

## Reconfigurable software-defined radar testbed with built-in validation

Juan Carlos Martínez Quintero, Edith Paola Estupiñán Cuesta, Johan Stiven García Ramírez

Telecommunications Engineering Program, Universidad Militar Nueva Granada, Sade Calle 100, Bogota, Colombia

---

### Article Info

#### Article history:

Received Jun 23, 2021

Revised Dec 1, 2021

Accepted Feb 21, 2022

---

#### Keywords:

Continuous-wave radar  
Micro-doppler  
SDRadar characterization  
Software defined radar  
Testbed

---

### ABSTRACT

This article presents a radar testbed for speed detection through micro-doppler effect in a controlled environment using software defined radio (SDR) technology. The target moves along a conveyor belt with software-controlled speed. The speed is detected by an SDR radar, and it is possible to compare it to an encoder-based sensor implemented on the testbed. The testbed as well as the SDR radar are reconfigurable and a continuous wave (CW) radar was implemented for the validation of the testbed; however, the testbed is not limited to this implementation. The testbed can be remotely operated because it includes the mechanism to move the target and control its velocity. The article shows the way in which the testbed was designed and implemented, the generation and processing of the radar signal using a limeSDR, and the validation of the radar measurements compared to the encoder-based speed sensor. The maximum speed obtained by the target in the testbed is 15.69 cm/s. Results show a difference in the speed measured with the SDRadar of no more than 5% compared to the sensor measurement. Results obtained allow characterizing the behavior of the SDR platform in the detection of low speeds.

*This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.*



---

### Corresponding Author:

Juan Carlos Martínez Quintero

Telecommunications Engineering Program, Universidad Militar Nueva Granada

Sede Calle 100, Bogotá, Colombia

Email: [juan.martinezq@unimilitar.edu.co](mailto:juan.martinezq@unimilitar.edu.co)

---

## 1. INTRODUCTION

Radar systems have been widely studied and have a fine range of applicability in different sectors of the industry. Some radar applications allow the detection of missiles, aircraft monitoring, traffic speed monitoring for the detection of a vehicle speed, among others. The use of software defined radio (SDR) for research and development of radar applications has become attractive inasmuch as the flexibilization of the system increases by the facility in the change of (radio frequency) RF parameters such as frequency and gain [1], [2]. The integration of these two technologies is known as SDRadar and it has the advantage of being able to change the architecture of the radar by means of programming, while keeping the same hardware [3], [4]. SDRadar has the facility of modifying parameters of RF functioning through software and has allowed researchers and scholars to experiment different low-cost radar techniques with the help of free software such as GNU-radio. The testbed development for experimentation in the SDRadar area is critical to discover new functionalities and to test new algorithms of detection. A testbed allows having a controlled setting for the collection and further processing of data with the advantage of having repeatable experiments and being able to reconfigure scenarios to find optimal conditions. In addition to this, data obtained by means of SDRadar can be stored, labelled, and processed with different algorithms to, further on, compare their efficiency and to determine in which condition they work better.

This article presents the design, implementation, and validation of an SDRadar testbed for the detection of the speed of a target that is integrated on the testbed. The target is fixed on the surface of a conveyor belt with speed modifiable by software. The speed of the target is low, thus, the micro-doppler effect is evidenced. The conveyor belt has a sensor that allows measuring the speed of the target alternately to the SDRadar measurement, thus permitting the validation of all detections. The testbed validation was done by means of the implementation of a continuous wave (CW) radar with which the target speed detection was carried out and it was compared to the data delivered by the sensor. A maximum difference of 5% was found between measurements delivered by the sensor and the radar detections. In addition, the testbed allows remote manipulation, given that hardware components (electronic and mechanic) can be controlled by software. The testbed is set out for its reconfiguration and for future tests with different SDRadar algorithms in a controlled environment. The proposed development presents the following updates with regard to the literature found: i) the testbed has control of the target speed, detection of micro-doppler effect using SDRadar, and validation of the speed with an independent sensor, ii) a method to characterize the SDRadar in the detection of low speeds (micro-doppler effect) is shown, as well as the procedure with which lower errors are obtained, as compared to the values given by the validation sensor, iii) the behavior of the SDRadar built with a low cost LimeSDR platform in the detection of low speeds is characterized when configured as a continuous wave radar. In section 2, the article presents a review of previous works about the research topics, in section 3 the proposed method is presented, in section 4 discussion and results are shown and, finally, in section 5 conclusions are shared.

## 2. PREVIOUS WORKS

SDR is a technology that allows reconfiguration and reprogramming of a radio, which means that, changes in the functionality of the radio can be performed by software without any type of change to the hardware [5]-[7]. Ever since the emergence of SDR platforms, different developments, and research projects on SDRadar have been carried out. In [8], [9], a Doppler radar system are presented for the detection of movements of the human body. Jian *et al.* [10], a radar for drone detection, monitoring and classification was developed. Wang *et al.* [11] a radar system was implemented to recognize 6 movements of the human body proposing a classification method using recurrent neural network (RNN) and SDR. Research by Severino *et al.* [12], a radar was implemented for pedestrian recognition, by means of the micro-doppler effect and the support vector machine (SVM) method.

A testbed is considered a platform to carry out tests and aims to provide a realistic environment of hardware and software. Testbeds can be developed in areas such as: research, academic, agriculture, space technology, and industry in general [13]. Currently, there are many SDR testbeds, among the largest are: ORBIT at Rutgers, which is an emulator of radio network, its facilities are experimental and have an open access flexible to multiple users [14]. CORNET at Virginia Tech, which has 48 SDR nodes located in a building, access is free and remote, storing several users simultaneously [15]. Lastly, CorteXlab at INSA, which consists of 38 SDR nodes and 42 nodes of wireless sensors. This testbed allows users to remotely execute applications in real time [16].

The use of SDR has also enabled the development of testbed in different areas. Research by Gallardo and Fuentes [17], an SDR testbed was designed for the solution of synchronization problems in digital modulations. An electronic product code (EPC) radio-frequency identification (RFID) testbed was developed based on SDR as label reader [18]. An SDR testbed was designed to evaluate the performance of the Zigbee technology under Wifi in a real RF environment [19]. Interference is analyzed through the degradation of the bit error rate (BER).

An SDR and radar testbed has been developed for speed detection as well. Woo *et al.* [20] conducted a testbed using SDR to detect pendular motion of a mechanism built for this purpose. An RF module, a module of reconfigurable software, a module of signal processing, a patch antenna, and Matlab were used. By means of the spectrogram and micro-doppler detection, it was determined that the pendular movement of a metallic sphere can be detected and discriminated with precision. Christiansen *et al.* [21] conducted a testbed based on SDR is presented that allows the detection of commercial aircrafts using cognitive radar. A USRP-2952R was used, as well as an RF amplifier, 2 S-Band horn antennas, MATLAB, and an interface for software and hardware control. It was concluded that the testbed can be adapted to the selection of parameters in real time and that it has a detection range of 3.5 km for a Boeing 737-300-type large commercial aircraft. Christiansen and Smith [22] carried out the development of a testbed of cognitive radar using SDR is presented for the detection of aircraft. In its implementation, an Ettus USRP X310 with a UBX-160 daughter board was used, as well as an RF amplifier and a set of patch antennas. It was concluded that the maximum range of detection for large aircrafts is of 5.5 km, for small aircrafts of 2.8 km and for a

drone, around 350 m. The system, autonomously, can adapt its carrier frequency parameters, band width, pulse length, pulse repetition rate (PRF) and real time number of paused pulses.

### 3. PROPOSED METHOD

The method of development and implementation of the testbed is illustrated in Figure 1. The first step of the proposed method is the design and manufacture of the testbed. Design criteria include; i) the target whose speed is measured must be part of the testbed, ii) target speed must be reconfigurable by software, iii) the target must have continuous movement to secure large measurement time windows and improve the resolution in frequency of the fast fourier transform (FFT), iii) speed must be measurable by a method different to the one implemented with the SDRadar to perform validation, iv) the SDRadar must allow simultaneous transmission and reception of signals, v) as the testbed does not depend on the movement of external targets, it should be able to be remotely controlled. The testbed described in section “SDRadar testbed description” fulfills all the design criteria; in this section construction details are specified, as well as the procedure implemented for the measurement of the speed through the alternative method.

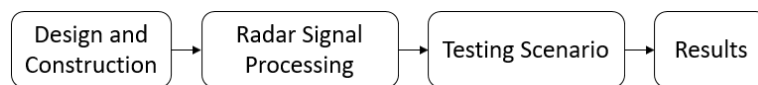


Figure 1. Method of development and implementation of the testbed

The second step of the proposed method is the processing of the radar signal, and it is presented in section “radar signal processing”. For the processing of the signal, an algorithm that includes all the aspects of the interactions between software and hardware must be proposed, it means, the way in which the signal is generated, the speed detection techniques in the signal captured using software, the way in which the signal is transmitted and received by the SDR platform, and how the different components of this platform affect its behavior. The third step is establishing a test scenario from the knowledge of the implemented algorithms and the reach and limits of the system configuration. The proposed scenario aims to obtain results regarding precision, accuracy and repeatability of the measurements carried out in the system, and the answer of the SDR platform when measuring small frequency displacements due to low speed of the target.

#### 3.1. SDRadar testbed description

An SDRadar testbed that permits measuring the velocity of a target adhered to the surface of a conveyor belt is presented. The movement of the conveyor belt is controlled by the user, and it is possible to modify the speed and the direction of the target. To measure the velocity of the target, algorithms are used to transmit a signal and to process the received echo. The LimeSDR platform is in charge of the transmission and reception of the RF signals. Some of LimeSDR characteristics are: low cost, open source, and it allows the implementation of any type of wireless communication standard. In Figure 2, a general block diagram of the LimeSDR can be observed [23].

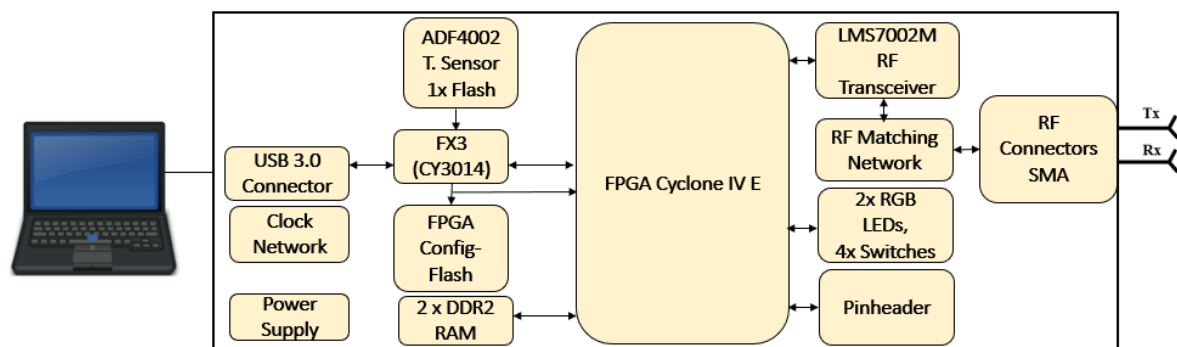


Figure 2. LimeSDR diagram [23]

As part of the testbed, a second tool to measure velocity is implemented to validate the results. The reading of an encoder located in one of the rollers of the belt, allows an Arduino to deliver the velocity being used in the validation. The architecture of the SDR radar testbed is shown in Figure 3.

The scheme shows all components of the system. The computer equipment supports software tasks and implements algorithms. Host 1 is a system with remote access through which the testbed and Host 2 can be controlled using remote desktop. The algorithm to generate and transmit the radar signal and the reception of the reflected signal is executed in Host 2. The LimeSDR platform transmits and receives the RF signals. The algorithm to determine the velocity from the reading of the encoder is executed in the Arduino.

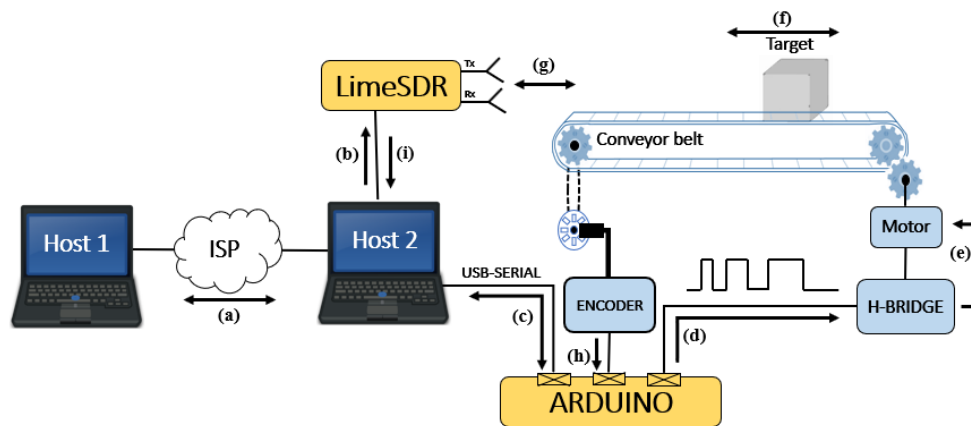


Figure 3. Testbed architecture

The information flow of the arrows shown in Figure 3 is characterized by:

- Host 1 and Host 2 communicate through remote access. Host 2 is in charge of: controlling the testbed, executing the transmission algorithms and signal reception, obtaining the speed data from Arduino and controlling the speed level and direction of the conveyor belt. The echo signal detected in host 2 is stored and can be processed and visualized in host 1 or in host 2 using Matlab.
- Host 2 sends the base band signal to the LimeSDR and configures the local oscillator and the gain. The digital base band signal turns into an analogue one, mixes with the local oscillator signal, and is filtered and amplified. This occurs in the LimeSDR and then, the signal is transmitted towards the target using one of the antennas.
- Host 2 sends the pulse width modulation (PWM) value and the direction of the turn to the Arduino (velocity control in open-loop). The Arduino returns the frequency and velocity values from the reading of the encoder.
- Arduino sends the PWM signal to the H-bridge and the signal of the change of direction of the target. It is possible to stop the movement of the conveyor belt with a pulse width equal to zero.
- H-Bridge provides the motor with the necessary current to make the conveyor belt advance or take a turn.
- The target moves linearly with different velocities and direction changes depending on the control signals applied to the motor.
- The LimeSDR transmits the RF continuous wave signal and receives the signal reflected by the target.
- The pulse detector delivers a squared signal to the Arduino. The signal frequency depends on the speed of the turn of the encoder and the conveyor belt velocity. Arduino reads the number of pulses per second and delivers the velocity.
- The LimeSDR sends the signal reflected in base band to Host 2 after being amplified, filtered, mixed, and transformed into digital.

Table 1 describes the main characteristics of the equipment used in the testbed. The testbed can work remotely through a remote desktop application, or it can also be configured directly from Host 2. Host 2 uses GNU-radio to configure the LimeSDR, allowing the processing of baseband signals. The testbed is designed to test some SDR radar algorithms, with the option of configuring the target at different speeds and positions. The development of a CW radar was proposed to verify the testbed's initial operation. Speed detection is only possible with this type of radar because the signal lag produced by the reflection on the target has a very low range of resolution. The encoder reading allows the user to verify the validity of the

algorithms implemented by using the SDR platform. Figure 4 shows the speed acquisition process using Arduino Uno.

Table 1. Describes the main characteristics of the equipment used in the testbed

Equipment	Technical specifications
Host 1-Host2	Windows 10 operating system. Remote desktop application. GNU radio software.
LimeSDR USB-Type A	MATLAB Software (Host1) Arduino software (Host2) RF LMS7002M FPRF transceiver. Altera Cyclone IV FPGA EP4CE40F23. 100KHz – 3.8GHz continuous frequency range. 61.44MHz bandwidth.
Arduino Uno	Power output (CW) above 10dBm. ATmega328 Microcontroller. 14 digital inlets/outlets, 6 PWM outlets of 8-bit.
Antenas	Type: In-house manufacturing Tin Can Waveguide. 2.4 GHz operation frequency
Conveyor belt	In-house manufacturing Transport distance: 60 cm Belt width: 5 cm, belt material: rubber Structure material: wood, plastic rollers.
Target	Surface: even Material: aluminum Height: 11 cm, width: 7 cm, thickness: 0.1 mm
HC-020-K encoder module	0.01 mm resolution. 100 KHz measuring frequency, 20-line disc
H-Bridge	L298N chip, 25W power output.
Motor	DC geared motor, 24v power supply voltage.
Signal characteristics	Type of wave: CW Signal frequency in base band: 10 KHz RF frequency: 2.4 GHz Power: Up to 10 dBm Decimation: 16384 Sample frequency: =5 Msps

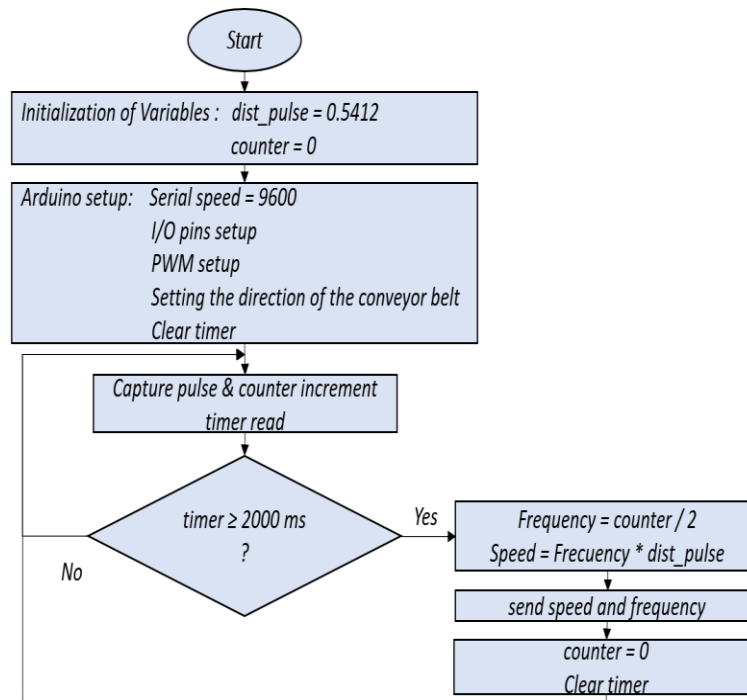


Figure 4. Speed acquisition using arduino uno flow chart

The “dist\_pulse” variable stores (in centimeters) the value of the distance traveled by the conveyor belt between the start of two slots in the encoder. In terms of the algorithm, counting a rising edge is equal to measuring 0.5412 cm. The pulse acquisition window was established at 2 s. A larger window improves resolution in speed measurement. To determine the speed of the conveyor belt, the number of pulses is divided by the size of the window (frequency  $f_p$  as result), and then multiplied by the value of the “dist\_pulse” variable. Measurement is cyclical, as shown in Figure 4. The PWM value and the band direction are established at the beginning of the algorithm. The PWM signal on the motor has an average voltage value proportional to the signal duty cycle. In this case, the supply voltage for the H-bridge is 24V. With a 50% duty cycle, an average voltage of 12 V will be obtained on the motor. This does not represent half the maximum speed, because the motor has a non-linear behavior. Figure 5 shows the appearance of the implemented testbed.

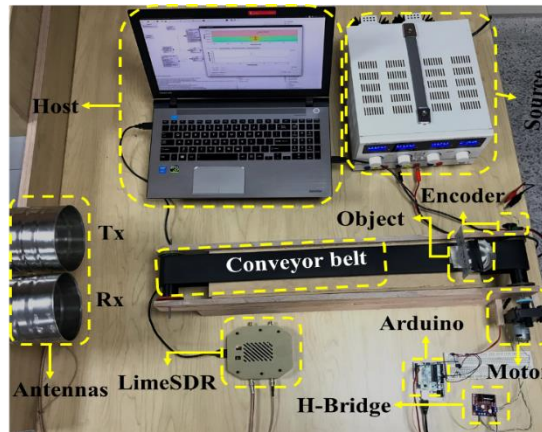


Figure 5. Description of the implemented testbed

### 3.2. Radar signal processing

A CW SDRadar transmits and detects at the same time. The transmission antenna generates a continuous sinusoidal oscillation in a single frequency. This signal changes its frequency through the Doppler effect by the reflection of a target in motion. Measurement and processing of the reflected signal allows determining the Doppler frequency [24].

The CW radar does not permit the measurement of the target range [25]. The CW SDRadar was implemented on GNU-radio. The CW signal has a frequency of approximately 2.4GHz and is transmitted by using the LimeSDR platform. The signal rebounds off the moving target located on the conveyor belt. The target speed produces a frequency shift in the reflected signal due to the doppler effect. The testbed produces low speeds and a frequency shift of a few Hertz, so a micro-doppler effect is observed in the echo signal. The block diagram for the generation, configuration, and processing of the radar signals is shown in Figure 6. The "signal source" block allows the generation of a complex baseband signal. The block output is represented in (1).

$$S(n) = S_0 e^{-j2\pi f_0 n T_s} \quad (1)$$

where,

$$T_s = \frac{1}{f_s} \quad (2)$$

$f_s$  is the sampling frequency and  $f_0$  is the baseband signal frequency. The real and imaginary components (I and Q) of the signal are sent to the hardware using the “LimeSuite sink (TX)” block. In (3) and (4) represent each component of the signal.

$$I(n) = \text{Re}\{S_0 e^{-j2\pi f_0 n T_s}\} = S_0 \cos(2\pi f_0 n T_s) \quad (3)$$

$$Q(n) = \text{Im}\{S_0 e^{-j2\pi f_0 n T_s}\} = -S_0 \sin(2\pi f_0 n T_s) \quad (4)$$



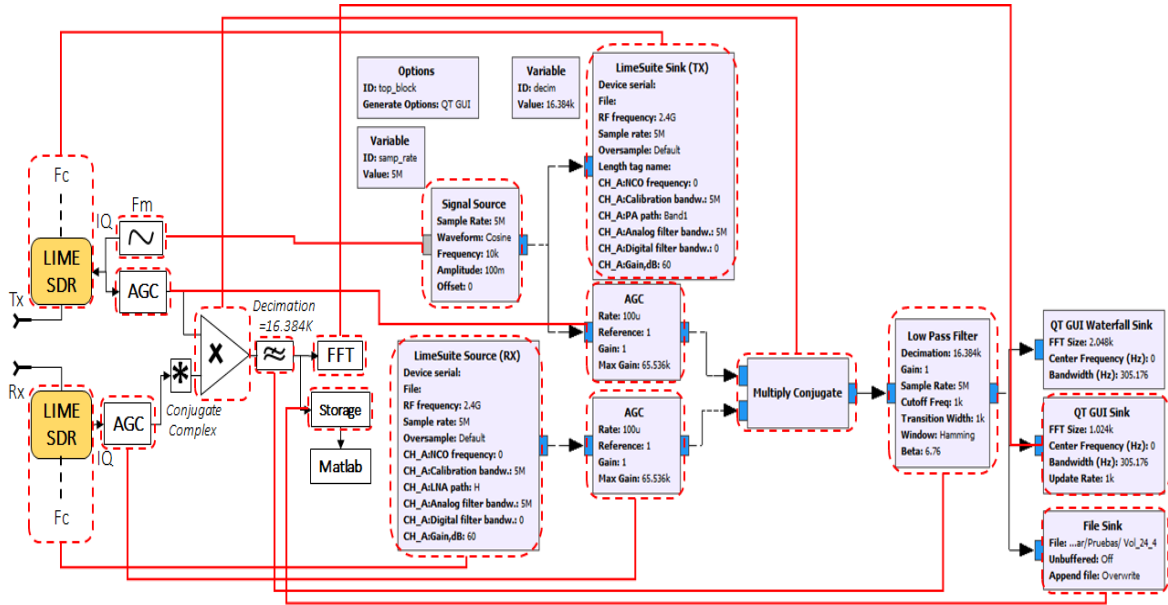


Figure 6. Block diagram for the GNURadio processing

Figure 7 shows the block diagram for one of the RF transmitters of the LMS7002M transceiver (see component in Figure 2). On this transceiver, each signal (I and Q) is converted from digital to analog on block DAC. Then they go through a low-pass filter (TXLPF). Continuous time signals after the filter are represented in (5) and (6).

$$I(t) = S_1 \cos(2\pi f_0 t) \tag{5}$$

$$Q(t) = -S_1 \sin(2\pi f_0 t) \tag{6}$$

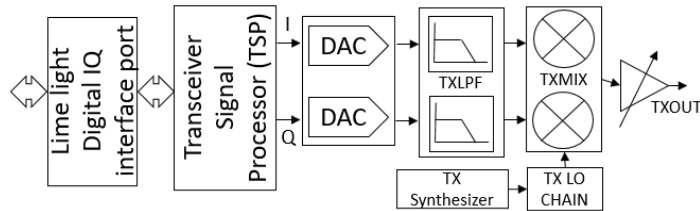


Figure 7. Block diagram for an RF transmitter of the LMS7002M transceiver

The signals are mixed in the quadrature modulator (TXMIX) with the one generated by the local oscillator of the transmitter (TX LO CHAIN). The local oscillator generates a signal as shown in (7).

$$C(t) = C_0 e^{j2\pi f_c t + \varphi} \tag{7}$$

Where  $C_0$  is the carrier peak amplitude,  $f_c$  is the frequency generated by the local oscillator and  $\varphi$  is the random initial phase related to the baseband signals. The real part of the  $C(t)$  signal is mixed with component I and results in (8) and, by trigonometric identity, expressed in (9).

$$E_I(t) = C_0 S_1 \cos(2\pi f_c t + \varphi) \cos(2\pi f_0 t) \tag{8}$$

$$E_I(t) = \frac{C_0 S_1}{2} [\cos(2\pi(f_c - f_0)t + \varphi) + \cos(2\pi(f_c + f_0)t + \varphi)] \tag{9}$$

Likewise, the imaginary part of  $C(t)$  is mixed with component Q resulting in (10) and (11).

$$E_Q(t) = -C_0 S_1 \sin(2\pi f_c t + \varphi) \sin(2\pi f_0 t) \tag{10}$$

$$E_Q(t) = \frac{C_0 S_1}{2} [-\cos(2\pi(f_c - f_0)t + \varphi) + \cos(2\pi(f_c + f_0)t + \varphi)] \tag{11}$$

The quadrature mixer output is equal to the sum of (9) and (11). The result is evident in (12) and (13).

$$E(t) = G_s L_s [E_I(t) + E_Q(t)] \tag{12}$$

$$E(t) = E_0 \cos(2\pi f_t t + \varphi) \tag{13}$$

With  $f_t = f_c + f_0$ . The signal defined by (13) is amplified and transmitted (TXOUT). The amplitude  $E_0$  considers the amplitudes  $C_0, S_1$ , the RF ( $G_s$ ) amplifier gain and the losses inherent to the system ( $L_s$ ). When the signal collides with a target, in this case moving as in Figure 3, the signal echo reaches the receiver of the SDR platform, having the shape of (14).

$$E_1(t) = K_1 \cos(2\pi(f_t \pm f_r)t + \varphi_1) \tag{14}$$

Where  $K_1$  is the amplitude of the reflected signal,  $f_r$  is the Doppler frequency shift, and  $\varphi_1$  a phase constant depending on the target range.

The value of  $K_1$  depends on the target radar cross section (RCS), the transmitted power, and the used antenna gain. A portion of the transmitted signal reaches the receptor as well, due to the proximity of the transmitting and receiving antennas as shown in Figure 5. The sum of the received signals is shaped as shown in (15).

$$E_r(t) = K_1 \cos(2\pi(f_t \pm f_r)t + \varphi_1) + K_2 \cos(2\pi f_t t + \varphi) \tag{15}$$

Where  $K_2$  is the peak amplitude of the signal that is received from the transmitter directly. If the target moves towards the radar, frequency  $f_r$  will be added as in Figure 8(a). If the target moves away from the radar, the frequency  $f_r$  will be subtracted as in Figure 8(b).

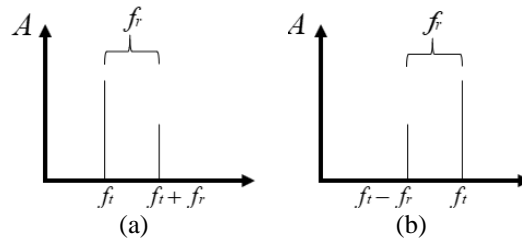


Figure 8. Doppler frequency (a) target approaching and (b) target moving away

Figure 9 corresponds to the block diagram for one of the RF transmitters of the LMS7002M transceiver (see component on Figure 2). The equation signal in (15) is amplified and mixed with the signal from the receiver local oscillator (RX LO CHAIN). This last signal is synchronized with the transmitter local oscillator, as shown in (7).

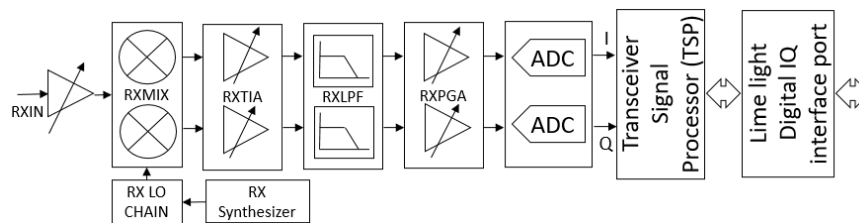


Figure 9. Block diagram for a receiver of the LMS7002M transceiver

The real part of the mix result takes channel I as shown in (16).

$$E_{I_r}(t) = C_0 \cos(2\pi f_c t + \varphi) * [K_1 \cos(2\pi(f_t \pm f_r)t + \varphi_1) + K_2 \cos(2\pi f_t t + \varphi)] \tag{16}$$



After filtering the high frequency components (RXLFP), (17) is achieved, considering the gain of the RXTIA amplifier and the filter.

$$E_{Irr'}(t) = \frac{c_0 K_1}{2} \cos(2\pi(f_0 \pm f_r)t + \varphi_1 + \varphi) + \frac{c_0 K_2}{2} \cos(2\pi f_0 t + 2\varphi) \quad (17)$$

Performing the same process for the mixing result in channel Q, we obtain:

$$E_{IQ'}(t) = \frac{-c_0 K_1}{2} \sin(2\pi(f_0 \pm f_r)t + \varphi_1 + \varphi) - \frac{c_0 K_2}{2} \sin(2\pi f_0 t + 2\varphi) \quad (18)$$

The complex signal formed by channels I and Q is shown in (19).

$$E_{rr'}(t) = \frac{c_0 K_1}{2} e^{-j(2\pi(f_0 \pm f_r)t + \varphi_1 + \varphi)} + \frac{c_0 K_2}{2} e^{-j(2\pi f_0 t + 2\varphi)} \quad (19)$$

The signal is amplified again (RXPGA) and converted from analog to digital (ADC), as evident in (20).

$$E_{r'd}(t) = K_3 e^{-j(2\pi(f_0 \pm f_r)nT_s + \varphi_1 + \varphi)} + K_4 e^{-j(2\pi f_0 nT_s + 2\varphi)} \quad (20)$$

Where  $K_3$  and  $K_4$  are the amplitudes of the amplified and digitalized signal.

The signal of (20) is the one delivered from the "lime suite source" block to the automatic gain control "AGC" block on Figure 6. "AGC" is used to compensate for losses between the transmitted and received baseband signal. The signal of (1) passes through an "AGC" block too, its amplitude changing to a value of A. After the "AGC" blocks, the complex conjugate of the received signal is obtained and mixed with the transmitted signal using the "multiply conjugate" block. This process removes the frequency from the baseband transmitted signal  $f_0$ , leaving the frequency shift. The resulting signal is shown in (21) and (22).

$$X(n) = A e^{-j2\pi f_0 nT_s} (B e^{j(2\pi(f_0 \pm f_r)nT_s + \varphi_1 + \varphi)} + C e^{j(2\pi f_0 nT_s + 2\varphi)}) \quad (21)$$

$$X(n) = A B e^{\pm j2\pi f_r nT_s + \varphi + \varphi_1} + A C e^{j2\varphi} \quad (22)$$

Where  $B$  and  $C$  are the amplitudes of the received signal after passing through AGC. The first term of (22) represents the Doppler signal, the second one is a phase-dependent constant.

Noise and speed changes at the edges of the conveyor belt were omitted throughout the analysis to simplify calculations. The signal of (22) is filtered and resampled (decimation) using the "low pass filter" block. Resampling allows the analysis to be focused on the low frequencies, since the frequency shift in the testbed is low due to the target low velocities. The sampling frequency after decimation appears in (23).

$$f_{s1} = \frac{f_s}{\text{decimation}} = \frac{5M\text{sps}}{16384} = 305,17\text{sps} \quad (23)$$

To measure the Doppler frequency shift, the fast fourier transform is applied as shown in (24). The "QT GUI Sink" block allows the FFT visualization.

$$X(k) = \frac{1}{N} \text{FFT}_N\{X(n)\} \quad (24)$$

The maximum peak of the FFT is obtained with (25). Figure 10 shows the maximum peak frequency of the FFT and represents the target on the conveyor belt moving at a speed of approximately 16.7 cm/s.

$$K_T = |X(k)| \quad (25)$$

The "file sink" block is used to store the signal from (22) to be locally or remotely analyzed later. Figure 10 was obtained from the signal stored and processed with MATLAB. With (26) the speed of the target on the band is calculated using the FFT maximum peak frequency,  $f_r = 2.663 \text{ Hz}$ .

$$v = \frac{c f_r}{2 f_t} \quad (26)$$

On Figure 10, a window of 5500 samples was taken for the computation of the FFT ( $N^\circ \text{ FFT Points}$ ). There is a better frequency resolution in the spectrum with a high number of samples per window, however, the resolution in time worsens. The equation for the window time is observed in (27). For Figure 10,  $t_w = 18.022\text{s}$ .

$$t_w = \frac{N^\circ \text{ FFT Points}}{f_{s1}} \quad (27)$$

The frequency resolution  $\Delta f$  is shown in (28) and determines the speed measurement accuracy.

$$\Delta f = \frac{f_{s1}}{N^{\circ} \text{ FFT Points}} \quad (28)$$

The frequency resolution for Figure 10 is  $\Delta f = 0.00327\text{Hz}$ . Using (26) a resolution is found in the speed measurement of 0.02 cm/s. Otherwise, if a time window  $t_w = 2\text{s}$  is taken, 610 points are obtained in the calculation of the FFT using (27).

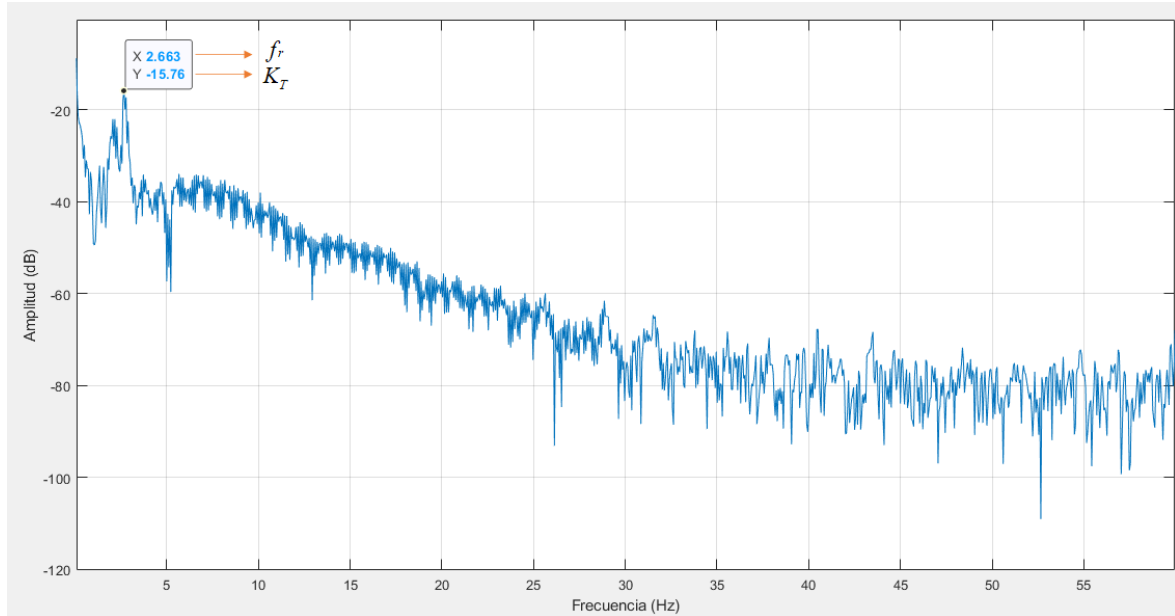


Figure 10. Highest peak in MATLAB (FFT 5500 points)

Figure 11 shows the FFT result for the same signal of Figure 10 but using only 610 points. In the case of peak frequency  $f_r = 2.501\text{ Hz}$ , a speed of 15.66 cm/s is obtained using (26), and with (28), a frequency resolution of  $\Delta f = 0.5\text{ Hz}$  can be found. The speed resolution is 3.13 cm/s, which is significant considering the testbed measures low speeds.

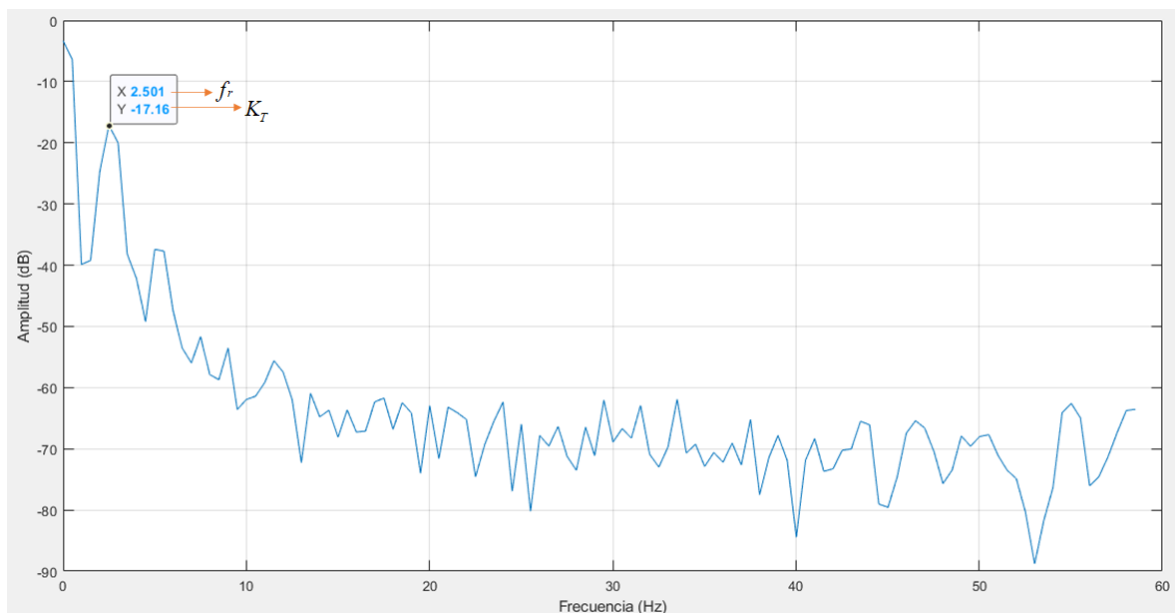


Figure 11. Highest peak in MATLAB (FFT 610 points)

### 3.3. Testing scenario

According to the analysis presented in the previous section, the following characteristics were defined to perform the tests; i) a window of 5500 samples in each measurement for good speed resolution, while sacrificing time resolution, ii) different duty cycle values of the PWM signal were established to have average voltages on the motor between 7 and 24V, with 1V intervals between each speed measurement, iii) for each average voltage level, 5 speed measurements were taken at different times. The motor starts with a speed of 0 cm/s at each measurement, but measurements are taken at stable state speeds, iv) speed measurements made with the encoder reading are taken, parallel to each measurement made with radar SDR.

## 4. RESULTS AND DISCUSSION

The five speeds obtained for each voltage were averaged. The standard deviation  $\sigma$  and percentage error were obtained. Table 2 summarizes the measurement results for each average voltage. Table 2 shows that the error between the data taken with the SDRadar compared to the data taken with the encoder reading is low and, in the worst case, less than 5%. On the other hand, the standard deviation value is also low, which guarantees the repeatability of the experiments. The frequency shift  $f_r$  due to the micro-doppler effect generated by the movement of the conveyor belt is less than 3 Hz, which is why a good frequency resolution of the FFT is necessary to perform more precise measurements.

Table 2. Results obtained after measuring

Volt (v)	SDR radar data			Encoder data		% E
	$f_r$ (Hz)	$v$ (cm/s)	$\sigma$ (cm/s)	$f_p$ (Hz)	$v$ (cm/s)	
7	0.659	4.128	0.25	8	4.33	4.7
8	0.787	4.931	0.36	9	4.87	1.3
9	0.897	5.619	0.22	10	5.41	3.9
10	1.007	6.306	0.14	12	6.49	2.8
11	1.135	7.109	0.29	13	7.04	1
12	1.269	7.951	0.31	14	7.58	4.9
13	1.349	8.448	0.55	16	8.66	2.4
14	1.501	9.403	0.58	17	9.2	2.2
15	1.556	9.748	0.63	18	9.74	0.1
16	1.648	11.321	0.52	20	10.82	4.6
17	1.813	11.353	0.45	22	11.91	4.7
18	1.953	12.232	0.25	23	12.45	1.7
19	2.087	13.073	0.32	24	12.99	0.6
20	2.270	14.220	0.26	26	14.07	1.1
21	2.332	14.602	0.42	27	14.61	0.1
22	2.447	15.329	0.41	29	15.69	2.3
23	2.633	16.488	0.64	30	16.24	1.5
24	2.645	16.563	0.17	31	16.78	1.3

\*% E: error rate

Figure 12 shows the short time fourier transform (STFT) for 4 signals taken with average voltages of 7 V, 15 V, 19 V and 22 V, respectively. The window for each signal has 1000 samples with  $t_w = 3.27s$ . Using this method, a maximum error of 13.26% was obtained in relation to the speed measured with the encoder. In the case of Figure 12, when applying the STFT to the signals, good results can be obtained with smaller time windows, however, the error is greater in relation to the method used with the data in Table 2, in which the time windows are much larger. According to table 2, the maximum speed of the testbed was measured at 16.78 cm/s using an average voltage on the motor of 24V.

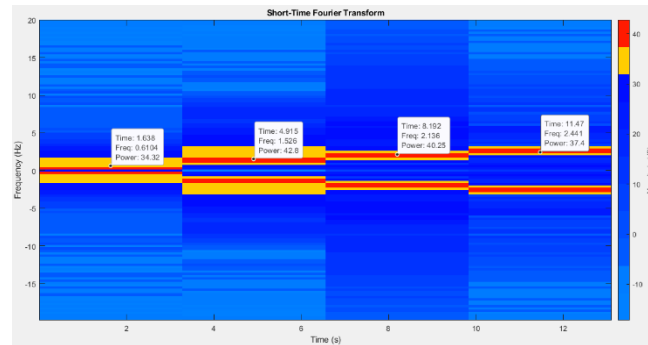


Figure 12. STFT for 4 signals taken with different speeds at different times

## 5. CONCLUSION

The results show the proper operation of the testbed for the detection of speed through micro-doppler effect and validate its use to carry out tests in academic and research environments. The results show a repeatability of the experiments and the speed measurement using the encoder reading, allowing to check the measurements made with SDRadar. The testbed guarantees the development and testing of SDRadar algorithms in a controlled speed environment. The use of the conveyor belt allows tests to be carried out with a target that maintains linear movement and with programmable speed, avoiding the movement of equipment and personnel to open spaces with more complex scenarios in taking measurements. The use of the testbed can be done remotely because it does not require any type of configuration on site, optimizing its use without time restrictions. Regarding SDRadar characterization in low speeds detection, Table 2 reflects that the speed measured with the SDR platform is very close to the sensor speed; nevertheless, it is evident that greater errors are in medium and low speeds of the table. In any case, the error is never above 5%, with which it is concluded that the evaluated platform (LimeSDR) can be used to detect micro-doppler effect in other applications with great reliability.

## ACKNOWLEDGEMENTS

This research was developed within the GISSIC research group as a by-product of the INV ING 2647 research project, funded by the vice-chancellor for research at the Universidad Militar Nueva Granada, and developed at the Maxwell hotbed for research.





## REFERENCES

- [1] ITU-R, "Definitions of Software Defined Radio (SDR) and Cognitive Radio System (CRS) SM Series," vol. 2152, 2009.
- [2] S. Feng, N. Mughees and V. Wollesen, "Reviewing the Application and Integration of Software Defined Radios to Radar Systems," *IEEE Radar Conference (RadarConf20)*, 2020, pp. 1-6, doi: 10.1109/RadarConf2043947.2020.9266396.
- [3] L. Zeng, C. Yang, Y. Zhao, M. Huang and C. Zhi, "Research on Evaluation Index System for Software Defined Radar (SDR)," *IEEE Radar Conference (RadarConf)*, 2019, pp. 1-6, doi: 10.1109/RADAR.2019.8835588.
- [4] A. F. Martone et al., "Practical Aspects of Cognitive Radar," *IEEE Radar Conference (RadarConf20)*, 2020, pp. 1-6, doi: 10.1109/RadarConf2043947.2020.9266646.
- [5] J. Mitola, "The software radio architecture," *IEEE Communications Magazine*, vol. 33, no. 5, pp. 26-38, 1995, doi: 10.1109/35.393001.
- [6] D. N. Grujić, P. Jovanović and M. Savić, "Using software defined radio for RF measurements," *Zooming Innovation in Consumer Electronics International Conference (ZINC)*, 2017, pp. 18-21, doi: 10.1109/ZINC.2017.7968652.
- [7] G. Manisha and M. Nikhil, "A Review on Automatic Signal Classification Techniques for Software Defined Radios," *5th International Conference on Signal Processing, Computing and Control (ISPCC)*, 2019, pp. 281-286, doi: 10.1109/ISPCC48220.2019.8988509.
- [8] B. Tan, K. Woodbridge and K. Chetty, "A real-time high resolution passive WiFi Doppler-radar and its applications," *International Radar Conference*, 2014, pp. 1-6, doi: 10.1109/RADAR.2014.7060359.
- [9] Y. Ma, Y. Zeng and S. Sun, "A Software Defined Radio based Multi-Function Radar for IoT Applications," *24th Asia-Pacific Conference on Communications (APCC)*, 2018, pp. 239-244, doi: 10.1109/APCC.2018.8633541.
- [10] M. Jian, Z. Lu and V. C. Chen, "Drone detection and tracking based on phase-interferometric Doppler radar," *IEEE Radar Conference (RadarConf18)*, 2018, pp. 1146-1149, doi: 10.1109/RADAR.2018.8378723.
- [11] M. Wang, Y. D. Zhang, and G. Cui, "Human motion recognition exploiting radar with stacked recurrent neural network," *Digital Signal Processing*, vol. 87, pp. 125-131, 2019, doi: 10.1016/j.dsp.2019.01.013.
- [12] J. V. B. Severino, A. Zimmer, T. Brandmeier, and R. Z. Freire, "Pedestrian recognition using micro-Doppler effects of radar signals based on machine learning and multi-objective optimization," *Expert Systems with Applications*, vol. 136, pp. 304-315, 2019, doi: 10.1016/j.eswa.2019.06.048.
- [13] P. Fortier and H. Michel, "Computer systems performance evaluation and prediction," Elsevier Science, 2003.





- [14] D. Raychaudhuri *et al.*, "Overview of the ORBIT radio grid testbed for evaluation of next-generation wireless network protocols," *IEEE Wireless Communications and Networking Conference*, 2005, vol. 3, pp. 1664-1669, doi: 10.1109/WCNC.2005.1424763.
- [15] N. Sharakhov, V. Marojevic, F. Romano, N. Polys, and C. Dietrich, "Visualizing real-time radio spectrum access with CORNET3D," *Web3D '14: Proceedings of the 19th International ACM Conference on 3D Web Technologies*, pp. 109-116, 2014, doi: 10.1145/2628588.2628598.
- [16] L. S. Cardoso *et al.*, "CorteXlab: A facility for testing cognitive radio networks in a reproducible environment," *9th International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CROWNCOM)*, 2014, pp. 503-507, doi: 10.4108/icst.crowncom.2014.255812.
- [17] S. P. Gallardo and H. A. Fuentes, "Design and implementation of a wireless softwaredefined radio testbed," *Institutional Repository of Universidad Nacional*, vol. 80, no. 180, pp. 67-76, 2013.
- [18] F. Galler, T. Faseth and H. Arthaber, "Implementation aspects of an SDR based EPC RFID reader testbed," *International EURASIP Workshop on RFID Technology (EURFID)*, 2015, pp. 94-97, doi: 10.1109/EURFID.2015.7332391.
- [19] J. Park, H. Yoon and B. Jang, "SDR testbed for analyzing frequency interference between unlicensed devices," *Asia-Pacific Microwave Conference (APMC)*, 2015, pp. 1-3, doi: 10.1109/APMC.2015.7413405.
- [20] I. Woo, J. Jung, M. Park and Y. Kwag, "Software defined radar platform testbed for micro-Doppler detection," *IEEE 5th Asia-Pacific Conference on Synthetic Aperture Radar (APSAR)*, 2015, pp. 50-53, doi: 10.1109/APSAR.2015.7306152.
- [21] J. M. Christiansen, G. E. Smith and K. E. Olsen, "USRP based cognitive radar testbed," *IEEE Radar Conference (RadarConf)*, 2017, pp. 1115-1118, doi: 10.1109/RADAR.2017.7944371.
- [22] J. M. Christiansen and G. E. Smith, "Development and Calibration of a Low-Cost Radar Testbed Based on the Universal Software Radio Peripheral," *IEEE Aerospace and Electronic Systems Magazine*, vol. 34, no. 12, pp. 50-60, 1 Dec. 2019, doi: 10.1109/MAES.2019.2953803.
- [23] L. M. Limited, "LimeSDR-QPCle v1.2 Quick Start Manual," vol. 0.02, pp. 1-107, 2018.
- [24] M. A. Mutschler, P. A. Scharf, H. Mantz, T. Walter and C. Waldschmidt, "Feature Extraction for Classification of Water Surfaces using a 24 GHz CW Radar," *16th European Radar Conference (EuRAD)*, 2019, pp. 105-108.
- [25] D. Uttamchandani, "Handbook of MEMS for wireless and mobile applications," Elsevier, 2013.

## BIOGRAPHIES OF AUTHORS







**Juan Carlos Martínez Quintero**     received the M.Sc. degree in autonomous systems of production from in Universidad Tecnológica de Pereria in 2013. He is currently profesor at Militar Nueva Granada University. His research interests include mobile networks, SDR, communication systems and digital signal processing. He can be contacted at email: [juan.martinezq@unimilitar.edu.co](mailto:juan.martinezq@unimilitar.edu.co).



**Edith Paola Estupiñan Cuesta**     received the M.Sc. degree in Electronic Engineering from Pontifical Xavierian University in 2013. She is currently working at Militar Nueva Granada University. Her research interests include mobile network, traffic analysis and data and management network. She can be contacted at email: [edith.estupinan@unimilitar.edu.co](mailto:edith.estupinan@unimilitar.edu.co).



**Johan Stiven García Ramírez**     is a Telecommunications Engineering Student from Universidad Militar Nueva Granada. During his major, he joined the GISSIC research group (Maxwell hotbed of research), which led him to learn about some research methods. He can be contacted at email: [u1401108@unimilitar.edu.co](mailto:u1401108@unimilitar.edu.co).