



Your Global Automation Partner

ASYGN AS3213T tags

Demo application – UHF temperature sensing tags

Hendrik Schnabel

Introduction

This document will provide the basic fundamentals of the UHF RFID sensor transponder and mainly cover the following topics:

- Types of available IC / Solutions
- Introduction of demo tools
 - ARGEE V3 demo
 - CODESYS V3.5 SP16

It will also contain ideas on potential applications, but no success stories or real world examples.

If i can be of assistance or you want to provide feedback to this document or technology, please do not hesitate to contact me.



Hendrik Schnabel

**Product Manager UHF RFID
systems**

+49 151 16773699

Hendrik.Schnabel@turck.com

AS3213T – Temperature

Key facts

Below some key facts about the temperature sensing element.

Parameter	Min	Typ	Max	Unit
Operating temperature range	-40		+125	°C
Storage temperature range	-70		+250	°C
Resolution		± 0.35		
Measurement time		140*		µs
Typ. accuracy (after calibration)		± 2**		°C

The sensor itself is capable to operate even in higher temperature environments than standard IC. Most standard IC on the market are only capable of maximum +80/+85°C operation temperature.

* A typical read operation of the tag itself lasts ~10ms, so compared to a standard read operation of an EPC, the duration of the measurement does not affect the total duration of a reading cycle!

** Accuracy can be recalibrated, but the operating temperature range than changes from



AS3213T

Temperature

The sensor IC is made by the company **asygn**, which is based in France (Grenoble). Their key focus is electronic system design for RF but also general sensor interfaces as well as RFID sensing IC.

You will find more information about the company on their website: www.asygn.com and the product itself in the attached PPT about the use cases.

Samples for initial tests can be provided via the product management – even if hard tags through our main tag supplier are available, the label tags (from identitytag) are easier to get.

Memory structure – UHF RFID temperature sensing

EPC memory bank 0x01:

EPC (up to 24bytes)																								ACQ				CONFIG						
																								Sens	Temp		0	1		2				
Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33
Content	64	21	00	00	00	00	00	00	00	00	cf													23	2a	0F	86	18	00	10	07	00	00	
Word	1	2		3		4		5		6		7		8		9		10		11		12		13	14		15		16		17			

Note: Word 0 of EPC contains „Stored CRC“ (byte 0) and „Stored PC“ (byte1)

USER memory bank 0x03:

ACQ				USER											
Sens		Temp		USER										CALIB_ACQ_TEMP	
Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13	
Content	23	2a	0F	86	00	00	00	00	00	00	00	00	0E	E3	
Word	0		1		2		3		4		5		6		

Process to read and calculate the sensor data:

Step 0 (optional): Read EPC with „Inventory“ command

Step 1: Read *USER* memory

- Word 1: ACQ Temp
- Word 6: CALIB_ACQ_TEMP

1. Calculation of the offset correction: **CALIB_OFFSET = 3860.27 – CALIB_ACQ_TEMP**

For example, if CALIB_ACQ_TEMP[12:0] = 0x0EE3 --> 3811 in decimal

→ CALIB_OFFSET = 3860.27 – 3811 ~ 49.27

Step 2 : Conversion of the sensor data in temperature [°C]

2. **ACQ_TEMP_CORRECTED = (ACQ_TEMP + CALIB_OFFSET) / 8**

3. **t° (in °C) = A x ACQ_TEMP_CORRECTED + B**

with A = 0.3378 and B = -133

For example, if ACQ_TEMP = 0x0F86 (10#3974) and CALIB_OFFSET = 49.27

ACQ_TEMP_CORRECTED = (3974 + 49.27) / 8 = 502.91

→ t° (in °C) = 0.3378 x 502.91 - 133 ~ 36.9 °C

Process to read and calculate the sensor data:

Step 3: Check if measurement was OK – the tag displays whether the temperature sensing process is done properly

Check if Bit 13 of ACQ_Temp is „1“:

For example:

ACQ Temp	
26	27
00100001	10101100

NSMPL	PWR_OK	Temp												
15:14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
00	1	0	0	0	0	1	1	0	1	0	1	1	0	0

1 = Power OK
0 = Power not OK

Example – Temperature sensing with ASYGN AS3213T IC

UHF reader demo setup (example)



Features and requirements

FEATURES

- Easy demo for usage of temperature sensor tags
- Read and calculate Integrated temperature sensor with RFID-U-interface
- Integrated HMI for easy demo purposes (ARGEES)
- Calculation formulas implemented in ARGEE and CODSYS V3

REQUIREMENTS

ARGEES

- „ASYGN_AS3213T_demo_V3.arg“
- ARGEE V3 programming environment
- ARGEE V3 compatible device with RFID-U-interface (*TBEN-S2-2RFID-4DXP, TBEN-Lx-4RFID-8DXP, TN-UHF-Q150-...-EN or similar*)

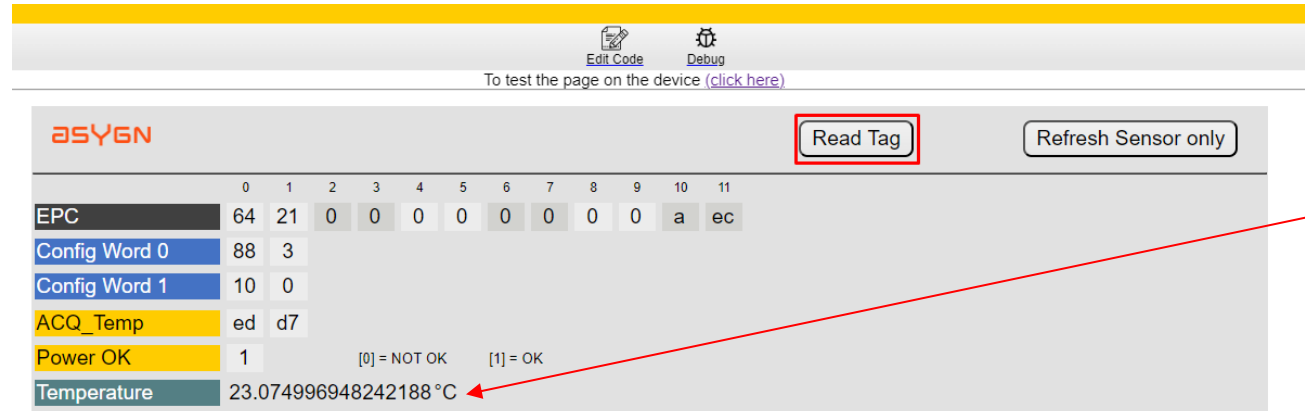
CODSYS

„Temperature_calculation_ASYGN_AS3213T.projectarchive“

- CODSYS V3.5 SP16 or later
- RFID-U-interface compatible devices (*TBEx-S2-2RFID-4DXP, TBEx-Lx-4RFID-8DXP, TN-UHF-Q150-..., TN-UHF-Qxxx-...-CDS or similar*)

ARGEE V3

NOTE: Load the ARGEE project into your device and press „RUN“



The screenshot shows the ARGEE V3 web interface. At the top, there are buttons for 'Edit Code' and 'Debug', and a link to 'To test the page on the device (click here)'. Below this, the 'asY6N' logo is visible. A red box highlights the 'Read Tag' button. To its right is a 'Refresh Sensor only' button. The main display area shows a table of sensor data with columns 0 through 11. The data is as follows:

	0	1	2	3	4	5	6	7	8	9	10	11
EPC	64	21	0	0	0	0	0	0	0	0	a	ec
Config Word 0	88	3										
Config Word 1	10	0										
ACQ_Temp	ed	d7										
Power OK	1											
Temperature	23.074996948242188 °C											

Below the table, there are two status indicators: '[0] = NOT OK' and '[1] = OK'. A red arrow points from the 'Temperature' row to the list of steps on the right.

1. „Run“ the program
2. Press „Read Tag“
3. Result displayed in „temperature“

CODESYS V3.5 SP16

NOTE: Please check all IP addresses and Profinet names ahead

The screenshot shows the CODESYS V3.5 SP16 interface. The 'Run' button is highlighted with a red box and labeled '1.'. The variable declaration table is shown with 'xStart' set to 'TRUE', highlighted with a red box and labeled '2.'. The ladder logic for temperature calculation is shown, with the 'TEMP_CELSIUS' variable highlighted with a red box and labeled '3.'.

Ausdruck	Datentyp	Wert	Vorbestimmter Wert	Adresse
xStart	BOOL	FALSE	TRUE	
PNV_OK	WORD	0		
acq_temp0	WORD	0		
acq_temp1	WORD	0		
W_Acq_TEMP	WORD	0		
calib_acq_temp0	WORD	0		
calib_acq_temp1	WORD	0		
W_CALIB_Acq_TEMP	WORD	0		
CALIB_OFFSET	REAL	0		
TEMP_CELSIUS	REAL	0		

```

31
32 40: IF vRES0[0] = 2 THEN
33
34   calib_acq_temp0[0] := %I2[0]
35   calib_acq_temp1[0] := %I2[0]
36   wWCD[0] := 0;
37   wLEN[0] := 0;
38   dwADDR[0] := 0;
39   iState[0] := 50;
40
41 END_IF
42
43 50: IF vRES0[0] = 0 THEN
44
45   FWR_ON[0] := (16#2000 AND W_Acq_TEMP[0]);
46
47   W_Acq_TEMP[0] := (SHL(16#000F AND acq_temp0[0], 0) + acq_temp1[0]);
48   W_CALIB_Acq_TEMP[0] := (SBL(calib_acq_temp0[0], 0) + calib_acq_temp1[0]);
49
50   CALIB_OFFSET[0] := ((3260.7 - W_CALIB_Acq_TEMP[0] + W_Acq_TEMP[0]) / 8;
  
```

1. „Run“ the program
2. Write „TRUE“ into variable „xSTART“
3. Result displayed