Application Note AN006 Sensor And Temperature Measurements

1 Introduction

Sensor tags based on Magnus®-S produce a Sensor Code which gives information about the sensor tag's environment. In addition, Magnus®-S3 chips can generate a Temperature Code indicating the chip temperature. To maximize the precision of sensor and temperature measurements, the user should understand and account for effects related to the reader transmission frequency, the amount of power received by the sensor tag, averaging, and command timing. This note describes best practices for maximizing the accuracy of measurements with Magnus®-S.

2 Sensor Code Measurements

2.1 Channel Frequency Effects

Legal regulations governing ISO 18000-6C communication forbid readers to continuously transmit on a single frequency for an unlimited time. Instead, readers typically transmit on a set of frequency channels, periodically hopping between them in a non-sequential, random-looking order.

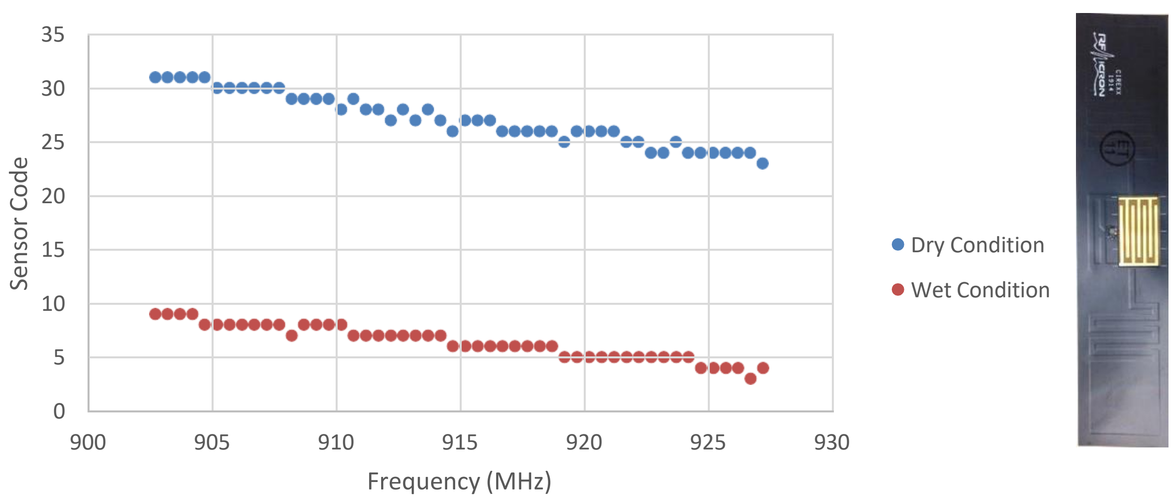

Figure 1 Sample Magnus®-S2 Sensor Codes for a moisture sensor in dry and wet conditions

The Sensor Code provided by Magnus®-S chips indicates changes in antenna impedance, which can be caused by factors the sensor tag is designed to sense, such as moisture, or the proximity of something metallic. But impedance also depends on frequency, which means that Magnus®-S can report different Sensor Codes when it is read repeatedly, as the reader changes its transmission frequency.

Typically, the Sensor Code will vary approximately linearly with frequency, and the line will shift up or down in response to a change in the sensing stimulus. For example, Figure 1 plots the measured Sensor Code as a function of frequency for a Magnus®-S2 sensor tag designed to sense moisture.

It is possible for Sensor Codes to saturate at their extreme values (0 and 31 for Magnus®-S2). It is a good idea to ignore readings at these extremes to ensure that only data within the dynamic range of the sensor are used in the measurement. Saturation is more likely for sensor tags which exhibit a Sensor Code vs. frequency plot with a large slope (Figure 2). Sensor tags designed to be placed on metal often have this feature.

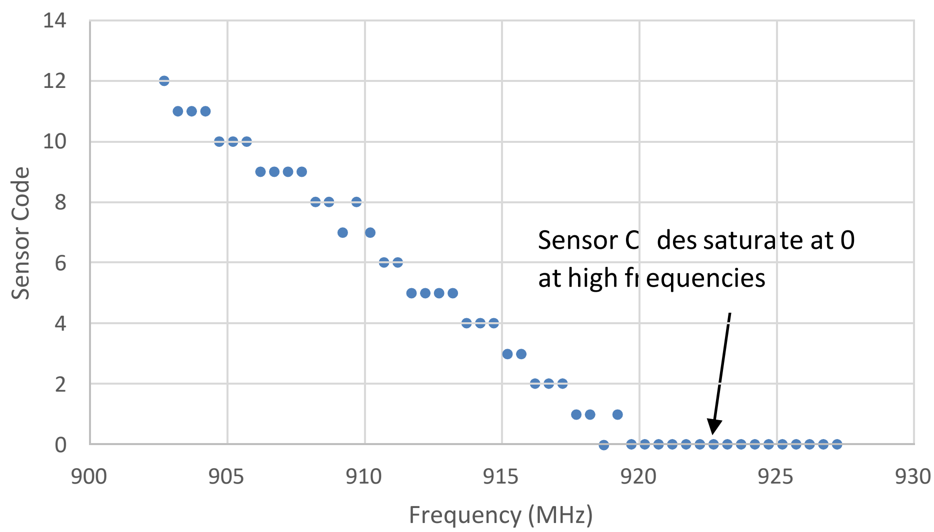

Figure 2 Example Magnus®-S2 Sensor Code results showing saturation at high frequencies

2.2 Summarizing A Set Of Sensor Code Measurements

When the Sensor Code is plotted against frequency, it is straightforward to visually recognize a change in sensor condition. However, it is often preferable to condense the data to a single number which eliminates frequency dependence and focusses entirely on the sensor environment. Combining results from different frequencies can also average out random noise and improve precision. Table 1 describes three possible approaches to dealing with frequency-dependence and reducing a series of readings to a single number.

Table 1 Techniques For Dealing With Frequency Dependence

Technique	Pros	Cons
Use Sensor Code value from one frequency only	• Simplest to implement	• Regulatory requirements prevent continuous transmission at a single frequency; compliance will limit the sample rate • Lack of averaging reduces precision.
Use the average Sensor Code value over the entire frequency band.	• Simple to implement • Averaging over frequency improves precision and reduces numerical noise	• Must collect enough data to ensure that the frequency range is adequately and evenly sampled to avoid biasing the results
Use regression analysis to fit the Sensor Codes to a line, then take the value of the line at some fixed frequency. (See Appendix for details.)	• Regression process improves precision and reduces numerical noise. • Can achieve good results even when sampling only a fraction of the frequencies in the band	• More complex to implement.

Note that different regulatory regimes have significantly different numbers of frequency channels in them. For example, in North America there are 50 frequency channels, each 500 kHz apart, between 902 and 928MHz. Under the European ETSI EN 302 208 specification, there are only 4 channels between 865 MHz and 868 MHz. So the time required – and precision gained – by reading the Sensor Code at every channel before producing a result depends significantly on regulatory requirements.

2.3 Power Distortion

When a Magnus®-S chip is receiving a low amount of power, the Sensor Code it generates is fairly independent of the precise power level. Once the received power increases beyond a certain threshold, the Sensor Code is distorted by the excess power: higher power levels produce artificially low Sensor Code values. Figure 3 shows a sample plot of the relationship between power and the Sensor Code value, averaged over frequency. (Keep in mind that power received by the sensor tag depends on many factors such as distance and antenna gain, not just reader output power.)

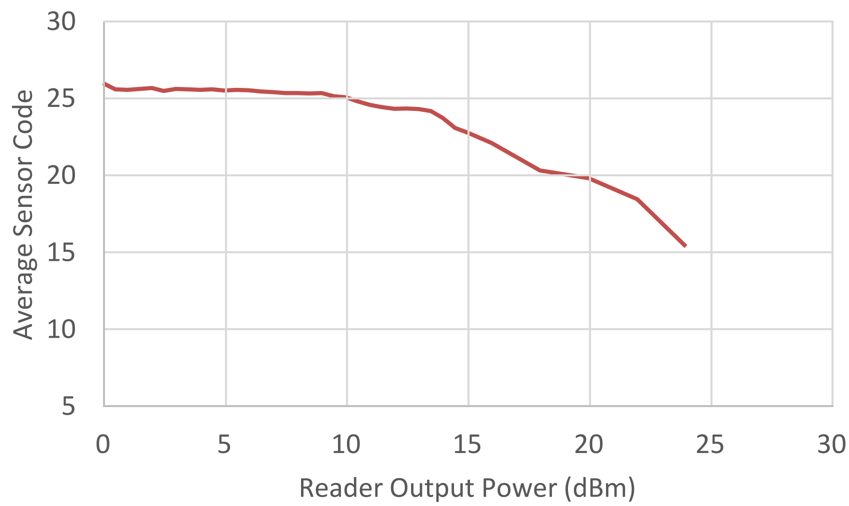

Figure 3 Sample Sensor Code vs. reader output power plot for a Magnus®-S2 sensor tag

For applications using low-gain antennas, low-power readers, large minimum separations between sensor tag and reader, or when high sensor precision is not needed, this effect may not be a concern. But in some cases, it will be desirable to ensure that the sensor tag does not receive enough power to avoid distortion. This can be achieved readily by making use of the On-Die RSSI Code.

The On-Die RSSI Code is a 5-bit value (0-31) which can be read from Magnus®-S and gives an indication of the amount of power it is receiving. Larger values correspond to higher power. If the On-Die RSSI Code is above the recommended upper threshold given in Table 2, the reader power should be reduced to avoid affecting the Sensor Code.

It is also desirable to avoid delivering very low amounts of power to Magnus®-S, mainly because this increases the chance of reading at some frequencies but not others. This is more likely to occur with readers and/or antennas which exhibit non-uniform radiated power across frequency. In some cases, very low power may also pull the Sensor Code lower, but by a maximum of only about 1 code value. If the On-Die RSSI Code is below the recommended lower threshold given in Table 2, the reader power should be increased, if possible.

Table 2 Recommended Ranges For On-Die RSSI Code When Taking Sensor Measurements

Recommended On-Die RSSI Values	Magnus®-S2	Magnus®-S3
Upper threshold (to avoid affecting Sensor Code)	21	21
Lower threshold, if achievable (to reduce the chance of missed reads)	16	13

In some applications, the power received by the sensor tag can be kept fairly constant (by fixing the placement of the sensor tag and reader and controlling interference in the transmission path). In those cases, the reader power can be preset to a level which achieves the codes in Table 2 and held constant. But often, the reader will be programmed to search automatically for a desirable power level and periodically adjust itself to account for changes in the environment that affect received power.

As noted earlier, higher On-Die RSSI Codes correspond to more received power, up to a maximum code of 31. However, if Magnus®-S is receiving very large amounts of power, the reader may fail to communicate properly with the tag. This should only occur when the reader is transmitting at or near the upper EIRP limit allowed by regulations, and only when the sensor tag is within a few feet of the reader. But for this reason, when searching for a desirable power level, the reader should start at a lower power and increase if necessary, rather than beginning at maximum power.

3 Temperature Measurements

Magnus®-S3 chips generate a 12-bit Temperature Code which can be translated into a measurement of the temperature of the chip. As with the Sensor Code, achieving accurate temperature readings requires consideration of a few factors.

3.1 Calibration

To account for manufacturing variability, Magnus®-S3 chips come with single-point calibration data pre-stored in the User memory bank, which is individually determined for each chip. By reading and applying this data, the user can translate the 12-bit Temperature Code into a temperature in degrees C. More information on the calibration data format is available in Application Note AN002.

3.2 Averaging

Unlike the Sensor Code, the Temperature Code does not depend on channel frequency. However, it does contain some inherent noise which can be reduced by averaging multiple readings. Table 3 gives the approximate number of samples that need to be averaged in order to obtain a particular confidence interval for the Temperature Code.

Table 3 Temperature Code Confidence Intervals

Temperature Code 95% Confidence Intervals	Number of Samples
+/- 3	1
+/- 2	3
+/- 1	9
+/- 0.5	35
+/- 0.25	139

Note that Table 3 refers to Temperature Code values, not actual temperatures. Temperature precision depends on multiple factors in addition to Temperature Code noise. In approximate terms, the Temperature Code will change by about 7.5 for every 1 degree C change in temperature.

3.3 Command Timing

The temperature sensor circuit is activated by a specific Gen2 UHF Select command (See Application Note AN002 for more details). After sending this Select command, the reader should emit a continuous wave for 3 ms. This will provide low-noise power for the temperature circuit and will maximize measurement accuracy. After the 3 ms of continuous wave, the commands for accessing the tag memory can be issued to retrieve the Temperature Code.

3.4 Power Distortion

As with the Sensor Code, the power received by the tag can affect the Temperature Code, so the user should always check the received power when taking readings. The amount of power the chip is receiving can be readily determined by reading the On-Chip RSSI Code (See Application Note AN002 for more information).

At very low received powers (On-Chip RSSI of 5 or lower) the Temperature Code is not reliable. At high received power levels, the Temperature Code can indicate a temperature higher than the actual value. The most accurate Temperature Code readings are available when the On-Chip RSSI Value is between 13 and 18, as indicated in Table 4.

Table 4 On-Chip RSSI Values For Temperature Measurements

Absolute Minimum	Preferred Minimum	Preferred Maximum
5	13	18

4 Notices