Application note

MLX90394 EVB

Contents

Application note

MLX90394 EVB

1. Scope

This document shows the schematic of the EVB and how it can be used easily to get started. Note that any of the code is provided as is, and is not guaranteed to be error free. It is only to be used as a demo.

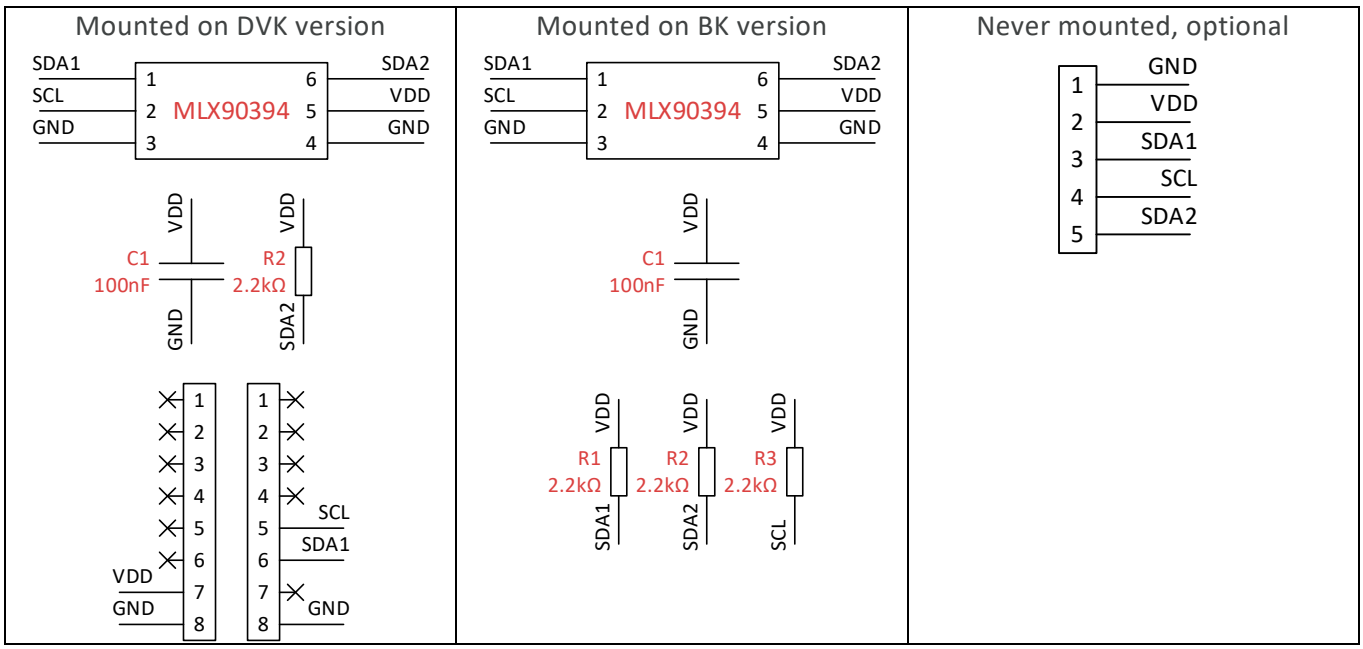
1.1. Different EVB Versions

Two different version of the EVB are made:

- **Breakout** version (BK): The IC is mounted, together with a 100nF decoupling capacitor and 3 pull-up resistors. The five ("useful") pins of the IC are directly routed to the 5-pin header footprint.
- **DVK Magnetic** version: The IC is mounted, together with a 100nF decoupling capacitor and 3 pull-up resistors. Two precision headers are mounted for compatibility with the DVK. Note that the drill holes of these are smaller than the standard headers.

1.2. Schematic

The schematic is for each version (Breakout and DVK) the same. Some components are never populated but can be placed by the user in case required. Selection of the I^2C address is done by selecting one of the two SDA lines and pulling the other one high through a pull-up resistor. This secondary SDA pin becomes then the INTB line. On the DVK, the primary SDA line is used for the communication, only R2 is mounted as on the DVK pull-ups to keep this line high. The I^2C pull-ups are by default on the DVK main board. For the breakout board, a pull-up is placed on both SDA lines and the SCL line.



1.3. Microcontroller and GUI Setup

Firmware is written for the below microcontroller. Note that the IC operates at 3.3V, so in case a 5V IO version is used, make sure to insert a level shifter.

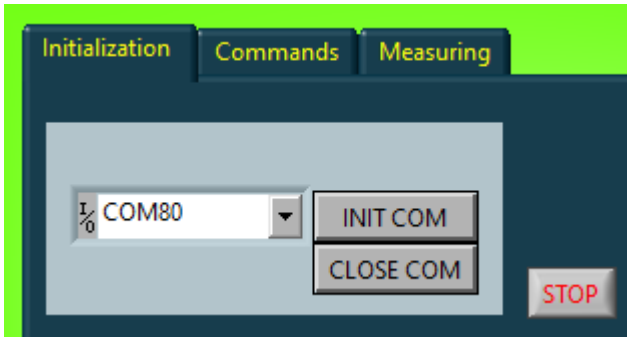
- Compatible Arduino®: https://www.arduino.cc/
    - An example code (.ino) is made implementing all the functions, executing them based on serial commands and returning the replies to the PC.
    - Install the software from the website to have also the drivers and to be able to upload the code to the board.

As GUI, an executable is available which implements the commands from section 3 of this document and visualizes the reply from the IC. As the GUI is written in LabVIEW™ software, make sure to install the runtime engine (2019, 32-bit) in order to run it, which can be downloaded free of charge.

2. GUI

In the demo, three tabs are present. All of them are shown in the following chapters.

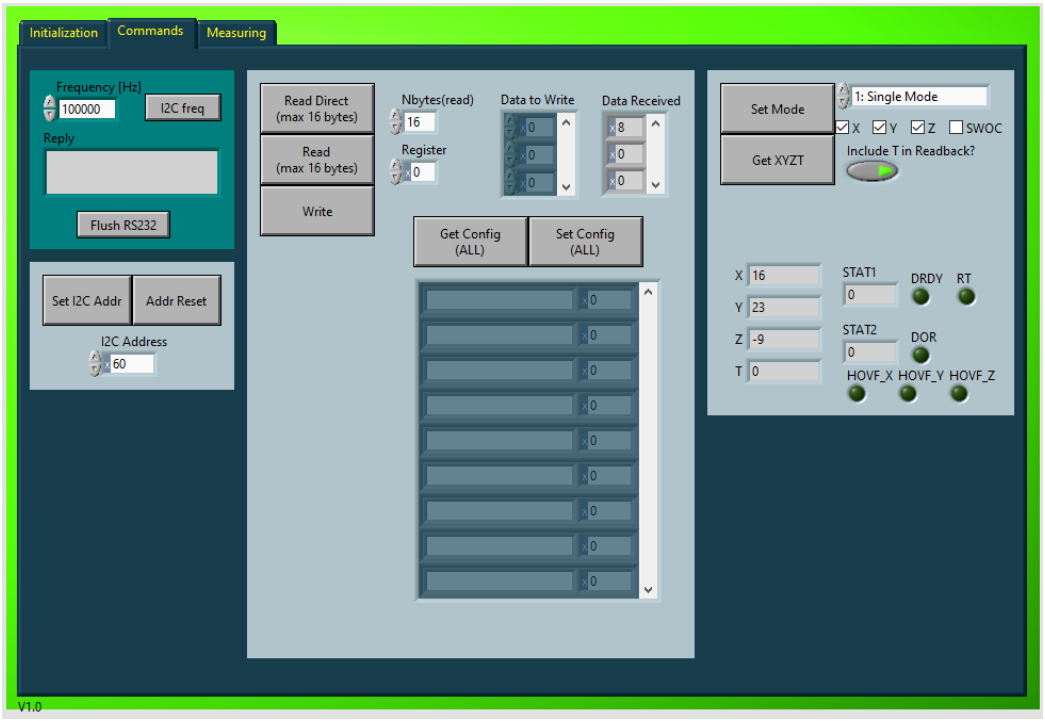
2.1. Initialization



Button	Action/Usage
INIT COM	Select the COM port in the drop down list and click this button to connect to the device.
CLOSE COM	To close to COM port, click this button.
STOP	This button stops the UI, there is no need to use this normally.

2.2. Commands

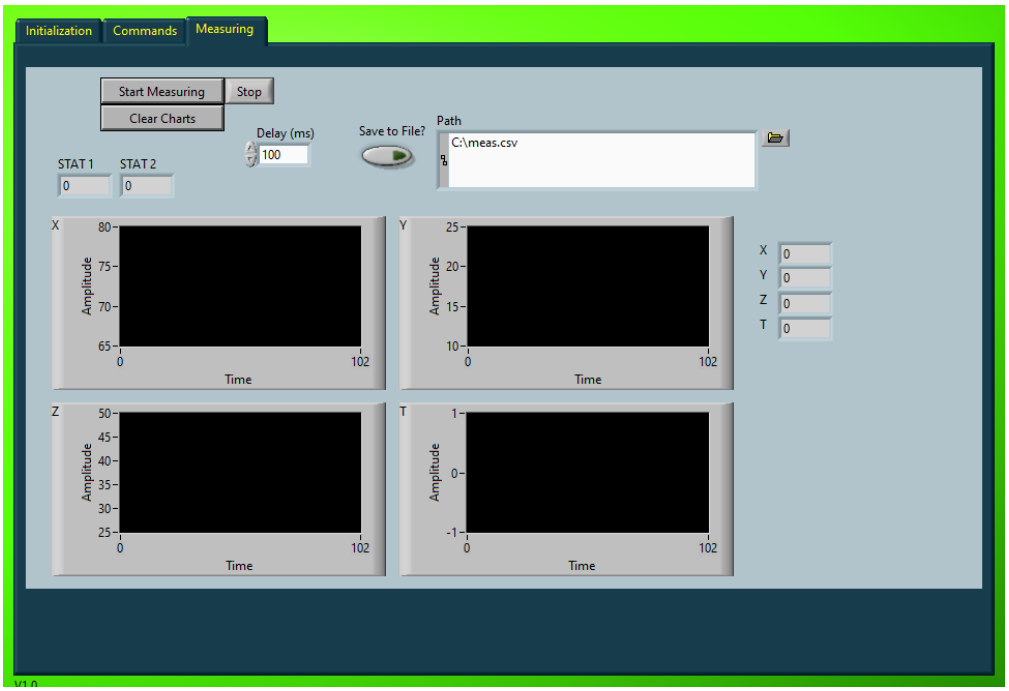
If a Nack is detected, the green border will turn red.



Button	Action/Usage
I2C freq	Change the I²C frequency. By default, the I²C frequency is set to 100kHz.
Flush RS232	Flush the RS232 buffers, click a couple of times in case of corrupted data.
Set I2C Addr	Change the I²C address in the microcontroller settings to the value in the “I2C Address” input box.
Addr Reset	Send a reset command to the device with I²C address as in the “I2C Address” input box.
Read Direct	Read a number of registers (specified by input box “Nbytes(read)”, starting from register 0x00. Answer shown in the “Data Received” output box.
Read	Read a number of registers (specified by input box “Nbytes(read)”, starting from the register specified by the input box “Register”. Answer shown in the “Data Received” output box.
Write	Write the content in the input box “Data to Write” to the IC, starting from the register specified by the input box “Register”.
Get Config	Reads registers 0x0E, 0x0F, 0x14, 0x15, 0x58-0x5D and decodes them in the section below.
Set Config	Writes registers 0x0E, 0x0F, 0x14, 0x15, 0x58-0x5D with the content of the section below.
Set Mode	Select a mode in the enumeration list next to it and click to set the IC in that mode. Selection of axis possible wit checkboxes.
Get XYZT	Get the magnetic measurement and status (and decoded status) from the IC, and show the result in the below output boxes. Temperature can also be read out if “Include T in Readback?” is set.

2.3. Measuring

If a Nack is detected, the green border will turn red.



Button	Action/Usage
Start Measuring	Starts sending in a loop single measurement commands to the IC, and asks after a delay specified by “Delay (ms)” the measurement results from the IC. The data is displayed in the graphs below. Enabling the option “Save to File?” allows for the data to be stored in a csv format. Specify the path first.
Stop	Stop the measurement loop, it allows for the last loop to be completed.
Clear Charts	Clears the graphs below.

3. Serial Commands

The commands are send on the serial bus with following settings:

Setting	Value
Baud rate	9600
# Data bits	8
# Stop bits	1
Parity	None
Flow control	No
End character (PC->microcontroller)	None needed (if ending is sent, execution starts before ending is received and ending is discarded)
End character (microcontroller->PC)	0x0A ("\n" or "Line Feed")

If more than one information is returned, each data section is ended by a ";" to allow for splitting, this includes the last section. In case of a single information, the data is not ended by a ";".

3.1. Read N Bytes

To read N bytes starting from a specified register, send to the microcontroller the following sequence. Note that setting Nbytes to 0 will equal 16 bytes to be read out.

Char #	Message to send	Example: Read 2 bytes, starting from register 0x14
1	**A** as character	A
2	**R[7:4]** as hex character [0-F]	1
3	**R[3:0]** as hex character [0-F]	4
4	**Nbytes** as hex character [0-F]	2

Data #	Data received	Example: Read 2 bytes, starting from register 0x14
1	**Nack;** >0 if I^2C NACK are observed	0;
2	**R_1[7:0];** as decimal value [0-255]	225;
3	**R_2[7:0];** as decimal value [0-255]	
…	**…**	
N+1	**R_N[7:0];** as decimal value [0-255]	181;
N+2	**"\n"**	\n

3.2. Read Direct N Bytes

To read N bytes starting from register 0x00, send to the microcontroller the following sequence. Note that setting Nbytes to 0 will equal 16 bytes to be read out.

Char #	Message to send	Example: Read Direct 16 bytes
1	**B** as character	B
2	**Nbytes** as hex character [0-F]	0

Data #	Data received	Example: Read Direct 16 bytes
1	**Nack;** >0 if I²C NACK are observed	0;
2	**R_1[7:0];** as decimal value [0-255]	0;
3	**R_2[7:0];** as decimal value [0-255]	0;
…	**…**	0;0;0;0;0;0;0;0;152;186;0;0;0;
17	**R_N[7:0];** as decimal value [0-255]	0;
18	**"\n"**	\n

3.3. Addressed Reset

To send an addressed reset on the bus, send to the microcontroller the following sequence.

Char #	Message to send	Example: Addressed Reset, 7-bit I²C address 0x0C
1	**D** as character	D
2	**I2CAddr[6:4]** as hex character [0-7]	0
3	**I2CAddr[3:0]** as hex character [0-F]	C

Data #	Data received	Example: Addressed Reset, 7-bit I²C address 0x0C
1	**Nack** >0 if I²C NACK are observed	0
2	**"\n"**	\n

3.4. Write N Bytes

To write N bytes starting from a specified register, send to the microcontroller the following sequence.

Char #	Message to send	Example: Write 0xE1 0xB5, starting from register 0x14
1	**E** as character	E
2	**R[7:4]** as hex character [0-F]	1
3	**R[3:0]** as hex character [0-F]	4
4	**D1[7:4]** as hex character [0-F]	E
5	**D1[3:0]** as hex character [0-F]	1
6	**D2[7:4]** as hex character [0-F]	B
7	**D2[3:0]** as hex character [0-F]	5
…	**…**	-
2*N+3	**DN[7:4]** as hex character [0-F]	-
2*N+4	**DN[3:0]** as hex character [0-F]	-
2*N+5	**T** as character (terminate transmission)	T

Data #	Data received	Example: Write 0xE1 0xB5, starting from register 0x14
1	**Nack** >0 if I^2C NACK are observed	0
2	**"\n"**	\n

3.5. Change I²C Frequency

To change the I²C frequency, this command has to be sent. The length of the frequency value (in Hz) is fixed to 7 decimals, so padding with "0" is required for frequencies smaller than 1MHz. The default I²C frequency is 100kHz.

Char #	Message to send	Example: Set I²C frequency to 100kHz
1	**W** as character	W
2	**F6** as dec character [0-9]	0
3	**F5** as dec character [0-9]	1
4	**F4** as dec character [0-9]	0
5	**F3** as dec character [0-9]	0
6	**F2** as dec character [0-9]	0
7	**F1** as dec character [0-9]	0
8	**F0** as dec character [0-9]	0

Data #	Data received	Example: Set I²C frequency to 100kHz
1	**"I2C freq now: xxxxx Hz\n"**	I2C freq now: 100000 Hz\n

3.6. Change I²C Address

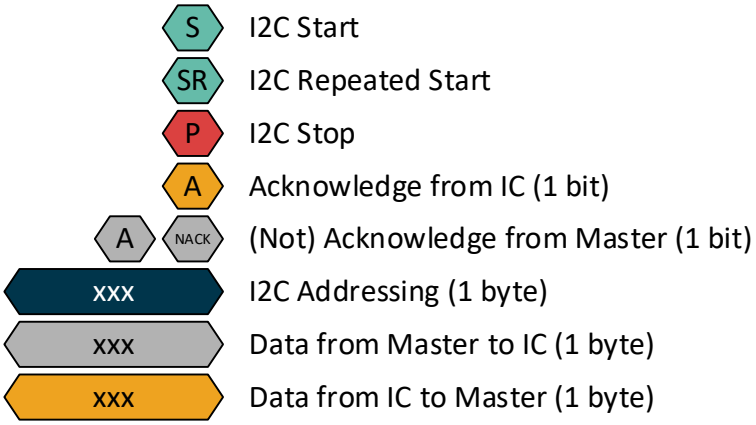
To change the I²C address, this command has to be sent. The length of the address is fixed to 2 characters (hexadecimal), so padding with "0" is required in case of an address below 0x10 (this shall not be the case in MLX90394). The default I²C address is 0x60.

Char #	Message to send	Example: Set I²C address to 0x60
1	**I** as character	I
2	**I2CAddr[6:4]** as hex character [0-7]	6
3	**I2CAddr[3:0]** as hex character [0-F]	0

Data #	Data received	Example: Set I²C address to 0x60
1	**"I2C address now: 0xab\n"**	I2C address now: 0x60\n

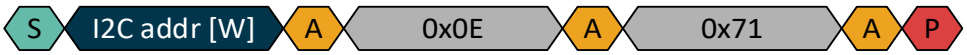
4. Examples

In the examples, the I²C communication is shown using the legend below:

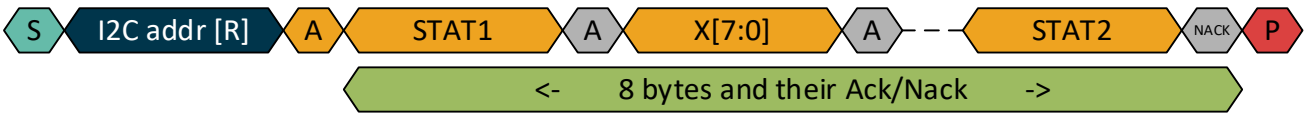


4.1. Performing a Single Measurement (XYZ)

First the mode '1' needs to be written into register 0x0E, the CTRL1 register, as well as the three axis enable bits:

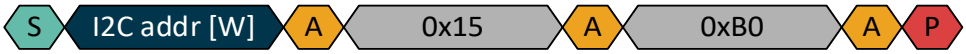


From register 0x00, the measurement data and status bytes can be read out after the measurement came to completion. A Read Direct is used here:

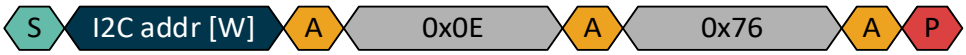


4.2. Performing a Continuous Measurement (XYZT, 100Hz), and Returning to Idle Mode

First the temperature needs to be enabled, and for this example the digital filter for the Z-axis is disabled. This is done by setting T_COMP_EN to register 0x15 (CTRL4).

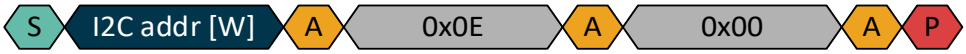


Then the mode ‘6’ (100Hz continuous) needs to be written into register 0x0E, the CTRL1 register, as well as the three axis enable bits:



From register 0x00, the measurement data and status bytes can be read out after the measurement came to completion. A Read Direct is used here. This can be done repeatedly to get new data.

To go back to idle, set the mode back to ‘0’:



5. 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.

IATF 16949 and ISO 14001 Certified

LabVIEW is a trademark of National Instruments. Neither Melexis, nor any software programs or other goods or services offered by Melexis, are affiliated with, endorsed by, or sponsored by National Instruments.

6. Revision History

Revision	Changes
001, 11 Dec 2023	Creation

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

For additional information, get in touch, www.melexis.com/contact