Scope

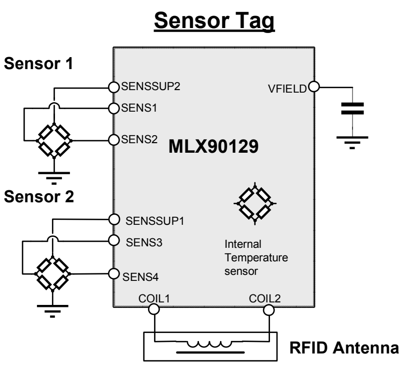
This application note gives some guidelines to detail the use of the MLX90129 for RFID sensor tag applications. The different command sequences to read a sensor and the power check system are described. An example to implement a temperature sensor tag is explained

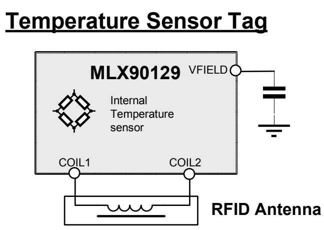
Applications

- ❑ Medical and health monitoring sensor tags
- ❑ Temperature sensor tag
- ❑ Industrial and residential control and monitoring

Related Melexis Products

Part No.	Temperature suffix	Package Code	Option code
MLX90129RGO	R (-40°C to 105°C)	GO [TSSOP 20]	-





Introduction

The MLX90129 is an RFID sensor tag. It senses its internal temperature sensor and/or external resistive sensor in battery less application. The energy is provided by an RFID reader. The minimalist application is a temperature sensor tag based on the MLX90129 internal temperature sensor. External differential resistive sensors such as pressure, strain gauge, light or gas can be directly interfaced through the MLX90192 sensor interface. There are several methods to get the sensor value. One method is based on the MLX90129 RFID custom commands. The other uses only ISO15693 native commands (which avoids any modification of the ISO15693 reader). For the applications with a MLX90129 sensor tag which can move rapidly over the RFID reader, a procedure which detects the electromagnetic field variation is described. This application note puts together all the necessary information, such as options description and basic configuration, in order to allow a quick and easy implementation of the MLX90129 sensor tag applications.

Contents

Note: The RFID commands described in this document use the Dual Subcarrier not addressed mode (flag =0x03) and the letters 'crc' means the CRC (cyclic redundancy check) has to be computed and added at the end of the command.

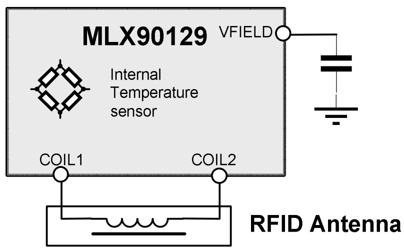
1.General information

1.1. hardware implementation

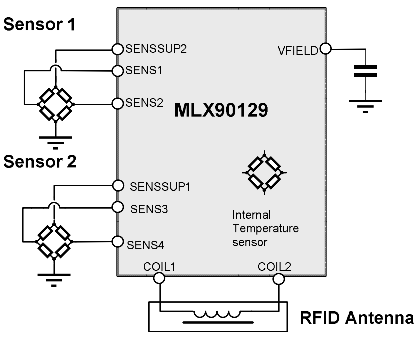
The MLX90129 embeds an internal temperature sensor and has the ability to connect up to two external differential sensors. It can be configured to process up to three sensors. The following schematics show the possible configurations.

Thanks to the MLX90129 internal temperature sensor, the temperature sensor tag application requires only an external capacitor and an RFID loop antenna.

Temperature Sensor Tag



Sensor Tag



If more environmental values need to be sensed, two external differential resistive sensors can be connected.

1.2. RFID reader

The MLX90129 used for sensor tag application is a battery less tag. The energy and the commands are provided by an off-the-shelf RFID reader compliant with the international RFID standard ISO15693. The FCC/CE certified USB desktop reader from Proxima RF is used in the MLX90129 development kit. Further information can be found on the ISO website www.iso.org and on the ProximaRF website www.proximarf.com.

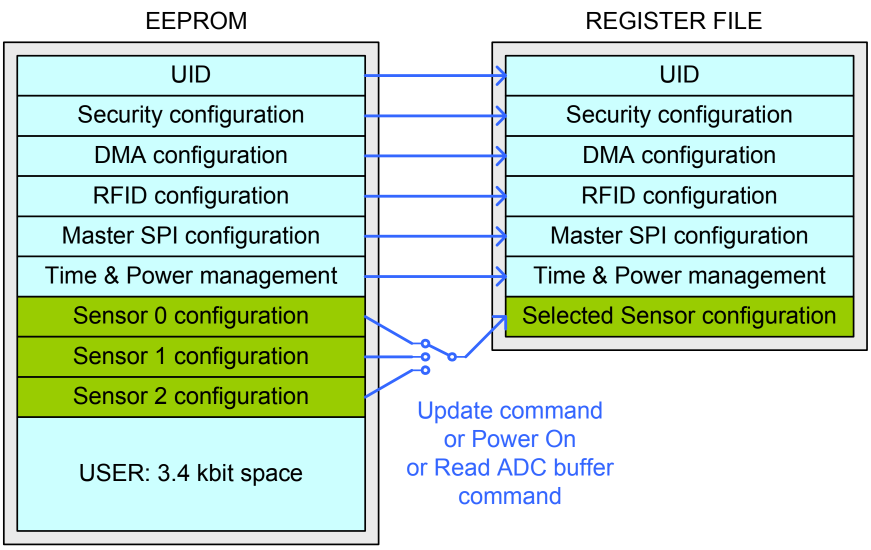
There are two possibilities to read MLX90129 sensors. One uses MLX90129 custom RFID command (Read Internal Device) and the second uses only the native commands defined in the standard ISO15693 (Read and Write Single Block). MLX90129 customs commands allows more flexibility but requires to be implemented in the reader. Most of the off-the-shelf ISO15693 reader allows implementing easily such kind of commands. When the ISO15693 native commands are used, there is no need to implement new set of commands. The protocol using the custom is detailed in the following paragraph. The protocol using only ISO15693 native commands is the object of the specific paragraph: 4.Read sensor value with ISO15693 native command.

1.3. Sensor configuration in MLX90129 memory

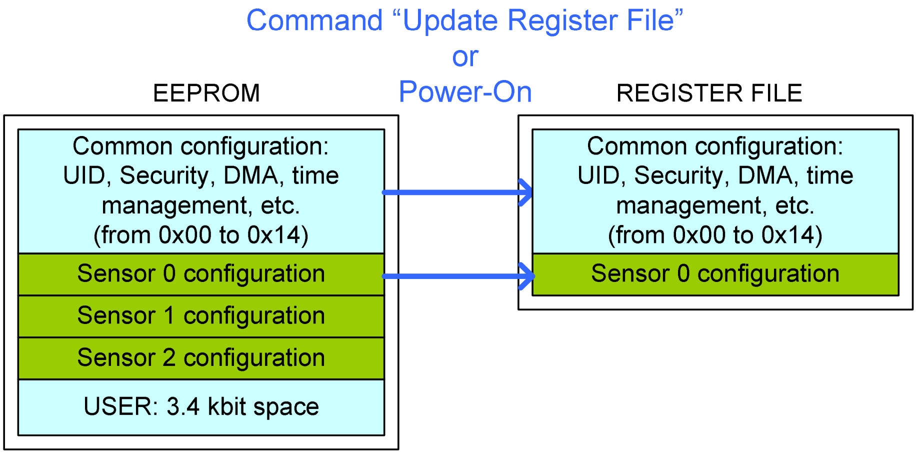
The MLX90129 EEPROM memory offers three areas to configure the sensors (sensor 0, sensor 1, and sensor 2). Each sensor configuration area allows to set the ADC settings, the gain, the offset, the threshold, etc. Two registers are used to configure the common part of the sensors. More information about the sensor configuration can be found in the MLX90129 datasheet. The following table summarizes the memory areas addresses.

Address (hex)	Description
0x12 - 0x14	Configuration in common for all 3 sensors
0x15 - 0x1A	Sensor 0 configuration
0x1B - 0x20	Sensor 1 configuration
0x21 - 0x26	Sensor 2 configuration

The sensor used is the one loaded into the Register file area. The register file is the RAM of the MLX90129. In addition to the MLX90129 settings saved in the EEPROM (UID, DMA, timer, SPI, time and power management settings), only one sensor configuration is loaded in the Register file. The following picture illustrates this load from EEPROM to register file. The selection of the sensor loaded in the register file (sensor 0, sensor 1 or sensor 2) depends of the state of MLX90129 or the command sent.



After power–on or after the RFID command "Update Register File", the configuration of the sensor 0 in EEPROM (0x15-0x1A) is copied into the register file.

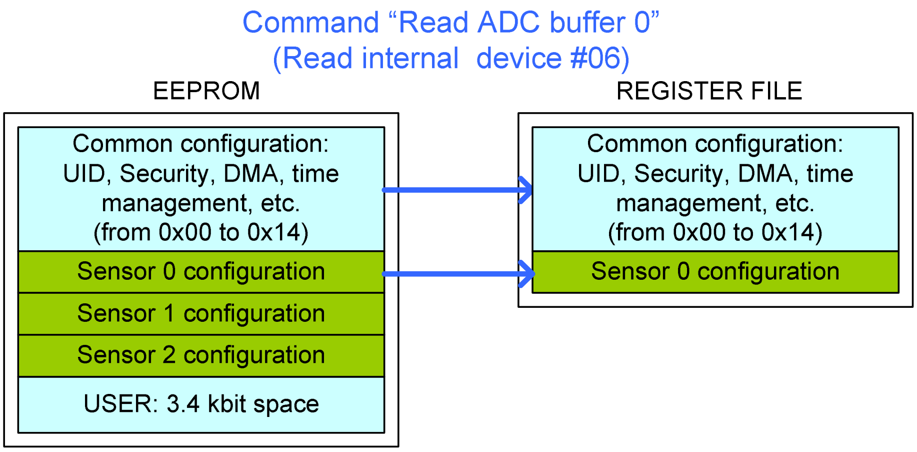


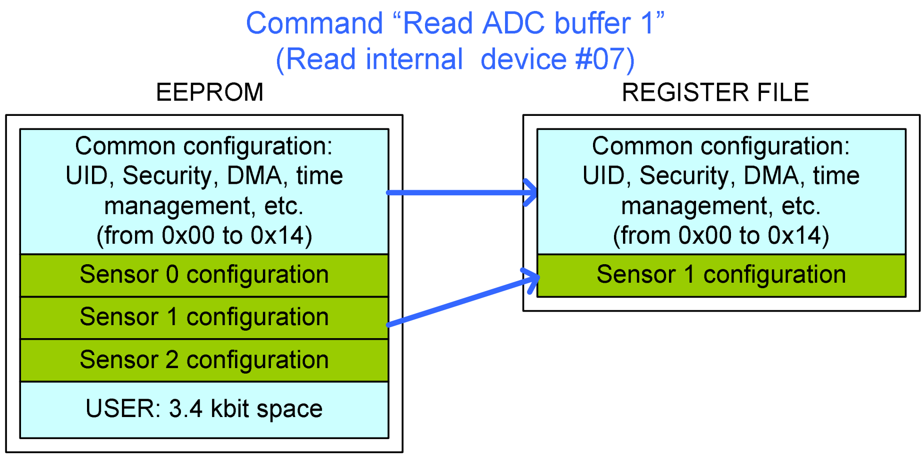
To load a specific sensor, the associated ADC buffer available in the internal device domain of the MLX90129 has to be read. The RFID command code to read the internal device domain is "0xA21F".

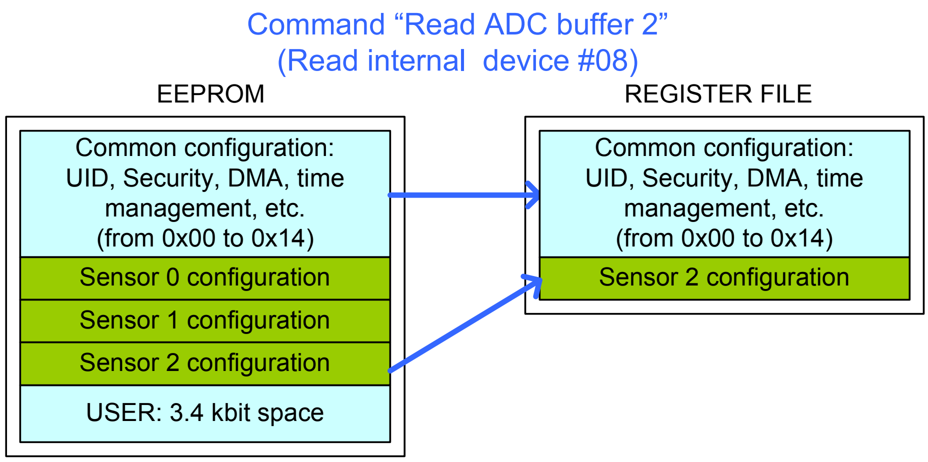
ADC buffer	Address	RFID commands
ADC buffer sensor 0	0x06	03A21F06 crc
ADC buffer sensor 1	0x07	03A21F07 crc
ADC buffer sensor 2	0x08	03A21F08 crc

Notes: In the following paragraphs, the expression "Read ADC buffer 0" is considered as similar as the RFID command "Read Internal device 06" and respectively with ADC buffer sensor 1 and ADC buffer sensor 2.

The following pictures illustrate what is loaded from the EEPROM to the Register file after each "Read ADC buffer" commands







/!\ the RFID response after a Read Internal Device 0x06, 0x07 or 0x08 command is always the error code 0xA1 (The selected Device is busy). It is due to the fact that this command loads the sensor configuration from EEPROM to Register Files and start the ADC conversion. The RFID answer is sent after about 300 us whereas the ADC conversion takes some milliseconds, so when answering the MLX90129 is busy so the error code 0xA1. The following paragraphs explain how to get the output of the ADC conversion.

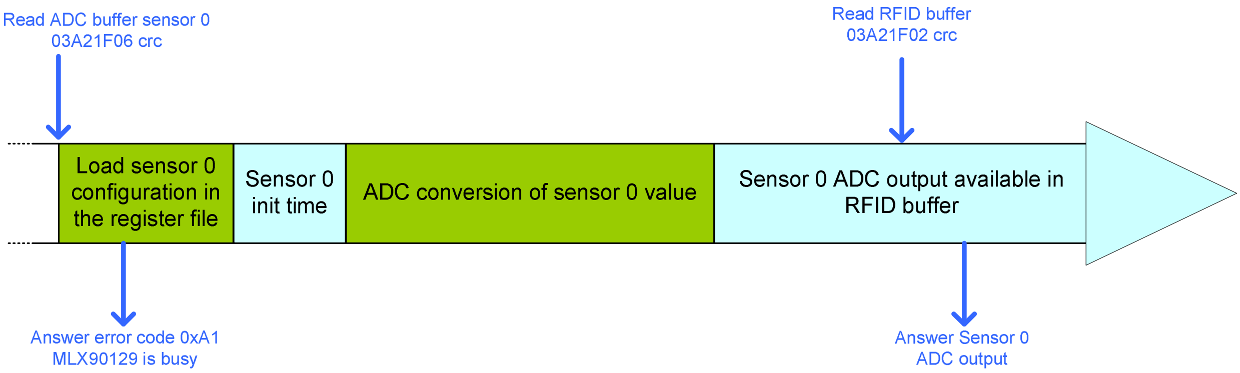
1.4. Sensor conversion chronology

This paragraph describes the several steps of the sensor conversion. Whatever sensor 0, sensor 1 or sensor 2 is used the steps are the same.

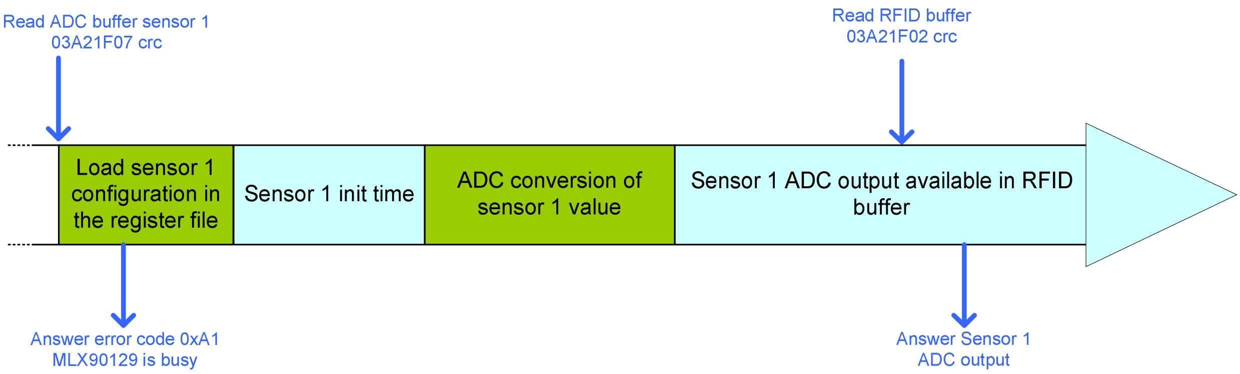
- The command reads internal device 0x06, 0x07 or 0x08 loads the register files with the EEPROM image of the selected sensor configuration.
- The sensor acquisition starts by the waiting time defined in the register "Sensor control word" (addresses #15, #1B, #21) bits b12 and b13. During this time, the sensor is supplied but the ADC conversion did not start. This time can last from some microseconds up to more than one hundred milliseconds depending the settings.
- After the waiting time, the ADC conversion starts. The last of the ADC conversion depends of the ADC settings defined in the register "Sensor control word" (addresses #15, #1B, #21). The ADC mode and the number of averaging allow to balance ADC accuracy with conversion time. Depending the settings the acquisition time can last from 3.2 ms up to 1.5 seconds.
- After the ADC conversion the output of this conversion is available in the "RFID local buffer" at the address 0x02 of the internal device domain.
- To get by RFID this value the command read internal device 0x02 has to be sent.

ADC buffer	Address	RFID commands
RFID local buffer (read only)	0x02	03A21F02 crc

The following timeline illustrates the previous steps. Note that the timing proportions are not respected.



It would be exactly similar if sensor 1 configuration was used except that the setting for initialization time and for the ADC may be different.



2.Get sensor value

As shown in the previous paragraph about sensor conversion chronology, the output of the sensor conversion can be read by RFID in the RFID buffer (internal device #02). However the reader needs to know when the data is valid and available. Two techniques can be implemented.

2.5. Constant timing technique

This technique can be implemented when the user knows the settings of the sensor or when the sensing rate is not critical. It is simply based on a fixed waiting time. With the settings of the sensor, specifically the initialization time, the ADC mode and the averaging, the conversion time can be determined. In addition it is recommended to take a margin of 10 or 15 %.

Examples:

Sensor Control word (#15, #1B, #21) = 0x1002
ADC low power mode disabled
ADC mode = 00 => 1 conversion lasts 3.2 ms
Initialization time => 01 => about 2 ms
Averaging = 10 = mean of 8 samples

ADC acquisition time = 2 + 8*3.2 = 27.6 ms

Sensor Control word (#15, #1B, #21) = 0xD000
ADC low power mode disabled
ADC mode = 1 => 1 conversion lasts 21 ms
Initialization time => 01 => about 2 ms
Averaging = 00 = single sample

ADC acquisition time = 2 + 21 = 23 ms

Consequently the sequence of commands to obtain the value of the sensor is:

Actions	RFID commands	Comments
Field ON		The MLX90129 is supplied by RF field
Read Internal device 0x06	03A21F06 crc	Read ADC buffer sensor 0
Wait the time needed for the ADC to do the conversion		
Read Internal device 0x02	03A21F02 crc	Read the ADC sensor output

To cover the entire possible ADC configuration, a reader would need to wait the longest possible acquisition time which is:
Sensor Control word (#15,#1B, #21) = 0xF83
ADC low power mode enabled
ADC mode = 11 => 1 conversion lasts 42 ms
Initialization time => 11 => about 128 ms
Averaging = 11 = mean of 32 samples

ADC acquisition time = 128 + 32*42 = 1.47 s

It could be limiting for the application to have to wait about 1.5 seconds if in reality the conversion time is much shorter. So this technique is recommended only when the acquisition time of the sensor is known. Otherwise there is the polling technique which is more flexible.

2.6. Polling technique

The polling technique checks IRQ flags meaning that the MLX90129 is still busy. As soon as these flags are cleared the output of the ADC conversion can be read in the RFID buffer.

When the MLX90129 is busy, the bits 0 and 1 of the RFID status word at the address 0x01 of the internal device domain are set to 1.

Bits	Name	Description
#01 – RFID interrupt & status word		
1	Last_Transaction_Status	This bit indicates whether the last request has been processed (‘0’) or not (‘1’). In this latter case, the MLX90129 ignores any new request.
0	Core_Main_Status	The system is busy with an internal operation and the request from RFID cannot be processed.

To get the status of these bits the internal device domain at the address 0x01 has to be read with the following command

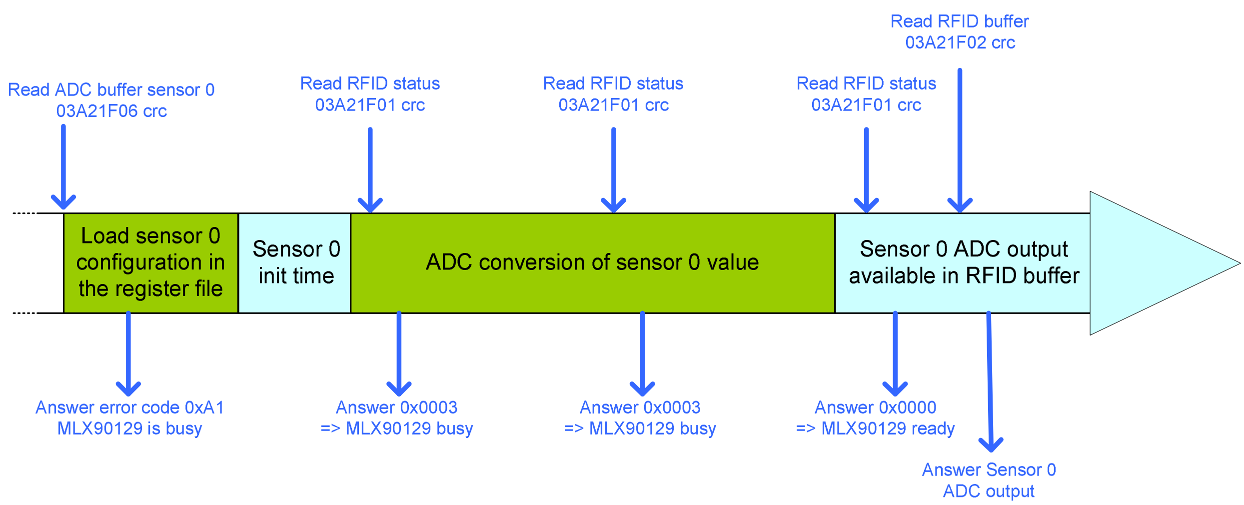
Action	Address	RFID commands
Read RFID interrupt & status word	0x01	03A21F01 crc

In order to limit the number of commands sent to the MLX90129 it is advised to implement delay between polling sequences. Consequently the sequence of commands to obtain the value of the sensor is:

Actions	RFID commands	Comments
Field ON		The MLX90129 is supplied by RF field
Read Internal device 0x06	03A21F06 crc	Read ADC buffer sensor 0
Optional: Wait at least 3.2ms (minimal ADC conversion time)		
Do {		
Read Internal device 0x01	03A21F01 crc	Read RFID interrupt & status word
Optional: Wait 1 ms or more between each polling		
} While (bit 0 and bit 1 are not cleared)*		
Read Internal device 0x02	03A21F02 crc	Read the ADC sensor output

*to avoid infinite loop in case of problem it is advised to implement a watchdog timer.

The following timeline illustrates the protocol.



The protocol is similar when several sensors are used as described in the following execution example:

```
Iso15 02A21F06 crc -> // Read ADC buffer sensor 0
  <- OK - Command has been sent
  <- TAG Resp 01A11CA2 CRC OK // error code 0xA1

Iso15 02A21F01 crc -> // Read RFID interrupt & status word
  <- OK - Command has been sent
  <- TAG Resp 000300A4EC CRC OK // status word =0x0003 => bits 0 and 1 equal to 11 => device busy

Iso15 02A21F01 crc ->
  <- OK - Command has been sent
  <- TAG Resp 000300A4EC CRC OK // status word =0x0003 => device busy

Iso15 02A21F01 crc ->
  <- OK - Command has been sent
  <- TAG Resp 000000CCC6 CRC OK // status word =0x0000 => device ready

Iso15 02A21F02 crc -> // Read RFID buffer
  <- OK - Command has been sent
  <- TAG Resp 000D93A6D0  CRC OK  // sensor 0 ADC value = 0x3AD9

Iso15 02A21F07 crc -> // Read ADC buffer sensor 1
  <- OK - Command has been sent
  <- TAG Resp 01A11CA2  CRC OK // error code 0xA1

Iso15 02A21F01 crc -> // Read RFID interrupt & status word
  <- OK - Command has been sent
  <- TAG Resp 000000CCC6  CRC OK // status word =0x0000 => device ready

Iso15 02A21F02 crc ->// Read RFID buffer
  <- OK - Command has been sent
  <- TAG Resp 00346FFF8C  CRC OK // sensor 1 ADC value = 0x6F34
```

3. Weak field detection

In the application where the MLX90129 sensor tag can be used in movement or at different distance of the reader it is possible that the variation of the supply due to the weak electromagnetic field corrupts the sensor value. For this reason, weak field detection can be implemented. This method is based on an internal block and requires only a very simple calibration.

3.7. Sensor settings

The following settings have to be applied in one of the sensor configuration area in the MLX90129 EEPROM (sensor 0, sensor 1 or sensor 2). It corresponds to a configuration where the bits to enable the power check are set to '1' in the "Connection register". The following example sets the weak field detection settings in sensor 0 area in EEPROM.

Content	#	Hexa	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Control word	15	0070	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0
Low threshold	16	0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
High threshold	17	0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Conditioner config.	18	0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Connection config.	19	E000	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Resistance network	1A	0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

/!\ The weak field check works only when the MLX90129 is used in high voltage mode (not the default setting).

To switch the MLX90129 in high voltage mode, the bit 3 of the EE-Latches word 1 (Internal device 0x04) has to be reset. So the command Write Internal Device #04 with data 0x0003 has to be sent. After this command the MLX90129 needs to be reset.

Actions	RFID commands	Comments
Field ON		The MLX90129 is supplied by RF field
Write 0x0003 in internal device 0x04	03A31F040300 crc	No responds from the tag
Field OFF		
Field ON		
Read internal device 0x04	03A21F04 crc	Answer 0x0003 => High Volt Mode selected

3.8. Weak field threshold

The weak field detector is used as any sensor. The methods described in the previous paragraph to get the sensor value can be applied. It is necessary to define a threshold which corresponds to the limit of the correct supply of the MLX90129. To define the threshold, place the tag in the operating volume of the reader. Close to the antenna is usually the best position. Then read the sensor which corresponds to the weak field detection settings. For example if you have set it in sensor 0, read ADC buffer sensor 0.

The number of significant bits used in the threshold will impact the application. The reading distance, the number of valid data, etc will be impacted. So the threshold value has to be chosen carefully after characterization. The value has to be stored in the MLX90129 EEPROM. In the following example it is stored in address 0x2A. For the internal temperature sensor, 8 significant bits are enough.

Iso15 02A21F06 crc -> // Read ADC buffer sensor 0 (weak field detector)
<- OK - Command has been sent
<- TAG Resp 01A11CA2 CRC OK

Iso15 02A21F01 crc -> // Read RFID interrupt & status word
<- OK - Command has been sent
<- TAG Resp 000000CCC6 CRC OK

Iso15 02A21F02 crc -> // Read RFID buffer
<- OK - Command has been sent
<- TAG Resp 00949A2283 CRC OK // sensor 0 ADC value = **0x9A94**
// 8 MSbits from **0x9A94 = 0x9A00**

Iso15 42212A009A crc -> // data 0x9A00 is written in EEPROM 0x0A
<- OK - Command has been sent
<- TAG Resp 0078F0 CRC OK

Iso15 02202A crc -> // The Weak field threshold is read back in EEPROM 0x0A
<- OK - Command has been sent
<- TAG Resp 00009A1FFD CRC OK // Vbandgap EEPROM #2A : **0x9A00**

3.9. Implementation with a sensor

To illustrate how the weak field detection can be implemented, the internal temperature sensor is taken as an example. The weak field detection settings are in sensor 0 and internal temperature sensor settings are in sensor 1. If the movement of the tag over the reader is very fast, a weak field detection has to be implemented before and after the sensor conversion, if it is not the case a simple check before or after may be enough to ensure the validity of the data. **The data of the sensor is valid only if the output of the weak field detection is above the threshold.**

In the following sequence, the field is checked before and after the conversion of the temperature sensor.

Actions	RFID commands	Comments
Field ON		The MLX90129 is supplied by RF field
Read EEPROM 0x2A	03202A crc	Read the Field_Threshold
Read Internal device 0x06	03A21F06 crc	Read ADC buffer sensor 0 (weak field detector)
Read Internal device 0x02	03A21F02 crc	Get the Field_value_1
Read Internal device 0x07	03A21F07 crc	Read ADC buffer sensor 0 (internal temperature sensor)
Read Internal device 0x01	03A21F01 crc	Check interrupt. Continue if device is no busy
Read Internal device 0x02	03A21F02 crc	Get temperature value
Read Internal device 0x06	03A21F06 crc	Read ADC buffer sensor 0 (weak field detector)
Read Internal device 0x02	03A21F02 crc	Get the Field_value-2
The temperature value is valid only if : “Field value 1” > “Field_Threshold” and if “Field_value_2” > “Field_Threshold”		

The following example illustrates the different RFID commands:

Iso15 02202A crc -> // Read EEPROM 0x2A
<- OK - Command has been sent
<- TAG Resp 00009A1FFD CRC OK //Threshold = **0x9A00**

Iso15 02A21F06 crc -> // Read ADC buffer sensor 0 (weak field detector)
<- OK - Command has been sent
<- TAG Resp 01A11CA2 CRC OK

Iso15 02A21F02 crc ->
<- OK - Command has been sent
<- TAG Resp 00969A92B0 CRC OK // Field value 1 = **0x9A96**

Iso15 02A21F07 crc -> / Read ADC buffer sensor 1 (internal temperature sensor)
<- OK - Command has been sent
<- TAG Resp 01A11CA2CRC OK

Iso15 02A21F01 crc -> // Read RFID interrupt & status word
<- OK - Command has been sent
<- TAG Resp 000000CCC6 CRC OK // device not busy

Iso15 02A21F02 crc ->
<- OK - Command has been sent
<- TAG Resp 0093945427 CRC OK // Get internal temperature sensor value **0x9493**

Iso15 02A21F06 crc -> // Read ADC buffer sensor 0 (weak field detector)
<- OK - Command has been sent
<- TAG Resp 01A11CA2 CRC OK

Iso15 02A21F02 crc ->
<- OK - Command has been sent
<- TAG Resp 00989A822A CRC OK // Field value 2 = **0x9A98**

// Field value 1 (**0x9A96) >** Threshold (**0x9A00)**
// Field value 2 (**0x9A98) >** Threshold (**0x9A00)**
// So,the internal temperature sensor value (**0x9493)** is valid

4.Read sensor value with ISO15693 native command

The following paragraph gives a method to read MLX90129 sensors when the application does not allow implementing the MLX90129 custom commands such as “Update register” or “Read Internal Device”. The principle is to configure the MLX90129 to save the sensor data in MLX90129 memory. As the MLX90129 memory can be accessed with ISO15693 native commands Read and Write Single Block, there is no need to implement additional custom commands.

4.1. Memory settings

This method is based on the datalogging features of the MLX90129. Bits 12, 13, 14 of EEPROM 0x09 will allow selecting which sensor will be saved in the internal memory. The number of sensor saved will impact the value of EEPROM 0x0C. Finally the final location of the sensor conversion output in the MLX90129 memory is determined by the EEPROM 0x0B. In the following example the Sensor 0 will be saved in EEPROM 0x0B (/!\ the address has to be above 0x29 otherwise it will overwrite the configuration data)
For more information please report to the MLX90192 datasheet and the application note about datalogging.

The following settings have to be written if only sensor 0 is used

Content	#	Hexa	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Configuration	09	1071	0	0	0	1	0	0	0	0	0	1	1	1	0	0	0	1
source start address	0A	0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
destination start address	0B	002B	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	1
processing length	0C	0001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

The following settings have to be written if sensor 0 and sensor 1 are used (data from sensor 0 will be saved in EEPROM 0x2B and sensor 1 data will be saved in EEPROM 0x2C)

Content	#	Hexa	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Configuration	09	3071	0	0	1	1	0	0	0	0	0	1	1	1	0	0	0	1
source start address	0A	0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
destination start address	0B	002B	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	1
processing length	0C	0002	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

The following settings have to be written if sensor 0, sensor 1 and sensor 2 are used (data from sensor 0 will be saved in EEPROM 0x2B, sensor 1 data will be saved in EEPROM 0x2C and sensor 2 data will be saved in EEPROM 0x2D)

Content	#	Hexa	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Configuration	09	7071	0	1	1	1	0	0	0	0	0	1	1	1	0	0	0	1
source start address	0A	0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
destination start address	0B	002B	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	1
processing length	0C	0003	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Then the settings of the sensor are to be written in the MLX90129 memory. Care should be taken to set the bits 4, 5, 6 of the sensor control word (address 0x15, 0x1B, 0x21) as shown in the following example for internal temperature sensor:

Content	#	Hexa	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Control word	15	C070	1	1	0	0	0	0	0	0	0	**1**	**1**	**1**	0	0	0	0
Low threshold	16	0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
High threshold	17	0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Conditioner config.	18	0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Connection config.	19	0231	0	0	0	0	0	0	1	0	0	0	1	1	0	0	0	1

Resistance network	1A	8000	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

4.2. Reading protocol

Once programmed it is necessary to follow a defined protocol in order to read the sensor value. The sensor values available in the MLX90129 memory (address 0x2B, 0x2C, 0x2D in the previous example) are updated at each reset of the sensor tag. The tag is reset by switching off the electromagnetic field of the RFID reader. After a reset the tag has to be supplied long enough to perform the conversion of the sensors. During this time the electromagnetic field of the reader should remain switch on continuously. To ensure that sensor data have been correctly updated it is advised to overwrite previous data with data 0xFFFF. The following table summarizes the protocol.

Actions	RFID commands	Comments
Field ON		The MLX90129 is supplied by RF field
Write EEPROM 0x2B	43212BFFFF crc	Overwrite the previous sensor data
Field OFF		The MLX90129 is reset
Wait at least 10 ms		
Field ON		The MLX90129 is supplied by RF field the datalogging starts
Wait at least the conversion time (to be adapted depending the sensor configuration)		
Read EEPROM 0x0B	03202B crc	sensor 0 value

4.3. Example with the internal temperature sensor

In the following example, the whole sequence of commands sent to configure and read a MLX90129 configured to work as a temperature sensor tag with a weak field detection are described.
The internal temperature sensor configuration is implemented in sensor 1 and the weak field detector is implemented in sensor 0 and 2 in order to check the field strength before and after the conversion of the internal temperature sensor. We assume that the weak field threshold have been saved in the EEPROM at the address 0x2A.

Configuration: (executed only once)

Actions	RFID commands	Comments
Field ON		The MLX90129 RFID communication is switched on
DMA and Power Configuration (MLX90129 EEPROM)		
Write EEPROM #09 with 0x7071	4321097170 crc	Datalogging with the 3 sensors
Write EEPROM #0A with 0x0000	43210A0000 crc	
Write EEPROM #0B with 0x002B	43210B2B00 crc	Sensor data are saved in 0x2B, 0x2C and 0x2D
Write EEPROM #0C with 0x0003	43210C0300 crc	3 data are saved
Write EEPROM #12 with 0x20FF	432112FF20 crc	Set power settings
Sensor 0 Configuration with weak field settings (MLX90129 EEPROM)		
Write EEPROM #15 with 0x0070	4321157000 crc	Sensor 0 configuration with weak field settings (cf 3. Weak field detection)
Write EEPROM #16 with 0x0000	4321160000 crc	
Write EEPROM #17 with 0x0000	4321170000 crc	
Write EEPROM #18 with 0x0000	4321180000 crc	
Write EEPROM #19 with 0xE000	43211900E0 crc	
Write EEPROM #1A with 0x0000	43211A0000 crc	
Sensor 1 Configuration with Internal temperature settings (MLX90129 EEPROM)		
Write EEPROM #1B with 0xC070	43211B70C0 crc	Sensor 1 configuration Internal temperature sensor settings (cf EVB90129 user manual)
Write EEPROM #1C with 0x0000	43211C0000 crc	
Write EEPROM #1D with 0x0000	43211D0000 crc	
Write EEPROM #1E with 0x0000	43211E0000 crc	
Write EEPROM #1F with 0x0231	43211F3102 crc	
Write EEPROM #20 with 0x8000	4321200080 crc	
Sensor 2 Configuration with weak field settings (MLX90129 EEPROM)		
Write EEPROM #21 with 0x0070	4321217000 crc	Sensor 0 configuration with weak field settings (cf 3. Weak field detection)
Write EEPROM #22 with 0x0000	4321220000 crc	
Write EEPROM #23 with 0x0000	4321230000 crc	
Write EEPROM #24 with 0x0000	4321240000 crc	

Write EEPROM #25 with 0xE000	43212500E0 crc	
Write EEPROM #26 with 0x0000	4321260000 crc	

Initialization sequence: (executed before each measurement session)

Actions	RFID commands	Comments
Field ON		The MLX90129 RFID communication is switched on
Read weak field threshold		
Read EEPROM #2A	03202A crc	Get the threshold value of the weak field detector
Initialize sensor value area		
Write EEPROM #2B with 0xFFFF	43212BFFFF crc	Initialize sensor 0 EEPROM value
Write EEPROM #2C with 0xFFFF	43212CFFFF crc	Initialize sensor 1 EEPROM value
Write EEPROM #2D with 0xFFFF	43212DFFFF crc	Initialize sensor 2 EEPROM value
Field OFF		The MLX90129 is reset

Sensor measurement sequence: (to be repeated continuously to get new sensor value)

Actions	RFID commands	Comments
Field ON		The MLX90129 RFID communication is switched on
Wait at least the conversion time (to be adapted depending the sensor configuration) (100 ms – 1.5 sec)		
Read EEPROM #2B	03202B crc	Get sensor 0 value
Read EEPROM #2C	03202C crc	Get sensor 1 value
Read EEPROM #2D	03202D crc	Get sensor 2 value
Write EEPROM #2B with 0xFFFF	43212BFFFF crc	Initialize sensor 0 EEPROM value
Write EEPROM #2C with 0xFFFF	43212CFFFF crc	Initialize sensor 1 EEPROM value
Write EEPROM #2D with 0xFFFF	43212DFFFF crc	Initialize sensor 2 EEPROM value
Field OFF		The MLX90129 is reset
Wait at least 10 ms		

Temperature value validation:
According to the configuration, the temperature measured with sensor 1 is valid only if sensor 0 and sensor 2 values are above weak field threshold.

5. Trouble shooting notes

In case of failure in the implementation of a sensor tag application the following list has to be checked in order to identify the issue.

<u>The MLX90129 is busy (error code 0xA1):</u>

- ❑ Check the configuration of the ADC for each sensor and estimate the conversion time. Then add delays between commands in your implementation in order to match with the conversion time.

<u>The ADC values are not updated:</u>

- ❑ Add delays between each RFID commands in order to let the time to the MLX90129 to process the conversion.

<u>The weak field detector reduces the reading range:</u>

- ❑ It is normal, the energy needed to do a sensor conversion is more important than to process a simple inventory command. In order to minimize the impact the threshold value has to be affined.

<u>The sensor data are not saved in the MLX90129 memory:</u>

- ❑ Check the configuration of the address 0x09, 0x0A, 0x0B and 0x0C to know how many sensor and where sensor data will be saved in EEPROM.
- ❑ Check that the bits 4, 5, 6 of the sensor control word (address 0x15, 0x1B, 0x21) are set to ‘1’.

6. Conclusion

This application note allows a fast implementation of the MLX90129 sensor tag applications. The settings have to be adapted according to the application. The flexibility of the MLX90129 allows using any kind of resistive sensor with any kind of ISO15693 RFID reader. More information and documentation can be found on the Melexis website, www.melexis.com.