-bit, Most Significant Word)
Read-only	0x241E	EE_Cb [15:0]	Cb calibration constant (16-bit, Least Significant Word)
Read-only	0x241F	EE_Cb [31:16]	Cb calibration constant (16-bit, Most Significant Word)
Read-only	0x2420	EE_Da [15:0]	Da calibration constant (16-bit, Least Significant Word)
Read-only	0x2421	EE_Da [31:16]	Da calibration constant (16-bit, Most Significant Word)

Read-only	0x2422	EE_Db [15:0]	Db calibration constant (16-bit, Least Significant Word)
Read-only	0x2423	EE_Db [31:16]	Db calibration constant (16-bit, Most Significant Word)
Read-only	0x2424	EE_Ea [15:0]	Ea calibration constant (16-bit, Least Significant Word)
Read-only	0x2425	EE_Ea [31:16]	Ea calibration constant (16-bit, Most Significant Word)
Read-only	0x2426	EE_Eb [15:0]	Eb calibration constant (16-bit, Least Significant Word)
Read-only	0x2427	EE_Eb [31:16]	Eb calibration constant (16-bit, Most Significant Word)
Read-only	0x2428	EE_Fa [15:0]	Fa calibration constant (16-bit, Least Significant Word)
Read-only	0x2429	EE_Fa [31:16]	Fa calibration constant (16-bit, Most Significant Word)
Read-only	0x242A	EE_Fb [15:0]	Fb calibration constant (16-bit, Least Significant Word)
Read-only	0x242B	EE_Fb [31:16]	Fb calibration constant (16-bit, Most Significant Word)
Read-only	0x242C	EE_Ga [15:0]	Ga calibration constant (16-bit, Least Significant Word)
Read-only	0x242D	EE_Ga [31:16]	Ga calibration constant (16-bit, Most Significant Word)
Read-only	0x242E	EE_Gb [15:0]	Gb calibration constant (16-bit)
Read-only	0x242F	EE_Ka [15:0]	Ka calibration constant (16-bit)
Read-only	0x2430	EE_Kb [15:0]	Kb calibration constant (16-bit)
Read-only	-	Melexis reserved	
R/W	0x2480	Melexis reserved	
R/W	0x2481	EE_Ha [15:0]	Ha Customer calibration constant (16 bit)
R/W	0x2482	EE_Hb [15:0]	Hb Customer calibration constant (16 bit)
R/W	-	Melexis reserved	
R/W	0x24C0…0x24CF	Customer data	Customer data storage area
R/W	-	Melexis reserved	
R/W	0x24D4	EE_CONTROL	EEPROM Control register, measurement control
R/W	0x24D5	EE_I2C_ADDRESS	I^2C slave address >> 1 Example: standard address (= 0x003A) >> 1 = 0x001D
R/W	-	Melexis reserved	
R/W	0x24E1	EE_MEAS_1	Measurement settings 1 (see section Measurement settings)
R/W	0x24E2	EE_MEAS_2	Measurement settings 2 (see section Measurement settings)
R/W	-	Melexis reserved	

			REGISTER
R/W	0x3000	REG_I2C_ADDRESS	I^2C slave address >> 1
R/W	0x3001	REG_CONTROL	Control register, measurement mode
-	-	Melexis reserved	
R/W	0x3FFF	REG_STATUS	Status register: data available
			RAM
Read-only	0x4000	RAM_1	Raw data 1
Read-only	0x4001	RAM_2	Raw data 2
Read-only	0x4002	RAM_3	Raw data 3
Read-only	0x4003	RAM_4	Raw data 4
Read-only	0x4004	RAM_5	Raw data 5
Read-only	0x4005	RAM_6	Raw data 6
Read-only	0x4006	RAM_7	Raw data 7
Read-only	0x4007	RAM_8	Raw data 8
Read-only	0x4008	RAM_9	Raw data 9
…	…	…	…
Read-only	0x4033	RAM_52	Raw data 52
Read-only	0x4034	RAM_53	Raw data 53
Read-only	0x4035	RAM_54	Raw data 54
Read-only	0x4036	RAM_55	Raw data 55
Read-only	0x4037	RAM_56	Raw data 56
Read-only	0x4038	RAM_57	Raw data 57
Read-only	0x4039	RAM_58	Raw data 58
Read-only	0x403A	RAM_59	Raw data 59
Read-only	0x403B	RAM_60	Raw data 60

Table 7: Memory table

Important! The width of the EEPROM is 16 bit.
Some calibration parameters are 32 bit and split up into two 16 bit numbers in EEPROM.
The least significant 16 bits of the parameter starts on the address shown in the Memory table.
Example: To retrieve value EE_Aa (32bit) = EE_Aa_MS (at 0x2415) << 16 | EE_Aa_LS (at 0x2414)
(Section Example Temperature Calculation)

Important! The EEPROM needs to be unlocked before each write command, e.g. all addresses with R/W access need the customer key for modification.
(Section EEPROM unlock for customer access)

7.1. Product ID

A unique 48-bit product ID is stored in the EEPROM.

Addresses 0x2405 (ID0), 0x2406 (ID1) and 0x2407 (ID2) should be readout to know the ID of the product.

ProductID[47:0] = ID2[15:0] << 32 | ID1[15:0] << 16 | ID0[15:0]

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
								ProductID[15:0]							

Figure 4: ID0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
								ProductID[31:16]							

Figure 5: ID1
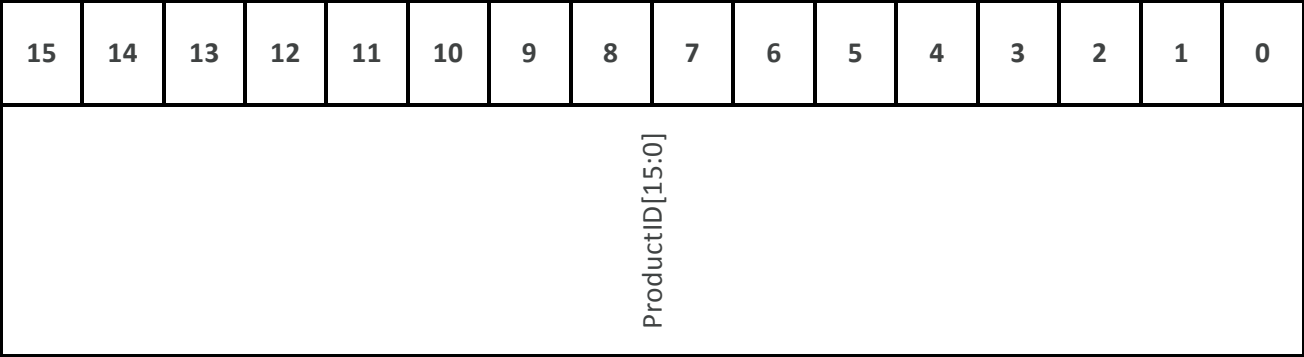
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
								ProductID[47:32]							
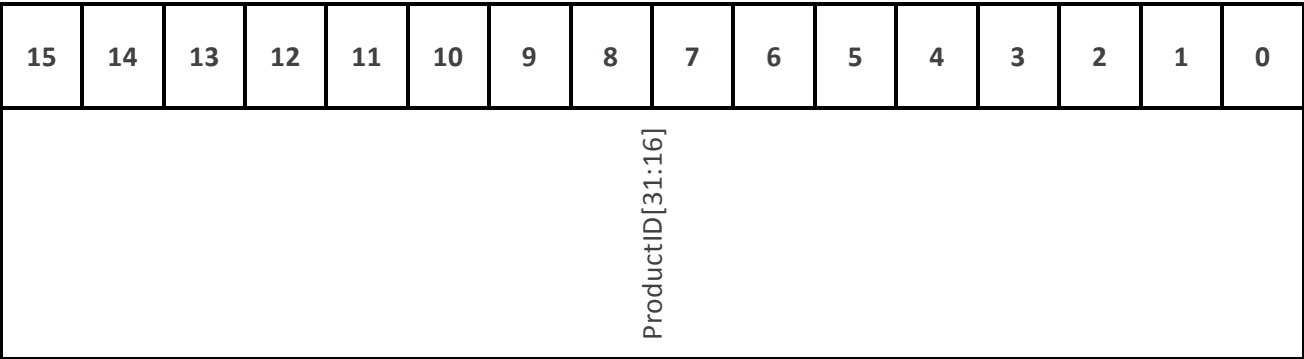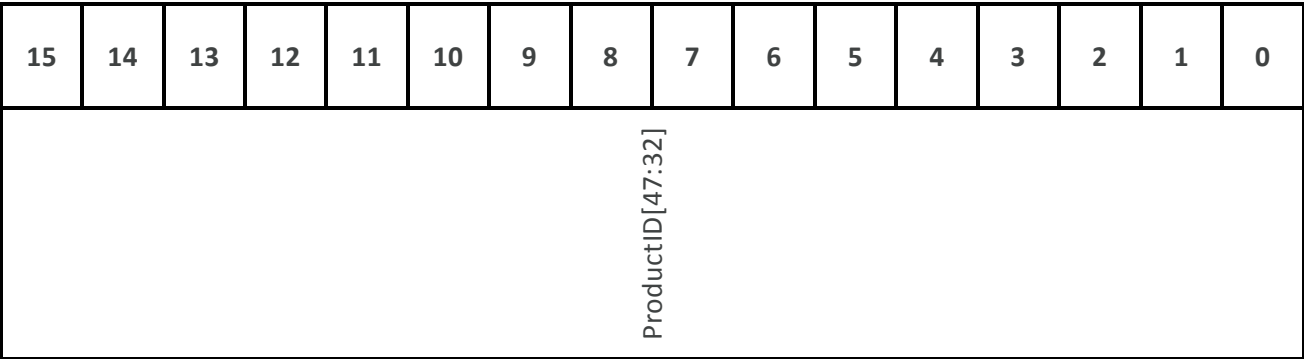
Figure 6: ID2

7.2. Product Code (0x2409)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Melexis reserved	Melexis reserved	Melexis reserved	Melexis reserved	Melexis reserved	Melexis reserved	FOV		Package			Accuracy range				

Figure 7 EE_PRODUCT_CODE

- FOV
  - ❑ 0: 50°
- Package
  - ❑ 0: -
  - ❑ 1: SFN 3x3
- Accuracy range
  - ❑ 0: -
  - ❑ 1: Medical
  - ❑ 2: Standard

7.3. Customer Data storage Area (0x24C0 to 0x24CF)

The EEPROM area dedicated for customer data storage consists of 16 EEPROM cell of 16-bit words in the address range from 0x24C0 to 0x24CF.

The customer data area in the MLX90632 EEPROM is not meant for storing intermittent data during production, calibration or normal application use. The purpose of having such an area is to enable the customer to store additional MLX90632 related calibration or tracking information. It is important that only the final data is written to the MLX90632 EEPROM in one go. In order to verify that the writing process was successful, the whole EEPROM data should be read-out prior to writing and compared to the EEPROM content after writing - only the customer data values should be different.

Important note: The maximum number of EEPROM re-writes is 10. Therefore, it is highly recommended that the writing of the data is done 1 time only.

8. Control and configuration

MLX90632 does multiple measurements in one table. Burst (SOB) enables automatic triggering of all measurement, while Single(SOC) allows one-by-one measurement triggering, thus more precise control. For more details refer to Application Note "AN MLX90632 measuring modes" available on Melexis Web.

Several bits in the EEPROM or register are available to control and configure the measurements:

8.1. Measurement control

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Melexis reserved	Melexis reserved	Melexis reserved	Melexis reserved	sob	Melexis reserved	Melexis reserved	meas_select					soc	mode		Melexis reserved

Figure 8: Register Measurement control settings

REG_CONTROL controls the measurement handling and data storage.
Changes will take immediate effect.

Bits	Parameter	Description	See section
11	sob	starts a full measurement table when being in (sleeping) step mode	Operating Modes
8:4	meas_select	select the type of measurement to be performed	Operating Modes
3	soc	starts a single measurement when being in (sleeping) step mode	Operating Modes
2:1	mode[1:0]	defines the operating mode	Operating Modes

Table 8: Register REG_CONTROL explained

Note that this register is initialized during POR by the EEPROM word EE_CONTROL.

Several measurement modes exist. These modes are controlled by bits mode[1:0] in register REG_CONTROL. In continuous mode the measurements are constantly running while in step mode the state machine will execute only one measurement which is initiated by soc bit. After finishing the measurement it will go in wait state until the next measurement is initiated by soc. The measurements are following the measurement sequence as defined in the measurement table.

The different possible operating modes are:

- mode[1:0] = 00: Enables the HALT mode. This is a special mode for EEPROM operations. Halt mode is aborting the current measurement. The device remains in active state all the time in this mode. Measurements are not executed.

- mode[1:0] = 01: Enables the sleeping step mode. In this mode the device will be in sleep. On request

(soc bit), the device will power-on, the state machine will do one measurement, will go into sleep and will wait for next command.

- mode[1:0] = 10: Enables the <u>step mode</u>. In this mode the state machine will do one measurement upon request (soc bit) and will wait for next command. The device remains in active state all the time in this mode.

- mode[1:0] = 11: Device is in <u>continuous mode</u>. Measurements are executed continuously. The device remains in active state all the time in this mode.

By default, the device is in continuous mode.

Switching between the step modes and continuous mode has immediate effect after the current measurement has finished (not waiting till end of measurement table was reached).

8.2. Measurement type select

For the medical version, there are two possible measurements to select from:

- meas_select[4:0] = 0x00: Enables the medical measurement. In order to calculate the correct temperatures, the appropriate raw data values and formulas should be used. Refer to the <u>medical measurement temperature calculations</u>

- meas_select[4:0] = 0x11: Enables the extended range measurement. In order to calculate the correct temperatures, the appropriate raw data values and formulas should be used. Refer to the <u>extended range measurement temperature calculations</u>

Note: If other values are being used for meas_select, the resulting calculated temperatures will be invalid.

In order to switch to the desired measurement type the following routine should be performed:

1. Send an addressed reset to the MLX90632 device

2. Read the register (REG_CONTROL) value

3. Modify the REG_CONTROL value to:
   a. mode[1:0] = 00
   b. meas_select = 0x00 for medical or 0x11 for extended range

4. Read the register (REG_CONTROL) value

5. Modify the REG_CONTROL value mode[1:0] to the desired mode

The next measurement will be of the type that was programmed.

By default, for the medical version of MLX90632 , the medical measurement is enabled.

8.3. Device status

Figure 9: Register Device status settings

REG_STATUS allows checking in which state the device is and indicates when measurements are finished. Changes will take immediate effect.

Bits	Parameter	Description	See section
10	device_busy	Read-only Flag indicating that a measurement is being executed (1 = measurement ongoing) In sleep mode, this flag is always low. In continuous mode, this flag is always high. In soc-step mode, this flag is high during one measurement. In sob-step mode, this flag is high till all measurements are finished.	Operating Modes
9	eeprom_busy	Read-only Flag indicating that the eeprom is busy (0: not busy) Eeprom being busy is defined as follows: - at start-up, the eeprom is busy and remains busy till the initialization phase (eeprom copy) has finished - during eeprom write/erase, the eeprom is busy	
8	brown_out	Bit is set to 0 Customer should set this bit to 1 at startup When the device would reset, the bit is set to 0 and a reset can be detected	
6:2	cycle_position	Read-only Indicates from which measurement (in the measurement table) the last written data is coming: - cycle_position[4:0]=x, corresponds to measurement x, x=0->31	Temperature calculation
0	new_data	Customer should set bit to 0 When a measurement is done and all data is available, the bit is set to 1 Customer can readout the data and reset the bit to 0	Operating Modes

Table 9: Register REG_STATUS explained

8.4. Measurement settings

8.4.1. Refresh rate

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Melexis reserved	Melexis reserved	Melexis reserved	Melexis reserved	Melexis reserved	Refresh rate			Melexis reserved	Melexis reserved	Melexis reserved	Melexis reserved	Melexis reserved	Melexis reserved	Melexis reserved	Melexis reserved

Figure 10: EEPROM Measurement settings

The refresh rate is the speed that the RAM will be updated with results and is configurable in "Measurement settings 1" and "Measurement settings 2".
The refresh rate can be set with 3 bits and is located in EEPROM addresses 0x24E1 and 0x24E2.
Changing the refresh rate will take immediate effect.
It is important to know that <u>the refresh rate must be kept the same for both measurements</u>.
The table below shows the available refresh rates and the corresponding result to be written in EEPROM addresses EE_MEAS_1 and EE_MEAS_2.

EE_MEAS_1[10:8] EE_MEAS_2[10:8]	Refresh rate [Hz]	Time [ms]	Burst meas time [ms]	EE_MEAS_1 (0x24E1)	EE_MEAS_2 (0x24E2)
0	0.5	2000	4000	0x800D	0x801D
1	1	1000	2000	0x810D	0x811D
2	**2**	**500**	**1000**	**0x820D**	**0x821D**
3	4	250	500	0x830D	0x831D
4	8	125	250	0x840D	0x841D
5	16	62.5	125	0x850D	0x851D
6	32	31.25	62.5	0x860D	0x861D
7	64	15.625	31.25	0x870D	0x871D

Table 10: EEPROM Refresh rate For Medical and Standard explained

EE_MEAS_17[10:8] EE_MEAS_18[10:8] EE_MEAS_19[10:8]	Refresh rate [Hz]	Time [ms]	Burst meas time [ms]	EE_MEAS_17 (0x24F1)	EE_MEAS_18 (0x24F2)	EE_MEAS_19 (0x24F3)
0	0.16	6000	6000	0x8000	0x8012	0x800C
1	0.32	3000	3000	0x8100	0x8112	0x810C
2	0.65	1500	1500	0x8200	0x8212	0x820C
3	**1.3**	**750**	**750**	**0x8300**	**0x8312**	**0x830C**
4	2.6	375	375	0x8400	0x8412	0x840C
5	5	200	200	0x8500	0x8512	0x850C
6	10	100	100	0x8600	0x8612	0x860C
7	20	50	50	0x8700	0x8712	0x870C

Table 11: Extended range EEPROM Refresh rate explained

9. I^{2}C commands

This device is based on I^{2}C specification Rev.5 – October 9th 2012. I^{2}C FM+ mode is supported.

The sensor has the following I^{2}C features:

- Slave mode only
- 7-bits addressing
- Modes: Standard-mode, Fast-mode, Fast-mode Plus
- Incremental addressing – allowing a block of addresses to be accessed inside one I^{2}C sequence

The following I^{2}C commands are implemented:

- Read/write access to internal memories and registers
    - ❏ Addressed write
    - ❏ Addressed read
- Global reset
- Addressed reset
- EEPROM unlock for CUST access
- Direct read

9.1. I²C address

By default, the device responds to the 7-bit slave address 0x3A.
Configuration of the 7-bit slave address is possible at EEPROM address 0x24D5.

The least significant bit (bit0) of the address is determined by the status of the ADDR-pin (either connected to ground or supply) and is taken in after power-up or a reset command if a change is made in EEPROM.
- Bit0 = '0' if ADDR-pin is connected to GND
- Bit0 = '1' if ADDR-pin is connected to VDD
The remaining 6-bits can be used to configure the I2C address of the device.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Melexis reserved	Melexis reserved	Melexis reserved	Melexis reserved	Melexis reserved	Melexis reserved	Melexis reserved	Melexis reserved	Melexis reserved	Configurable 6-bit slave address (0x1D)						External status of ADDR-pin
									I2C slave address(7-bit) = 0x3A if ADDR-pin=0(GND) =0x3B if ADDR-pin=1(VDD)						

Figure 11: EEPROM I2C address configuration

Important!	**The device will not respond if the I²C address is changed to 0 (and ADDR pin is low).** The only way to get the device to respond in this condition is to pull the ADDR pin high. The slave address will be changed to 1 and communication is possible.

Important!	The device shall not execute measurements when performing EEPROM memory operations (I²C read/erase/write instructions in EEPROM address range)! Hence, the device must be put in HALT or Step mode before doing EEPROM write/erase operations. EEPROM read can be done in HALT, STEP or Sleeping STEP mode.

9.1.1. Slave Address change process flow

⇨ Put device in Halt mode, use current slave address

- o Read Control Register

- o Mask Control register value – change only the power mode bits

- o Write the new value to Control register

⇨ Write Slave address to EERPOM, use current slave address

- o Send EEPROM unlock key

         o Write 0x0000 to address 0x24D5

Important! Do <u>not</u> power down the device at this point

- o Wait at least 5ms

- o Send EERPOM unlock key

- o Write New slave address [6:1] to address 0x24D5

- o Wait at least 5ms

- o It is advised to read back and verify the written address value at this point

  It is now safe to power-down the device

- ⇨ After power-down of the device the new slave address will be in operation.

9.1.2. Slave Address change example

```
#include <mlx90632.h>

#define OLD_SLAVE_ADDRESS 0x3A
#define NEW_SLAVE_ADDRESS 0x42
#define ADDR_PIN 0
//i2c read/write functions – i2ca parameter is the Slave address
int32_t mlx90632_i2c_read_SA(uint8_t i2ca, int16_t register_address, uint16_t *value)
int32_t mlx90632_i2c_write_SA(uint8_t i2ca, int16_t register_address, uint16_t value)

int32_t mlx90632_i2c_write_EE(uint8_t i2ca, int16_t register_address, uint16_t value)
{
 //Send EEPROM unlock key
 //Write 0x0000
 //Wait 5ms
 //Send EEPROM unlock key
 //write value
 //wait 5ms
}
int main()
{

 uint16_t rData;
//Put the device in Halt mode
mlx90632_i2c_read_SA(OLD_SLAVE_ADDRESS, MLX90632_REG_CTRL, &rData);
  rData = rData & (~MLX90632_CFG_PWR_MASK);
  rData |= (MLX90632_PWR_STATUS_HALT);

  mlx90632_i2c_write_SA(OLD_SLAVE_ADDRESS,MLX90632_REG_CTRL, rData);
//write new Slave address to EEPROM
  mlx90632_i2c_write_EE(OLD_SLAVE_ADDRESS, MLX90632_EE_I2C_ADDRESS, NEW_SLAVE_ADDRESS>>1);
//it is now safe to power down the device in order load the new Slave address
//After POR the new slave address should be used for i2c communication
 mlx90632_i2c_read_SA(NEW_SLAVE_ADDRESS, MLX90632_REG_CTRL, &rData);
 }
```

9.2. Addressed read

The addressed read command allows an incremental read-out, starting from any given address within the memory space.


Figure 12: Addressed read

Important!   An addressed read is only valid when combining directly communicating the start address and a direct read through a repeated START condition. In case the read and write part are separated by a STOP condition, or in case the read is not directly following the write, or in case the slave address is not identical for both, the command will not be seen as an addressed read. As a result, the second read will in practice act as a direct read.
As soon as incremental addressing leaves the address space, the slave will respond with all 8'hFF.

Figure 13 Addressed read - Oscilloscope Plot

9.3. Addressed write

The addressed write command allows doing an incremental write, starting from any given address within the memory space.
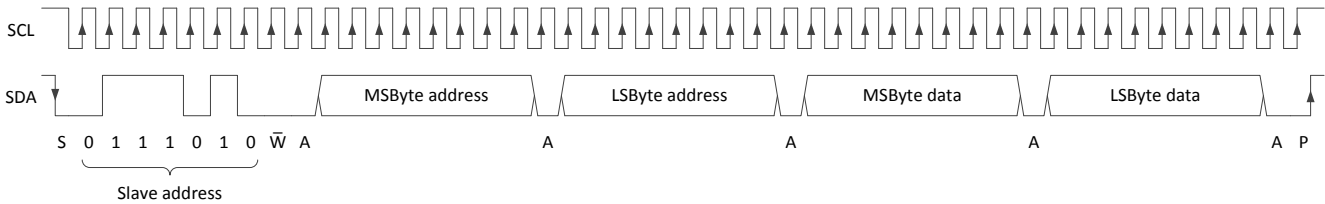
Figure 14: Addressed write

Important! The slave is sending ACK/NACK based on the fact whether it was able to write data (timing, end of register space, access rights).
The slave will automatically increment the address of the write byte, independent if it gave an ACK or a NACK to the master. It is up to the master to re-write the byte afterwards.

The device shall not execute measurements when performing EEPROM memory operations (I^2C read/erase/write instructions in EEPROM address range)! Hence, the device must be put in HALT or Step mode before doing EEPROM write/erase operations. EEPROM read can be done in HALT, STEP or Sleeping STEP mode.

Before writing to EEPROM it is necessary to erase the specific address location in EEPROM. This is done by first writing 0x0000. Then the new data can be written.

When the device is busy with the write operation to EEPROM, new write commands will be ignored. A read operation will return invalid data. The fact that the device is busy is indicated via the bit device_busy in REG_STATUS.

9.4. Global reset

This command resets all devices on the I^2C bus (based on the general call address 0x00).
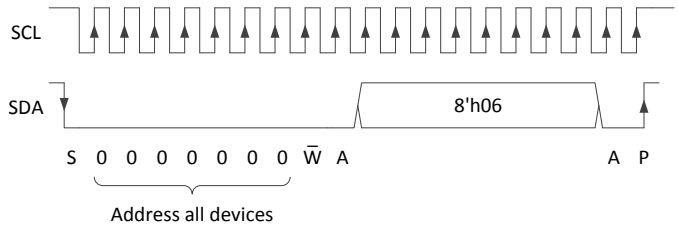
Figure 15: Global reset

Note: After this command, a delay at least 150us is needed before the next communication with the device.

9.5. Addressed reset

This command resets the addressed device only (based on the I^{2}C address).
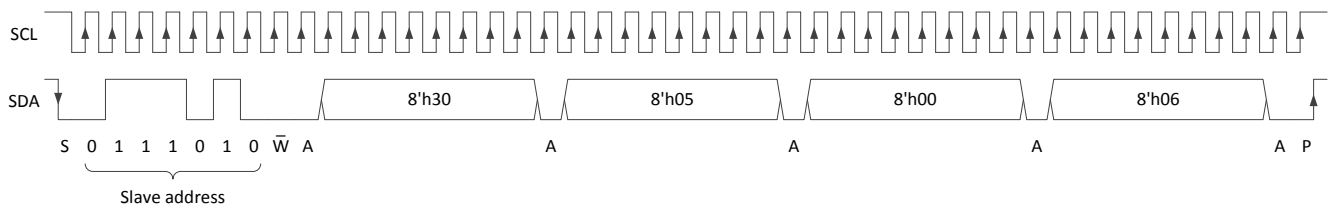

Figure 16: Addressed reset

Note: After this command, a delay at least 150us is needed before the next communication with the device.

9.6. EEPROM unlock for customer access

This command unlocks the EEPROM allowing only one write operation to an EEPROM word in the customer part of the EEPROM.
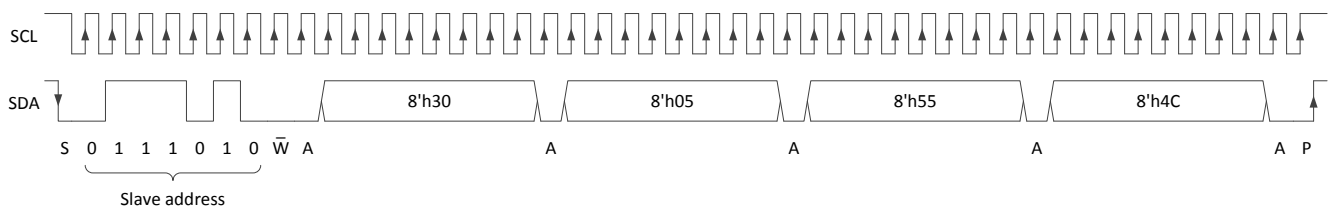After the EEPROM write, the EEPROM access goes back to the "NoKey" access mode.

Figure 17: EEPROM unlock

9.7. Direct read

The direct read command allows an incremental read out at a default start address.
This default start address is fixed to the register location REG_STATUS (0x3FFF).
According to the I^{2}C specification, the master will keep sending an acknowledge (A) until it want to stop. This is indicated by sending a NAK. As a result, the slave will stop driving the SDA-bus as soon as a NAK is received by the master.
As soon as the incremental addressing leaves the address space, the slave will respond with all 8'hFF.
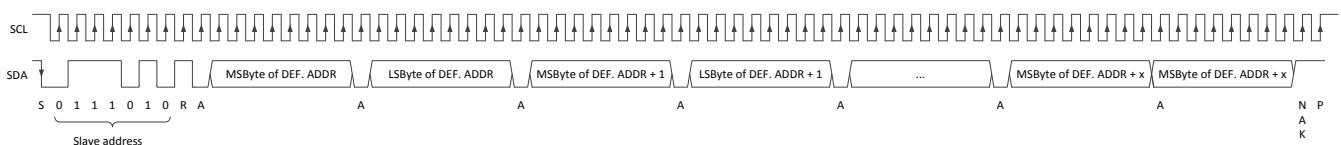
Figure 18: Direct read

9.8. 1.8V I2C communication device

When using such device, it is recommended to start with a Stop condition or a dummy command before any other command after powering up the device.

10. Operating Modes

The device has 2 states of operation: **sleep state** and **active state**.

▪ **Sleep state**

In this state, most of the circuitry is disabled to limit the current consumption to a few uA.

▪ **Active state**

In this state, the sensor is active. The current consumption is I_{DD} (about 1mA).

10.1. Measurement modes

Several measurement modes exist. These modes are controlled by bits mode[1:0] in register REG_CONTROL[2:1]. In continuous mode the measurements are constantly running while in step mode the state machine will execute only one measurement which is initiated by the soc bit or a full measurement set initiated by the sob bit. After finishing the measurement(s) the sensor will go in a wait state until the next measurement is initiated by the soc or sob bit. If soc is used to initiate a measurement, the measurements are following the measurement sequence as defined in the measurement table.

▪ mode[1:0] = 00: Enables the <u>HALT mode</u>. This is a special mode for EEPROM operations. Halt mode is aborting the current measurement. The device remains in active state all the time in this mode. Measurements are not executed.

The different possible measurement modes are:

▪ mode[1:0] = 01: Enables the <u>sleeping step mode</u>.

The device will be in sleep mode. On request (soc or sob bit), the device will power-on, the state machine will perform one measurement(soc) or the full measurement table (sob), will go into sleep and wait for the next command.

In the sleeping step mode all the measurements from the measurement table will be performed so that all data necessary for the calculations is refreshed. The two ways of using the device in this mode are:

• SOB bit

The SOB bit initiates a full measurement table measurement. Once the measurement is started, the SOB bit is cleared and the *device_busy* bit is set internally in the MLX90632. When all the measurements from the measurement table are performed, the *device_busy* bit is cleared indicating the end of measurements – the new data can be read. The flow should be:

1. Set SOB bit

2. Wait for all the measurements from the measurement table to finish - depending on the refresh rates (see Table 10 and Table 11)

3. Make sure that the device_busy bit is cleared

4. Read out the data

5. Calculate the temperatures

• SOC bit

The SOC bit initiates a single measurement from the measurement table. The measurements are being performed consecutively as set in the measurement table. Once the measurement is started, the SOC bit is cleared internally in the MLX90632 and could be set again so that the next measurement from the measurement table is started right after the current one is done. When the current measurement is done, the *new_data* bit is set – the new data can be read and the bit should be cleared. The flow should be:

1. Set SOC bit

2. Wait for the 1st measurement from the measurement table to finish - depending on the refresh rate (see Table 10 and Table 11)

3. Make sure that the new_data bit is set and clear it

4. Set SOC bit

5. Wait for the 2nd measurement from the measurement table to finish - depending on the refresh rate (see Table 10 and Table 11)

6. Make sure that the new_data bit is set and clear it

7. If medical mode is selected proceed with step 11 (e.g. skip 8..10)

8. Set SOC bit

9. Wait for the 3rd measurement from the measurement table to finish - depending on the refresh rate (see Table 10 and Table 11)

10. Make sure that the new_data bit is set and clear it

11. Read out the data

12. Calculate the temperatures

- mode[1:0] = 10: Enables the step mode.

   The state machine will do one measurement upon request (soc bit) and will wait for the next command. The device remains powered all the time in this mode.

- mode[1:0] = 11: Device is in continuous mode.

   Measurements are executed continuously. The device remains powered all time in this mode.

By default, the device is in continuous mode.

Switching between the step modes and continuous mode has immediate effect after the current measurement has finished (not waiting until the end of the measurement table was reached, i.e. it is not guaranteed the whole measurement table is executed).

10.2. Measurement types

There are two possible measurement types to select from:

- meas_select[4:0] = 0x00: Enables the medical measurement. In order to calculate the correct temperatures, the appropriate raw data values and formulas should be used. Refer to the medical measurement temperature calculations

- meas_select[4:0] = 0x11: Enables the extended range measurement. In order to calculate the correct

temperatures, the appropriate raw data values and formulas should be used. Refer to the extended range measurement temperature calculations

Note: If other values are being used for meas_select, the resulting calculated temperatures will be invalid.

11. Temperature calculation

11.1. Medical measurement

To calculate the ambient and object temperature, a set of 2 measurements is required:

- Measurement 1: RAM_4, RAM_5, RAM_6;

- Measurement 2: RAM_7, RAM_8, RAM_9;

One should notice this requires double the measurement time than specified (= 2 * 500ms) for the very first calculation.
After the first calculation, TA and TO can be calculated with the next measurement.

Example:

t0:	Measurement 1 (cycle_pos = 1)	=>	no calculation of TA or TO possible because not all parameters are known		
t1:	Measurement 2 (cycle_pos = 2)	=>	calculate TA (RAM_6, RAM_9) calculate TO (RAM_7, RAM_8, RAM_6, RAM_9)	=>	1s.
t2:	Measurement 3 (= 1) (cycle_pos = 1)	=>	calculate TA (RAM_6, RAM_9) calculate TO (RAM_4, RAM_5, RAM_6, RAM_9)	=>	0.5s.
t3:	Measurement 4 (= 2) (cycle_pos = 2)	=>	calculate TA (RAM_6, RAM_9) calculate TO (RAM_7, RAM_8, RAM_6, RAM_9)	=>	0.5s.
t4:	…				

To calculate the new ambient and object temperature RAM_6 and RAM_9 have to be used.
The choice between [RAM_4 and RAM_5] or [RAM_7 and RAM_8] depends on the current measurement.
REG_STATUS[6:2] (= "cycle_pos") returns the current position of the measurement defined in the measurement table.
Using the current data and the data from measurement (x-1), TA and TO can be calculated every 500ms.
The complete measurement sequence can be automated by using the new_data bit in combination with cycle_pos bits.
The sequence should look like the following:

- Write new_data = 0

- Check when new_data = 1

- Read cycle_pos to get measurement pointer

    - If cycle_pos = 1

        - Calculate TA and TO base on RAM_4, RAM_5, RAM_6, RAM_9

    - If cycle_pos = 2

- Calculate TA and TO base on RAM_7, RAM_8, RAM_6, RAM_9
- Return to top

11.1.1. Pre-calculations

11.1.1.1. Ambient

$$VR_{TA} = RAM\_9 + Gb * \frac{RAM\_6}{12}$$

$$Gb = EE\_Gb * 2^{-10}$$

$$AMB = \left[\frac{RAM\_6}{12}\right]/VR_{TA} * 2^{19}$$

The parameter EE_Gb is a signed 16-bit number.

11.1.1.2. Object

$$S = \frac{RAM\_4 + RAM\_5}{2}$$

OR

$$S = \frac{RAM\_7 + RAM\_8}{2}$$

$$VR_{TO} = RAM\_9 + Ka * \frac{RAM\_6}{12}$$

$$Ka = EE\_Ka * 2^{-10}$$

$$S_{TO} = \left[\frac{S}{12}\right]/VR_{TO} * 2^{19}$$

The parameter EE_Ka is a signed 16-bit number.

11.1.2. Ambient temperature

$$\mathbf{Ta\ (sensor\ temperature\ in\ °C) = P\_O + \frac{AMB - P\_R}{P\_G} + P\_T * (AMB - P\_R)^2}$$

With:
Ta in degrees Celsius
P_R   = EE_P_R * 2^{-8}
P_O   = EE_P_O * 2^{-8}
P_G   = EE_P_G * 2^{-20}

$$P\_T = EE\_P\_T * 2^{-44}$$

The parameters EE_P_R, EE_P_O, EE_P_G and EE_P_T are signed 32-bit numbers.

11.1.3. Object temperature

TO (object temperature in °C)

$$= \sqrt[4]{\frac{S_{TO}}{\varepsilon * Fa * Ha * (1 + Ga * (TO_{DUT} - TO_0) + Fb * (TA_{DUT} - TA_0))} + Ta_{[K]}^4} - 273.15 - Hb$$

With:

$Fa = EE\_Fa * 2^{-46}$

$Fb = EE\_Fb * 2^{-36}$

$Ga = EE\_Ga * 2^{-36}$

$Ha = EE\_Ha * 2^{-14}$

$Hb = EE\_Hb * 2^{-10}$

$TO_0 = 25°C$

$TA_0 = 25°C$

$$TA_{DUT} = \frac{(AMB - Eb)}{Ea} + 25$$

$Ea = EE\_Ea * 2^{-16}$

$Eb = EE\_Eb * 2^{-8}$

$Ta_{[K]} = TA_{DUT} + 273.15$ in Kelvin

$TO_{DUT} = $ Object temperature in 25°C

$\varepsilon = 1 = $ Object Emissivity parameter (not stored in EEPROM, but part of the 'app')

The parameters EE_Ea, EE_Eb, EE_Fa, EE_Fb, EE_Ga are signed 32-bit numbers.
The parameters EE_Gb, EE_Ka, EE_Ha and EE_Hb are signed 16-bit numbers.

Note:
One can see that to compute "To (object temperature)", "To" already needs to be known.
"To (object temperature)" is computed in an iterative manner. In the first iteration "To" is assumed to be 25°C.
In the 2nd iteration the result of first iteration is used, and in the 3rd iteration the end result is obtained.
(See example on next page).

11.1.4. Example Medical measurement Temperature Calculation

Assumed are the following calibration parameters read from EEPROM:

ADDR	PARAM	DATA (hex)	hex to dec	Conversion to use in formula
0x240C	EE_P_R [15:0]	0103	$EE\_P\_R = 005D0103_{hex} = 6095107_{dec}$	$P\_R = 6095107 * 2^{-8} = \textbf{23809.01}$
0x240D	EE_P_R [31:16]	005D		
0x240E	EE_P_G [15:0]	FAE5	$EE\_P\_G = 051CFAE5_{hex} = 85785317_{dec}$	$P\_G = 85785317 * 2^{-20} = \textbf{81.81125}$
0x240F	EE_P_G [31:16]	051C		
0x2410	EE_P_T [15:0]	0000	$EE\_P\_T = 00000000_{hex} = 0_{dec}$	$P\_T = 0 * 2^{-44} = \textbf{0}$

0x2411	EE_P_T [31:16]	0000			
0x2412	EE_P_O [15:0]	1900	$EE\_P\_O = 00001900_{hex} = 6400_{dec}$	$P\_O = 6400 * 2^{-8} = \textbf{25}$	
0x2413	EE_P_O [31:16]	0000			
0x2424	EE_Ea [15:0]	CFAE	$EE\_Ea = 0051CFAE_{hex} = 5361582_{dec}$	$Ea = 5361582 * 2^{-16} = \textbf{81.81125}$	
0x2425	EE_Ea [31:16]	0051			
0x2426	EE_Eb [15:0]	0103	$EE\_Eb = 005D0103_{hex} = 6095107_{dec}$	$Eb = 6095107 * 2^{-8} = \textbf{23809.01}$	
0x2427	EE_Eb [31:16]	005D			
0x2428	EE_Fa [15:0]	6351	$EE\_Fa = 03506351_{hex} = 5559995_{dec}$	$Fa = 55599953 * 2^{-46} = \textbf{7.9E-07}$	
0x2429	EE_Fa [31:16]	0350			
0x242A	EE_Fb [15:0]	71F1	$EE\_Fb = FE2571F1_{hex} = -31100431_{dec}$	$Fb = -31100431 * 2^{-36} = \textbf{-0.00045}$	
0x242B	EE_Fb [31:16]	FE25			
0x242C	EE_Ga [15:0]	A7A4	$EE\_Ga = FDFFA7A5_{hex} = -33577052_{dec}$	$Ga = -33577052 * 2^{-36} = \textbf{-0.00049}$	
0x242D	EE_Ga [31:16]	FDFF			
0x242E	EE_Gb [15:0]	2600	$EE\_Gb = 2600_{hex} = 9728_{dec}$	$Gb = 9728 * 2^{-10} = \textbf{9.5}$	
0x242F	EE_Ka [15:0]	2A00	$EE\_Ka = 2A00_{hex} = 10752_{dec}$	$Ka = 10752 * 2^{-10} = \textbf{10.5}$	
0x2481	EE_Ha [15:0]	4000	$EE\_Ha = 4000_{hex} = 16384_{dec}$	$Ha = 16384 * 2^{-14} = \textbf{1}$	
0x2482	EE_Hb [15:0]	0000	$EE\_Hb = 0000_{hex} = 0_{dec}$	$Hb = 0 * 2^{-10} = \textbf{0}$	

Table 12: Example EEPROM calibration parameters

The returned values from the RAM (0x4000 to 0x4008):

ADDR	PARAM	DATA (hex)	DATA (dec)
0x4003	RAM_4	FF9B	-101
0x4004	RAM_5	FF9D	-99
0x4005	RAM_6	57E4	22500
0x4006	RAM_7	FF97	-105
0x4007	RAM_8	FF99	-103
0x4008	RAM_9	59D8	23000

Table 13: Example RAM data

11.1.4.1. Ambient temperature calculation

$$VR_{TA} = RAM\_9 + Gb * \frac{RAM\_6}{12} = 23000 + 9.5 * \frac{22500}{12}$$

$$VR_{TA} = 40812.5$$

$$AMB = \left[\frac{RAM\_6}{12}\right] / VR_{TA} * 2^{19} = \left[\frac{22500}{12}\right] / 40812.5 * 2^{19}$$

$$AMB = 24086.73813$$

$$\textbf{Ta } (\text{sensor temperature in } °C) = P\_O + \frac{AMB - P\_R}{P\_G} + P\_T * (AMB - P\_R)^2$$

$$\textbf{Ta} = 25 + \frac{24086.73813 - 23809.01}{81.81125} + 0 * (24086.73813 - 23809.01)^2$$

$$Ta = 28.395°C$$

$$\textbf{Ta} = \textbf{28.4°C}$$

11.1.4.2. Object temperature calculation

$$S = \frac{RAM\_4 + RAM\_5}{2} = \frac{(-101) + (-99)}{2}$$

$$S = -100$$

OR

$$S = \frac{RAM\_7 + RAM\_8}{2} = \frac{(-105) + (-103)}{2}$$

$$S = -104$$

Assumed is that RAM_4 and RAM_5 are updated lastly by the device (cycle_pos = 1)

$$VR_{TO} = RAM\_9 + Ka * \frac{RAM\_6}{12} = 23000 + 10.5 * \frac{22500}{12}$$

$$VR_{TO} = 42687.5$$

$$S_{TO} = \left[\frac{S}{12}\right] / VR_{TO} * 2^{19} = \left[\frac{-100}{12}\right] / 42687.5 * 2^{19}$$

$$S_{TO} = -102.35$$

$$\begin{aligned} TO_0 &= 25°C \\ TA_0 &= 25°C \end{aligned}$$

$$TA_{DUT} = \frac{(AMB - Eb)}{Ea} + 25 = \frac{(24086.73813 - 23809.01)}{81.81125} + 25 = 28.3947$$

$$Ta_{[K]} = TA_{DUT} + 273.15 = 28.3947 + 273.15 = 301.5447$$

To (object temperature in °C)

$$= \sqrt[4]{\frac{S_{TO}}{\varepsilon * Fa * Ha * \left(1 + Ga * (TO_{DUT} - TO_0) + Fb * (TA_{DUT} - TA_0)\right)} + Ta_{[K]}^4} - 273.15 - Hb$$

The emissivity parameter (Ɛ) is controlled by the user and is assumed in this example equal to 1.
TO_{DUT} = 25 for the first calculation

$$To = \sqrt[4]{\frac{-102.35}{1 * (7.9E - 07) * 1 * \left(1 + (-0.00049) * (25 - 25) + (-0.00045) * (28.3947 - 25)\right)} + (301.5447)^4} - 273.15 - 0$$

$$To = 27.2048027°C$$

The object temperature needs to be calculated 3 times in order the get the end result.
Next object temperature calculation uses previous obtained object temperature.

$$To = \sqrt[4]{\frac{-102.35}{1 * (7.9E - 07) * 1 * \left(1 + (-0.00049) * (\mathbf{27.2048027} - 25) + (-0.00045) * (28.3947 - 25)\right)} + (301.5447)^4} - 273.15 - 0$$

$$To = 27.2035098°C$$

$$To = \sqrt[4]{\frac{-102.35}{1 * (7.9E - 07) * 1 * \left(1 + (-0.00049) * (\mathbf{27.2035098} - 25) + (-0.00045) * (28.3947 - 25)\right)} + (301.5447)^4} - 273.15 - 0$$

$$To = 27.20351053°C$$

$$\mathbf{To = 27.2°C}$$

11.2. Extended range measurement

This measurement type option is implemented in order to give additional range to the medical devices.

When using the extended range measurement the following should be done:

1.   Switch the device to extended range measurement mode

2.   Wait for the whole measurement to finish

3.   Use the following routine to read the data of interest and calculate the temperatures.

All the necessary functions are available at https://github.com/melexis/mlx90632-library

To calculate the ambient and object temperature, a set of 3 measurements is required:

▪   Measurement 1: RAM_52, RAM_53, RAM_54;

▪   Measurement 2: RAM_55, RAM_56, RAM_57;

▪   Measurement 3: RAM_58, RAM_59, RAM_60;

All three measurements should be available for proper temperature calculation.

11.2.1. Pre-calculations

11.2.1.1. Ambient

$$VR_{TA} = RAM\_57 + Gb * \frac{RAM\_54}{12}$$

$$Gb = EE\_Gb * 2^{-10}$$

$$AMB = \left[\frac{RAM\_54}{12}\right]/VR_{TA} * 2^{19}$$

The parameter EE_Gb is a signed 16-bit number.

11.2.1.2. Object

$$S = \frac{RAM\_52 - RAM\_53 - RAM\_55 + RAM\_56}{2} + RAM\_58 + RAM\_59$$

OR

$$VR_{TO} = RAM\_57 + Ka * \frac{RAM\_54}{12}$$

$$Ka = EE\_Ka * 2^{-10}$$

$$S_{TO} = \left[\frac{S}{12}\right]/VR_{TO} * 2^{19}$$

The parameter EE_Ka is a signed 16-bit number.

11.2.2. Ambient temperature

$$\mathbf{Ta\ (sensor\ temperature\ in\ °C)} = \mathbf{P\_O} + \frac{\mathbf{AMB} - \mathbf{P\_R}}{\mathbf{P\_G}} + \mathbf{P\_T} * (\mathbf{AMB} - \mathbf{P\_R})^2$$

With:
Ta in degrees Celsius
P_R = EE_P_R * 2^{-8}
P_O = EE_P_O * 2^{-8}
P_G = EE_P_G * 2^{-20}
P_T = EE_P_T * 2^{-44}

The parameters EE_P_R, EE_P_O, EE_P_G and EE_P_T are signed 32-bit numbers.

11.2.3. Object temperature

$$\mathbf{TO\ (object\ temperature\ in\ °C)}$$
$$= \sqrt[4]{\frac{S_{TO}}{\varepsilon * \frac{\mathbf{Fa}}{2} * \mathbf{Ha} * \left(1 + \mathbf{Ga} * (\mathbf{TO_{DUT}} - \mathbf{TO_0}) + \mathbf{Fb} * (\mathbf{TA_{DUT}} - \mathbf{TA_0})\right)} + \mathbf{Ta_{[K]}}^4 - \mathbf{273.15} - \mathbf{Hb}}$$

With:
Fa   = EE_Fa * 2^{-46}
Fb   = EE_Fb * 2^{-36}
Ga   = EE_Ga * 2^{-36}
Ha   = EE_Ha * 2^{-14}
Hb   = EE_Hb * 2^{-10}
TO_0 = 25°C
TA_0 = 25°C
$$TA_{DUT} = \frac{(AMB - Eb)}{Ea} + 25$$
Ea   = EE_Ea * 2^{-16}
Eb   = EE_Eb * 2^{-8}
$Ta_{[K]}$ = TA_{DUT} + 273.15 in Kelvin
TO_{DUT} = Object temperature in 25°C
ε   = 1 = Object Emissivity parameter (not stored in EEPROM, but part of the 'app')

The parameters EE_Ea, EE_Eb, EE_Fa, EE_Fb, EE_Ga are signed 32-bit numbers.
The parameters EE_Gb, EE_Ka, EE_Ha and EE_Hb are signed 16-bit numbers.

Note:

One can see that to compute "To (object temperature)", "To" already needs to be known.
"To (object temperature)" is computed in an iterative manner. In the first iteration "To" is assumed to be 25°C.
In the 2nd iteration the result of first iteration is used, and in the 3rd iteration the end result is obtained.
(See example on next page).

11.2.4. Example Extended range measurement Temperature Calculation

Assumed are the following calibration parameters read from EEPROM:

ADDR	PARAM	DATA (hex)	hex to dec	Conversion to use in formula
0x240C	EE_P_R [15:0]	0103	EE_P_R = 005D0103$_{hex}$ = 6095107$_{dec}$	**P_R** = 6095107 * 2^{-8} = **23809.01**
0x240D	EE_P_R [31:16]	005D		
0x240E	EE_P_G [15:0]	FAE5	EE_P_G = 051CFAE5$_{hex}$ = 85785317$_{dec}$	**P_G** = 85785317 * 2^{-20} = **81.81125**
0x240F	EE_P_G [31:16]	051C		
0x2410	EE_P_T [15:0]	0000	EE_P_T = 00000000$_{hex}$ = 0$_{dec}$	**P_T** = 0 * 2^{-44} = **0**
0x2411	EE_P_T [31:16]	0000		
0x2412	EE_P_O [15:0]	1900	EE_P_O = 00001900$_{hex}$ = 6400$_{dec}$	**P_O** = 6400 * 2^{-8} = **25**
0x2413	EE_P_O [31:16]	0000		
0x2424	EE_Ea [15:0]	CFAE	EE_Ea = 0051CFAE$_{hex}$ = 5361582$_{dec}$	**Ea** = 5361582 * 2^{-16} = **81.81125**
0x2425	EE_Ea [31:16]	0051		
0x2426	EE_Eb [15:0]	0103	EE_Eb = 005D0103$_{hex}$ = 6095107$_{dec}$	**Eb** = 6095107 * 2^{-8} = **23809.01**
0x2427	EE_Eb [31:16]	005D		
0x2428	EE_Fa [15:0]	6351	EE_Fa = 03506351$_{hex}$ = 5559995$_{dec}$	**Fa** = 55599953 * 2^{-46} = **7.9E-07**
0x2429	EE_Fa [31:16]	0350		
0x242A	EE_Fb [15:0]	71F1	EE_Fb = FE2571F1$_{hex}$ = -31100431$_{dec}$	**Fb** = -31100431 * 2^{-36} = **-0.00045**
0x242B	EE_Fb [31:16]	FE25		
0x242C	EE_Ga [15:0]	A7A4	EE_Ga = FDFFA7A5$_{hex}$ = -33577052$_{dec}$	**Ga** = -33577052 * 2^{-36} = **-0.00049**
0x242D	EE_Ga [31:16]	FDFF		
0x242E	EE_Gb [15:0]	2600	EE_Gb = 2600$_{hex}$ = 9728$_{dec}$	**Gb** = 9728 * 2^{-10} = **9.5**
0x242F	EE_Ka [15:0]	2A00	EE_Ka = 2A00$_{hex}$ = 10752$_{dec}$	**Ka** = 10752 * 2^{-10} = **10.5**
0x2481	EE_Ha [15:0]	4000	EE_Ha = 4000$_{hex}$ = 16384$_{dec}$	**Ha** = 16384 * 2^{-14} = **1**
0x2482	EE_Hb [15:0]	0000	EE_Hb = 0000$_{hex}$ = 0$_{dec}$	**Hb** = 0 * 2^{-10} = **0**

Table 14: Example EEPROM calibration parameters

The returned values from the RAM (0x4033 to 0x403A):

ADDR	PARAM	DATA (hex)	DATA (dec)
0x4033	RAM_52	FE64	-412
0x4034	RAM_53	FEAB	-341
0x4035	RAM_54	57E4	22500
0x4036	RAM_55	FEA3	-349
0x4037	RAM_56	FE6A	-406
0x4038	RAM_57	59D8	23000
0x4039	RAM_58	000B	11
0x403A	RAM_59	0009	9

Table 15: Example RAM data

11.2.4.1. Ambient temperature calculation

$$VR_{TA} = RAM\_57 + Gb * \frac{RAM\_54}{12} = 23000 + 9.5 * \frac{22500}{12}$$

$$VR_{TA} = 40812.5$$

$$AMB = \left[\frac{RAM\_54}{12}\right]/VR_{TA} * 2^{19} = \left[\frac{22500}{12}\right]/40812.5 * 2^{19}$$

$$AMB = 24086.73813$$

$$\mathbf{Ta}\ (\text{sensor temperature in } °C) = P\_O + \frac{AMB - P\_R}{P\_G} + P\_T\ * (AMB - P\_R)^2$$

$$\mathbf{Ta} = 25 + \frac{24086.73813 - 23809.01}{81.81125} + 0\ * (24086.73813 - 23809.01)^2$$

$$Ta = 28.395°C$$

$$\mathbf{Ta = 28.4°C}$$

11.2.4.2. Object temperature calculation

$$S = \frac{RAM\_52 - RAM\_53 - RAM\_55 + RAM\_56}{2} + RAM\_58 + RAM\_59$$

$$S = \frac{(-412) - (-341) - (-349) + (-406)}{2} + 11 + 9$$

$$S = -44$$

$$VR_{TO} = RAM\_57 + Ka * \frac{RAM\_54}{12} = 23000 + 10.5 * \frac{22500}{12}$$

$$VR_{TO} = 42687.5$$

$$S_{TO} = \left[\frac{S}{12}\right] / VR_{TO} * 2^{19} = \left[\frac{-44}{12}\right] / 42687.5 * 2^{19}$$

$$S_{TO} = -45.034$$

$$TO_0 \quad = 25°C$$
$$TA_0 \quad = 25°C$$

$$TA_{DUT} = \frac{(AMB - Eb)}{Ea} + 25 = \frac{(24086.73813 - 23809.01)}{81.81125} + 25 = 28.3947$$

$$Ta_{[K]} = TA_{DUT} + 273.15 = 28.3947 + 273.15 = 301.5447$$

To (object temperature in °C)

$$= \sqrt[4]{\frac{S_{TO}}{\varepsilon * \frac{Fa}{2} * Ha * \left(1 + Ga * (TO_{DUT} - TO_0) + Fb * (TA_{DUT} - TA_0)\right)} + Ta_{[K]}^4} - 273.15 - Hb$$

The emissivity parameter (Ɛ) is controlled by the user and is assumed in this example equal to 1.
TO_{DUT} = 25 for the first calculation

$$To = \sqrt[4]{\frac{-45.034}{1 * \frac{7.9E - 07}{2} * 1 * \left(1 + (-0.00049) * (25 - 25) + (-0.00045) * (28.3947 - 25)\right)} + (301.5447)^4} - 273.15 - 0$$

$$To = 27.34837117°C$$

The object temperature needs to be calculated 3 times in order the get the end result.
Next object temperature calculation uses previous obtained object temperature.

$$To = \sqrt[4]{\frac{-45.034}{1 * \frac{7.9E - 07}{2} * 1 * \left(1 + (-0.00049) * (\mathbf{27.34837117} - 25) + (-0.00045) * (28.3947 - 25)\right)} + (301.5447)^4} - 273.15 - 0$$

$$To = 27.34715755°C$$

$$To = \sqrt[4]{\frac{-102.35}{1 * \frac{7.9E - 07}{2} * 1 * \left(1 + (-0.00049) * (\mathbf{27.34715755} - 25) + (-0.00045) * (28.3947 - 25)\right)} + (301.5447)^4} - 273.15 - 0$$

$$To = 27.34715818°C$$

$$\mathbf{To = 27.35°C}$$

12. Performance characteristics

12.1. Accuracy

All accuracy specifications apply under settled isothermal conditions only.

12.1.1. Ambient temperature

The calculated ambient temperature has an accuracy of ±3°C between -20°C and 85°C of ambient temperature. Between 15°C and 45°C the accuracy is ±1°C.

12.1.2. Standard Accuracy object temperature
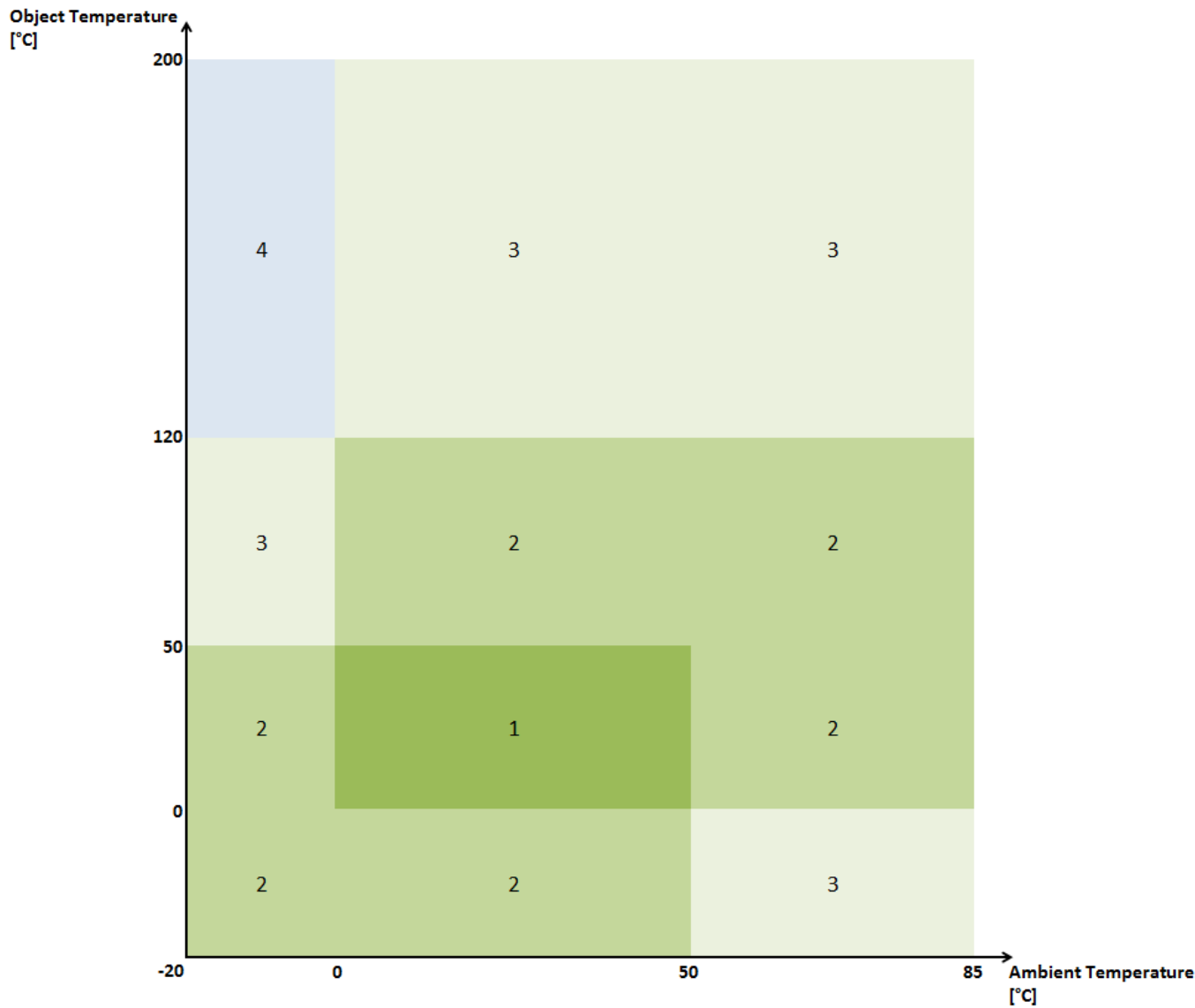
Figure 19: Standard accuracy table

12.1.3. Medical Accuracy object temperature
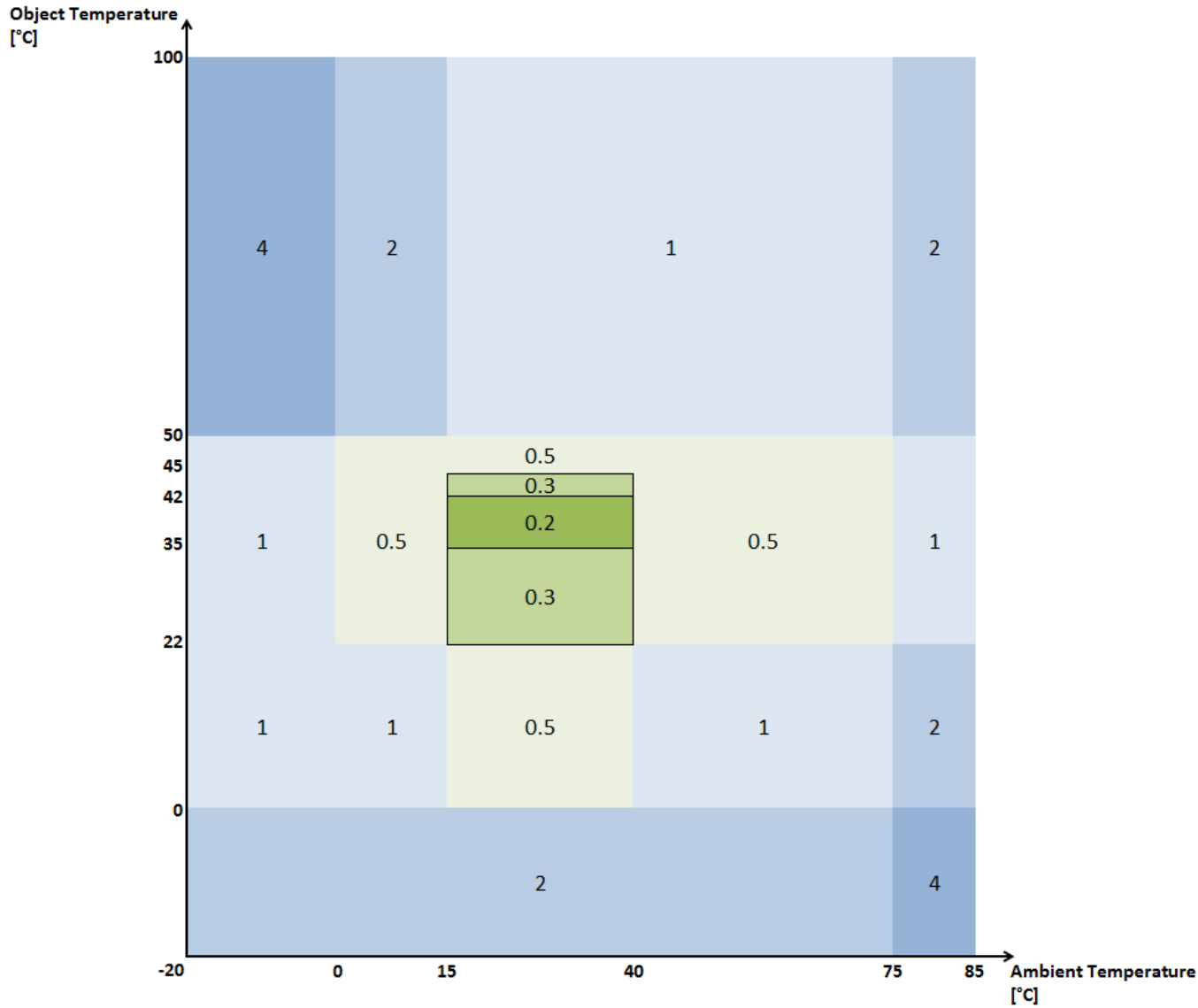
Figure 20: Medical accuracy table

The version MLX90632SLD-DCB complies with the ASTM standard section 5.4 (Designation: E1965 – 98 (Re-approved 2009) - Standard Specification for Infrared Thermometers for Intermittent Determination of Patient Temperature.

12.2. Field of View (FoV)
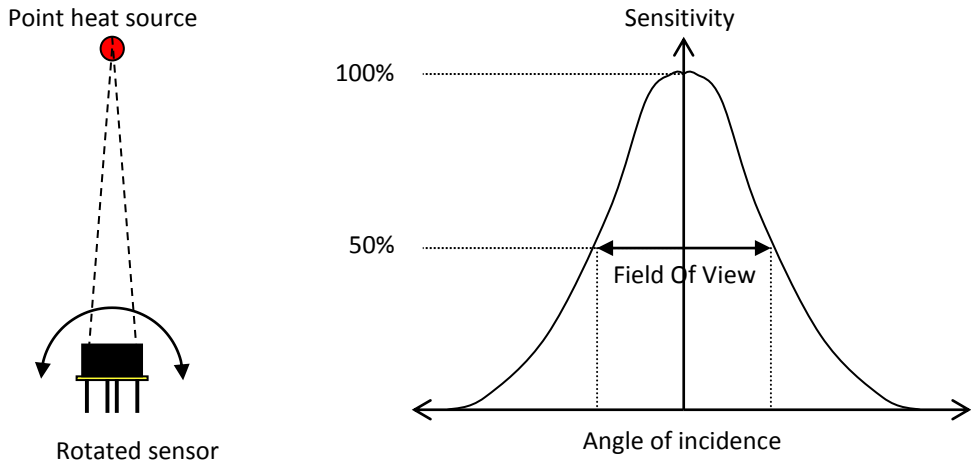
Figure 21: Field of View measurement principle

Parameter	50% of maximum	10% of maximum	Unit
Field Of View	50	70	° (angular degrees)

Table 16: Field Of View of the MLX90632
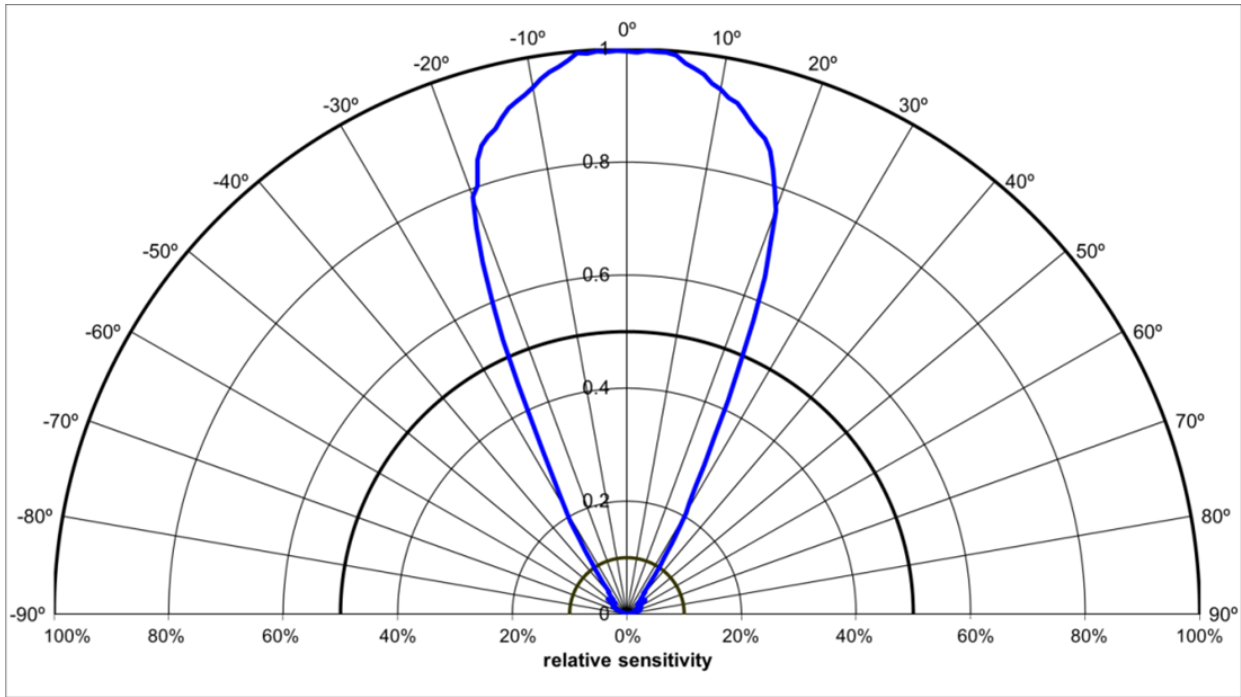
Figure 22: Field of View of MLX90632 (FoV = 50˚)

The 50° is measured at the 50% level of sensitivity.
For high accuracy applications, one should take care that the field of view is not obstructed by the enclosure of the application. For this, one has to take care that no obstruction is in a cone of at least 70° wide. For medical applications the obstacle free zone should be at least 110° wide.

12.3. Noise

Measurement conditions for noise performance are To = Ta = 25°C.

Note:
Due to the nature of thermal infrared radiation, it is normal that the noise will decrease for high object temperatures and increase for lower temperatures.
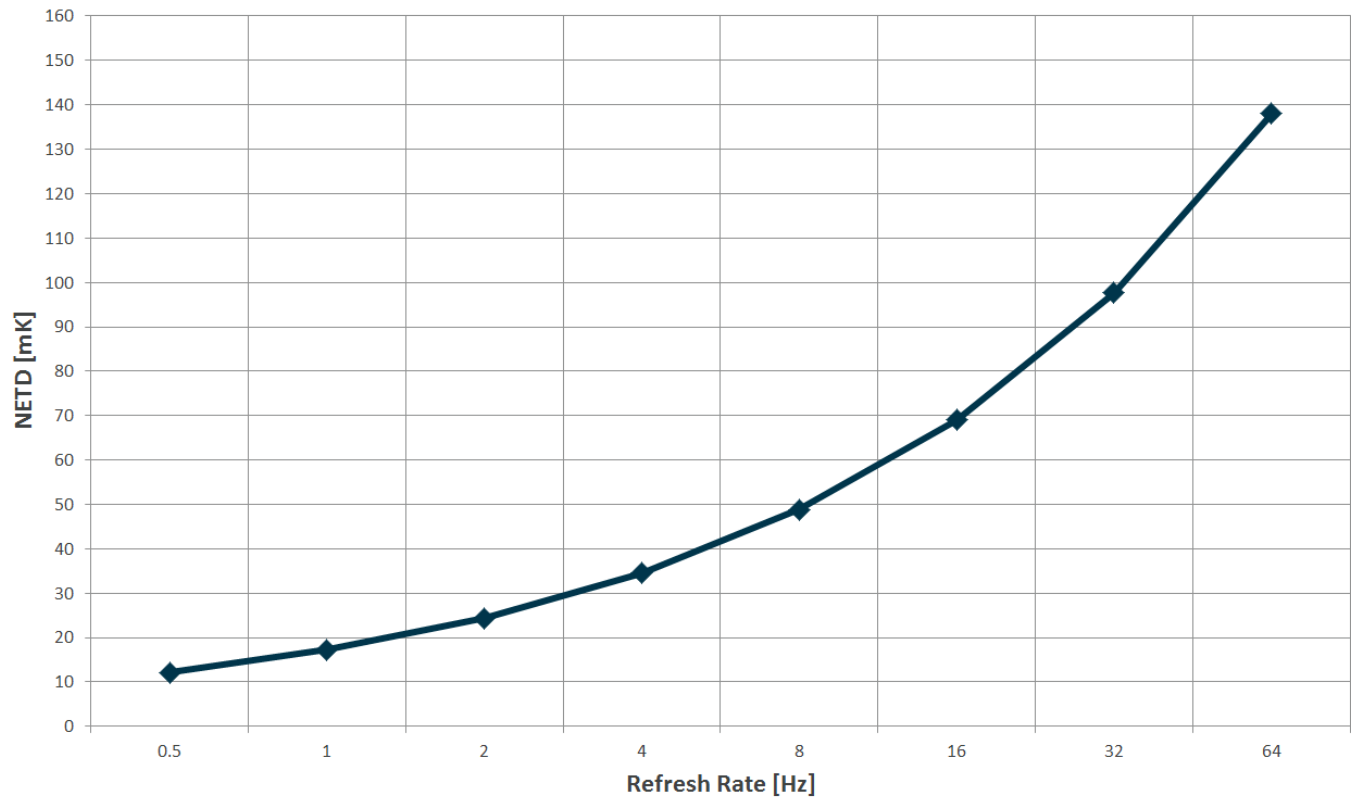
Figure 23: NETD vs. Refresh rate

13. Mechanical Drawing

13.1. Package dimensions

Marking:

.xxxxxD

"xxxxx"= last 5 digits from lot #
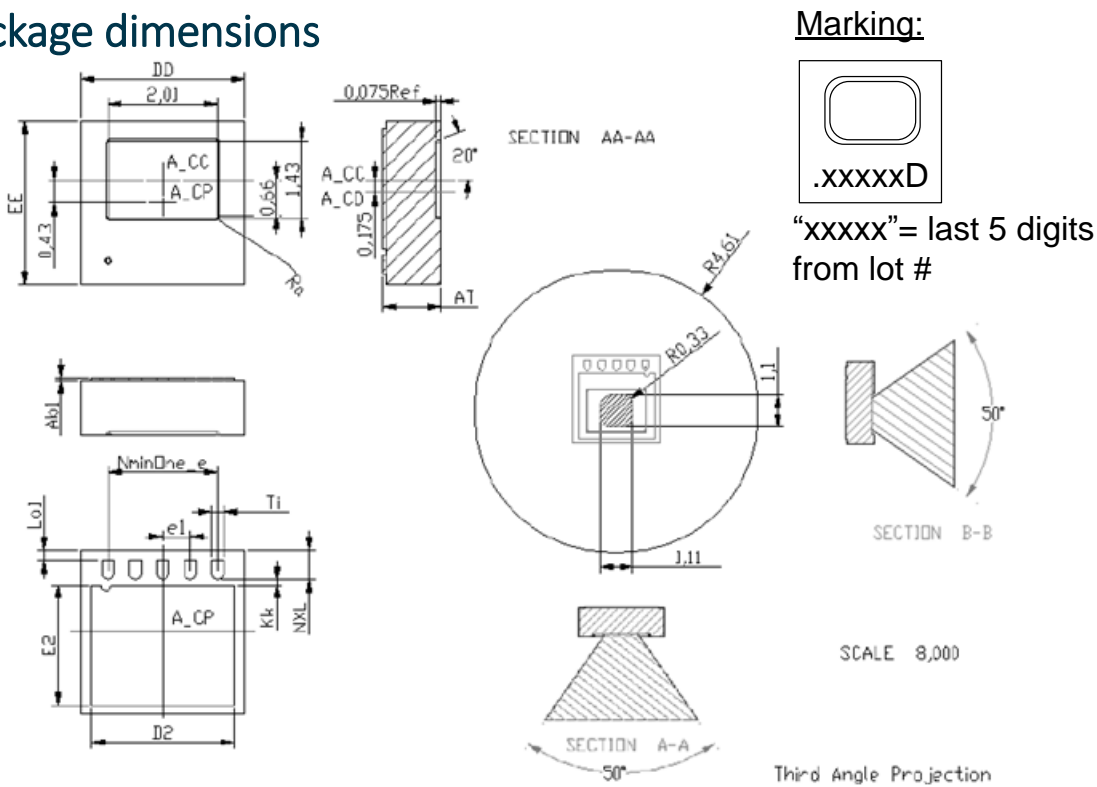
Figure 24: Package dimensions for MLX90632 (FoV = 50°)

Symbol	Min	Nom	Max
DD=EE	2.9	3.0 BSC	3.1
AT	0.90	0.95	1.00
Ra	0.05		
D2	2.40	2.50	2.60
E2	2.00	2.10	2.20
Lo1	0.15 Max		
Kk	0.20	0.30	--
NXL	0.35	0.40	0.45
e1	0.50 BSC		
NminOne_e	(5-1)*e1		
Ti	0.18	0.25	0.30
Tolerance (A_CC – A_CP)	-0.15		0.15
Tolerance (A_CC – A_CD)	-0.1		0.1

Table 17: Package dimensions for MLX90632 (FoV = 50°)

*BSC Ξ basic dimension   *A_CC = Center of silicon Cap   *A_CD = Center of Die frame   *A_CP = Center of Package

13.2. PCB footprint
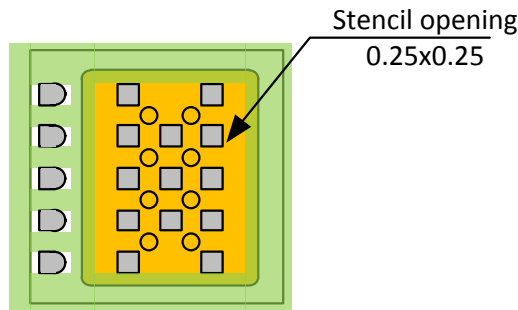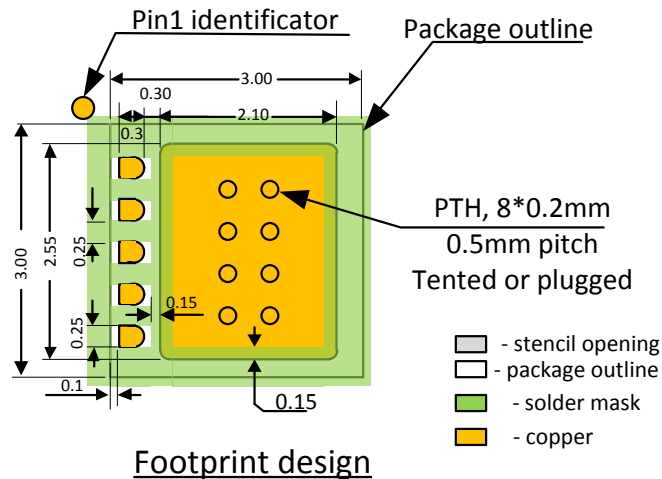
Footprint design

Stencil design

Recommended 1)PCB finish: OSP, ENIG, ENEPIG
2)Stencil thickness max 100um
3)Solderpaste – noclean , halogen free

Figure 25: PCB footprint and stencil design for MLX90632
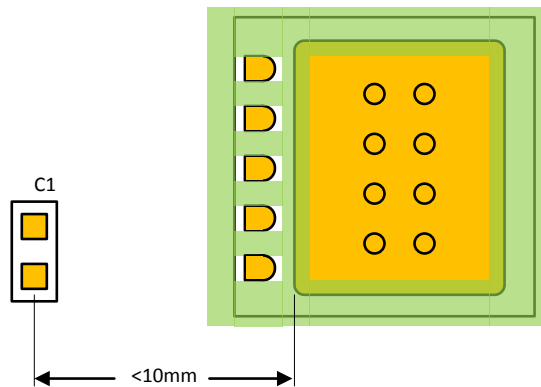
Figure 26a: Decoupling capacitor C1 placement

14. Application schematic

14.1. 3V3 I^{2}C mode
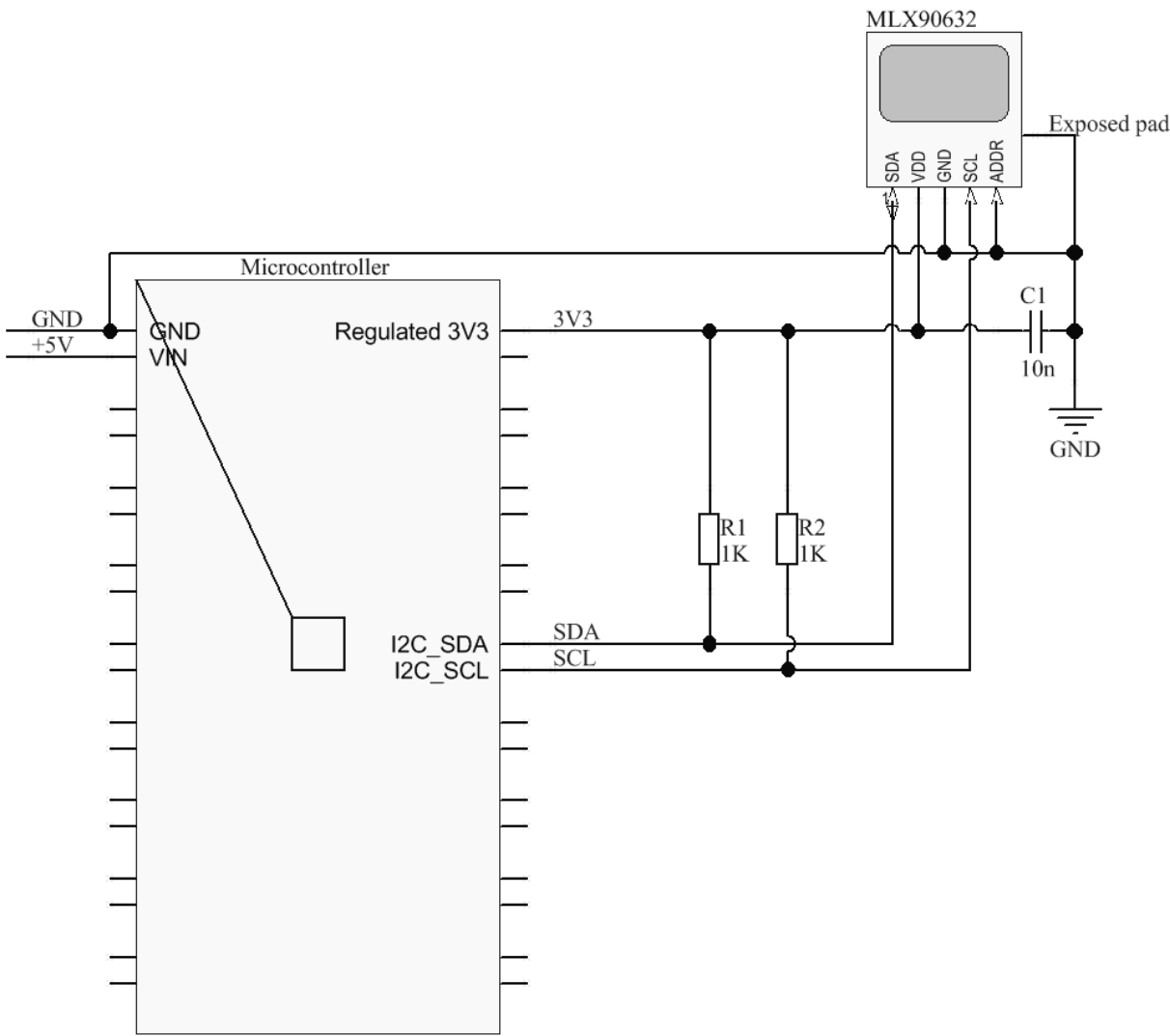
Figure 27: Typical application schematic for 3V3 I^{2}C communication with MLX90632

Note: C1 should be placed within 10mm from the device

14.2. 1V8 I^{2}C mode
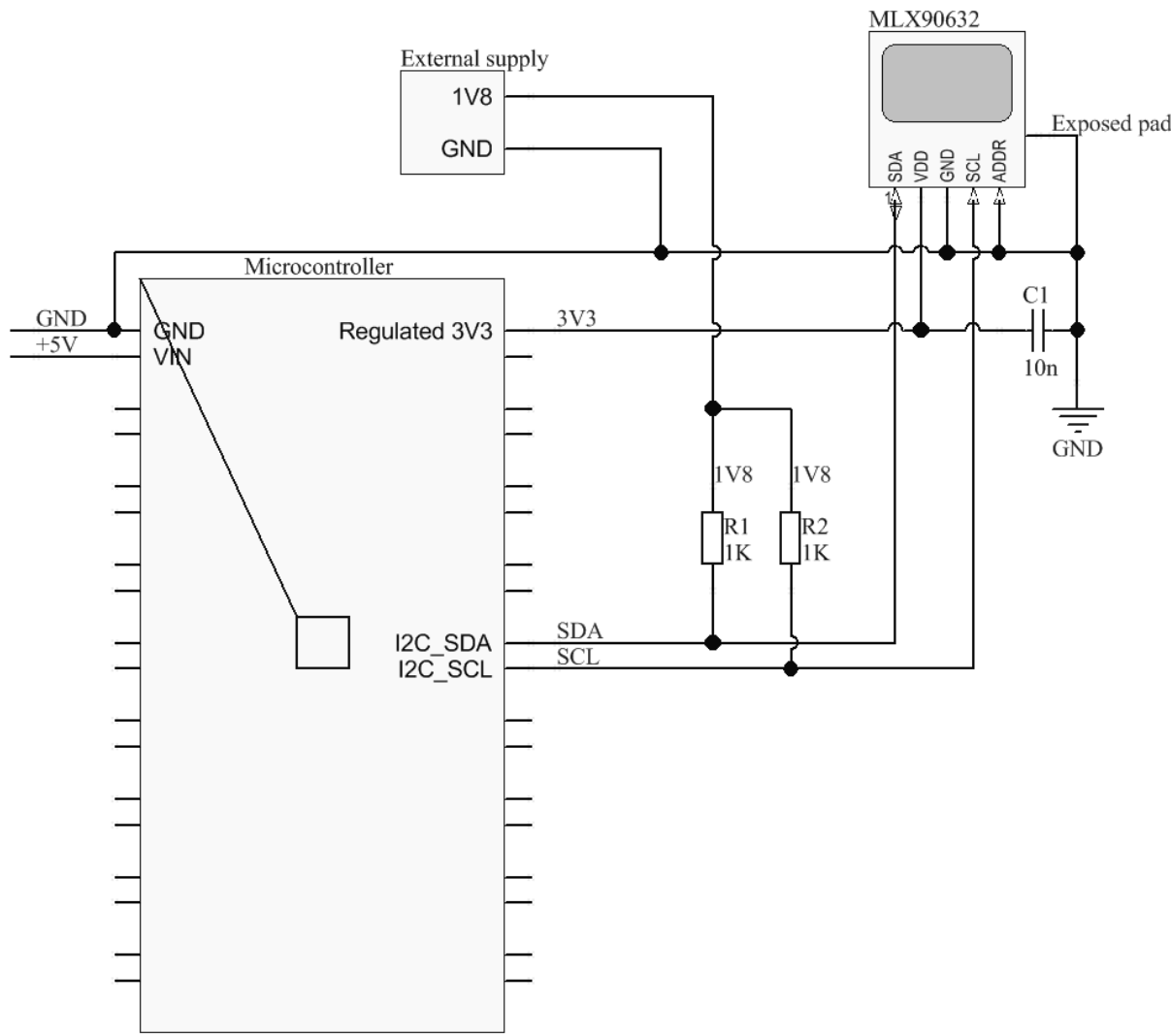
Figure 28: Typical application schematic for 1V8 I^{2}C communication with MLX90632

Note: C1 should be placed within 10mm from the device

15. Software

MLX90632 library on Github:
https://github.com/melexis/mlx90632-library

Example usage of the MLX90632 Library with Keil IDE:
https://github.com/melexis/mlx90632-example

Evaluation board EVB90632:
https://www.melexis.com/en/documents/tools/tools-evb90632-software-exe

16. Standard information regarding manufacturability of Melexis products with different soldering processes

The MLX90632 is a MSL-3 device.
Our products are classified and qualified regarding soldering technology, solderability and moisture sensitivity level according to following test methods:

Reflow Soldering SMD's (Surface Mount Devices)

- IPC/JEDEC J-STD-020

  Moisture/Reflow Sensitivity Classification for Nonhermetic Solid State Surface Mount Devices
  (Classification reflow profiles according to table 5-2)

- EIA/JEDEC JESD22-A113

  Preconditioning of Nonhermetic Surface Mount Devices Prior to Reliability Testing
  (Reflow profiles according to table 2)

Wave Soldering SMD's (Surface Mount Devices) and THD's (Through Hole Devices)

- EN60749-20

  Resistance of plastic- encapsulated SMD's to combined effect of moisture and soldering heat

- EIA/JEDEC JESD22-B106 and EN60749-15

  Resistance to soldering temperature for through-hole mounted devices

Iron Soldering THD's (Through Hole Devices)

- EN60749-15

  Resistance to soldering temperature for through-hole mounted devices

Solderability SMD's (Surface Mount Devices) and THD's (Through Hole Devices)

- EIA/JEDEC JESD22-B102 and EN60749-21

  Solderability

For all soldering technologies deviating from above mentioned standard conditions (regarding peak temperature, temperature gradient, temperature profile etc.) additional classification and qualification tests have to be agreed upon with Melexis.

The application of Wave Soldering for SMD's is allowed only after consulting Melexis regarding assurance of adhesive strength between device and board.

Melexis recommends reviewing on our web site the General Guidelines soldering recommendation (http://www.melexis.com/Quality_soldering.aspx).

Melexis is contributing to global environmental conservation by promoting lead free solutions. For more information on qualifications of RoHS compliant products (RoHS = European directive on the Restriction Of the use of certain Hazardous Substances) please visit the quality page on our website:
http://www.melexis.com/quality.aspx

17. ESD Precautions

Electronic semiconductor products are sensitive to Electro Static Discharge (ESD).
Always observe Electro Static Discharge control procedures whenever handling semiconductor products.

18. Application comments

1. Significant contamination at the optical input side (sensor filter) might cause unknown additional filtering/distortion of the optical signal and therefore results in unspecified errors.

2. IR sensors are inherently susceptible to errors caused by thermal gradients. There are physical reasons for these phenomena and, in spite of the careful design of the MLX90632, it is recommended not to subject the MLX90632 to heat transfer and especially transient conditions.

3. The MLX90632 is designed and calibrated to operate as a non-contact thermometer in settled conditions.

4. Upon power-up the MLX90632 passes embedded checking and calibration routines. During these routines the output is not defined and it is recommended to wait for the specified POR time before reading the module. Very slow power-up may cause the embedded POR circuitry to trigger on inappropriate levels, resulting in unspecified operation which is not recommended.

5. Capacitive loading on an I^2C bus can degrade the communication. Improvement is possible with use of current sources compared to resistors in the pull-up circuitry. Further improvement is possible with specialized commercially available bus accelerators.

6. A sleep mode is available in the MLX90632. This mode is entered and exited via the I^2C compatible 2-wire communication.

7. A power supply and decoupling capacitor is needed as with most integrated circuits. The MLX90632 is a mixed-signal device with sensors, small analog signals, digital parts and I/O circuitry. In order to keep the noise low, power supply switching noise needs to be decoupled. High noise from external circuitry can also affect the noise performance of the device. In many applications a 10nF SMD ceramic capacitor close to the Vdd and Vss pins would be a good choice. It should be noted that not only the trace to the Vdd pin needs to be short, but also the one to the Vss pin.

8. Do not perform measurements in oily or helium environments

19. Table of figures

20. **Disclaimer**

The content of this document is believed to be correct and accurate. However, the content of this document is furnished "as is" for informational use only and no representation, nor warranty is provided by Melexis about its accuracy, nor about the results of its implementation. Melexis assumes no responsibility or liability for any errors or inaccuracies that may appear in this document. Customer will follow the practices contained in this document under its sole responsibility. This documentation is in fact provided without warranty, term, or condition of any kind, either implied or expressed, including but not limited to warranties of merchantability, satisfactory quality, non-infringement, and fitness for purpose. Melexis, its employees and agents and its affiliates' and their employees and agents will not be responsible for any loss, however arising, from the use of, or reliance on this document. Notwithstanding the foregoing, contractual obligations expressly undertaken in writing by Melexis prevail over this disclaimer.

This document is subject to change without notice, and should not be construed as a commitment by Melexis. Therefore, before placing orders or prior to designing the product into a system, users or any third party should obtain the latest version of the relevant information.
Users or any third party must determine the suitability of the product described in this document for its application, including the level of reliability required and determine whether it is fit for a particular purpose.

This document as well as the product here described may be subject to export control regulations. Be aware that export might require a prior authorization from competent authorities. The product is not designed, authorized or warranted to be suitable in applications requiring extended temperature range and/or unusual environmental requirements. High reliability applications, such as medical life-support or life-sustaining equipment or avionics application are specifically excluded by Melexis. The product may not be used for the following applications subject to export control regulations: the development, production, processing, operation, maintenance, storage, recognition or proliferation of:
1. chemical, biological or nuclear weapons, or for the development, production, maintenance or storage of missiles for such weapons;
2. civil firearms, including spare parts or ammunition for such arms;
3. defense related products, or other material for military use or for law enforcement;
4. any applications that, alone or in combination with other goods, substances or organisms could cause serious harm to persons or goods and that can be used as a means of violence in an armed conflict or any similar violent situation.

No license nor any other right or interest is granted to any of Melexis' or third party's intellectual property rights.

If this document is marked "restricted" or with similar words, or if in any case the content of this document is to be reasonably understood as being confidential, the recipient of this document shall not communicate, nor disclose to any third party, any part of the document without Melexis' express written consent. The recipient shall take all necessary measures to apply and preserve the confidential character of the document. In particular, the recipient shall (i) hold document in confidence with at least the same degree of care by which it maintains the confidentiality of its own proprietary and confidential information, but no less than reasonable care; (ii) restrict the disclosure of the document solely to its employees for the purpose for which this document was received, on a strictly need to know basis and providing that such persons to whom the document is disclosed are bound by confidentiality terms substantially similar to those in this disclaimer; (iii) use the document only in connection with the purpose for which this document was received, and reproduce document only to the extent necessary for such purposes; (iv) not use the document for commercial purposes or to the detriment of Melexis or its customers. The confidentiality obligations set forth in this disclaimer will have indefinite duration and in any case they will be effective for no less than 10 years from the receipt of this document.
This disclaimer will be governed by and construed in accordance with Belgian law and any disputes relating to this disclaimer will be subject to the exclusive jurisdiction of the courts of Brussels, Belgium.
The invalidity or ineffectiveness of any of the provisions of this disclaimer does not affect the validity or effectiveness of the other provisions.
The previous versions of this document are repealed.
Melexis © - No part of this document may be reproduced without the prior written consent of Melexis. (2023)

IATF 16949 and ISO 14001 Certified

For the latest revision of this document, visit www.melexis.com/90632

Happy to help you! www.melexis.com/contact