AS321X Application Note – AN04-0
Access to sensor data

AS3213S.3 – AS3213RH.3 - AS3213L.3 - AS3213C.3
AS3212.5 – AS3211.6

Revision History

Revision	Date	Comment
2.0	2023-02-10	First version with revision history

Table of Contents

1. Description

The purpose of this document is to help users of ASYGN AS321X integrated circuits to collect and interpret sensor data. It explains how data is organized and formatted inside the memory of the IC.

This document is valid for the following references:
AS3213S.3 – AS3213RH.3 – AS3213L.3 – AS3213C.3 – AS3212.5 – AS3211.6

2. How to obtain the sensor data

First of all, it is important to note that sensor data can be obtained using a simple standard EPC Gen2 *Read* command in the USER bank at addresses 0x0 and 0x1. The memory map of this bank is shown below.

Memory map of the USER bank

Label			LogAddr	Bank	NVM mapped
USER7	0x	7	7	USER	YES
USER6	0x	6	6		YES
USER5	0x	5	5		YES
USER4	0x	4	4		YES
USER3	0x	3	3		YES
USER2	0x	2	2		YES
ACQ_TEMP	**0x**	**1**	**1**		**NO**
ACQ_SENS	**0x**	**0**	**0**		**NO**

Note – Sensor words are stored in a volatile register

Even though sensor data words are mapped into the USER bank, they are not written in the non-volatile memory (NVM) of the IC but in a volatile register. As a consequence, the sensor words are only available when the tag is powered by the RF field. Each interrogation round provides a new set of data.

3. Sensor data format

The resolution of the ADC embedded into the IC is 10 bits. Acquired sensor data is available as a 16-bit word as shown below. This format is the same for ACQ_SENS and ACQ_TEMP data.

	ADC_NSMPL	PWR_OK	ACQ_SENS / ACQ_TEMP DATA											
# of samples	15:14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	ADC_NSMPL	PWR_OK				D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
2	ADC_NSMPL	PWR_OK			D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	D-1
4	ADC_NSMPL	PWR_OK		D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	D-1	D-2
8	ADC_NSMPL	PWR_OK	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	D-1	D-2	D-3

PWR_OK (bit 13 of this sensor data word) is a status bit indicating that the power supply remained within the specified limits during the measurement, which guarantees the accuracy and precision of the measurement.

ADC_NSMPL (bits <15:14> of this sensor data word) gives the number of acquisitions performed by the ADC. This parameter is set in CONFIG1 register (CONFIG1<1:0>), available at address 0xF in the EPC bank (please refer to the datasheets of the related ICs).

ADC_NSMPL is explained in the table below:

ADC_NSMPL	Number of samples acquired and converted in a row, for averaging. '00': Only one acquisition is performed '01': 2 acquisitions '10': 4 acquisitions '11': 8 acquisitions

Depending on the value of ADC_NSMPL, used for averaging, the samples taken are summed in a binary way and written in the register.

Specifically:

- if ADC_NSMPL = 0x0 (1 acquisition), the recorded data is 10-bit long, ACQ_XXX<9:0>, because only one sample is stored in the data register.
- if ADC_NSMPL = 0x1 (2 acquisitions), the recorded data is 11-bit long, ACQ_XXX<10:0>, because 2 successive samples are added in a binary sum and stored in the data register.
  Similarly:
- if ADC_NSMPL = 0x2 (4 acquisitions), the recorded data is 12-bit long, ACQ_XXX<11:0>,
- if ADC_NSMPL = 0x3 (8 acquisitions), the recorded data is 13-bit long, ACQ_XXX<12:0>.

Each time the ADC performs an acquisition the data is summed with the previous one.
It is therefore possible to reduce noise by averaging as explained below:

- For 1 acquisition, take ACQ_XXX<9:0> and divide it by 1 (2^ADC_NSMPL --> 2^0)
- For 2 acquisitions, take ACQ_XXX<10:0> and divide it by 2 (2^ADC_NSMPL --> 2^1)
- For 4 acquisitions, take ACQ_XXX<11:0> and divide it by 4 (2^ADC_NSMPL --> 2^2)
- For 8 acquisitions, take ACQ_XXX<12:0> and divide it by 8 (2^ADC_NSMPL --> 2^3)