# Windowing Functions Comparison for Peak Detection of Amplitud Coded SAW ID-tags

Luis C. Diaz, Ernesto A. Rincón, Sebastian Velasquez,
Juan C. Bohórquez, Fredy E. Segura, Nestor Peña and Luis E. Muñoz.
Department of Electrical and Electronic Engineering
{lc.diaz12, ea.rincon, se-velas, jubohorq, fsegura, npena, lui-mono}@uniandes.edu.co
Universidad de los Andes
Bogotá, Colombia

***Abstract*—This paper presents an interrogation scheme for passive RFID SAW tags. In such interrogation scheme, the filtering stage is critical to achieve the identification. To evaluate this stage performance, different windowing functions are applied to the tag's response signal. This stage performance represents a higher efficiency in the tag's code identification. This efficiency is evaluated by two parameters: noise floor level and the peak amplitude from the reflections. This work presents the comparison between several windowing functions. These windowing functions were applied to the amplitude and phase response obtained from a vector network analyzer that was used to interrogate the tag. To avoid any spurious interference, all tests were performed in an anechoic chamber. These windowing functions were implemented in $MATLAB$®.**

## I. Introduction

RFID is a widely used technology. Inside this technology, SAW RFID tags are devices that contain a unique code encrypted in a pattern of reflectors on a piezoelectric substrate. The code is obtained by identifying the time and amplitude of each reflection due to the different reflectors. This scheme is presented in Figure 1 [1].

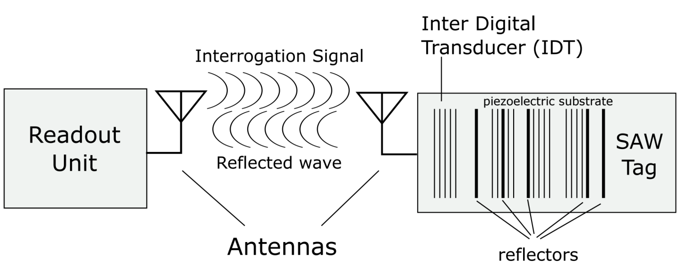


Fig. 1. General Scheme

To obtain such reflections, a readout unit generates an interrogation RF signal is transmitted through a reading antenna. This signal travels to the receiving antenna of the SAW tag, which converts the electric interrogation unit into a surface acoustic wave through a transducer. This acoustic wave travels through the reflectors, and is returned back to the transducer. This signal already contains the information of the reflectors, and it is converted to an electric signal by the transducer. Finally, the antenna transmits the response signal to the readout unit.

There are two main interrogation schemes used in this interrogation units: time and frequency domain sampling schemes. Both are widely used, but time sampling requires of high speed platforms, which are commonly expensive [2]. On the contrary, frequency approaches can be implemented in low cost radio based platforms. The most common frequency sampling scheme is FSCW (Frequency Stepped Continuous Wave), as shown in Figure 2 [3].

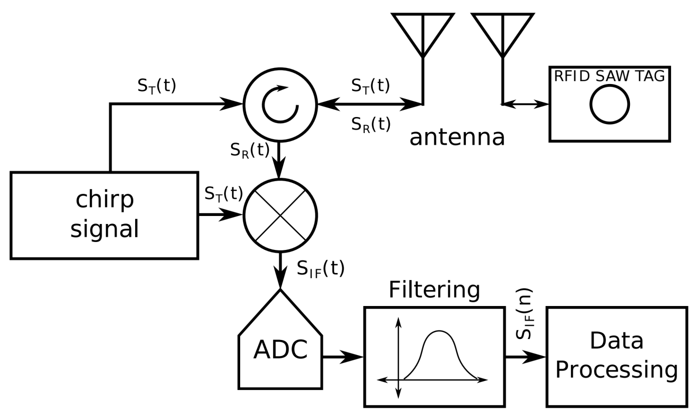


Fig. 2. Architecture for a readout unit

Although the reflectors are time encoded, it is possible to identify the position in time of each reflector by detecting the frequency difference between the interrogation signal and the reflected signal. Based on this, the main blocks of the interrogation unit are: a frequency generator a mixer that synthesizes the received signal, a low pass filter that removes the high frequency component generated by the mixer and a digitizer. The signal of the digitizer is passed to a data processing module which retrieves the code. According to the scheme presented in Figure 2, the interrogation signal is given by:

$$S_T(t,n) = A_T \cdot cos(2\pi(f_0 + \frac{K}{2} \cdot n)t) \tag{1}$$

Where, $f_0$ is the initial frequency and $K/2$ corresponds to the frequency step $\Delta F$, (see Figure 3).

Copyright © 2015 for the individual papers by the papers' authors. Copying permitted for private and academic purposes. This volume is published and copyrighted by its editors. *Latin American Workshop On Communications*' 2015 Arequipa, Peru Published on CEUR-WS: http://ceur-ws.org/Vol-1538/

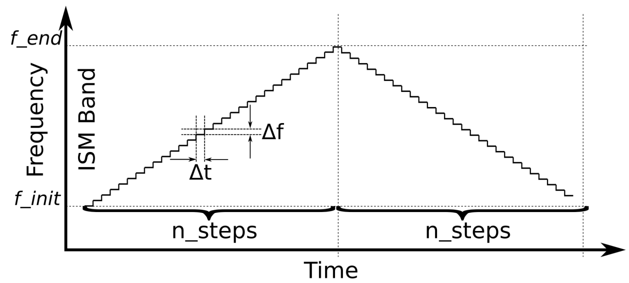


Fig. 3. FSCW Interrogation Signal

After going through the mixer, the filter and the analog to digital converter, the intermediate signal, which is in the sample domain is given by [4]:

$$S_{IF}(n) = \sum_{i=1}^{p} A_i \cdot cos(2\pi f_i n + 2\pi f_0 \tau_i) \tag{2}$$

Where $p$ is the number of reflectors, $A_i$ and $\tau_i$ are, respectively, amplitude and time delay associated to each tag reflector. Because the sample domain is analogue to the frequency domain, the received signal, $R[n]$, was obtained by multiplying $S_{IF}(n)$ by $H[n]$, which is the sample domain response of the window, as shown in Equation 3.

$$R[n] = S_{IF}(n) \cdot H[n] \tag{3}$$

The intermediate signal, $S_{IF}(n)$, can be obtained through the reflection parameter received through one of the ports of a vector network analyzer (VNA). A reader antenna is connected to this port and the S11 parameter is extracted. The response of the tag loads the frequency response of the antenna, which resembles the operation of the FSCW scheme. CTR SAW tags were the RFID tags used for these tests.

Basically, what the VNA does is generate a chirp signal, returning phase and amplitude measures of the reflected signal by the single-port network. Sweeping in a frequency points collection, will feed the same information searched by the architecture depicted in figure 2.

The data processing module may present different structures [4], but one of the most studied processing schemes is the inverse Fourier transform. If the inverse Fourier transform is applied to the signal, it is possible to obtain the time location of each reflector.

These system implemented is shown in figure 4. In order to reduce external noise sources, tests were made in an anechoic chamber using a R&S®$ZVB$ Vector Network Analyzer, a saw tag from CTR [5] is interrogated using a circular polarized antenna.

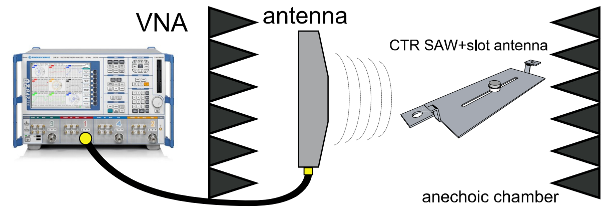


Fig. 4. VNA based readout unit

This paper is organized as follows: Section II presents the digital data processing stage, by considering many types of windowing filters. Section III presents a comparison between the implementation of these filters. Finally, Section IV presents a conclusion on these results.

## II. Digital Data Processing

Each data extraction at the vector network analyzer had the following parameters: 2500 points in the $2.25GHz$ up to $2.65GHz$ range. The port's power was set at $10dBm$. Data from the analyzer was extracted through text files that contained the magnitude($P$) and phase($\phi$) of the S11 parameter per frequency($F$) point. $S_{IF}$ was obtained as follows:

$$S_{IF}[n] = A[n]e^{\frac{j\phi[n]}{180}}[V] \tag{4}$$

where:

$$A[n] = 10^{\frac{P[n]}{20}}[V]$$

Different $R[n]$ signals were obtained by applying several windowing functions $H[N]$ to $S_{IF}(n)$. $R[n]$ is extended by performing zero padding, and by applying the Inverse Fast Fourier Transform (IFFT) [4] the time response was obtained. This time response contains the information of the reflections of the tag. To evaluate the efficiency of the window, the $R[n]$ signal for different windowing functions is presented. The experimental setup used in this work is presented in Figure 5. The distance between the antenna and the tag is $20cm$, and losses due to cables and propagation path are included.

## III. Digital Processing and Windowing Function Comparison

Unfortunately, using the discrete fourier transform in both directions in digital processing techniques produces the Gibbs phenomenon. This phenomenon results from the frequency discontinuities of the spectrum obtained through the instrument, which generate time oscillations that result in inaccurate tag information [6].

To reduce the effect of the Gibbs phenomenon, windowing techniques are used. Windowing functions modify the amplitude of the signal to be processed in some portions of the overall data vector [7]. This is why it is necessary to apply windowing functions to the intermediate signal $S_{IF}$. Hence, windowing increases the accuracy of the code detection; thus, the code detection efficiency.

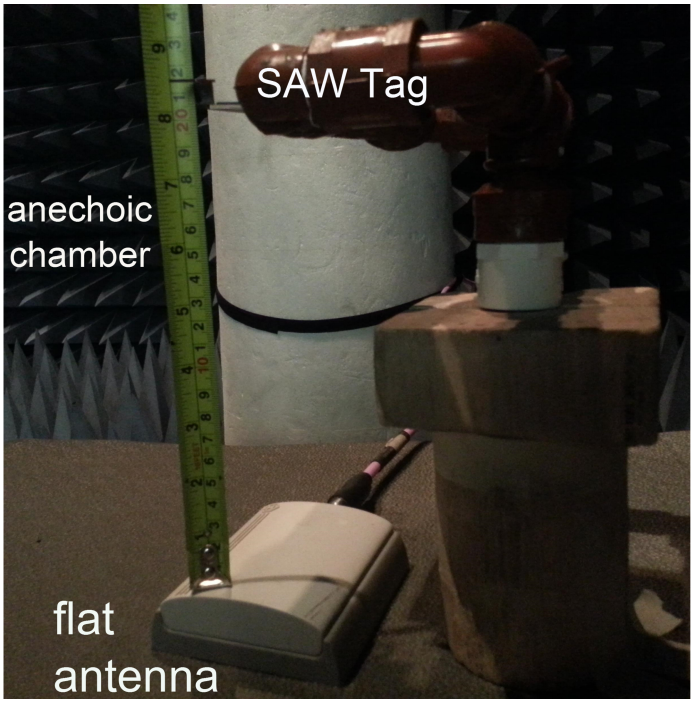


Fig. 5. Anechoic chamber at Universidad de los Andes

### A. Theoretical Windowing Function Comparison

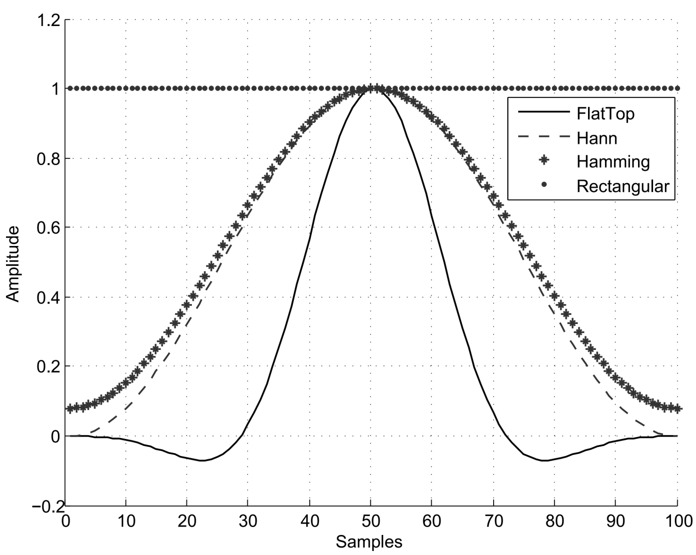


Fig. 6. Windowing Functions. Comparison in Time Domain.

Several windowing functions result from the combination of three parameters: the application restrictions, filtering level and frequency range [8]. This windowing represents a higher amplitude of the reflection peaks and a lower noise level in $R[n]$ if selected correctly.

Commonly, these applications use Hann and Hamming windowing functions. These functions are obtained by adding one period of a cosine signal to a fixed frequency to a rectangular signal. The comparison between the Han, Hamming and rectangle signals in the time domain are depicted in Figure 6. As it can be seen, Hann , Hamming and Flat top windows provide component attenuation at the border while preserving the signal information in the center of the signal. Additionally, Flat Top and Hann offer a smooth transition to zero at the window border, while Hamming has a discontinuity ”slam to zero” at borders. The rectangular window has a special behavior compared to the other function. Its response is constant until it reaches an abrupt transition at the border of the window [9].

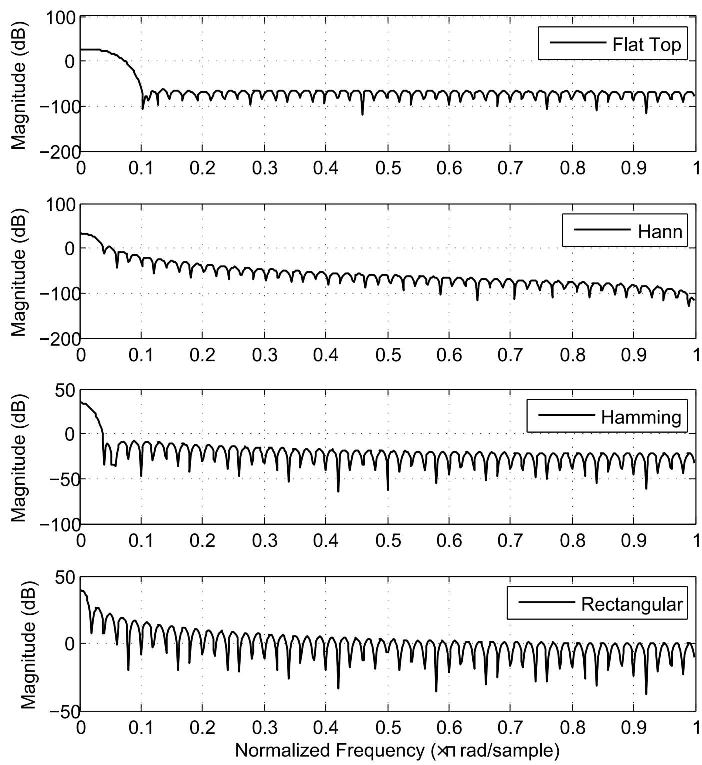


Fig. 7. Windowing Functions. Comparison in Frequency Domain. Window length=100

To select a windowing function, the frequency response and the roll-off parameters of the function must be analyzed. While the frequency response defines the bandwidth of the filter, the roll off parameter evaluates the transition in narrow band response. This eases the separation of near frequency components. According to Figure 7.

The Hann window has a Roll-off approximately -18 dB per octave, while the Hamming has approximately -6 dB per octave Roll-off. The first side lobe for each of the windowing functions are the following: 2.337dB for Hann function, while Hamming first side lobe is = -11.2dB and Flat Top first side lobe is = -80dB. Consequently, flat top function has a greater rejection compared to Hann and Hamming windows. Hamming and flat top have side lobes that tend to ”equal ripple”, while the Hann window has a variable response that depends on frequency changes. Therefore, flat top window has a higher and more stable response over the whole frequency range compared to the other analyzed windowing functions [10].

Regarding the rectangular windowing function, this function presents the lowest performance of all the analyzed functions.

### B. Practical Windowing Function Comparison

To perform a quantitative comparison of the windowing functions to evaluate efficiency, a $MATLAB$® algorithm that calculates the maximum of the IFFT response was implemented. Each peak corresponds to a code digit of the SAW tag. The MATLAB algorithm obtains the "Peak Amplitude [dB]", which is calculated according to equation 5. The threshold values obtained by the algorithm are illustrated in Figure 8.

$$PA = MP - (MV + SD)[dB] \tag{5}$$

Where:
PA:Peak Amplitude
MP:Maximum Peak
MV:Median Value
SD:Standard Deviation

| Windowing | Median Value (dB) | Standard Deviation (dB) | Maximum Peak (dB) | Peak Amplitude (dB) |
|---|---|---|---|---|
| flattopwin | -191 | 8,76 | -164 | 18.6 |
| triang | -188 | 8.3 | -162 | 18.2 |
| bartlett | -189 | 8.33 | -162 | 18.2 |
| blackman | -189 | 8.78 | -162 | 18.2 |
| bohmanwin | -189 | 8.83 | -162 | 18.2 |
| parzenwin | -189 | 8.83 | -162 | 18.2 |
| nuttallwin | -189 | 8.88 | -162 | 18.1 |
| blackmanharris | -189 | 8.91 | -162 | 18.1 |
| hann | -188 | 8.49 | -161 | 18 |
| barthannwin | -188 | 8.52 | -162 | 18 |
| tukeywin | -186 | 7.89 | -160 | 17.5 |
| chebwin | -187 | 8.16 | -162 | 17.2 |
| gausswin | -174 | 6.83 | -159 | 8.32 |
| hamming | -169 | 6.93 | -157 | 5.31 |
| taylorwin | -156 | 6.83 | -147 | 1.62 |
| kaiser | -148 | 6.96 | -139 | 1.43 |
| rectwin | -147 | 6.97 | -139 | 1,43 |

TABLE I
COMPARISON RESULTS BETWEEN DIFFERENT WINDOWING FUNCTIONS, APPLIED TO SAME SAW ID-TAG DATA VECTOR.

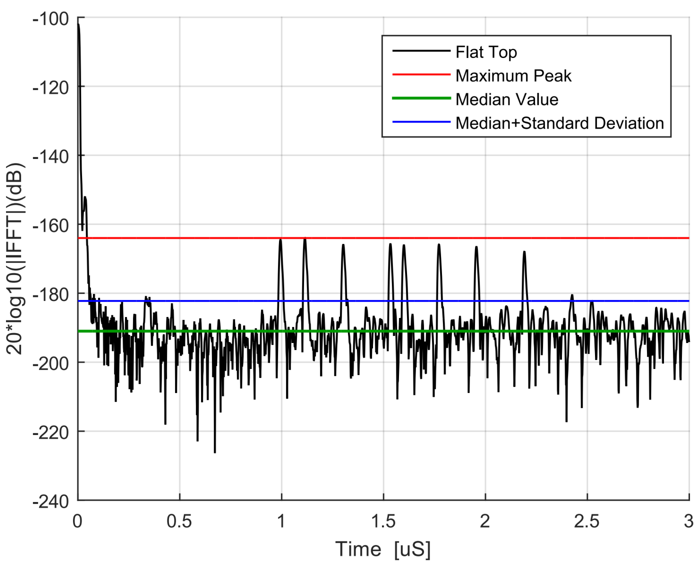


Fig. 8. Interest Points result of "Peak Amplitude (dB)" calculation over SAW ID-Tag Data processed by Flat Top windowing function. Maximum Peak=-164 dB, Median Value=-191 dB, Standard Deviation=8.76 dB, Peak Amplitude=18.6 dB

The comparison of the peak amplitude for each window are presented in Table I. As mentioned previously in the initial theoretical analysis, the best windowing function is the flat top function while the poorest is the rectangular function. The graphical comparison for different $R[n]$ signals that result from different windowing functions are presented in figures 9 to 11

As seen in these figures, it is not possible to detect the tags reflections from a rectangular windowing function. The rectangular function is not capable of treating the error introduced by a finite Fourier transform when seeking for the time response. Regarding the Hamming function, it does not allow for an adequate code detection. A slight difference of 2dB compared to the noise level may not be accurately detected by conventional platforms used for code retrieval. However, the flat top and Hann windowing functions are more adequate for this purpose. The evident difference of 20dB is enough for latter code detection. It is interesting to notice how this specific application presents such a sensitivity to the windowing function definition. In this particular case, the flat top window is should be selected for adequate code identification.

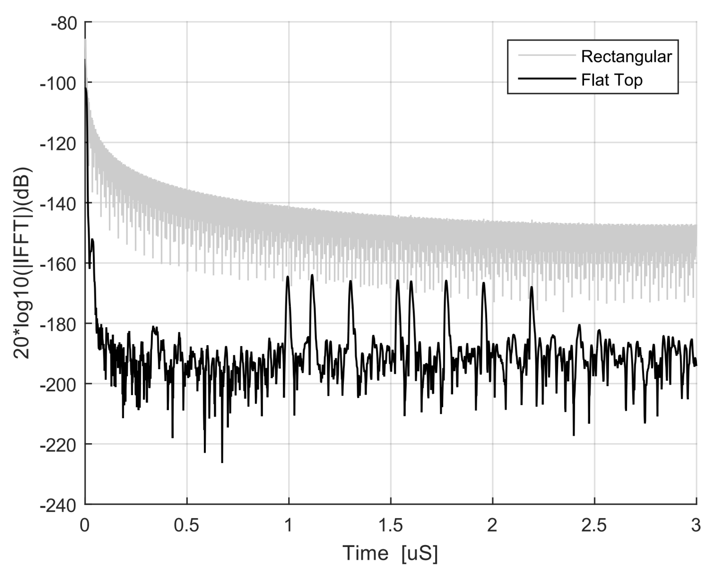


Fig. 9. Flat Top Vs Rectangular Window Functions Comparison over Reflections Data.

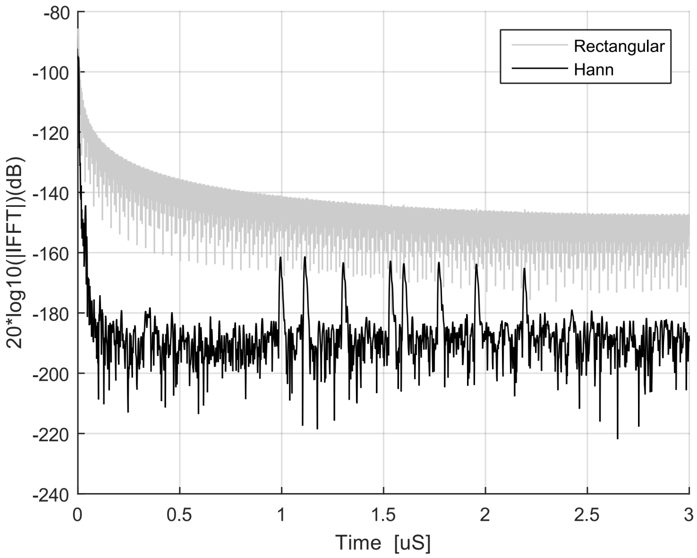


Fig. 10. Hann Vs Rectangular Window Functions Comparison over Reflections Data.

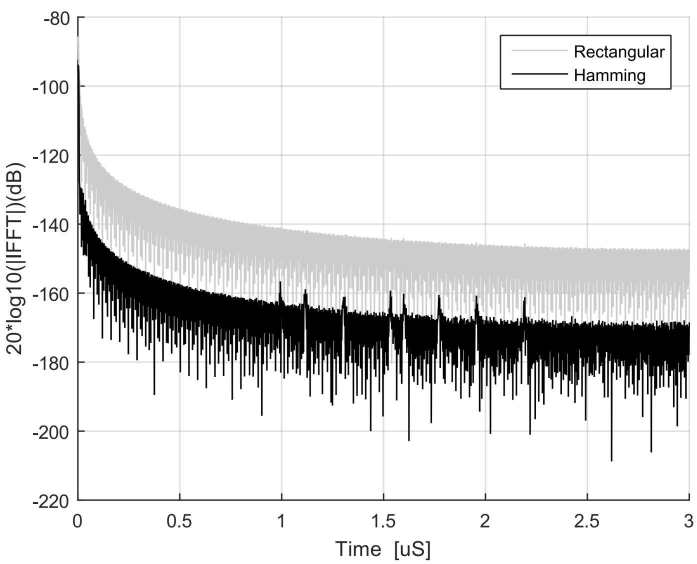


Fig. 11. Hamming Vs Rectangular Window Functions Comparison over Reflections Data.

## IV. Conclusions

By considering the FSCW scheme, and using a vector network analyzer to obtain the raw data from a SAW tag, different windowing functions were compared. Efficiency was defined as the peak amplitude level achieved by each of the functions. The flat top window presents the higher efficiency, which represents an enhancement of the SNR level when performing a code interrogation. By using this windowing function in digital processing units at the interrogation scheme, it is possible to easily identify the code by selecting an adequate threshold level at the processing unit.

## V. Acknowledgment

We acknowledge B. Moreno, F. De Milleri and H. Rojas for their assistance, technical discussion and analysis in this work. This project was funded by COLCIENCIAS, CODENSA S.A E.S.P and Universidad de los Andes according to the RC. No. 0582-2013 agreement.

## References

[1] A. Stelzer, G. Bruckner, L. Maurer, L. Reindl, R. Teichmann, and R. Hauser, "A Low-Cost Interrogation Unit and Signal Processing for a SAW-Identification Tag for a Pressure Sensor," in *XVII International Measurement Confederation World Congress, IMEKO 2003*, 2003, pp. 22–27. [Online]. Available: http://www.imeko.org/publications/wc-2003/PWC-2003-TC4-051.pdf

[2] A. Stelzer, S. Schuster, and S. Scheiblhofer, "Readout unit for wireless SAW sensors and ID-tags," in *International Workshop on SiP/SoC Integration of MEMS and Passive Components with RF-ICs*, 2004, pp. 37–43. [Online]. Available: http://www.te.chiba-u.jp/ ken/Symp/Symp2004/PDF/1B4.PDF

[3] S. Scheiblhofer, S. Schuster, and A. Stelzer, "Modeling and performance analysis of saw reader systems for delay-line sensors," *Ultrasonics, Ferroelectrics, and Frequency Control, IEEE Transactions on*, vol. 56, no. 10, pp. 2292–2303, October 2009.

[4] A. Stelzer, M. Pichler, S. Scheiblhofer, and S. Schuster, "Identification of SAW ID-tags using an FSCW interrogation unit and model-based evaluation," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 51, no. 11, pp. 1412–1420, Nov. 2004.

[5] CTR, *RadFIT Reader Units*, 2nd ed., CTR Carinthian Tech Research AG, www.ctr.at, 1 2009.

[6] C. Pan, "Gibbs phenomenon removal and digital filtering directly through the fast Fourier transform," *IEEE Transactions on Signal Processing*, vol. 49, no. 2, pp. 444–448, Feb. 2001.

[7] R. W. Hamming, *Digital filters*. Courier Corporation, 1989.

[8] K. R. Rao, D. N. Kim, and J. J. Hwang, *Fast Fourier Transform-Algorithms and Applications*. Springer Science & Business Media, 2011.

[9] J. O. Smith, *Spectral audio signal processing*. W3K, 2011.

[10] F. J. Harris, "On the use of windows for harmonic analysis with the discrete fourier transform," *Proceedings of the IEEE*, vol. 66, no. 1, pp. 51–83, 1978.