Channel Leakage Cancellation for Software Defined Radio (SDR) Narrowband Radar Interferometry Using GNU Radio

Victor Cai, Jon Kraft
Analog Devices Inc., Longmont, CO, USA

victorjcai@gmail.com
jon.kraft@analog.com

Abstract

Because the SDR has both an RF transmitter and a receiver integrated in the same module, leakage from transmit into receive path is inevitable. Without proper compensation, the received radar signal is combined with this unintentional leakage signal from the transmit path creating unintended distortion in phase and amplitude. This type of accuracy degradation has been reported in the previous publication when a Multiple Frequency Continuous Wave (MFCW) distance sensing radar was created with SDR.[1] A traditional Pulse or FMCW radar needs 1GHz bandwidth (BW) to achieve 15cm resolution for short range applications, making a typical SDR's 4-60MHz BW seem far from adequate for distance sensing. However, in this project, a new interferometry radar solution is demonstrated to make short range distance sensing possible using bandwidth-limited SDR. This new concept opens the doors to many short-range radar applications using this low cost SDR technology and will help to overcome the high power and harsh interference associated with traditional ultrawide band radars. Major improvement in distance sensing accuracy has been achieved by introducing two all-software solutions in GNU Radio for leakage cancellation and automatic erroneous result correction. This article will highlight how GNU Radio was used not only as a simulator in the debug phase but also as the end solution to the actual leakage cancellation in the radar operation.

1. Introduction

Traditional radars were typically associated with high-power long-distance sensing over a range measured in kilometers. However in the last decade, lower-power radars with meter range measurements have become quite useful for autonomous vehicles [2], smart homes [3], medical sensing [4], imaging [4], and tomography for security with a "see through the wall" capability [5]. Because the traditional "Pulse Radar" and Frequency Modulated Continuous Wave (FMCW) radar are based on transmits and receive a narrow pulse using the time-of-travel to calculate distance, an Ultra Wideband (UWB) radar needs a 1GHz bandwidth (BW) to achieve a 15cm distance resolution.[6] This high bandwidth demand directly translates to higher power and harsh interference with other wireless devices for short meter-range applications requiring centimeter accuracy.

Proceedings of the 11th GNU Radio Conference, Copyright 2021 by the author(s).

Strict licenses are required to operate these kinds of UWB radar, effectively preventing amateur exploration outside the short license-free 2.4~2.5GHz citizen band. Thus, a different design must be implemented to work around civilian bandwidth restrictions.

Software Defined Radio is an active research area featuring a software-configurable hardware and enabling users to construct Radio Frequency (RF) systems through software. While an RF circuit design requires advanced college-level engineering knowledge, the SDR platform enables any amateur to explore any RF system with transmit and receive capabilities, including walkie-talkie, satellite receiver, and even Doppler radar. All these systems can be constructed through software codes as an ideal platform for this new radar algorithm research.[7]

Low cost (<$200) SDR transceivers are readily available for purchase such as the Analog Devices Pluto-Adalm or LimeSDR to be used to construct a radar system. A typical SDR's 4-60MHz in BW is far from adequate for UWB distance sensing using traditional methods. Based on personal experimentation, even at just 4MHz and above, the interface from SDR to computer will occasional drop data samples. For a typical SDR with 4MHz BW, it will have a theoretical accuracy limit of 3750cm (i.e. inaccuracy equivalent to the size of a large house). [6][8] Therefore it may only be useful for detecting large object far away such as an airplane or a ship, and become essentially blind within a short 10-meter range or less.

In GRCON 2020, we introduced a feasibility study of an SDR-based, narrow-band, low-power radar technology capable of distance sensing in meter range while maintaining a reasonable accuracy. [1] However, there were two issues identified – erroneous distance calculation as a phase modulo of 2π from linear phase subtraction, and an undesirable systemic error oscillating around the ideal value. In this article, we introduce two pure software solutions featured in GNU Radio and successfully resolve the two issues.

With GNU Radio's data flow chart style programming, there is no "if" or other conditional data processing to process the live data stream in a selective manner. The erroneous distance issue from the 2π phase modulo was resolved with phase domain transposing using existing GNU modulo block as shown in section 2.

With further investigation, the systemic error oscillating around the ideal value was identified to be a result of leakage signal from transmitter to the receiver within the same SDR channel. As shown in Fig.1, for a two channel SDR radar system [1], there are two types of leakage or crosstalk – cross channel leakage and single channel leakage. As a result, the radar signal received from the target reflection is contaminated with the leakage signals. This paper will not only introduce a way to quantify the leakage, but also to create the appropriate signal to mathematically cancel out the leakage signal and achieve even higher distance sensing accuracy.

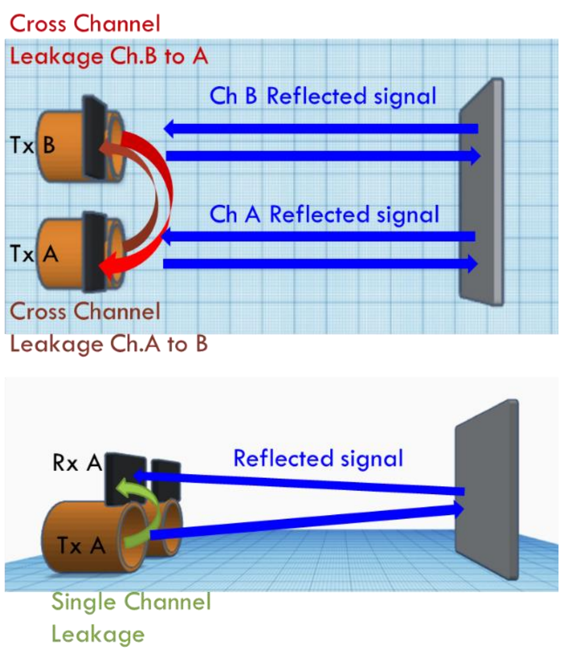

Fig. 1 In a two channel SDR radar system, there are two types of leakage. Besides the obvious cross channel leakage from Ch.A to Ch.B and vice versa, the single channel leakage is the transmit signal coupled into the receiver path. (In the actual setup, the receive panels are swapped to reduce single channel leakage. This is not reflected by Fig. 1 for simplicity.)

2. Narrowband Multiple Channel Distance Sensing Technology

This project developed a new interferometry radar to make distance and speed sensing possible on bandwidth limited SDR technology, where phase delta from two different frequencies is used to calculate target distance in a dual-channel Multiple Frequency Continuous Wave (MFCW) Radar [9]. In the next two sections, the unique challenges and solutions associated with this SDR-based design are illustrated.

2.1 Operation Principle of Multiple Frequency Continuous Wave (MFCW) Radar

The key realization of this project was that the low-bandwidth MFCW radar algorithm paired with a low-cost, low-power SDR system would meet all the requirements for a meter-range, centimeter-accuracy radar. As shown in Fig.2, the MFCW radar uses two or more frequencies of Continuous Wave (CW) simultaneously instead of a pulse. Distance is calculated from the phase difference between the two CW signals, so it is an extended type of interferometry radar relying on phase measurement. Fig.2 shows an MFCW radar operating with two different CW frequencies with different wavelengths. At any given distance, the phase difference info for each of the frequencies is uniquely determined. Using the known frequency difference, Δf, and measured phase difference, $\Delta\emptyset$, the distance can be calculated.

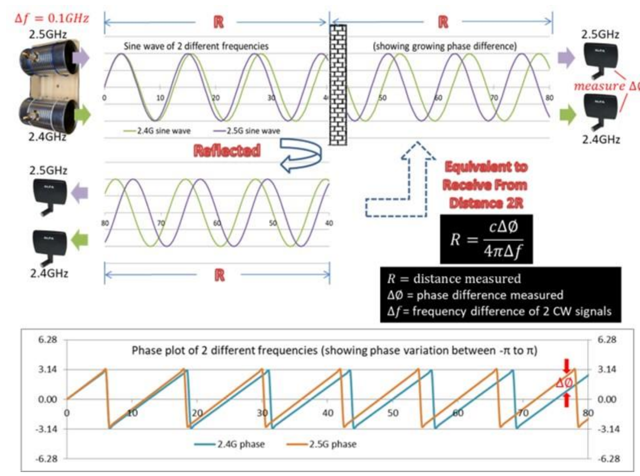

Fig. 2 MFCW radar operates with two different frequencies of CW waves with slightly-different wavelengths. For a specific distance, the phase difference between the two frequencies is used to determine the distance of the object.

The phase plot of the two frequencies in Fig.2 showed triangular curves with the phase linearly increasing between -π to +π as a periodic function. Since the phase is a modulo of 2π, the range becomes ambiguous when $\Delta\emptyset = 2\pi m$, where m is a positive integer. The maximum unambiguous range occurs when m=1, with c being the speed of light: [9]

$$R_{max\_unambiguous} = \frac{c}{2\Delta f}$$

For a 2.5GHz wave (wavelength = 12cm) and a 2.4GHz wave (wavelength = 12.5cm), the phases will line up at 0 again at a 300cm round trip distance because 25x12cm=24x12.5cm=300cm. Therefore, the MFCW radar maximum unambiguous range is 150cm with 2.5GHz and 2.4GHz frequencies. Using 2.5GHz and 2.45GHz will double the max range to 300cm by halving the Δf.

The advantage of MFCW is the super low kHz bandwidth to achieve the superior ~0.15m distance sensing accuracy of an expensive traditional "pulse radar" or FMCW radar with ultrawide 1000MHz bandwidth. This is because interferometry only requires two small slivers of bandwidth at specific frequencies as opposed to a whole range of frequencies. However, MFCW radar requires multiple RF transmit and receive channels with stable phase relations.

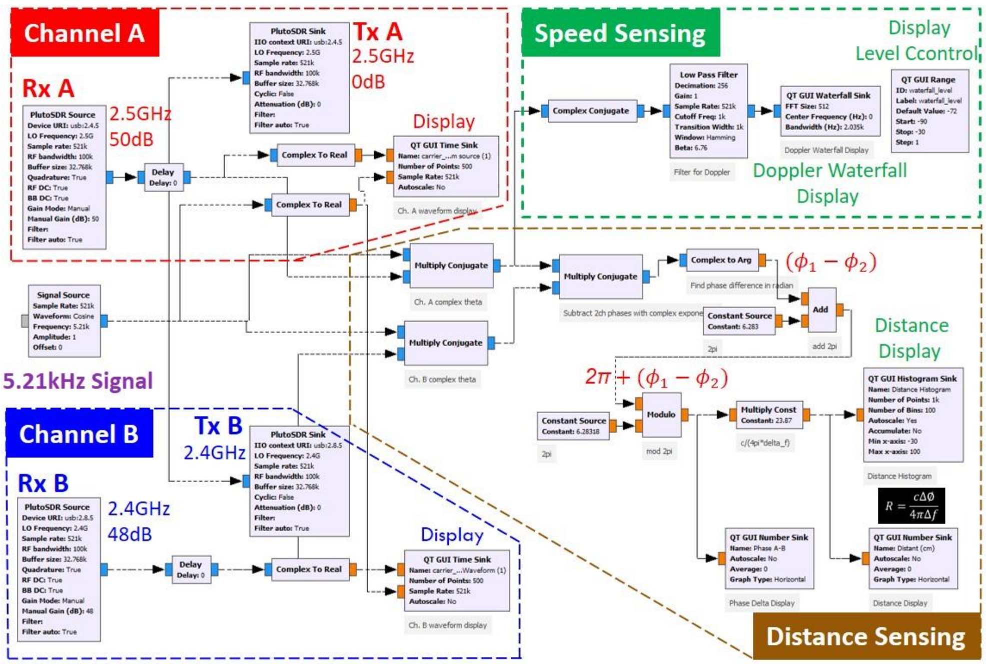

Fig. 3 GNU Radio program developed for Dual-SDR MFCW radar design with two frequency channels (2.4GHz and 2.5GHz).

2.2 Building the 2 Channel MFCW SDR Radar

The MFCW radar is developed in the intuitive GNU Radio with a data flow chart style GUI interface as shown in Fig.3. The program uses a single 5.21kHz signal modulated on the channel-A's 2.5GHz carrier, which is transmitted and received to produce a phase ϕ_1 carrying the info of distance R. The same operation is done with 2.4GHz channel-B. Then, the distance R is derived from Δf and $\Delta\emptyset$. [1]

This block diagram styled GNU Radio code looks straightforward with various math operations encapsulated in functional blocks such as Signal-Source, Modulation, Demodulation, and Multiply-conjugate phase measurement. The Dual SDR radar system software in Fig.3 can be clearly partitioned into two nearly independent sections for two SDRs. They are essentially duplicated bocks running at two different frequencies (i.e. 2.5GHz and 2.4GHz). The two sections are connected by a shared 5.21kHz signal source to synchronize the two SDRs and distance sensing block for data process and result display. The math operation is more than just a simple subtraction between the two phase measurements. The new algorithm developed here is a key to the success of this project. The 521kHz sampling rate is selected as almost the lowest programmable value for Pluto SDR to minimize the data traffic to the computer. The signal of 5.21KHz is selected to provide 100 samples in each period. Two separated USB addresses are used to control both Pluto SDRs using the same program. In the command window, USB addresses can be found using "iio_info -s" command in Windows Command Prompt. This address can increment if the USB is unplugged and plugged in again.

In the GNU Radio software, a real time GUI interface was utilized for development and demonstration purposes. It not only displays the final distance and speed sensing results but also displays some of these intermediate calculations and even the waveforms and phase graphs, as shown in Fig. 4.

As for the radar hardware, two tin can radars are made for the two SDR transmitters, which can be traced back to the MIT Open Courseware on "Coffee Can Radar." [10] The tin can antenna design dimensions is shown in Fig. 5, which is optimized for 2.45GHz operation between the two operating frequencies (2.4 and 2.5GHz). Two baked bean tin cans were selected with 84mm diameter, and two N-Type female connectors are used to hold the quarter wavelength monopole wire as the radiating source in the tin can. [11] The transmit frequency is selected in the 2.4-2.5GHz frequency band in the license free "citizen band."

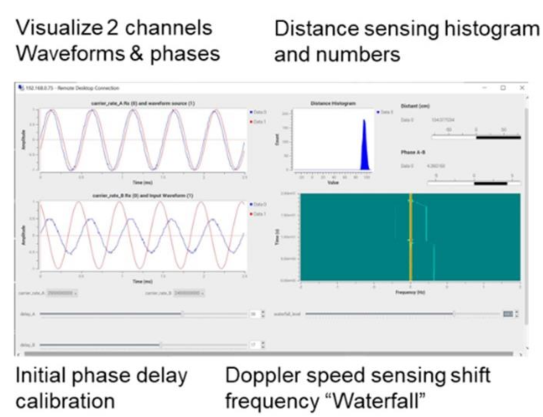

Fig. 4 A GUI interface was developed for debug and demonstration purposes with both distance and speed sensing results and waveforms from the two radar channels.

To minimize the increase in size, two compact directional 2.4GHz panel antennas were added for the two SDR receivers. These are commercially available for WiFi and drone applications. As shown in Fig.6, the panel antennas are much smaller than the tin can antennas. The two tin can antennas are better kept as transmitters because of their higher directionality and isolation, which can direct more of the energy towards the target in the front and minimize coupling into adjacent antennas.

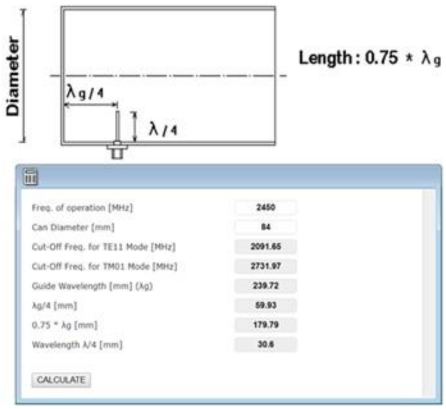

Fig. 5 Screenshot of free calculator [11] used to determine optimal coffee can dimensions for the operating frequency range.

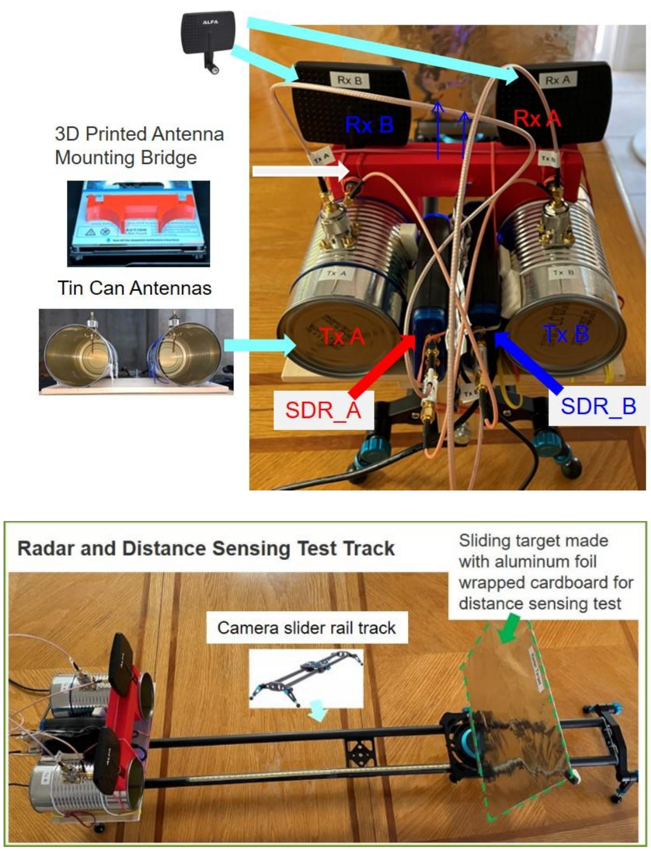

Fig. 6 Dual SDR radar system developed with two WiFi panel antennas added on top of a 3D-printed mounting bridge over the two tin can antennas. Sliding target made with aluminum foil wrapped cardboard for distance sensing test is attached on a camera slider rail for cm accuracy movement.

Stably mounting the swivel antennas on top of the round cylindrical cans was a challenge. A custom-made antenna bridge was built using 3D printing with PLA plastic. The bridge shown in Fig.6 illustrates the functions of this bridge not only to hold the four antennas together but also to provide some distance between antennas to reduce the magnetic coupling between adjacent antennas.

The full experiment platform to study radar with centimeter resolution requires a reliable setup with accurate distance control. As shown in Fig.6 this is by mounting the target on a camera slider rail. The radar is mounted at a fixed location on the camera slider. The SDR transmit port is connected to one antenna with a coaxial cable, and the receive port is similarly connected to the other antenna. The radar must first be calibrated by setting the target at a fixed position and “zero-ing” the transmit and receive to be perfectly aligned in phase. All distances are measured from this zero point, which was set at 30 cm from the ends of the TX/RX cans.

The Doppler speed sensing test setup in Fig. 7 uses spinning fan blades to emulate a constant-velocity object moving toward or away from radar. Although moving the sliding target in Fig. 8A can generate positive and negative Doppler frequency shift on this short track, this method using the pedestal fan blades produces a long lasting stable speed reading for better testing and demonstration. The fan with speed control can be used to induce different Doppler frequency shifts in a very repeatable manner for the experiment.

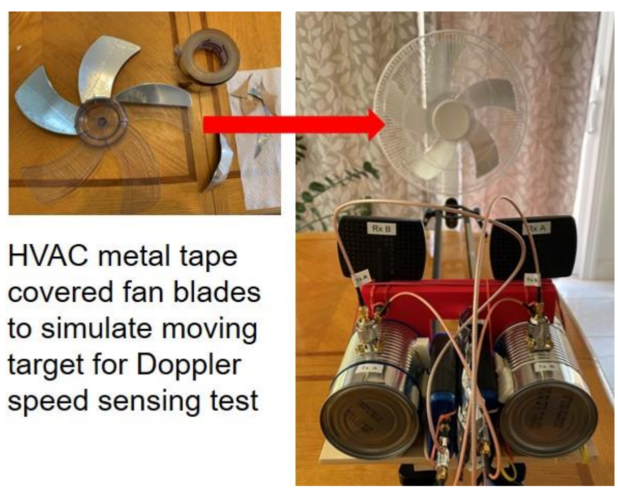

Fig. 7 For the Doppler speed sensing test, a pedestal fan is used. The plastic fan blade is covered with HVAC metal tape making it reflective to the radar signal.

2.3 Distance Test Results and Erroneous Result Correction Algorithm for MFCW Radar

The procedure used to test radar accuracy was done by moving the target from 30cm to 80cm away from the antenna in 1cm increments. Although the MFCW theory demonstrated in the publication [9] showed a simple linear phase subtraction, the earlier version of the distance sensing algorithm showed an issue generating impossible negative values for the calculated distance in specific ranges in Fig. 8.

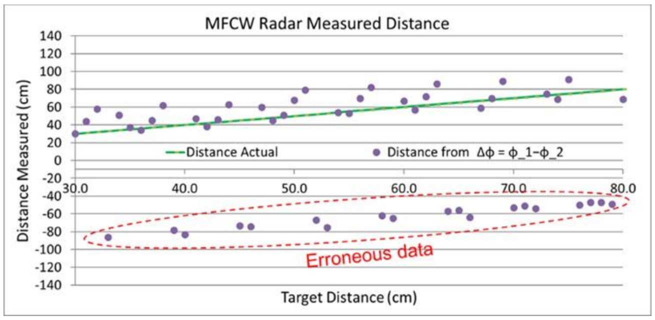

Fig. 8 Erroneous result from MFCW distance sensing algorithm using a simple linear phase subtraction.

As a simulation, the phase, phase difference, and calculated distance were plotted over target distance as illustrated in Fig. 10. Each signal's phase is a periodic function of distance as a sawtooth wave from -π to π. Analysis revealed that erroneous delta phase results were exactly lowered by the amount of 2π, which caused the calculated distance to be negative. In the previous year's project, the first improvement attempt to this algorithm was to add a second display with every raw data shifted up by 2π. As shown as the bottom plot of Fig. 9, when the two data streams are combined, the distance reported will become a straight line in the middle with points coming from both curves. In other words, the desired line in the middle was formed by some points from the regular $\Delta\phi = \phi_1 - \phi_2$ (when $\phi_1 \geq \phi_2$) and some other points from $\Delta\phi' = \phi_1 - \phi_2 + 2\pi$ (when $\phi_1 < \phi_2$).

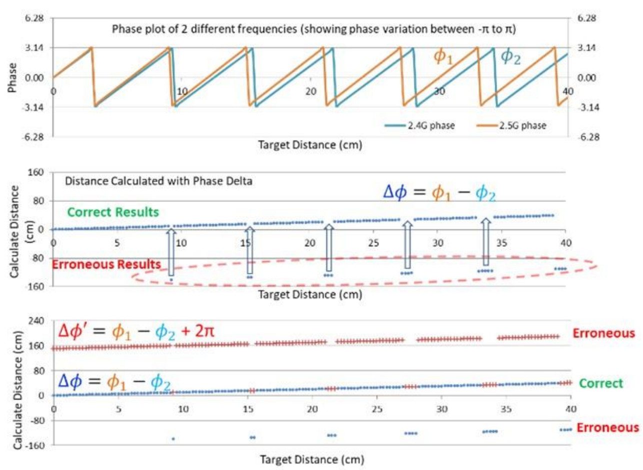

Fig. 9 A simulation to illustrate the erroneous result from MFCW distance sensing algorithm using a simple linear phase subtraction. Each signal's phase as a periodic function of distance forms a sawtooth wave from -π to π, the erroneous distance calculation will show up when $\phi_1 < \phi_2$, lowered by the amount of 2π.

With GNU Radio's data flow chart style programming, there is no "if" or other conditional data processing to process the live data stream in a selective manner. After several mathematic modeling trials, an algorithm was successfully created to automatically merge raw and corrected data streams and only output the correct distance value with disturbing the GNU Radio data flow. First, a complex exponential "Multiply Conjugate", $e^{i(\phi_1)} * e^{i(-\phi_2)} = e^{i(\phi_1-\phi_2)}$, was used instead of linear phase subtract. This helped the delta phase domain stay in [-π, π] instead of the [-2π, 2π], although this will still create negative phases. The next step is to add 2π to $\phi_1 - \phi_2$ which transposes the domain up. As a result of this 2π transpose, the original correct data in the [0, π] range will be placed at [2π, 3π], then a "modulo of 2π" is applied to folding it back down to the correct [0, π] position. On the other hand, the original erroneous data in the [-π, 0) range will be placed in the corrected [π, 2π) range by the first step of adding 2π. The subsequent "modulo of 2π" operation will have no effect in the [π, 2π) range. In summary the follow math operation can dissect the phase measurement by two ranges and position them to the proper value in the [0, 2π) range for desired distance calculation:

$$Corrected\ Phase = Modulo\ [(2\pi+(\phi_1-\phi_2)),\ 2\pi]$$

The Fig. 10 showed the final implementation of this phase to distance calculation algorithm in GNU Radio program. In this case, only the correct phase will be display. In other words, the phase correction algorithm created in this project can patch up the "holes" in the radar distance sensing range.

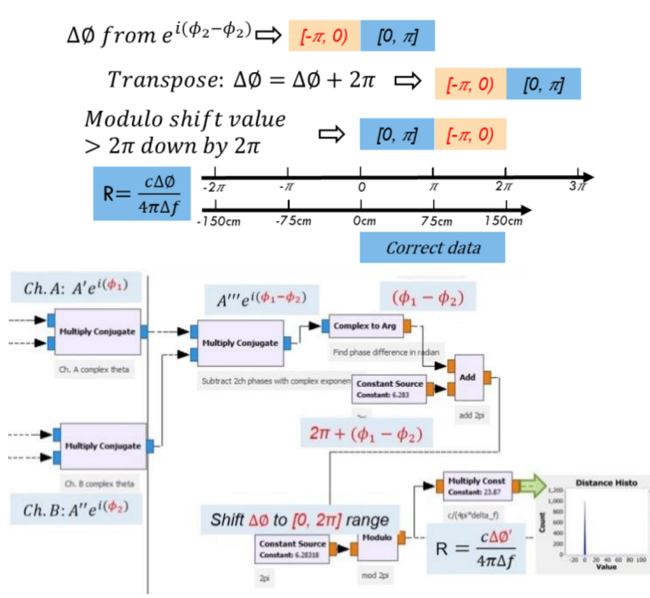

Fig. 10 An algorithm to automatically merged the raw and corrected data streams using Modulo [(2π+($\phi_1 - \phi_2$), 2π].

The distance sensing data shown in Fig. 11 demonstrated the complete removal of erroneous values. The MFCW radar achieved a 25cm distance sensing accuracy. This

25cm accuracy is already incredible considering its SDR can only provide a 4MHz bandwidth (not to mention that very little of that bandwidth is used in MFCW). Using other algorithms, such as Pulse and FMCW radars, would give this SDR radar a much worse theoretic range resolution of 3750cm. However, the measured distance showed a periodic oscillation that was actually a result of SDR Tx to Rx leakage which can be removed to achieve even better accuracy.

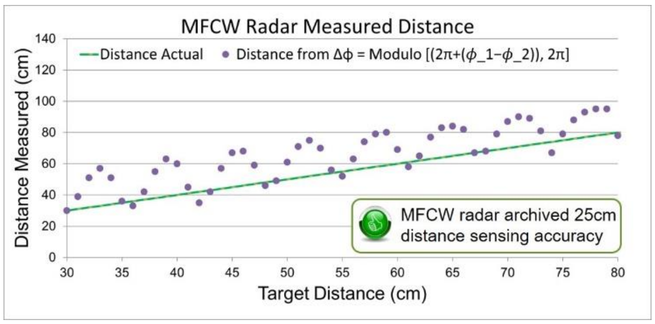

Fig. 11 Raw data from the automatic phase correction algorithm showing no more erroneous negative distance value and only a single display with the correct phase.

2.4 Distance Test Accuracy Improvement with Leakage Cancellation Calibration

As shown in Fig.1, for a two channel MFCW SDR radar system [1], there are two types of leakage or crosstalk – cross channel leakage and single channel leakage. As a result, the radar signal received from the target reflection is contaminated with the leakage signals. The cross channel leakage can be easily quantified by reduce Ch. A Tx gain or even turning it off and observe the amplitude variation in Ch. B, and vice versa. In this MFCW radar setup, this cross channel variation is negligibly small (as the SDRs are transmitting and receiving distinct frequencies). The dominant leakage is the single channel leakage from the Tx to Rx within the same SDR.

To simulate the impact of the leakage signal, the simulation capabilities in GNU Radio Companion are used. As shown in Fig. 12, the Pluto SDR source and sink are replaced with a direct connection to the signal source with programmable delay based on a variable called "target_distance_cm" and 2.5GHz/2.4GHz carrier rate to emulate the received signal. On the other hand, a leakage signal is generated using the same signal source as Tx multiplied by a 20% or 50% factor to set the leakage signal amplitude, and then a delay stage to adjust leakage phase. The "adder" sums up the received the signal and the leakage signal for the phase/distance calculation blocks. By sweeping the target_distance_cm value in 1cm increment, the distance calculation is recorded just like how the real radar accuracy is evaluated. The distance accuracy degradation from these single channel leakage scenarios are periodic and with 14cm maximum deviation as shown in Fig. 12.

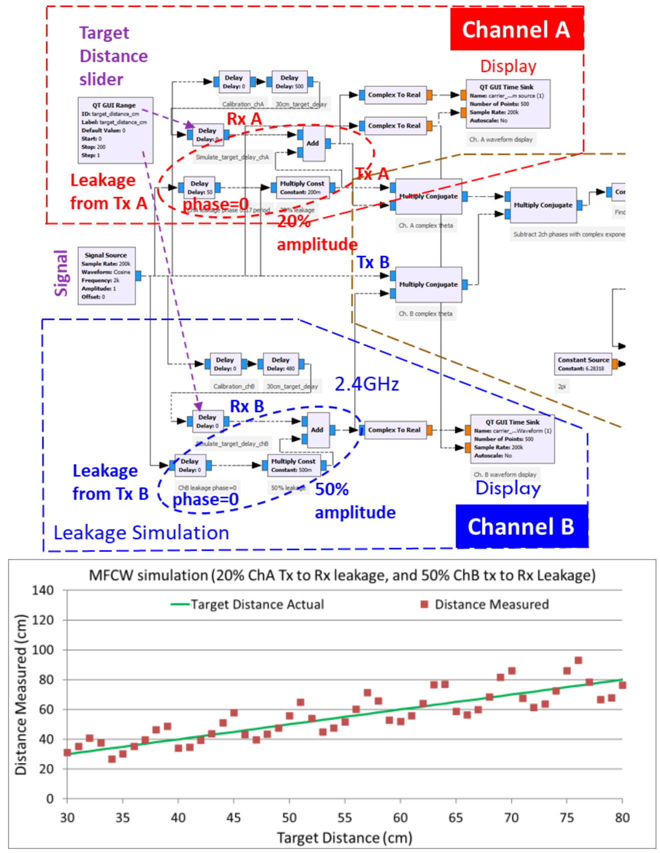

Fig. 12 Simulation program in GNU Radio Companion to estimate SDR single channel leakage. For Ch.A, a 20% leakage is added. For Ch.B, a 50% leakage is added. The simulated received signals are created with programmable delays based on a variable called "target_distance_cm" and carrier rates (Ch.A=2.5GHz, Ch.B=2.4GHz). The leakage caused a max 14cm error.

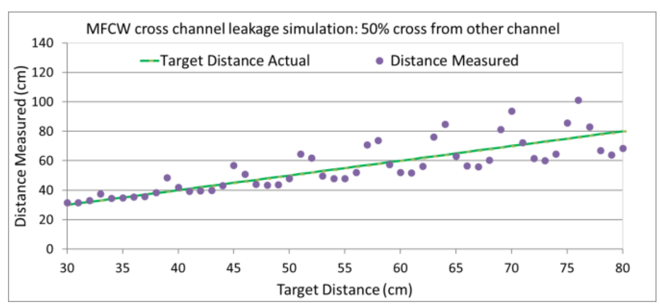

Fig. 13 Simulation program in GNU Radio Companion to estimate cross channel leakage. For Ch.A receiver, a 50% leakage from Ch.B added. For Ch.B receiver, a 50% leakage from Ch.A is added.

Similarly, a cross channel leakage simulation is performed simply by directing the Ch.B Tx signal into Ch.A Rx, and vise versa. The result in Fig.13 showed a periodic error that increase with the target distance with a maximum of 24cm.

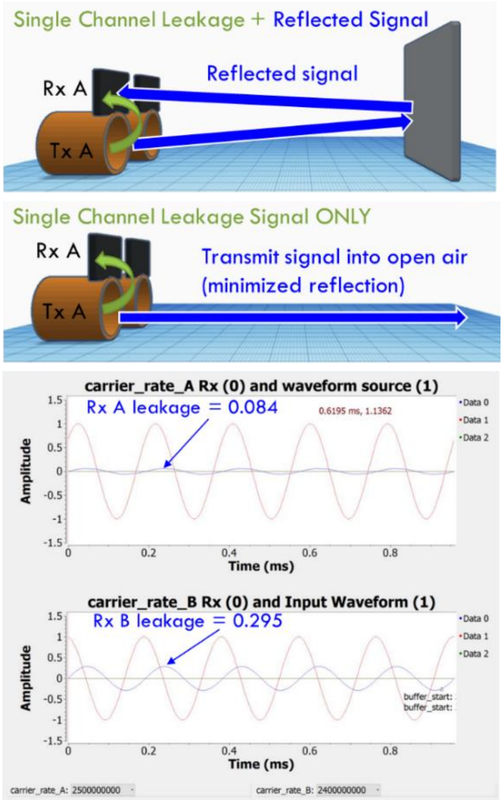

Fig. 14 Proposed technique to quantify the leakage signal within the single SDR is to remove the target and observe the pure leakage signal.

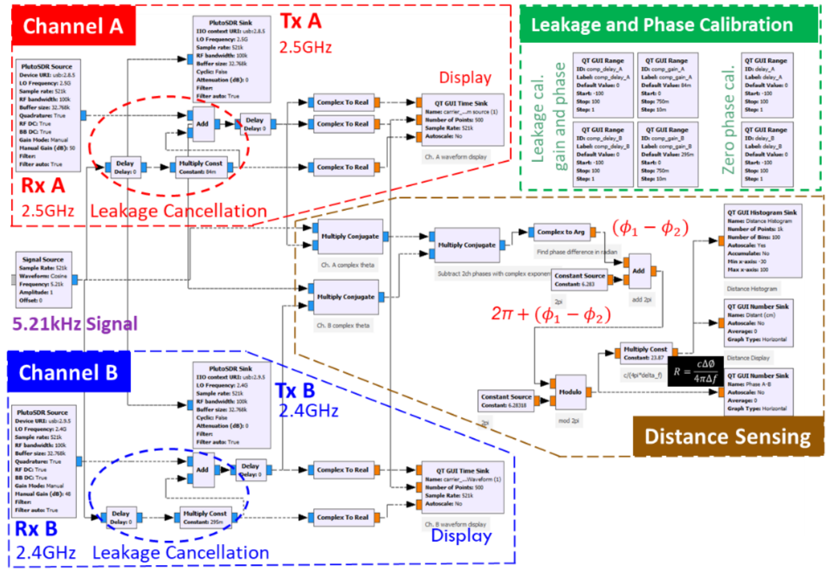

Fig. 15 Improved MFCW radar algorithm with leakage compensation logic added in the two oval circles. The negative image used for leakage cancellation is controlled by the gain and phase GNU GUI Range Sliders in green rectangles during the open air leakage calibration process.

When the target is in place, the signal at the Rx is a combination of the reflected and the leakage signals as shown in the upper diagram of Fig. 14. Single channel leakage can be revealed by remove the metal target on the slider and let the radar transmit the signal into open air as shown in lower diagram in Fig. 14. By removing the target, the reflected signal is minimized and the Rx signal displayed is primarily the leakage signal, so the Ch.A and Ch.B signals captured in Fig. 14 are the undesirable leakage that should be cancelled out.

Because these leakage signals are the same frequency as the 5.21kHz signal used to modulate the two SDR, we can simply add a new offset signal equal in amplitude, but 180° phase shifted. This is the principle behind the leakage cancellation algorithm that we created and implemented in GNU Radio as shown in Figure 15. The enhanced design added a compensation signal with amplitude and phase control by GNU Range Sliders during the open air calibration. First the phase is adjusted to make the compensation signal 180° in phase with the leakage, which will minimize the received signal (i.e. blue curve) in the display. Second, by properly adjusting the compensation signal amplitude to match the leakage signal, the Rx signal will become a completely flat line (i.e. no reflection detected, as desired). As shown in Fig. 16, the blue Rx signal is flatten out, the green signal is the negative image of the leakage signal being added to cancel the leakage.

After this open air leakage calibration/cancellation process, the target is placed starting at a known location at 30cm to perform the usual zero phase calibration, and then followed by a sweep of the distance from 30cm to 80cm in 1cm increments to evaluate distance sensing accuracy. The distance accuracy is significantly improved with the leakage cancellation as shown in Fig.17. Radar-measured distance data after Leakage cancellation performed with open air calibration showed a 12cm accuracy in 30~80cm range.

To quantify the single channel leakage at 30cm away from the radar, the received Rx signal after leakage cancellation and the leakage signal are plotted for comparison in Fig.18., indicating a -14dB leakage in channel A, and -6.9dB leakage in channel B. In terms of amplitude, they are 20%-45% of reflected signal, which proved why the phase measurement were distorted by these strong leakage terms and caused the distance measurement error if not properly compensated. That explained why in Fig 12, even at just 3cm away from the 30cm calibration mark, a large 25cm error occurred. In comparison of Fig 17 with the leakage cancellation, from 30~55cm range near the 30cm calibration point the measurement error is less than 5cm.

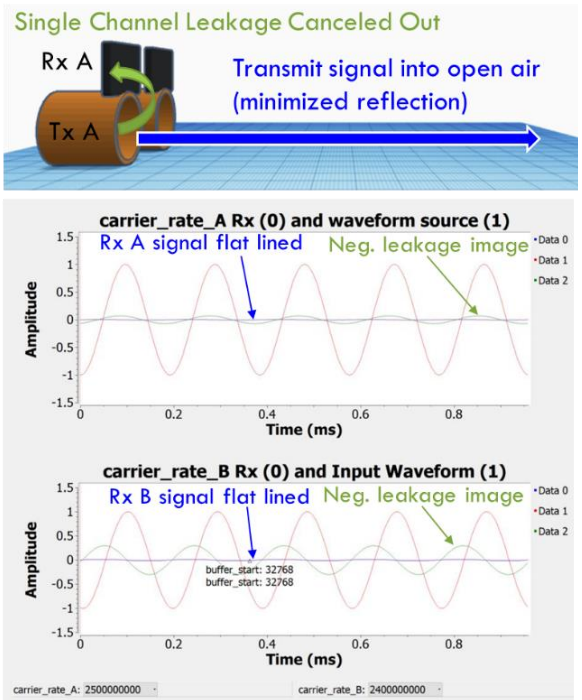

Fig. 16 After this open air leakage calibration/cancellation process, the blue Rx signal is flattened out. The green signal is the negative image of the leakage signal being added to cancel the leakage to indicate the actual leakage signal strength before being removed.

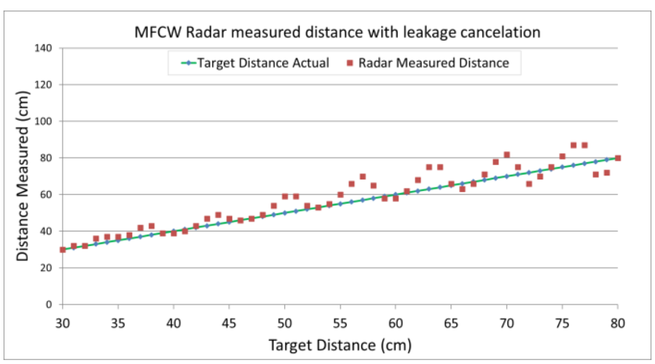

Fig. 17 Radar measured distance data after Leakage cancellation performed with open air calibration with target removed. The distance sensing accuracy significantly improved to 12cm in 30~80cm range.

Beyond the 55cm mark, this almost linear curve started to show a small ripple near the far end. This residual ripple at the far end of the track is believed to be the second order effect from so called short-range leakage. Besides leakage is within the radar device itself, signal reflections from a fixed object in front of the antennas besides the target will cause this short-range leakage. In this case, although the camera slider track is made from non-conducting carbon fiber tubes, the base of the camera/target mount is made of metal that can cause reflection. When the open air calibration is used to remove leakage, although the target is removed, the metal base is still in the 30cm location in front of the radar. Therefore, the metal base reflection created a short-range leakage [12] that is also cancelled out by the open air calibration. However, the metal base is moving with the target away from the radar. When the same value is used for leakage cancellation that was derived with base at 30cm mark, at the far end 50~80cm mark with the short range leakage from the metal base reduced, a slight over compensation effect started to show up with ripple effect gradually increase with the distance.

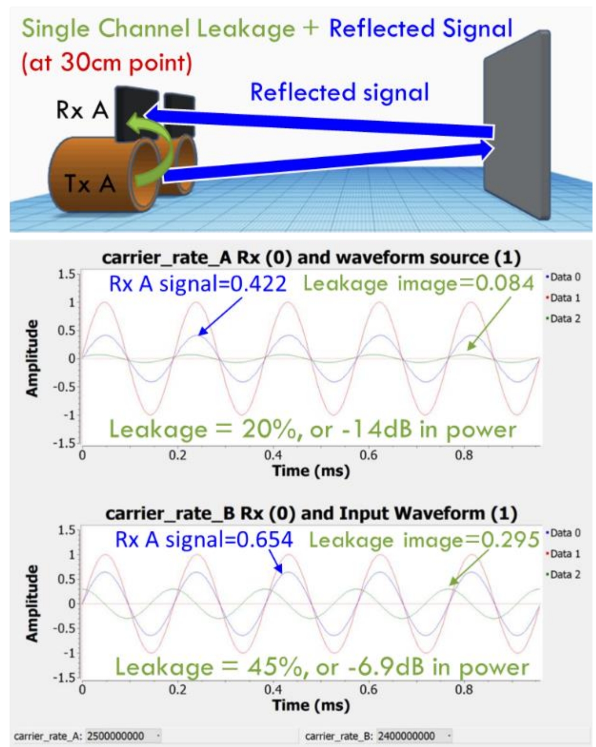

Fig. 18 To quantify the single channel leakage, the received Rx signal after leakage cancellation and the leakage amplitude is plotted in comparison. Indicating a -14dB leakage in channel A, and -6.9dB leakage in channel B.

2.5 Test Results for MFCW Radar Speed Sensing Capability

Doppler shift is the frequency shift of the reflected wave when compared to the transmit signal, which contains the velocity info for the target. As shown in Fig. 18, if the target is moving towards the radar, a positive frequency shift will be detected. For this project, as shown in Fig. 7, the Doppler speed sensing test uses a pedestal fan. The plastic fan blade was covered with HVAC metal tape making it reflective to the radar signal.

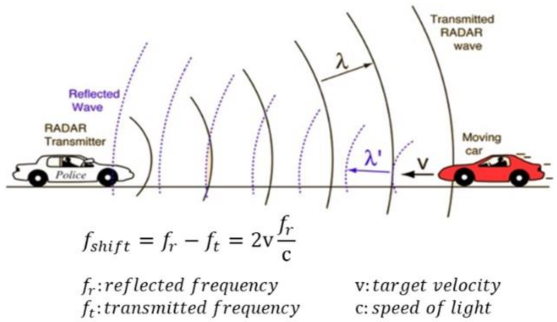

Fig. 18 The principle of Doppler radar operation and equation to link speed to frequency shift. [13]

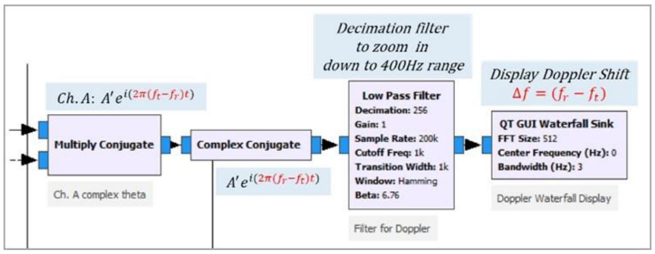

Fig. 19 The Doppler algorithm tagged on the 2.5GHz channel of MFCW for speed sensing.

Since the MFCW radar has two continuous wave channels and Doppler radar only needs one, adding the speed sensing capability to MFCW radar is simply a task of reuse the radar signal from any of the two channels to detect frequency shift instead phase shift used for distance sensing. As shown in Fig. 3 and Fig. 19, there are only 3 blocks used in the GNU Radio program for Doppler radar. The same "Multiply Conjugate" blocked used for derive channel-A phase delta is used for finding the Doppler frequency shift, and the next "Complex Conjugate" is really just an inverse the sign to get $\Delta f = (f_r - f_t)$. The decimation filter is the key to Doppler design. Because the frequency shift is very small, we need the decimation to get the FFT operation to zoom in down to the <400MHz range. The GNU Radio's "Frequency Waterfall" display is simply a way to plot spectrum history in color as shown in Fig. 20. The old spectrum data keeps rolling up, while new spectrum data appears at the bottom. Fig. 20 records the following events: the fan was turned on to half speed, switched to full speed, turned off, and then switched blowing direction to point away from the radar at half speed. At maximum fan speed, an 84Hz frequency shift was detected which translates to 5 meters per second in velocity. When the fan switched direction to blow away from the radar, the blade movement generates a negative frequency shift as expected.

In summary, the MFCW radar can produce velocity sensing simultaneously with distance sensing. It only involves additional computation blocks without any new hardware required. This is the advantage of the SDR systems where different radar algorithms can be explored with just a change to the software.

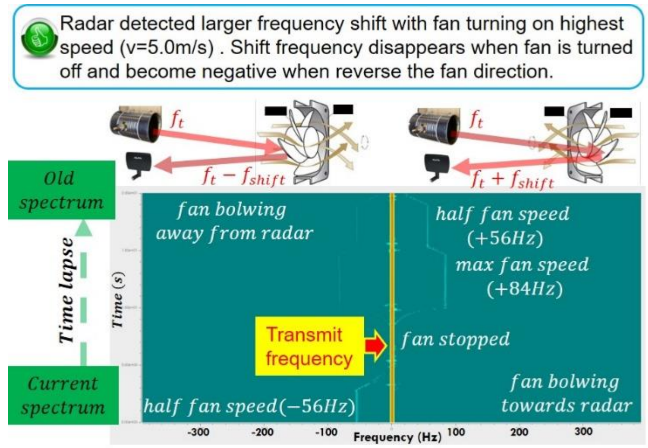

Fig. 20 The Doppler radar speed sensing result. Frequency waterfall display is used to record frequency shift for speed calculations.

4. Conclusion

In conclusion, a new interferometry radar solutions MFCW is demonstrated to make distance sensing possible using very bandwidth limited SDR, enabling more accurate and low cost SDR based short-range radar distance and speed sensing applications. This first MFCW distance sensing radar using low cost SDR achieved range resolution of 15cm while only using a few KHz BW. In comparison, a traditional Pulse or FMCW radar would have to use almost 1GHz BW to achieve the resolution. The achievements from this project include 5 magnitudes less bandwidth usage, an innovative distance sensing algorithm avoiding erroneous phase results, and a simultaneous distance and speed sensing system.

5. Acknowledgements

In November 2020, the coauthor Victor Cai was fortunate enough to find a Co-Op position and a great mentor Jon Kraft at Analog Devices Inc. (ADI). After reviewing my independent dual-SDR MFCW research, he guided me into simulating channel leakage with GNU, which sparked my leakage cancellation idea in MFCW to resolve the lingering periodic variation in the distance measurement.

References:

[1] Cai, V. (2020). Designing a Narrowband Radar using GNU Radio and Software Defined Radio for Tomography and Indoor Sensing. Proceedings Of The GNU Radio Conference, 5(1). Retrieved from https://pubs.gnuradio.org/index.php/grcon/article/view/67.

[2] Suleymanov, S. (2016). Design and implementation of an FMCW radar signal processing module for

automotive applications [Unpublished master's thesis]. U of Twente.

[3] Adib, F., Mao, H., Kabelac, Z., Katabi, D., & Miller, R. C. (2015). Smart homes that monitor breathing and heart rate. Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems.

[4] Marimuthu, J., Bialkowski, K., & Abbosh, A. (2016). Software-Defined radar for medical imaging. IEEE Transactions on Microwave Theory and Techniques, 64(2), 643-652.

[5] Adib, F., & Katabi, D. (2013). See through walls with wi-fi! SIGCOMM 2013.

[6] Wolff, C. (n.d.). Range resolution. Radartutorial. Retrieved June 17, 2020, from http://radartutorial.eu/01.basics/Range%20Resolution.en.html

[7] Collins, T. F., Getz, R., Pu, D., & Wyglinski, A. M. (2018). Software-Defined radio for engineers. Artech House Mobile Communications.

[8] Zhu, J. (2017). Low cost, software defined FMCW radar for observations of drones [Unpublished master's thesis]. U of Oklahoma.

[9] Anderson, M. G. (2018). Design of multiple frequency continuous wave radar hardware and micro-doppler based detection and classification algorithms. University of Texas at Austin: Texas Scholar Works. Retrieved October 10, 2019, from https://repositories.lib.utexas.edu/handle/2152/4000

[10] Charvat, G., Williams, J., Fenn, A., Kogon, S., & Herd, J. (2011, 01). Build a small radar system capable of sensing range, doppler, and synthetic aperture radar imaging. MIT Open Course Ware. Retrieved 09 10, 2019, from https://ocw.mit.edu/resources/res-ll-003-build-a-small-radar-system-capable-of-sensing-range-doppler-and-synthetic-aperture-radar-imaging-january-iap-2011/

[11] Frank, A. (n.d.). Cantenna Calculator Aka Circular Waveguide Antenna [Computer program]. https://www.changpuak.ch/electronics/cantenna.php

[12] Melzer, A., Onic, A., Startzer, F., & Huemer, M. (2015). Short-range leakage cancelation in FMCW radar transceivers using an artificial on chip target. Journal of Selected Topics in Signal Processing, Vol. 9, No. 8.

[13] Freitas, L. (2018, November 23). Software defined doppler radar with limesdr mini. Software Defined Doppler Radar with LimeSDR Mini. Retrieved October 10, 2019, from https://luigifreitas.me/2018-11-23/software-defined-radar-cw-doppler-radar-with-limesdr