Russell Silva

5/15/2018

AMRUPT, Spring ‘18

**Final Report Draft**

I. Executive Summary

Animal Movement Research Using Phase-based Trilateration (AMRUPT) is a technology being designed for the localization of small animals in the field of ecology. This is used for the study of flight patterns, social interactions, or other biological attributes to most species. The system utilizes Phase Interferometry for use in estimating the Angle of Arrival (AOA) of radio signals. These systems are substantially more accurate than other common methods, although performance scales strongly with the spatial scale of the receiver network. Because many researchers are interested in small-scale movements of animals within populations, such a system may be extremely useful. To accomplish this, a low weight radio tag is being developed to transmit signals to radio base stations. These tags will transmit sub 1-GHz UHF frequencies. The phase information of these RF signals will be calculated on the RTL SDR. This requires multiple receive antennas connected to an RF switch which is attached to the RTL SDR which communicates the I & Q values of the received to a Raspberry Pi for angle of arrival calculation.

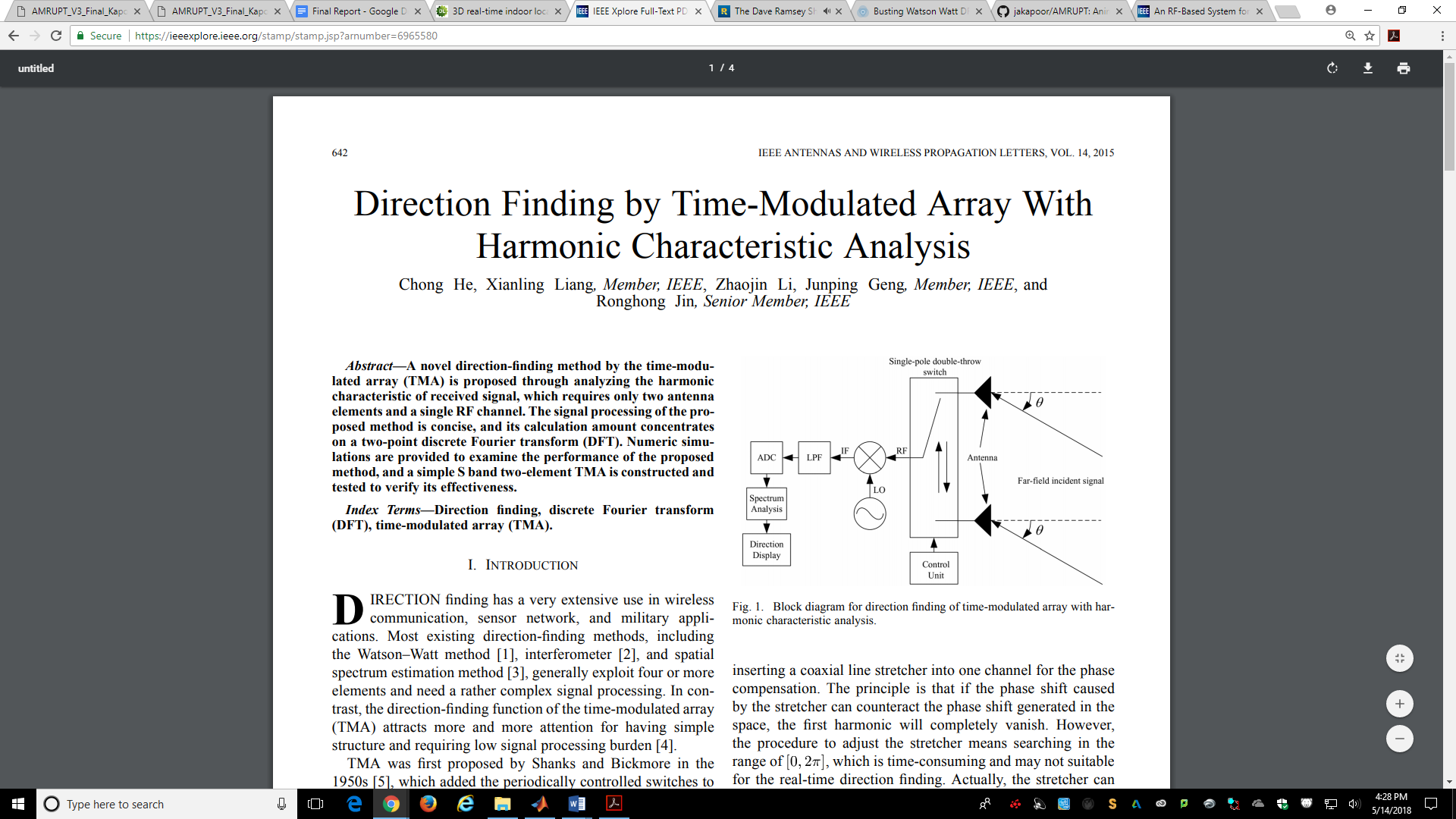
II. Introduction

The localization of small animals in the field of ecology is imperative to determining the flight patterns, social interactions, or other biological attributes to most species. Many attempts[1,2,3] have been made to determine the positioning of animals temporally and spatially in the past, but have been either inaccurate (errors over five meters) or have required constant manual human intervention. Since direction finding requires wireless telecommunication, measurements have been thwarted by multipath interference from vegetation, electromagnetic interference, or other environmental conditions. Our objective is to develop a cost effective and automated system to track animal movements within the range of five meters while taking into account expected causes of error. Our proposed system consists of a receiver architecture that is built specifically for phase interferometry direction finding to facilitate accurate measurements from radio tags on tracked individuals.

III. Review of Literature

Many different techniques have been explored to achieve localization. Transmitted signals at antenna array elements can be quantized at receivers to provide phase difference information such as in phase interferometry [4,6]. Phase based measurements can be skewed by multipath effects in the environment by constructive and destructive interference for line of sight signals [6]. Received signal strength can be also be used for localization through several different algorithms however these metrics have been found to be unreliable [5] and thus do not have high accuracy. Time Difference of Arrival (TDoA) [1] where the position of a transmitter is determined cooperatively by a group of receiver stations from the differences in the time at which a short-lived transmission reaches. TDOA systems are not as susceptible to multipath effects; however, obtaining precise positioning from close proximity transmitters in TDOA is difficult because nanosecond synchronization is required to compare lightspeed propagated signals. This flaw in time difference of arrival can be mitigated by subsample interpolation at signal correlation peaks [1]. Another bottleneck with TDOA is sampling rate as our timing difference measurements are only as accurate as our maximum sampling rate.

Although lightspeed propagation substantially helps with obtaining real time results, it adversely affects the collection of synchronized data at antenna array elements in the radio frequency direction finding systems mentioned. More intensive hardware synchronization can be avoided by using a time-modulated array to switch between antennas in a direction finding system [7], shown in Figure 1.



**Fig. 1. Block diagram for direction finding of time-modulated array with harmonic characteristic analysis**

In coherent receivers running on a single clock signal, synchronization errors from clock skew and other delays can be corrected by incorporating a signal generator input to each channel [8]. In addition, clock drifts and bulk delays can be mitigated by implementing cross correlation with virtual sources on a software level [9].

A phase interferometry system with real time operation on multiple receivers driven by a common local oscillator was able to identify an angle of arrival within degrees for all emitter distances from 1 km to 100 km [1]. Because this system was driven by a common local oscillator, the resulting intermediate frequency at each receiver could be compared without phase-offset synchronization. The error .5± 2 threshold from this system is less than the degree angle of arrival threshold error determined ± 5 in the time-modulated approach [21]. Although, this error rate is promising, this receiver system was designed to receive line of sight airborne signals. This does not account for the adverse effects of a ground environment, which will add additional factors to AoA error such as multipath interference.

The phase interferometry system in [4] was primarily developed to handle UHF frequencies for airborne sources. Because the frequency of this system was relatively high for phase interferometry, antennas were spaced at distances larger than half the wavelength of transmitted signals. Thus, the system was optimized to handle phase ambiguity, which is explored more in the technical section of this proposal. Testing protocols and optimization were handled in Matlab and C, modeling the effects of antenna spacings on AOA accuracy under worst-case conditions. In the hardware setup, three antennas were used to resolve phase ambiguities and determine the azimuthal AOA in a synchronized three channel system with RF mixers driven by a common local oscillator. The design of this system provides a basis for a three antenna architecture as discussed in the technical section of this proposal. The block diagram of this system is shown in Figure 2.

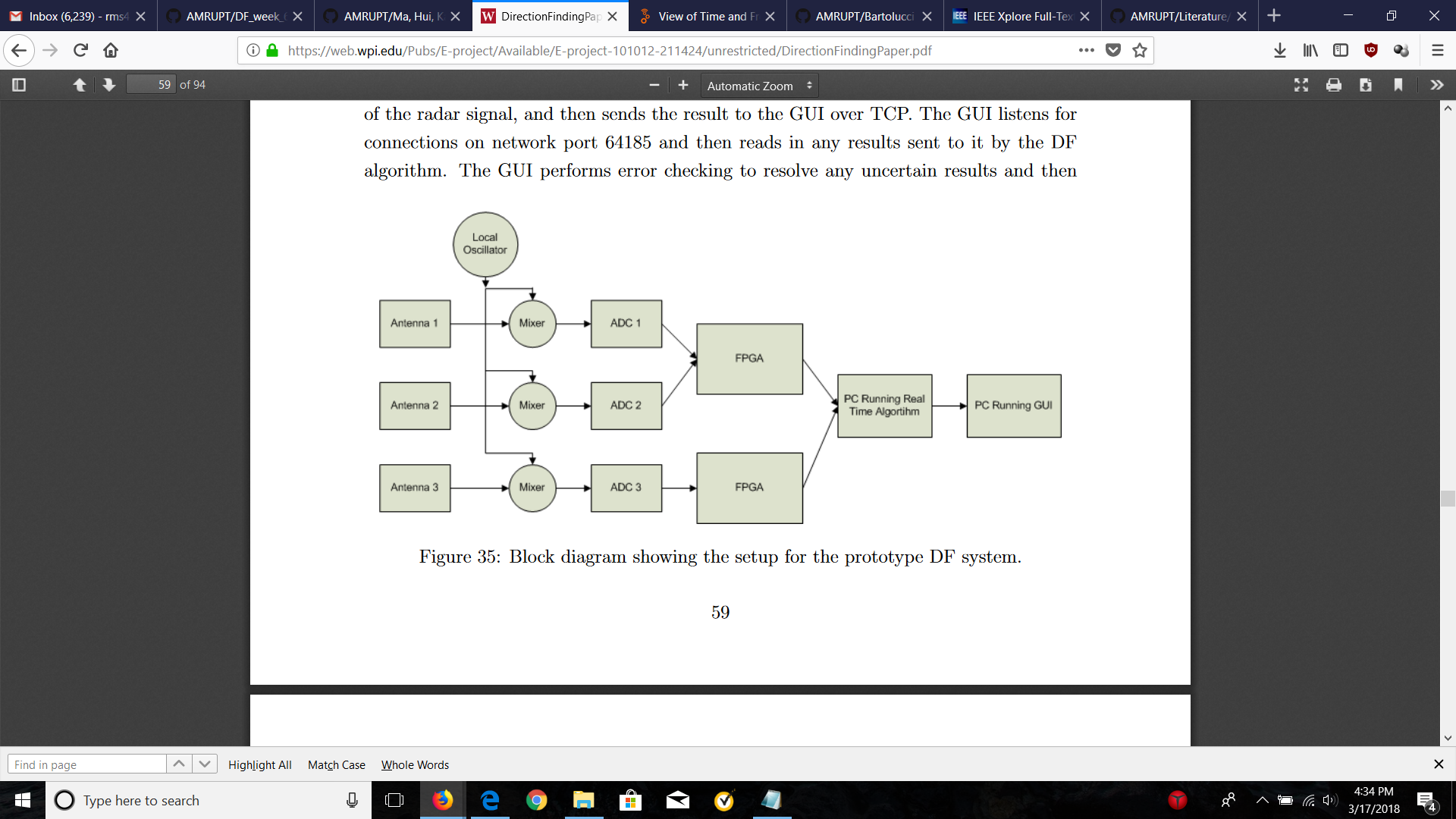


Figure 2: Block diagram for Phase Interferometry in Guerin, Jackson, and Kelly [4]

Ma, Hui and Kan [6] proposes a 3D indoor passive tag localization method with an accuracy of a few centimeters in a multi-frequency identification system. The paper leverages nonlinear elements in passive devices to generate second or higher-order harmonics which are used for location detection. This paper introduces a novel approach in mitigating multipath interference, defined as the occurrence when radio waves reach a receiver via two or more paths. This causes a constructive and destructive interference of the signal, as well as phase induced error in Figure 3.

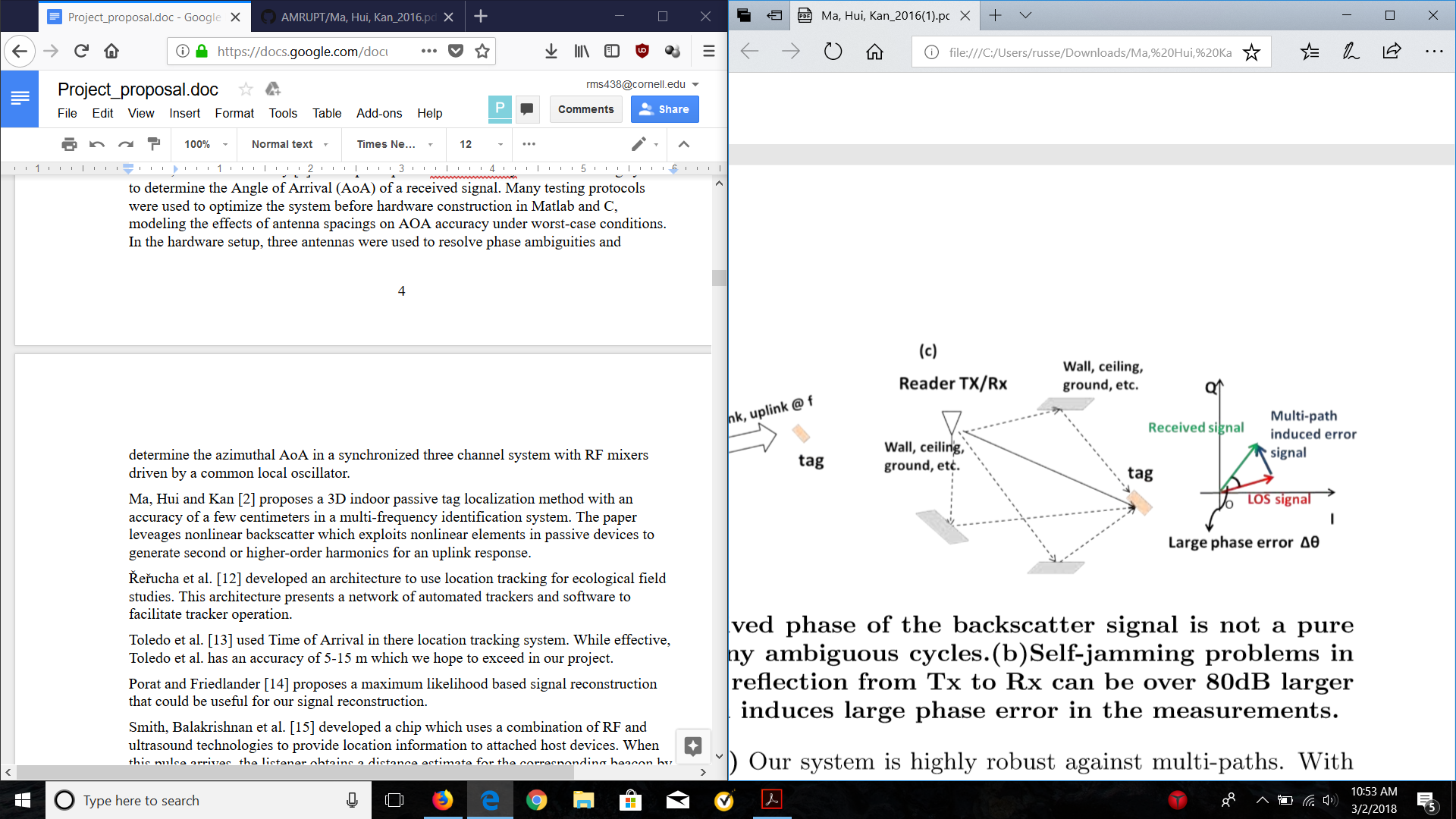


Figure 3: Dense indoor multi-path induced phase error [6]

In order to combat multipath interference, a phase error threshold is used. An algorithm is implemented which uses heuristic multi-frequency continuous wave (HMFCW), where the frequency is dynamically adjusted to identify an optimal frequency based on the phase error threshold. The optimal frequency allows an undistorted line of sight path. HMFCW ranging correctly pins down a phase cycle integer by transmitting a sequence of frequencies determined by a genetic algorithm to maximize an error tolerance equal to the percent bandwidth of the signal. The larger the bandwidth of the multi-frequency transmission, the more robust the system will be to multi-path induced phase error. The forward compatibility of this system to ours is discussed in the design objectives section of this proposal.

In order to improve the cost effectiveness of direction finding, [1] has used low-cost RTL SDRs to extract in-phase and quadrature samples from incoming radio signals for TDOA and AoA calculations respectively. Direction Finding Implementations using RTL SDRs are promising alternatives to more expensive options by achieving up to 3.5m accuracy in TDOA [1] and by having an extensive hobbyist base with multiple Github repositories such as this one [10], demoed here. The advantages of having this repository available to us is that it will provide us with a point of reference when implementing our code and hardware. This specific repository was a precursor to the RTL SDR system developed by Sam Whiting in [11].

The TDOA approaches discussed above have been promising, as an ultra wide bandwidth transmission would be less susceptible to signal refractions from a cluttered environment. The TDOA approach implemented with RTL SDRs in [1] does have error thresholds within 5 meters; however, the extensive digital signal processing (DSP) used to achieve these error thresholds will likely be too computationally expensive for portable embedded devices such as the Raspberry Pi, and will result in more power hungry measurement analysis (~3.7 Watts). We choose the less DSP heavy phase interferometry approach for its potential for high accuracy with low frequency transmissions in cluttered environments, which can be further improved upon by the multi-frequency techniques in [6]. Furthermore, triangulation error can be scaled lower with an increased number of receivers per unit area with angle of arrival calculations. This option will be effective with low cost and on site programmable receiver units such as the RTL SDR.

IV. Design Requirements

We have proposed the following objectives in the design:

1. The receiver system is low-power and can track up to 50 lightweight and low-power radio tags
2. System architecture is resilient in cluttered environment (unsusceptible to multipath interference, electromagnetic interference, and other environmental conditions)
3. System is able to achieve two dimensional high spatial accuracy (error for triangulation results is limited within 5 meters) with a 100-300m distance between receivers
4. Forward compatibility: Must be compatible with and adaptable to a multi-frequency-phase-integer-disambiguation approach for future versions.
5. System is cost-efficient (almost all components are commercially off-the-shelf)

The first objective is to successfully track the locations of 50 individuals in the testing environment. We need to design the tags as lightweight as possible since the individuals are small in size and heavy tags may affect the individuals’ biological activities. To allow for the least possible human intervention during the tracking process, both the receivers and tags need to operate with minimal power consumption to increase automatic tracking period. In addition, both the transceivers (ground nodes) and tags (mobile nodes) follow a communication protocol in which the mobile nodes will go to sleep when they are not communicating with the ground nodes to reduce power consumption.

The communication protocol is an intended route for development, but has not yet been designed. It specifies that mobile nodes wake up every 5 minutes to prepare for data transmission to the ground nodes. The mobile node will receive a 5-second countdown signal once it wakes up. As soon as the mobile node is verified to be within the receiver’s range and has good link, it will be synchronized to global time before it is given a scheduled transmission time by the receiver or sent back to sleep again. If the mobile node is not within range of any receiver, it will go to sleep and wake up every 5 minutes to check whether it’s within range again. The complexity of the ground-mode to mobile-node communication protocol will be governed by how accurate our receivers are when taking angles of arrival measurements. If angle of arrivals from a at least three base stations intersect to a triangulation area of no more than 5 meter error (discussed further) over the specified tracking area, then tags will not have to be linked to different receivers depending on location and the same set of receivers can be used for all tags. The communication protocol will also be used for a multi-frequency system, which is a possibility in the future of this project.

Furthermore, the system must be able to obtain accurate results in a cluttered environment. We agreed that a real environment would have substantial multiple interference as there will be trees and rocks that can reflect a wireless signal. Multipath interference could result a false transmit signal which would give us wrong information about the location of the tags. The effects of multipath are more amply discussed in the literature review section.

We agreed to set the tracking accuracy of our system to 5 meters because this is a minimum requirement to monitor the social interactions and movements of small mammal species and is already much more accurate than existing systems mentioned in the literature section. We propose a triangulation algorithm for Phase 1 of the project in order to acquire this accuracy: Phase 1 will be the development of the necessary base station algorithms and hardware setup to achieve an accurate angle of arrival measurement. In a future semester, we plan to modify our system to better overcome the effects of multipath interference by frequency hopping to obtain minimum variation results (Phase 2). If necessary, we plan to implement a multi-frequency phase integer disambiguation system based on [6] that trilaterates positions of mobile nodes if a 5-meter accuracy level has not been achieved by previous efforts (Phase 3). Figure 4 illustrates error minimization with relation to AOA calculations:

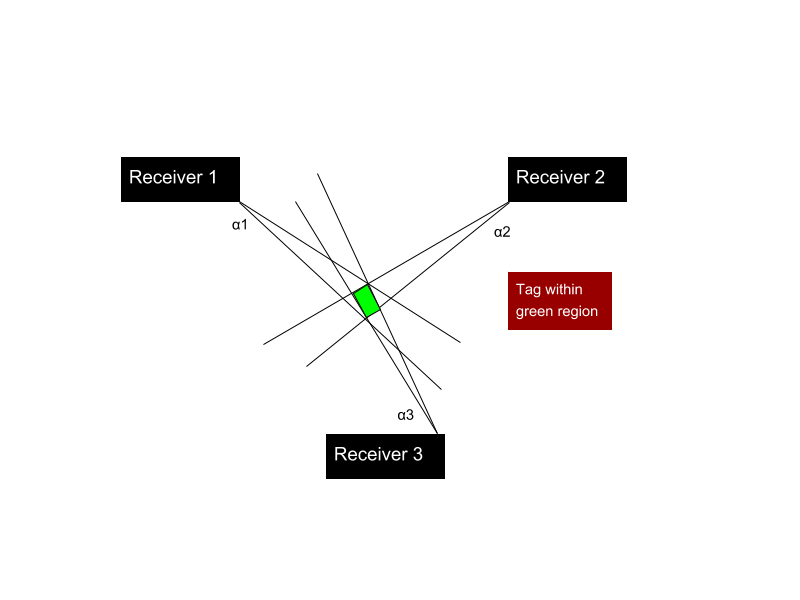


Figure 3: Error in triangulated area

In order to accomplish at least a 5 meter accuracy, a line of more than five meters cannot be drawn within the triangulated area of error . This area of error will be determined by α1, α2, and α3 (Figure 5) which resemble the angle of arrival error from receiver 1, 2, and 3 respectively. α1, α2, and α3 will be determined by phase difference errors from a transmitting RF signal to multiple antennas. Sources of AOA error are further discussed in the technical section of this proposal, and simulations have been planned to find algorithms that can make additional steps in minimizing this error.

To address forward compatibility, we must have a system in mind to distinguish tags. Code-division multiple access (CDMA) is a very popular method for doing this due to the coding infrastructure. CDMA, after digitizing data, spreads it out over the entire available bandwidth. All of the users transmit in the same wide-band chunk of spectrum. Each user's signal is spread over the entire bandwidth by a unique spreading code. At the receiver, that same unique code is used to recover the signal. CDMA is a form of spread spectrum, which simply means that data is sent in small pieces over a number of the discrete frequencies available for use at any time in the specified range. However it is unclear if CDMA is compatible with making phase-based AOA measurements. CDMA codes are mutually orthogonal to each other, so it is possible to distinguish multiple bit streams from multiple tags at the same time. However, these codes may not be able to differentiate phases.

V. Design Implementation

V.i. Architecture Overview

In order to coherently receive RF signals from a radio tag sources, we planned to use RTL SDRs connected to a common clock for synchronous data extraction. A company at coherent-receiver.com has constructed a multichannel coherent receiver system with the required components necessary for accurate direction finding. Figure 4 displays this device. The clock card is used to send a common clock signal to all four RTL SDRs. The noise generator (which can be replaced by an external function generator) is utilized during noise switching. In the noise switching process, antenna switches synchronously transition from a common noise signal and individual antenna signals. This antenna switching aids in cross correlating samples from all four single channel receivers.

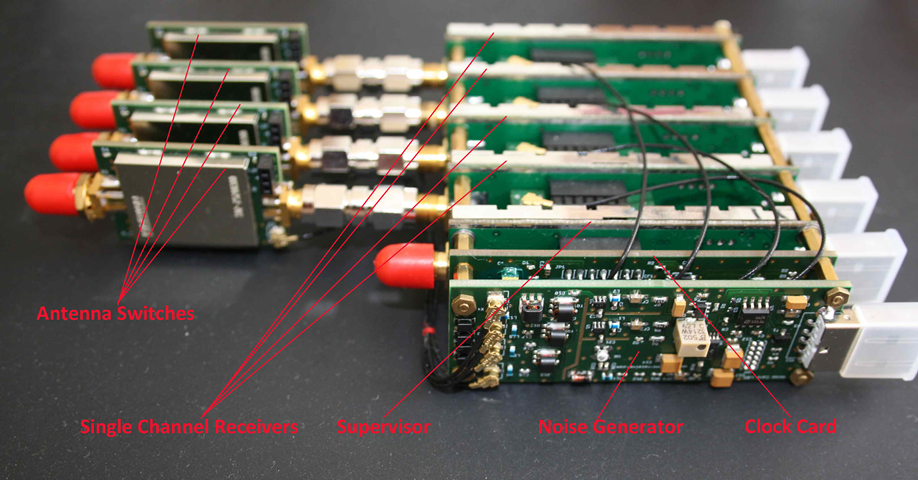


Figure 4. Coherent Receiver Components

In addition to the coherent receiver, low loss male to female SMA Connector wires were used to to separate antennas connected to the RTL SDRs on the coherent receivers. The lengths of the connectors' insulated wires accommodated distances up to one half the wavelength of the incoming VHF RF signal (anywhere from 0.5 to 2.5 meters for VHF signals).

An RF signal was continuously transmitted at 434 mHz at a narrow bandwidth from CC1310s during testing. This 434 mHz frequency was used as the lowest default option for transmitter frequencies in SmartRF Studio.

V.ii. RTL-SDR I/Q Extraction and Setup

After setting up the system described in section V.i., parallel I/Q extraction from multiple RTL SDR receivers can be tested in two ways: writing to a file from and Ubuntu terminal or by a Graphical user interface within GNU Radio.

The following code will be used for parallel I/Q data extraction within GNU Radio:

rtl\_sdr -d0 -f 1125000000 -g 35 -s 2500000 -n 50000000 -N FMcapture0-2.dat &

rtl\_sdr -d1 -f 1125000000 -g 35 -s 2500000 -n 50000000 -N FMcapture1-2.dat &

rtl\_sdr -d2 -f 1125000000 -g 35 -s 2500000 -n 50000000 -N FMcapture2-2.dat &

rtl\_sdr -d2 -f 1125000000 -g 35 -s 2500000 -n 50000000 -N FMcapture2-2.dat &

The following setup is used to view I/Q samples on a graphical user interface within GNU Radio:

Figure 5 shows the basic block diagram to display the real/imaginary values of an incoming RF signal (WVBR Ithaca radio station) in the time domain. Figure 6 shows the result of the block diagram in Figure 5.



Figure 5. Simple I/Q Extraction from RTL SDRs on GNU Radio



Figure 6. I/Q Samples in the Time Domain

V. iii. Ubuntu and OutOfTreeModules Installation

During the four week implementation period, a large portion of the time was dedicated to installing and understanding the workflow of GNU Radio’s OutOfTreeModules. We discovered that properly creating and installing out of tree modules in GNU Radio (<https://wiki.gnuradio.org/index.php/OutOfTreeModules>) required an understanding of gr\_modtool and cmake, which are much more efficient in Ubuntu file systems. In order to save us time in obtaining rudimentary AoA measurements, we decided to use custom blocks already developed by Ettus Research.

The entire installation guide is listed below:

Important Note: Steps 2 and onwards use the terminology “generic cmake instructions” to refer to this set of instructions to install github libraries into Ubuntu:

$ git clone “*github library here”*

$ cd “*installed library name”*

$ mkdir build

$ cd build

$ cmake ..

$ make

$ make test

$ sudo make install

$ sudo ldconfig

It is critical to know that **Ubuntu versions 14.04 and 16.04** are the only compatible distributions with the Ettus Doa library at this time.

0. Make Sure to update the operating system before the following installations:

pi@raspberrypi ~ $ sudo apt-get update  
pi@raspberrypi ~ $ sudo apt-get upgrade

1. Install GNU Radio by using the command sudo apt-get install gnuradio-dev Install the RTL SDR drivers, the following commands have been included in this report in case this page (https://gist.github.com/floehopper/99a0c8931f9d779b0998) gets lost in the future:

pi@raspberrypi ~ $ cat <<EOF >no-rtl.conf  
blacklist dvb\_usb\_rtl28xxu  
blacklist rtl2832  
blacklist rtl2830  
EOF  
pi@raspberrypi ~ $ sudo mv no-rtl.conf /etc/modprobe.d/  
  
pi@raspberrypi ~ $ sudo apt-get install git-core  
pi@raspberrypi ~ $ sudo apt-get install git  
pi@raspberrypi ~ $ sudo apt-get install cmake  
pi@raspberrypi ~ $ sudo apt-get install libusb-1.0-0-dev  
pi@raspberrypi ~ $ sudo apt-get install build-essential  
  
pi@raspberrypi ~ $ git clone git://git.osmocom.org/rtl-sdr.git  
pi@raspberrypi ~ $ cd rtl-sdr/  
pi@raspberrypi ~/rtl-sdr $ mkdir build  
pi@raspberrypi ~/rtl-sdr $ cd build  
pi@raspberrypi ~/rtl-sdr/build $ cmake ../ -DINSTALL\_UDEV\_RULES=ON  
pi@raspberrypi ~/rtl-sdr/build $ make  
pi@raspberrypi ~/rtl-sdr/build $ sudo make install  
pi@raspberrypi ~/rtl-sdr/build $ sudo ldconfig  
pi@raspberrypi ~/rtl-sdr/build $ cd ~  
pi@raspberrypi ~ $ sudo cp ./rtl-sdr/rtl-sdr.rules /etc/udev/rules.d/  
pi@raspberrypi ~ $ sudo reboot

After sending the above commands, you should be able to perform a simple test by typing in the following command:

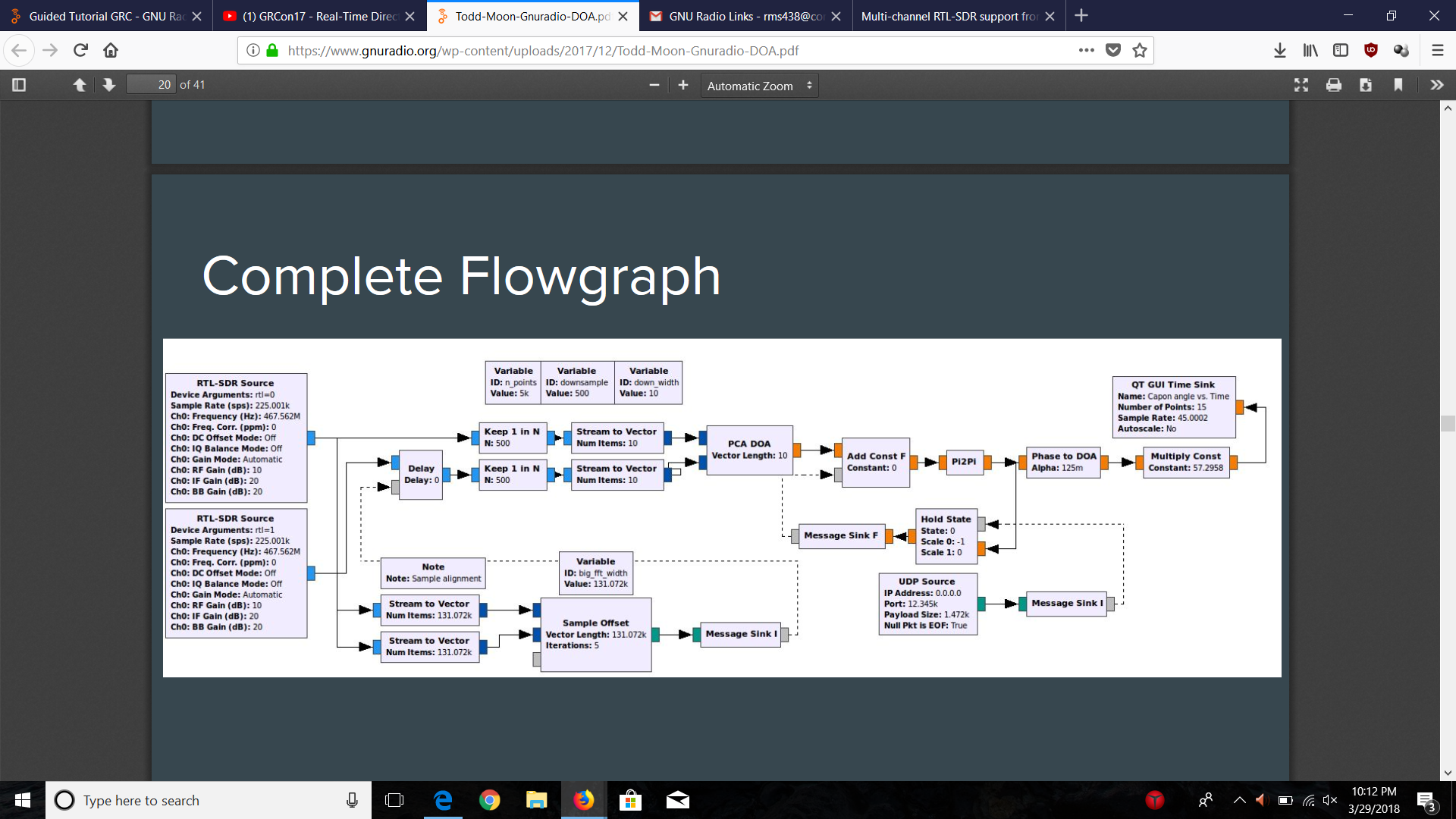
pi@raspberrypi ~ $ rtl\_test  
Found 1 device(s):  
 0: Generic, RTL2832U, SN: 77771111153705700

1. Install the bias\_tee software from <https://github.com/rtlsdrblog/rtl_biast> (this will be used to turn the bias\_tee on/off when performing RF Switching (not yet completed)) using the generic cmake instructions.
2. Install armadillo-code from https://github.com/conradsnicta/armadillo-code/ using the generic cmake instructions.
3. Install EttusResearch gr-doa custom blocks from https://github.com/EttusResearch/gr-doa using the generic cmake instructions. During the cmake ../ step, the terminal might show a list of missing dependencies that cmake was not able to compile. You will need to install these dependencies if they are missing (with the exception of DOxygen).
4. Use the relevant GRC files from <https://github.com/jakapoor/AMRUPT>. The master flowchart for obtaining rudimentary AoA measurements is in the GitHub called musicRTL.grc.

V. iv. GNU Radio Protocols

A protocol has been developed in [23] to synchronize multiple channels in a coherent radio receiver platform using SDRs. The flowchart below defines a sequence of DSP functions that need to be implemented to correct time and frequency offsets from RTL SDR measurements. This protocol can be adapted to a platform with four different SDR receivers, but has been truncated to 2 input SDR blocks for the means of simplicity. This protocol is shown in Figure 7.

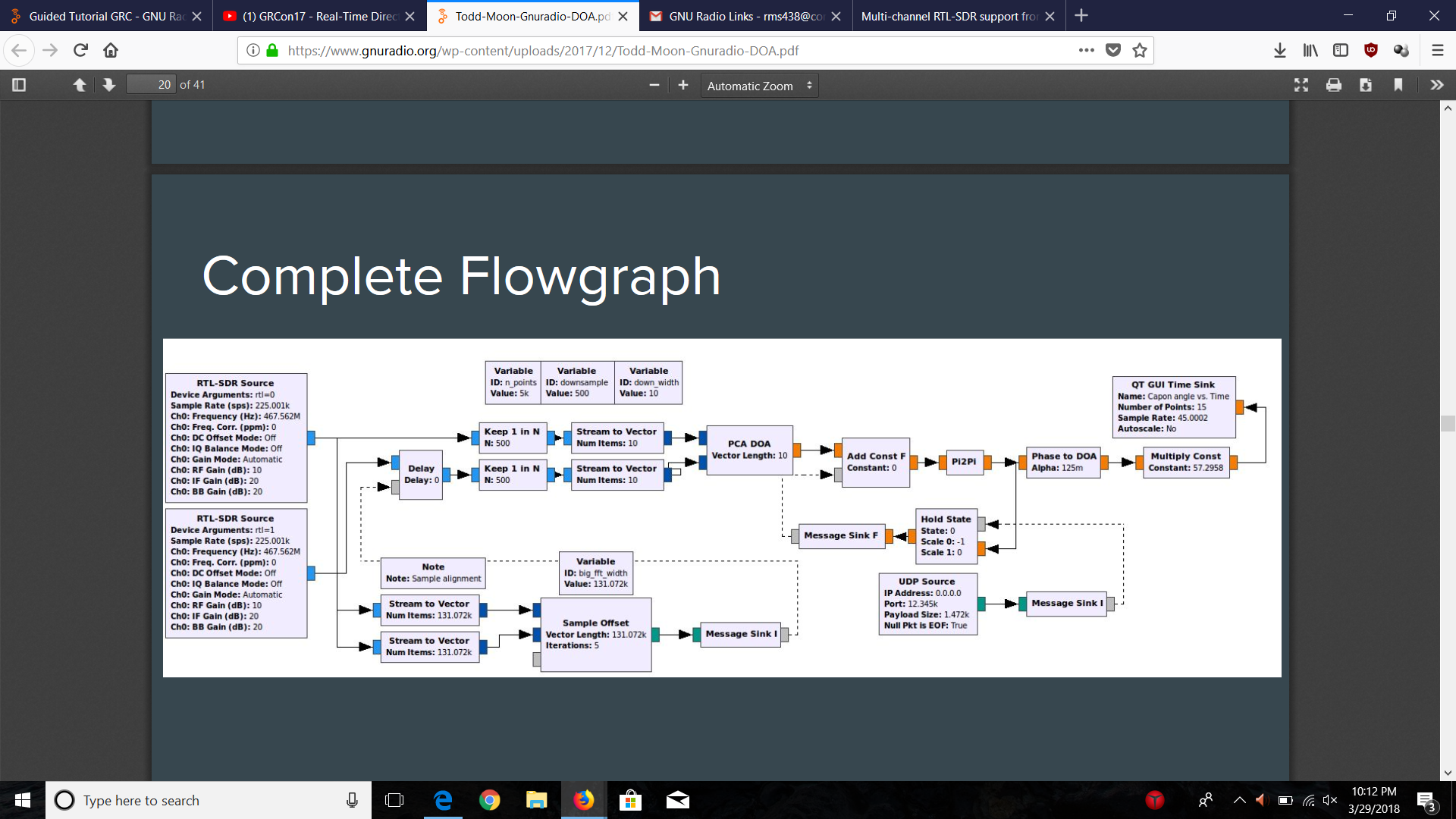
We have decided to opt with a GNU Radio flowchart design using custom blocks from Ettus Research (<https://github.com/EttusResearch/gr-doa>) instead because of this library’s more streamlined installation and explanatory documentation specific to the GNU Radio protocols available in this repository. This flowchart is shown in Figure 8.



1

2

3



1

2

3

Figure 7: GNU Radio Digital Signal Processing Flowchart - Sam Whiting, et. al. After synchronizing both receiver channels from the cross correlating samples in the Sample Offset block, the two input signals will be downsampled for more cost efficient AOA calculations. The subsampling of the synchronized signals did not reduce the integrity of the system described in [23]. The PCA DOA block computes the phase difference from the downsampled signals.

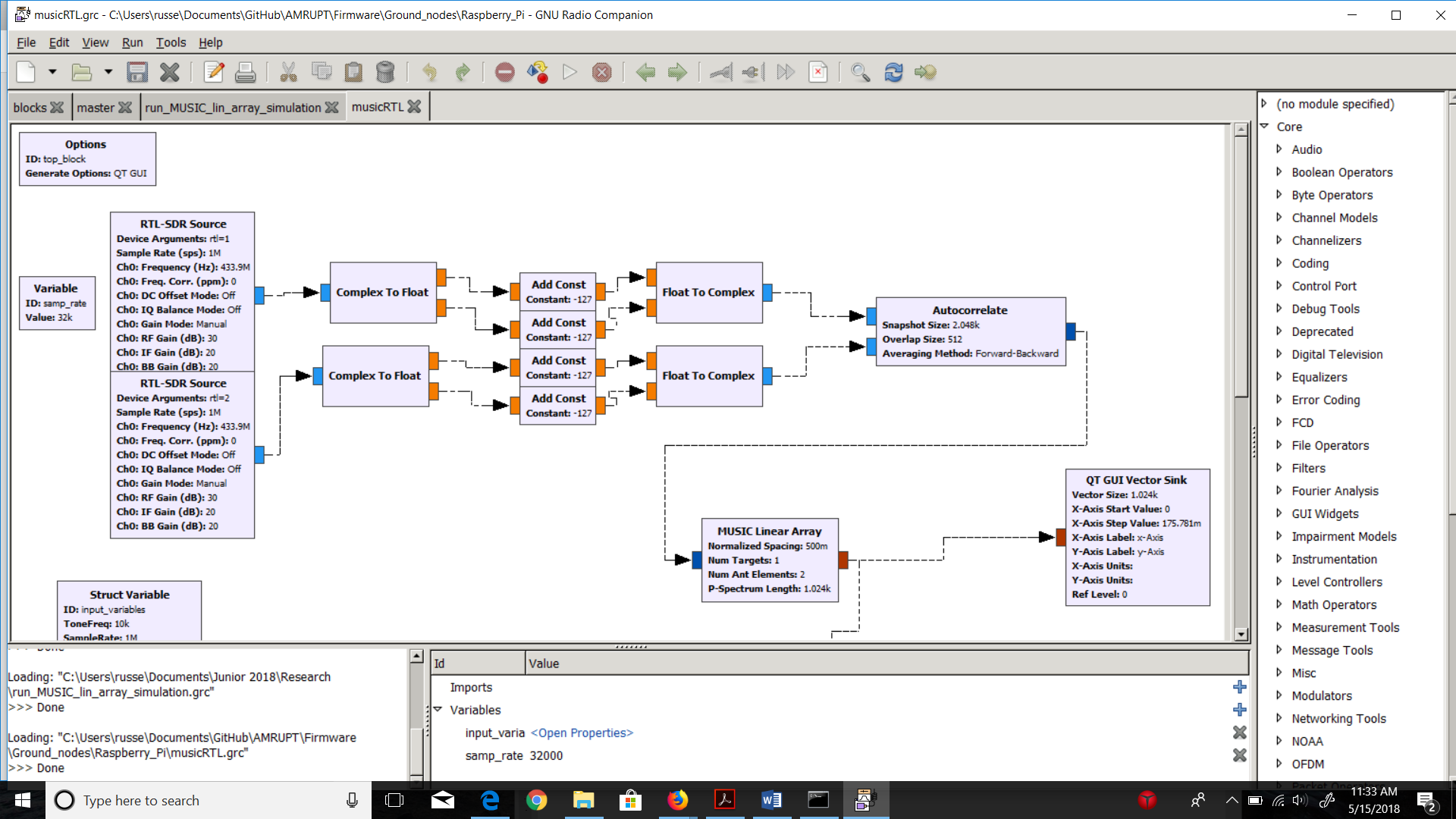
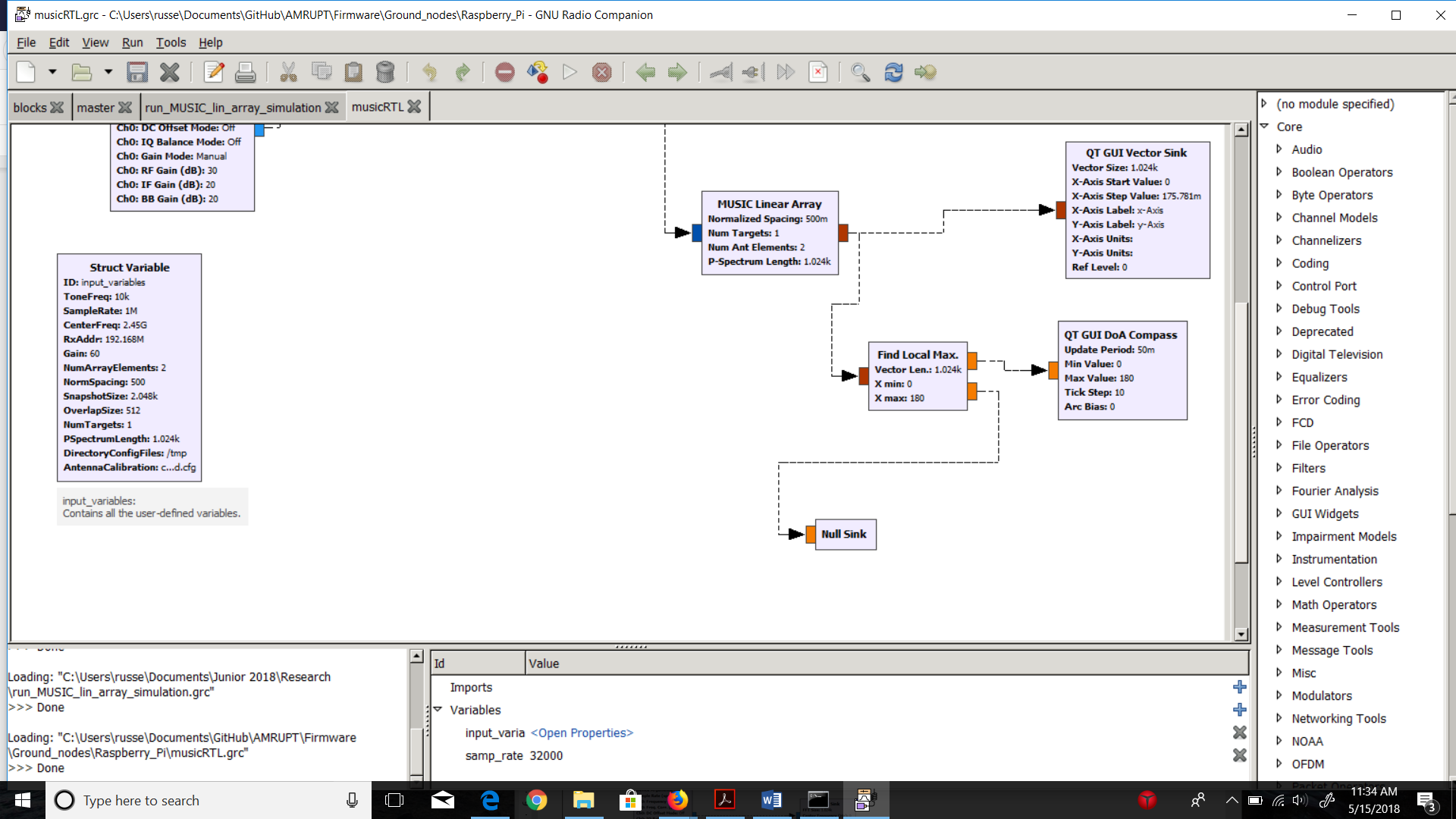


Figure 8. GNU Radio Digital Signal Processing Flowchart - Based on Ettus Research. Two RTL-SDR sources transmit complex I/Q values in real time to an Autocorrelation block. The autocorrelation block corrects offsets from two I/Q streams using Forward-Backward averaging.

The MUSIC direction finding algorithm is used in this approach. Although more computationally expensive, MUSIC can compute highly accurate AoA results when implemented correctly.

In order to accommodate the above protocol for two RTL SDR receivers instead of four, the autocorrelation overlap size has been adjusted to 1024k instead of 512k (2048k total vector size/2 rtl-sdr receivers = 1024k overlap size) and the MUSIC Linear Array’s Num\_Ant\_Elements variable to 2 instead of 4.

The current issue we are facing is that an angle of arrival appears for a very short time, and then disappears to 0/-1. We speculate that the source of this problem might be from RTL-SDR sources transmitting data in real-time to other blocks instead of file transfer from a remote receiver to computer via TCP/ZMQ (the original Ettus Research flowchart). The GUI compass updates around every 50 milliseconds, therefore a sampling offset could result in autocorrelation malfunctions after a first iteration of sampling collection. That is, the autocorrelation would receive samples numbered at perhaps [0 – 2000], and then [2000 – 4000] which would be inconsistent with the 2048k snapshot size of the autocorrelation block. A possible solution to this would be an integrated “head” block which take finite samples, iterated through an endless loop to allow for continuous angle of arrival collection.

VI. Conclusion

In order to accomplish the localization of small animals, we plan to develop a cost effective and automated system to track animal movements within the range of five meters while taking into account expected causes of error. Our proposed system consists of a receiver architecture that is built specifically for phase interferometry direction finding to facilitate accurate measurements from radio tags on tracked individuals. In order to accomplish this, a low weight radio tag is being developed to transmit signals to radio base stations. These tags will transmit sub 1-GHz UHF frequencies.

Appendix A. Works Cited

[1] Krüger, S W. “An Inexpensive Hyperbolic Positioning System for Tracking Wildlife Using off-the-Shelf Hardware.” May 2017.

[2] He, Chong, et al. “Direction Finding by Time-Modulated Array With Harmonic Characteristic Analysis.” IEEE Antennas and Wireless Propagation Letters, vol. 14, 2015.

[3] Řeřucha, Š., Bartonička, T., Jedlička, P., Čížek, M., Hlouša, O., Lučan, R. and Horáček, I. (2015). The BAARA (Biological Automated Radiotracking) System: A New Approach in Ecological Field Studies. PLOS ONE, 10(2), p.e0116785.

[4] D. Guerin, S. Jackson, and J. Kelly, “Passive Direction Finding: A Phase Interferometry Direction Finding System for an Airborne Platform,” Oct. 10, 2012. https://web.wpi.edu/Pubs/E-project/Available/E-project-101012-211424/unrestricted/DirectionFi ndingPaper.pdf.

[5] D. Zhang, J. Ma, Q. Chen and L. M. Ni, "An RF-Based System for Tracking Transceiver-Free Objects," Fifth Annual IEEE International Conference on Pervasive Computing and Communications (PerCom'07), White Plains, NY, 2007, pp. 135-144.

[6] Y. Ma, X. Hui, and E. Kan, “3D Real-time Indoor Localization via Broadband Nonlinear Backscatter in Passive Devices with Centimeter Precision,” Oct. 3, 2016. https://dl.acm.org/citation.cfm?id=2973754.

[7] He, Chong, et al. “Direction Finding by Time-Modulated Array With Harmonic Characteristic Analysis.” IEEE Antennas and Wireless Propagation Letters, vol. 14, 2015.

[8] Bartolucci, Marco, et al. “Synchronisation of Low-Cost Open Source SDRs for Navigation Applications.” 2016 8th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC), 2016.

[9] Whiting, Sam, et al. “Time and Frequency Corrections in a Distributed Network Using GNURadio.” 2017.

[10]“Tejeez/rtl\_coherent.” GitHub, 6 July 2016, github.com/tejeez/rtl\_coherent

[11]Whiting, Sam, et al. “Time and Frequency Corrections in a Distributed Network Using GNURadio.” 2017