AMRUPT PROJECT PROPOSAL:

Russell Silva - rms438
Mei Yang - mny8
Peidong Qi -pq32
Justin Cray - jgc232
Electrical and Computer Engineering, Cornell University
2/21/2018

Submitted to—
Dr. Julian Kapoor, Prof. Joe Skovira
Behavior/Electrical and Computer Engineering, Cornell University

This page is intentionally left blank

Contents

I. Executive Summary

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 basestations. These tags will transmit sub 1-GHz UHF frequencies.

RF signals phase information is calculated on the CC1310. This requires multiple receive antennas connected to an RF switch which is attached to the CC1310 which communicates the I & Q values of the received to a Raspberry Pi for angle of arrival calculation. A three antenna system is used for angle of arrival. An angle of arrival will be computed between Antenna 1 and Antenna 2 that is ambiguous between Quadrants 1&2 and Quadrants 3&4. An angle of arrival will be computed between Antenna 1 and Antenna 3 that is ambiguous between Quadrants 2&3 and Quadrants 1&4. The angle of arrival will be determined to be in the quadrant that contains both AoAs.

II. Statement of Problem

II. i. The Localization of Small Animals in the Field of Ecology

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 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.

II ii. Literature Review

Radio frequency direction finding systems have been implemented from amplitude-based (Watson-Watt), Phase Interferometry, or Time Difference of Arrival approaches.

Guerin, Jackson and Kelly [1] developed a phase interferometry direction finding system to determine the Angle of Arrival (AoA) of a received signal. Many testing protocols were used to optimize the system before hardware construction 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.

Ma, Hui and Kan [2] proposes a 3D indoor passive tag localization method with an accuracy of a few centimeters in a multi-frequency identification system. The paper leveages nonlinear backscatter which exploits nonlinear elements in passive devices to generate second or higher-order harmonics for an uplink response. 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 (Figure 1). In order to combat multipath interference, a phase error threshold is used within a HMFCW ranging algorithm to find an optimal frequency combination that generates an undistorted line of sight path.

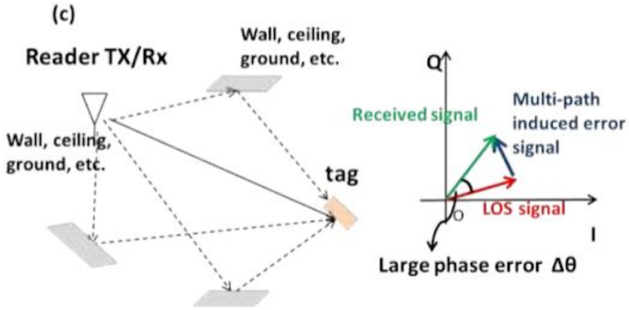

(c)
Reader TX/Rx
Wall, ceiling, ground, etc.
Wall, ceiling, ground, etc.
tag
Q
Received signal
Multi-path induced error signal
LOS signal
o
I
Large phase error Δθ

Figure 1: Dense indoor multi-path induced phase error [2].

Řeřucha et al. [12] developed an architecture to use location tracking for ecological field studies. This architecture presents a network of automated trackers and software to facilitate tracker operation.

Toledo et al. [13] used Time of Arrival in there location tracking system. While effective, Toledo et al. has an accuracy of 5-15 m which we hope to exceed in our project.

Porat and Friedlander [14] proposes a maximum likelihood based signal reconstruction that could be useful for our signal reconstruction.

Smith, Balakrishnan et al. [15] developed a chip which uses a combination of RF and ultrasound technologies to provide location information to attached host devices. When this pulse arrives, the listener obtains a distance estimate for the corresponding beacon by taking advantage of the difference in propagation speeds between RF (speed of light) and ultrasound (speed of sound).

In Zhang et al. [16], an algorithm is used to directly estimate a human’s position from RSS (received signal strength) measurements. When a link measures RSS variation above a threshold, it is assumed that the person is located within a rectangle centered at the midpoint of the line between the transmitter and receiver. A “best cover algorithm” then estimates the person’s position, which is inputted into a tracking filter. This method again does not generate data within the accuracy we require.

III. Design Objectives

In order to accomplish the goals listed in the problem statement, 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) System is cost-efficient (almost all components are commercially off-the-shelf)
(5) Forward compatibility: Must be compatible with and adaptable to a multi-frequency-phase-integer-disambiguation approach for future versions.

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 to node communication protocol will be governed by how accurate our receivers are when taking angle of arrival measurements. If angle of arrivals from a couple of basestations 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. 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 basestation 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 that trilaterates positions of mobile nodes if a 5-meter accuracy level has not been achieved by previous efforts (Phase 3).

The diagram below is used to better understand error minimization with relation to AOA calculations.

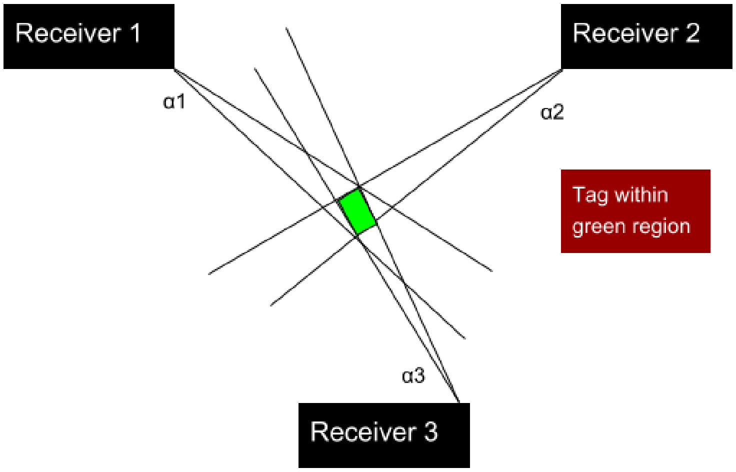


Figure 2: Error in triangulated area

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 1) 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.

Last, but not least we have devised a system that is composed of cost-effective, off-the-shelf components. This is done to make this setup more reproducible in future works and more accessible to ecological hobbyists/researchers.

IV. Technical Approach

The entirety of the proposed direction finding system consists of radio transmitters and receivers. This section will focus primarily on receiver design as the lightweight radio tags are being developed by another party. In order to achieve our design objectives, the receiver architecture will require the most development.

IV. i. Receiver Architecture

We first propose a receiver architecture that consists of an embedded device to simplify wireless communication and improve the cost effectiveness of this project.

The ideal embedded device would include the following:

1. A sub 1-GHz device for VHF or UHF frequencies transmitted from radio tags. We choose a lower frequency band (relative to most RF applications) to mitigate multipath interference and better determine the phase difference of signals. Previous systems have used ~150 MHz as the operating frequency of transmitters because of the impact of large trees on multipath interference.
2. A very high sample frequency during the analog to digital conversion of RF signals. This is essential for mitigating adverse effect from noise when determining accurate phase differences from radio waves moving at the speed of light. However, a sampling frequency above twice the radio frequency (constant in a non-frequency modulated signals) is not needed. Sampling rates are further discussed in section IV. vii. “RF Wave Reconstruction and Matlab Simulation”.
3. Ample UART/I2C/SPI/GPIO connections for data logging and transfer
4. Contains every component necessary for receiving an RF signal from an external antenna - ADC, local oscillator, etc.
5. Extremely high RF sensitivity and blocking performance
6. Programmable and highly used by the public - helpful for finding more tutorials and readily available information on the device
7. Low power and low cost

From this list of specifications, the CC1310 was chosen. The CC1310’s specifications are collated within the datasheet: http://www.ti.com/lit/ds/symlink/cc1310.pdf

- Suitable sampling rate and clock frequency: 200KHz ADC sample rate and runs on a 24MHz clock.
- 30 GPIO Pins with I2C and UART capability
- 128 Kb flash and 20 Kb ram
- The CC1310 transceiver chip is the smallest available chip in the market (critical for low weight tags on small animals)

With the CC1310 in place, we refer to a simplified representation of our receiver system:

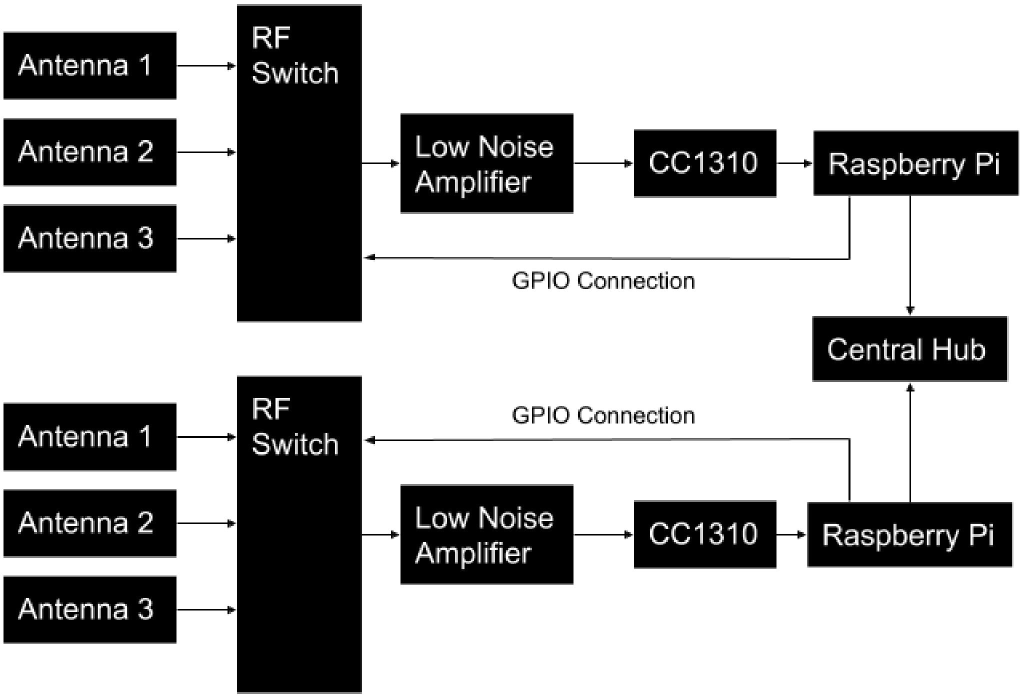


Figure 3: Receiver Architecture

We choose to use three antennas to delimit phase ambiguity to expand the phase interferometry system's capabilities from determining AOAs from $-\pi < \theta < \pi$ to $-\pi n < \theta < \pi n$ where $n > 0$. The RF Switch was placed very early in the design to combine multiple receiving channels into one streamlined input. With this system, only offsets from the antenna geometry or the wire lengths/connectors from each antenna to the switch could contribute to phase difference errors between each antenna. Solutions for this type of error is discussed in section IV. vi. "Phase Disambiguation and Angle of Arrival Calculation". A low noise amplifier is proposed to increase the power of the incoming signal, even though we may find it unnecessary after preliminary testing. Low noise amplification takes place after the RF Switching stage to decrease the length variability of different antenna channels. The CC1310's primary responsibility will consist of the analog to digital conversion of the signal, and an extraction of in-phase and quadrature values which will be sent to the Raspberry Pi over a UART connection. The RF signal from each antenna will be reconstructed using embedded programming on the Raspberry Pi. From this reconstruction, the phase differences for each antenna will be calculated along with phase disambiguation (from comparing the phase difference from antennas 1 and 2 to the phase difference of antennas 2 and 3). Subsequently at this stage, the angle of arrival will be calculated, timestamped, and sent wirelessly (over a unique frequency to avoid interference with the whole system) to a central hub which will triangulate the source of the signal.

IV. ii. Antennas

We seek an antenna that is in our target frequency range (the UHF band) and also has an SMA connection for ease of interfacing with other components in our system. To accomplish this, an antenna such as the ANT700 is compact high gain antenna and operates in the expected frequency range.

IV. iii. RF Switch

An RF switch board will be created for this project. Leveraging a high isolation and low insertion loss RF switch such as the ADG918 we can switch between two antennas mounted on a PCB. The isolation and insertion loss parameters are dependent on the Raspberry Pi and other external hardware. To derive these numbers, mock RF signals will be inserted through the system to measure the system's loss. The ADG918 will operate in the UHF range making it a practical candidate for the project. The ADG918 will be controlled by the Raspberry Pi. By sending GPIO signals the Pi can toggle the ADG918 between two different antenna values. This setup would require the use of multiple GPIO ports and multiple chained switches to accomplish our goal of having 3 antennas. More research will be done on finding a three to one RF switch to prevent chaining if possible. Another factor in this design is the SMA connection from the antenna to the PCB. Many mounts and cables are available for these connections allowing us to connect the antenna to the RF switch to the CC1310 by using commercially available adapters.

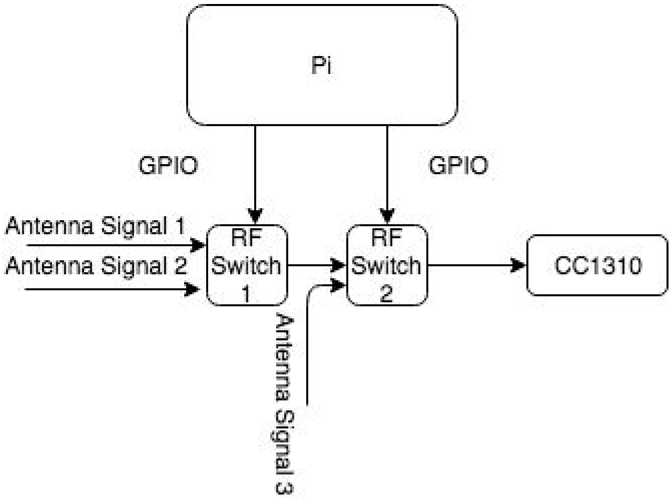


Figure 4: RF Switch Block Diagram

IV. iv. CC1310 I/Q Extraction

Since an RF wave can be essentially modeled as a sinusoidal function, the instantaneous phase, or the wave's offset from its from its origin (universally recognized as f(x) = 0, f'(x) > 0) can be more effectively determined by using the signal's real (I: In-phase) and imaginary (Q: Quadrature) components (Guerin, Jackson, Kelly 2012).

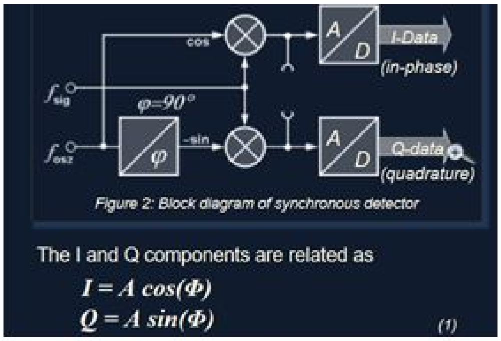
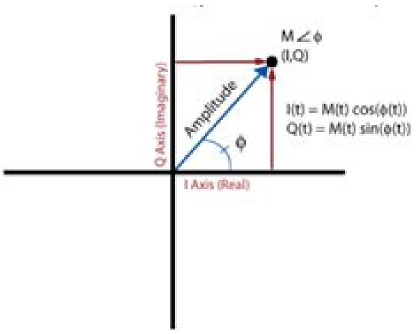


Figure 2: Block diagram of synchronous detector

The I and Q components are related as

$$I = A\cos(\Phi)$$
$$Q = A\sin(\Phi) \quad (1)$$



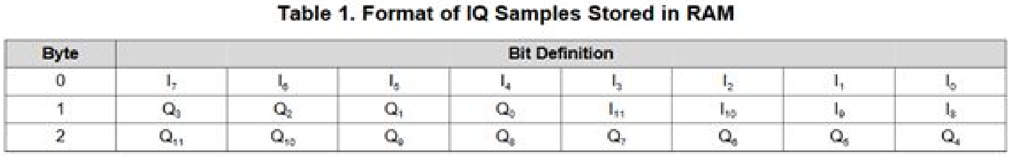


Table 1. Format of IQ Samples Stored in RAM

Byte	Bit Definition							
0	I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0
1	Q_3	Q_2	Q_1	Q_0	I_{11}	I_{10}	I_9	I_8
2	Q_{11}	Q_{10}	Q_9	Q_8	Q_7	Q_6	Q_5	Q_4

Figure 5: A radio frequency signal can be decomposed into a real and imaginary value top-right. The in phase and quadrature information can be obtained from the top-left setup through an I/Q Demodulator (The CC1310 employs a similar I/Q demodulation within its circuitry). I/Q Samples are stored in RAM using 3 byte data packages (bottom). Top-right: (National Instruments, 2016) [19], top-left: (Wolff, Date N/A) [18], bottom: (TI, 2017) [7].

If the I/Q data of an RF signal is received from two antennas in real time, the phase difference between these two signals can be determined by the following calculations:

Phase angle between Signal 1 (ph2) =atan(Q/I)

Phase angle between Signal 2 (ph1) =atan(Q/I)

Phase difference = ph1-ph2

An I/Q document detailing the steps to accessing the I/Q data collected by the CC1310 can be found in [7].

IV. v. CC1310 to Raspberry Pi UART Connection and Datalogging

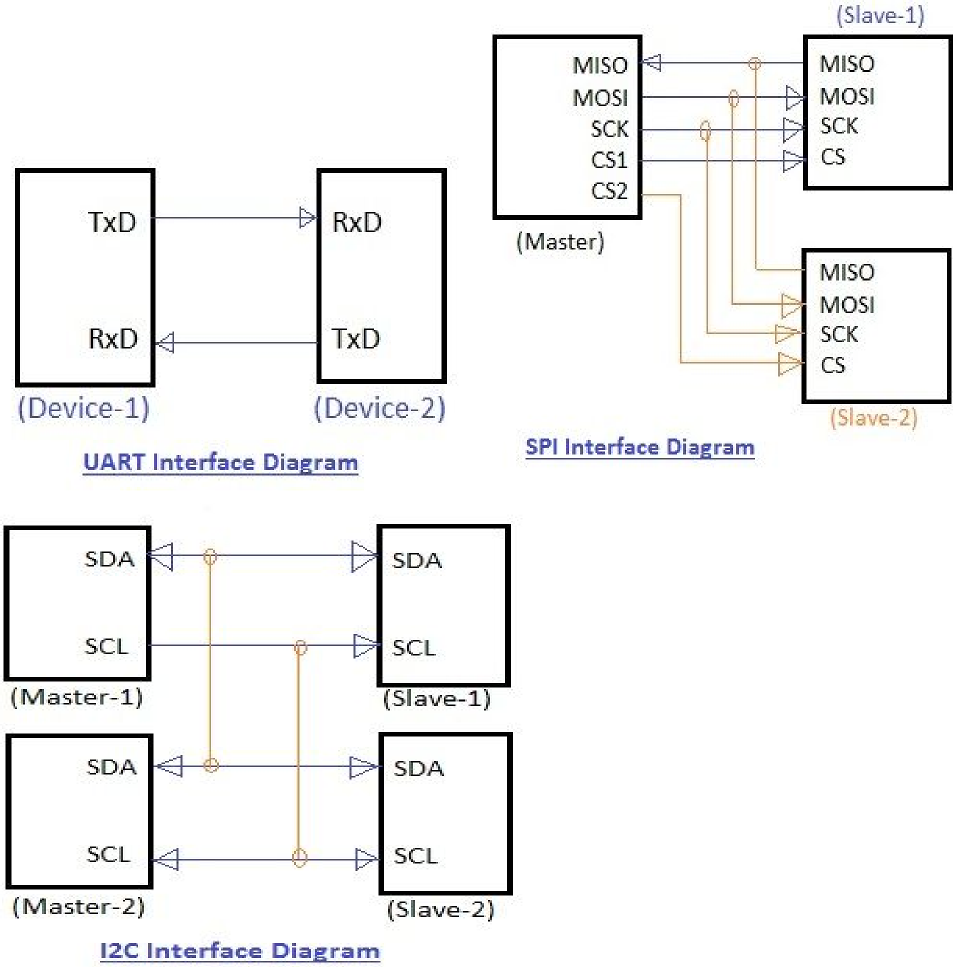


Figure 6: UART, SPI and I2C connection layout (rfwireless-world) [11]

The data transfer from the CC1310 to the Raspberry Pi can be done through different ways. There are three major ways: Uart, SPI and I2C. These connections are shown in Figure 5: SPI has the fastest data transfer rate and Uart is the easiest to set up. The priority of our technical approach is to determine the required sampling rate. So our first task is to usea Matlab simulation system to determine the sampling rate. We will modify our plan according to the maximum data transfer rate. For Phase 1 of the project, we will use Uart to communicate. For Uart, we need to use three ports on both the Raspberry-pi and the CC1310. The Raspberry-pi has those ports already setup and the CC1310 has RF pins which support data transfer. We need to first connect both GND together to ground, and then connect Tx to Rx and Rx to Tx. The Tx stands for transmit, the Rx stands for read. For communication, the Tx to Rx connection will allow data transfer from one device to another. Our overall goal is to transfer data from the CC1310 to the Raspberry-Pi.

Once we get the data from the CC1310, we will run a program on the Raspberry-pi to log the data. The program will build a file for the data we have received. Then the program is creates a tag label and a time label for the file. The tag label is going to indicate which tag will send this signal. There will be a real time clock onboard to monitor the time and help to create the time label. It will be a convenience for the user and system to track the tags. Once a file is required, the Raspberry-pi will use the file for the direction finding algorithm.

IV. vi. Phase Disambiguation and Angle of Arrival Calculation

The phase difference between two antennas models the extra distance that a radio wave has to travel to reach one antenna over the other. The geometry that demonstrates this phenomenon is displayed in Figure 5, where an angle of arrival is dependent on a signal's wavelength, the separation between two antennas, and the phase difference. Using these fundamental notions of phase interferometry, the angle of arrival will be calculated at this stage using the phase differences obtained from the CC1310.

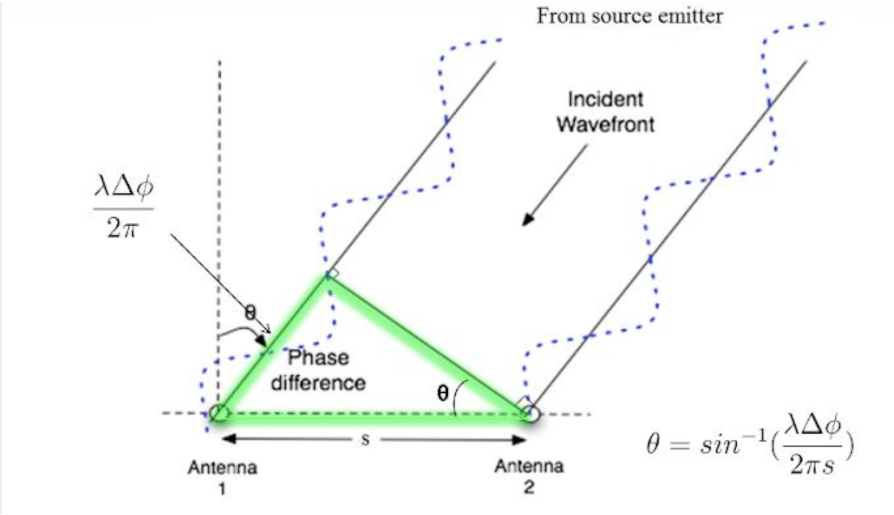

From source emitter
Incident Wavefront
$\frac{\lambda\Delta\phi}{2\pi}$
θ
Phase difference
θ
s
Antenna 1
Antenna 2
$\theta = sin^{-1}(\frac{\lambda\Delta\phi}{2\pi s})$

Figure 7: Diagram showing how the angle of arrival equation is formed using the phase difference of a signal received at two antennas, the distance between the antennas, and the wavelength of the signal. (Guerin, Jackson, Kelly 2012)

There are two types of phase disambiguation that will be covered in this section. One type of phase disambiguation results from mobile nodes that are flipped above and below the receiver's 0 degree horizontal axis reference. This will be defined as quadrant ambiguity. The second type of phase disambiguation results from antennas separated greater than $\lambda/2$ in which a phase difference is outside the $-\pi < \theta < \pi$ range. This will be defined as distance ambiguity. Distance ambiguity is not to be confused with the phase integer ambiguation that occurs when calculating differential distances from mobile to ground node as discussed in (Ma, Hui, Kan 2016).

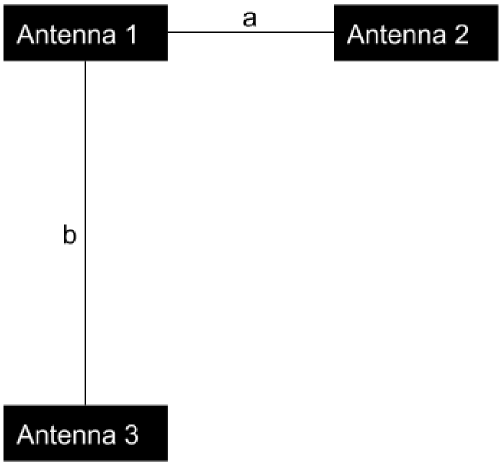


Figure 8: Antenna Setup with distance a between Antenna 1 and Antenna 2, distance b between Antenna 1 and Antenna 3.

We propose to solve quadrant ambiguity by adding a third antenna that is placed at a right angle from the line formed by antenna 1 and antenna 2 (Figure 6). An angle of arrival will be computed between Antenna 1 and Antenna 2 that is ambiguous between Quadrants 1&2 and Quadrants 3&4. An angle of arrival will be computed between Antenna 1 and Antenna 3 that is ambiguous between Quadrants 2&3 and Quadrants 1&4. The angle of arrival will be determined to be in the quadrant that contains both AoAs.

Since we are using sub-1GHz frequencies, placing the antennas at distances less than $\lambda/2$ is a feasible solution. For example, a 300 MHz signal will have a wavelength of 1 meter. So placing antennas around $0.4\text{m} < d < 0.5\text{m}$ from each other would not have a detrimental effect on signal gain, especially if PCB antennas are used. Frequency hopping at frequencies lower than 300 MHz will also be supported. This is important for forward compatibility with the multi frequency phase integer disambiguation approach used in (Ma, Hui, Kan 2016). .

If the above solution is found to be infeasible, and one of the antennas is placed greater than $\lambda/2$, distance ambiguity can be solved by making distance b > distance a (Figure 6) to a suitable offset determined by the signal wavelength. Since the CC1310 uses higher frequencies (315 - 930 mHz), a broadband frequency hopping system that includes all of these frequencies would have to support wavelengths of 0.32 meters ($c = \lambda f$). Therefore, our system would have to place antennas lower than 0.16 meters from each other, and near field radiation patterns may cause additional interference at these distances, especially if an equidistant beacon is used (discussed later in the section).

The following approach is proposed to solve distance ambiguity:

The number of phase ambiguities n is dependent on the following formula where s is the distance between two antennas and lambda is the wavelength of the signal. Theta resembles the maximum angle arrival of the system which in our case is pi.

$$n = \frac{s}{\lambda} sin\theta_{max},$$

(Guerin, Jackson, Kelly 2012)

We can formulate this equation for both pairs of antennas by replacing n by the normalizing phase difference from 0 to 1 (division by 2 π) added to the number of possible phase difference in each antenna pair.

$$\Delta\phi'_{12} + i_{12} = \frac{s}{\lambda} sin\theta_{AoA}$$
$$\Delta\phi'_{13} + i_{13} = \frac{s}{\lambda} sin\theta_{AoA}.$$

(Guerin, Jackson, Kelly 2012)

Finally we combine both equations to form a set of lines which correlates a disambiguated phase difference for each pair of antennas to a maximum likelihood by determining the number of full phases for a phase offset.

$$\Delta\phi'_{13} = \frac{s_{13}}{s_{12}} \Delta\phi'_{12} + \frac{s_{13}}{s_{12}} i_{12} - i_{13}.$$

(Guerin, Jackson, Kelly 2012)

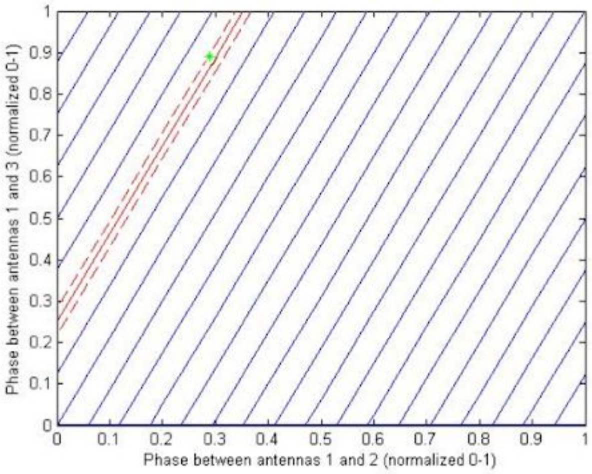


Figure 9: Phase Disambiguation Chart (Guerin, Jackson, Kelly 2012)

We do not believe that the 90 degree offset from quadrant disambiguation will affect the geometry of this solution.

After the phase disambiguation process, error from antenna wire lengths mentioned in section IV. i. will be corrected by connecting an additional GPIO pin on the Raspberry Pi 3 to an overpowered radio transmitter equidistant to the three known locations of each antenna. From this setup, each reconstructed wave should have no phase offset from each other. If there is a phase offset at antenna 2 or antenna 3 in relation to antenna 1, that amount will be subtracted from phases collected at antenna 2 or antenna 3 respectively for future AOA calculations.

A beacon placed this close to all three antennas will be in the near-field of those antennas and could potentially cause adverse interference effects. This can be fixed by putting the transmitter at a much higher vertical height relative to the horizontal distances between the beacon and each antenna. A lightweight mobile node can be used to act as an equidistant beacon, so that our receiver system is not difficult to setup or transport.

Instead of an equidistant beacon, a linear frequency modulated signal can be integrated into our receiver architecture in order to synchronize channel delays through equalization. Such a signal would have to be connected to the exact beginning of each antenna wire connection, and the leads of the LFM wires would have to be exactly the same distance from each other. Even though a closed system approach would have benefits over the wireless beacon strategy in terms of signal transmission control, it is more useful in systems that have seperate channels each containing a separate demodulator/ADC, so that delays from these components could be corrected. It would be difficult to integrate such a system to correct offsets singularly generated by varying antenna wire lengths.

IV. vii. RF Wave Reconstruction and Matlab Simulation

Matlab will be used as a tool for this project to reconstruct RF waveforms and simulate the proposed system by producing an estimate for the Angle of Arrival (AOA). Matlab software can be installed and utilized on the Raspberry Pi 3. A Matlab simulation from multiple signal generators (acting as separate antennas) will be used to verify the feasibility and correct functionality of the proposed solution. The proposed reconstruction is shown in Figure 8, in which the Raspberry Pi 3 reconstructs a sinusoid from one antenna at a time, while using the predicted continuation of signals at other antennas for phase difference comparisons.

In addition, Matlab simulations will follow the modular design principle that helps us build and organize the project as a whole. For RF waveform reconstruction, a signal generator will be created to simulate the RF waveforms at a certain sampling rate. We will consult with Dr. Kan to determine a high enough sampling rate such that the reconstructed RF waveforms are close to continuous as possible and are of the form $A\sin(\omega t + \varphi)$. The phase differences for each of the reconstructed waveform will be calculated along with phase disambiguation. The signals will then be used for future AOA calculation. We will consult with Dr. Kan and hopefully, can adapt some of his existing simulation models for direction finding verification.

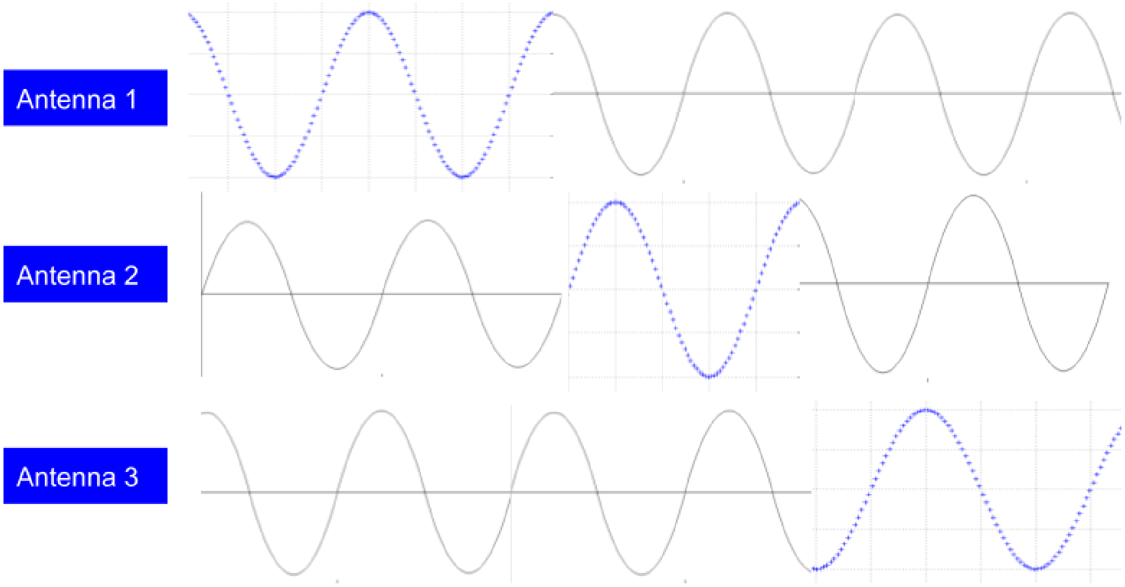


Figure 10: Blue regions represent areas that are being currently reconstructed by I/Q samples. Grey regions represent areas that continue the reconstructed signal and can be used for comparing phases to other antennas, but are not composed of collected data.

For ideal sinusoidal signals, it is important to note that the CC1310 does not have to sample antenna outputs at twice the frequency of the transmitted signals if the frequencies of the signals are constant.

Assuming our sampling rate is constant, sampling a sinusoidal wave lower than the Nyquist frequency will produce an equivalent reconstructed signal. This reconstructed signal will have the same wave pattern and structure of the original signal. The only difference is that this reconstructed wave will be at a lower frequency. The above holds true for infinitely slow sampling rates (stationary sources) .

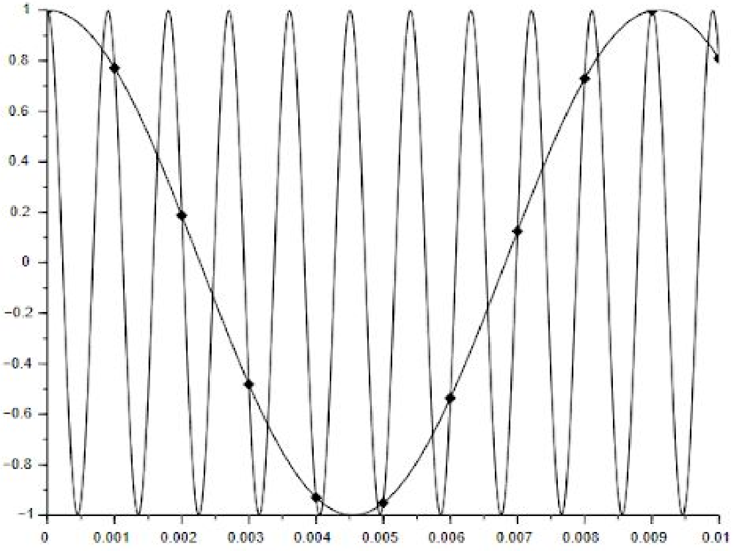


Figure 11: Sampling a sine wave lower than the Nyquist Frequency (Wescott, 2016)

If we take the phase difference between two sinusoidal waves that have been reconstructed from a lower sampling frequency than the Nyquist at any given time, their phase differences at that given time will be the same as if we were using the original sinusoidal function.

For this approach, we cannot use frequency modulated signals on radio tags. We will have to make sure that the transmitted signals represent sinusoidal waves of constant frequency in order for undersampling to work. This will not impede on a multi-frequency-phase-integer-disambiguation system as sequential frequency hopping is scheduled by the receiver (Ma, Hui, Kan 2016). For our active tag system, the receiver system will have to send a frequency hopping scheduling to the mobile nodes before a transmission period, so that it will know when frequency hops will occur (to prevent irregular measurements).

IV. viii. CC1310 Crystal Oscillator

The CC1310 crystal oscillator supports low-power consumption mode to reduce power consumption and is therefore chosen for the time synchronization of CC1310 in this project. The CC1310 has 24-MHz and 32.76-kHz crystal oscillators that are integrated on it. The 24-MHz oscillator generates the reference clock and the 32.76-kHz oscillator is used in the low-power standby mode. The datasheet discusses in great detail on the set of crystal parameter values for working with the CC13XX family. For example, the drive level, DL, specifies the maximum power level for the crystal to have proper long-time operation. In order to configure CC1310 for different crystals, the datasheet has provided sample code to set up the various parameters needed for the crystals working correctly. The datasheet provides detailed sample code to set up the crystal.

IV. ix. Raspberry Pi 3 Network with Central Hub

This section covers work that will likely be beyond the scope of this semester's work, but includes a very rough plan on how the positions of radio tags will be triangulated.

We will connect the Raspberry Pi 3 basestations wirelessly at a frequency outside the bandwidth of the rest of our system to prevent radio frequency interference. The Raspberry Pi 3 will send timestamped and identified angle of arrival measurements to a central hub/laptop. The accuracy requirements and wireless transmission protocol of the timestamping will be determined later on later on in the semester. The central hub will then triangulate the position of a radio tag (based on the AOA identifications) in two dimensional space.

Triangulation Algorithm Pseudocode Based on Three Receivers (Python):

```
class Coordinate:
     def _init_(self, x = 0.0, y = 0.0):
          self.x = x
          self.y = y

     #Finds intersection of two lines (y = mx + b)
     def intersect (m1, b1, m2, b2):
          intersection = Coordinate()
          xval = float(b2 - b1) / float(m1 - m2)
          yval = xval * m1 + b1
          intersection.x = xval
          intersection.y = yval
          return intersection

     #Finds the centroid of the triangulated area
     def centroid(vertex1, vertex2, vertex3):
          center = Coordinate()
          center.x = (vertex1.x + vertex2.x + vertex3.x) / 3.0
          center.y = (vertex1.y + vertex2.y + vertex3.y) / 3.0
          return center

     #Parameters are decimal angle of arrivals calculated at
     #each basestation and coordinate positions of each
     #basestation.
     def triangulated area (angle_of_arrival1,
     angle_of_arrival2, angle_of_arrival3, receiver1, receiver
     2, receiver3):

           #Converts angle degrees to slope
          slope1 = math.tan(angle_of_arrival1)
          slope2 = math.tan(angle_of_arrival2)
          slope3 = math.tan(angle_of_arrival3)

          b1 = receiver1.y - receiver1.x * slope1 #y = mx + b
```

```
b2 = receiver2.y - receiver2.x * slope2
b3 = receiver3.y - receiver3.x * slope3

intersection1 = intersect(slope1, b1, slope2, b2)
intersection2 = intersect(slope2, b2, slope3, b3)
intersection3 = intersect(slope1, b1, slope3, b3)

position = centroid(intersection1, intersection2,
intersection3)
return position
```

IV. x. Separate Demodulator/ADC (Plan B Solution)

Unfortunately, we have hit many roadblocks with the CC1310's I/Q extraction in the past. A report has been used as our most significant reference for debugging ([7]: I/Q Extraction Document Link). We found this source of information to be incomplete with the following statement, "The processing of the I/Q samples should be done outside the callback. It is not the scope of this application report to show how this can be done." The device's manufacturer, Texas Instruments, has been unresponsive to our questions regarding this issue.

If the problem persists we propose using a I/Q demodulator (with IF mixing) and an ADC that will replace the CC1310 in our block diagram.

Before moving forward with part descriptions, we define the demodulation/intermediate frequency mixing stage as recovering the information and wave structure of a received RF signal by the means of an electronic circuit. We will focus on phase demodulation or I/Q demodulation in order to interpret an incoming radio wave in terms of its in-phase and quadrature sinusoidal components. This stage will also require the use of intermediate frequency mixing in order to reconstruct the original signal with a cost effective and kHz range ADC. The analog to digital conversion stage will quantify the analog in-phase and quadrature signals for data processing on the Raspberry Pi 3.

We have looked into the following I/Q demodulator for our system:

EV9700 Evaluation Kit:

http://www.cmlmicro.com/products/EV9700_Evaluation_Kit/

CMX970:

http://www.cmlmicro.com/products/CMX970_IF_RF_Quadrature_Demodulator/

The CMX970 is an ideal demodulator for its low power consumption and its integrated intermediate frequency mixing. Its in-phase and quadrature analog output signals are displayed in the component's datasheet (Figure 10).

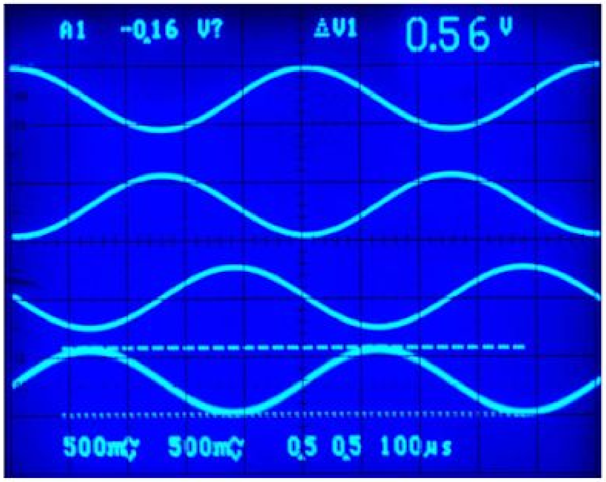


Figure 12: The output signals of the CMX970. They are from bottom to top: RXQ-, RXQ+, RXI-, RXI+.

Since, we are still using our RF switch and reconstruction method, local oscillator synchronization will not be necessary. However, if we decide to change to a multiple channel architecture, each CMX970 has an input to an external local oscillator (each output can be connected to the same local oscillator). We are currently looking in the DS1077, a programmable local oscillator that is cost-efficient and flexible to a wide range of output frequencies. There is also a lot of documentation and is widely used by hobbyists.

https://www.mouser.com/Passive-Components/Frequency-Control-Timing-Devices/Oscillators/Programmable-Oscillators/_/N-7jdx1/

It is out of stock at the moment on most electronic component databases. It is available at a very low price on Ebay: https://www.ebay.com/i/192140514584?chn=ps&fl=a

Since the Raspberry Pi 3 does not have a built in analog-to-digital converter, we propose to use an external multichannel ADC with a normal sampling rate such as the MCP3008 which has a maximum sampling rate of 200 ksps, more than enough for sampling sinusoidal functions in the KHz range (the frequency of the demodulated signal is much lower than the signal received at the antenna stage). This sampling rate will be sufficient for the sampling of 2 parallel data streams (1 in-phase, 1 quadrature). We will not need to sample all 6 data streams from each antenna at once because the Raspberry Pi controls which antenna it is listening to (RF switch control), and therefore knows which antenna the in-phase and quadrature data streams are coming from at a given time. From the

ADC, the digitally sampled in phase and quadrature signals will be connected to two different GPIO pins on the Raspberry Pi. A tutorial on this can be found in Appendix A.

We will use SPI to communicate from the external ADC to the Raspberry pi. The ADC's pinout diagram is shown below:

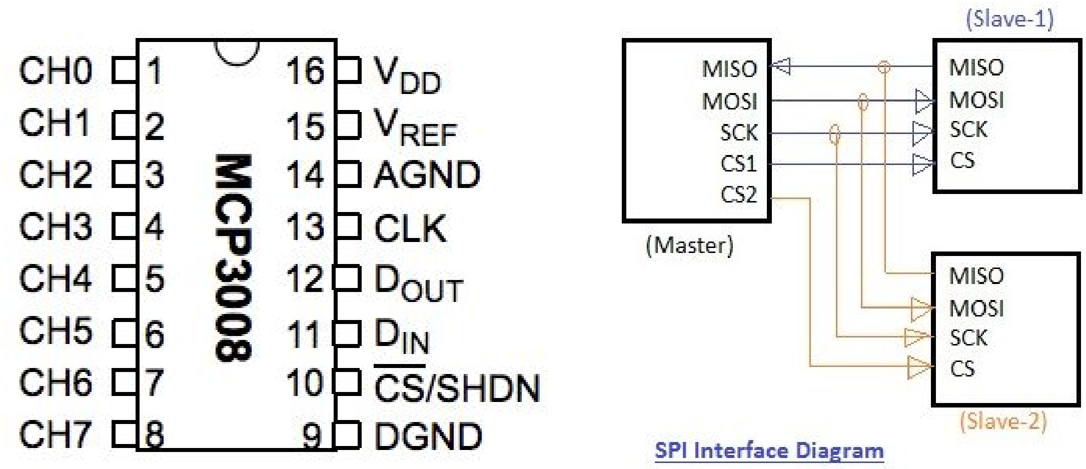


Figure 13: MCP3008 Pinout(https://cdn-shop.adafruit.com/datasheets/MCP3008.pdf) and SPI Interface Diagram(http://www.rfwireless-world.com/Terminology/UART-vs-SPI-vs-I2C.html)

As shown in Figure 10, the Raspberry Pi has SPI GPIO ports. We will use the Raspberry Pi as a master device and the MCP3008 as slave device. With the instruction of the SPI connection layout, we need to connect Dout to MISO and Din to MOSI. The analog signal will be converted to a digital signal by the ADC and then transferred to the Raspberry Pi through SPI.

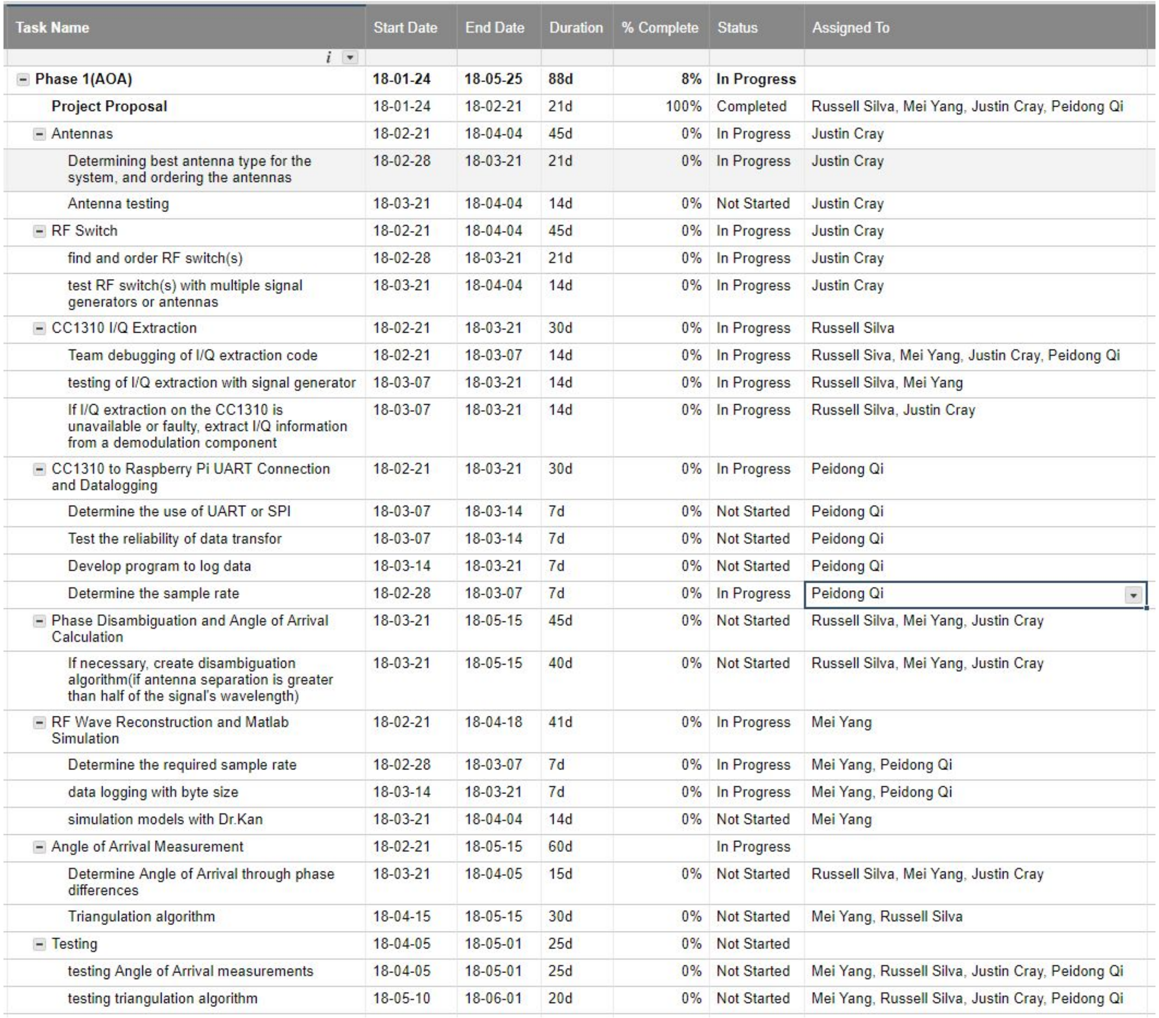
The manufacturer of the CMX970 has already been contacted, and have already expressed interest in our project/providing us with EV9700 evaluation kits for the demodulator. A PE0003 can be used to more easily manipulate the demodulator for a better signal to noise ratio through a GUI. This device is a universal interface card for all of the manufacturer's evaluation kits (see link below).

http://www.cmlmicro.com/products/PE0003_Universal_Interface_Card/

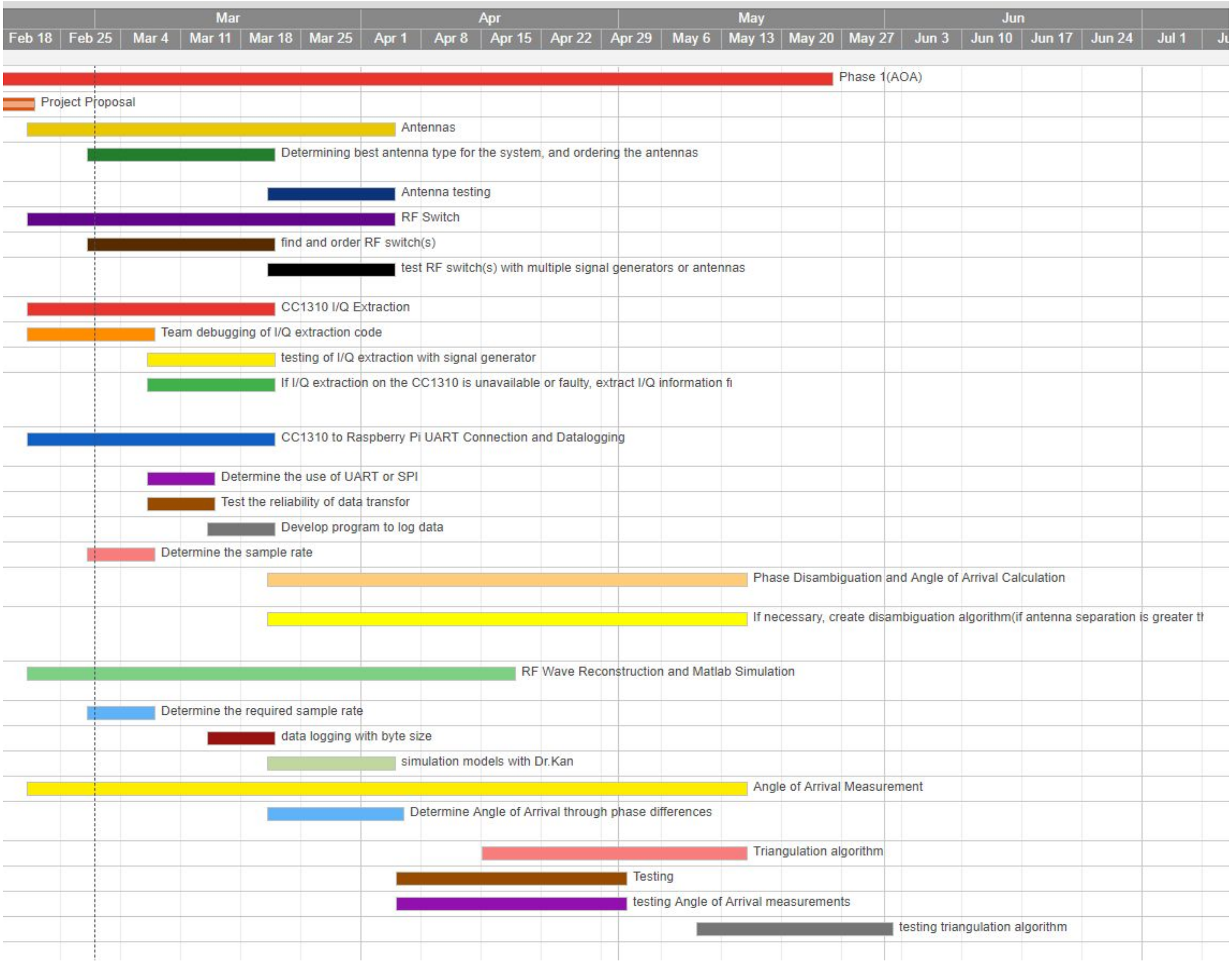
The manufacturer has not listed the cost of these components. It is expected that the cost of these components will be relatively low since they are often bought in large quantities by RF businesses and government organizations.

V. Project Management

Below is the proposed schedule for this semester's allocated work. Please note this is subject to change:



Task Name	Start Date	End Date	Duration	% Complete	Status	Assigned To
Phase 1(AOA)	**18-01-24**	**18-05-25**	**88d**	**8%**	**In Progress**	
Project Proposal	18-01-24	18-02-21	21d	100%	Completed	Russell Silva, Mei Yang, Justin Cray, Peidong Qi
Antennas	18-02-21	18-04-04	45d	0%	In Progress	Justin Cray
Determining best antenna type for the system, and ordering the antennas	18-02-28	18-03-21	21d	0%	In Progress	Justin Cray
Antenna testing	18-03-21	18-04-04	14d	0%	Not Started	Justin Cray
RF Switch	18-02-21	18-04-04	45d	0%	In Progress	Justin Cray
find and order RF switch(s)	18-02-28	18-03-21	21d	0%	In Progress	Justin Cray
test RF switch(s) with multiple signal generators or antennas	18-03-21	18-04-04	14d	0%	In Progress	Justin Cray
CC1310 I/Q Extraction	18-02-21	18-03-21	30d	0%	In Progress	Russell Silva
Team debugging of I/Q extraction code	18-02-21	18-03-07	14d	0%	In Progress	Russell Siva, Mei Yang, Justin Cray, Peidong Qi
testing of I/Q extraction with signal generator	18-03-07	18-03-21	14d	0%	In Progress	Russell Silva, Mei Yang
If I/Q extraction on the CC1310 is unavailable or faulty, extract I/Q information from a demodulation component	18-03-07	18-03-21	14d	0%	In Progress	Russell Silva, Justin Cray
CC1310 to Raspberry Pi UART Connection and Datalogging	18-02-21	18-03-21	30d	0%	In Progress	Peidong Qi
Determine the use of UART or SPI	18-03-07	18-03-14	7d	0%	Not Started	Peidong Qi
Test the reliability of data transfor	18-03-07	18-03-14	7d	0%	Not Started	Peidong Qi
Develop program to log data	18-03-14	18-03-21	7d	0%	Not Started	Peidong Qi
Determine the sample rate	18-02-28	18-03-07	7d	0%	In Progress	Peidong Qi
Phase Disambiguation and Angle of Arrival Calculation	18-03-21	18-05-15	45d	0%	Not Started	Russell Silva, Mei Yang, Justin Cray
If necessary, create disambiguation algorithm(if antenna separation is greater than half of the signal's wavelength)	18-03-21	18-05-15	40d	0%	Not Started	Russell Silva, Mei Yang, Justin Cray
RF Wave Reconstruction and Matlab Simulation	18-02-21	18-04-18	41d	0%	In Progress	Mei Yang
Determine the required sample rate	18-02-28	18-03-07	7d	0%	In Progress	Mei Yang, Peidong Qi
data logging with byte size	18-03-14	18-03-21	7d	0%	In Progress	Mei Yang, Peidong Qi
simulation models with Dr.Kan	18-03-21	18-04-04	14d	0%	Not Started	Mei Yang
Angle of Arrival Measurement	18-02-21	18-05-15	60d		In Progress	
Determine Angle of Arrival through phase differences	18-03-21	18-04-05	15d	0%	Not Started	Russell Silva, Mei Yang, Justin Cray
Triangulation algorithm	18-04-15	18-05-15	30d	0%	Not Started	Mei Yang, Russell Silva
Testing	18-04-05	18-05-01	25d	0%	Not Started	
testing Angle of Arrival measurements	18-04-05	18-05-01	25d	0%	Not Started	Mei Yang, Russell Silva, Justin Cray, Peidong Qi
testing triangulation algorithm	18-05-10	18-06-01	20d	0%	Not Started	Mei Yang, Russell Silva, Justin Cray, Peidong Qi



VI. Deliverables

There will be individual weekly updates starting from the week of March 5th. The weekly update will include sections on problems, goals, general approach, solutions/testing strategy, and planned course of action with any references used to solve the problem. At the end of the semester, there will also be an individual final report on the semester's work.

VII. 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 basestations. These tags will transmit sub 1-GHz UHF frequencies.

VIII. References

[1] 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/DirectionFindingPaper.pdf.

[2] 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.

[3] "Sub-1 GHz and 2.4 GHz Antenna Kit for LaunchPad and SensorTag," May 3, 2016. http://www.ti.com/tool/CC-ANTENNA-DK2.

[4]: "CC1310 LaunchPad Default Antenna," Nov. 14, 2016. https://e2e.ti.com/support/wireless_connectivity/proprietary_sub_1_ghz_simpliciti/f/156/t/554880.

[5] "CC1310 LaunchPad Design," Jul. 28, 2016. https://e2e.ti.com/support/wireless_connectivity/proprietary_sub_1_ghz_simpliciti/f/156/p/532331/1938371.

[6] "CC1310 SimpleLink Ultra-Low-Power Sub-1 GHz Wireless MCU," http://www.ti.com/lit/ds/symlink/cc1310.pdf.

[7] "CC1310 IQ Samples," http://www.ti.com/lit/an/swra571/swra571.pdf.

[8] "CC13XX Antenna Diversity," http://www.ti.com/lit/an/swra523b/swra523b.pdf.

[9] "Crystal Oscillator and Crystal Selection for the CC26XX and CC13XX Family of Wireless MCUs," http://www.ti.com/lit/an/swra495f/swra495f.pdf.

[10] "The Raspberry Pi UARTS," https://www.raspberrypi.org/documentation/configuration/uart.md.

[11] "UART vs SPI vs I2C," http://www.rfwireless-world.com/Terminology/UART-vs-SPI-vs-I2C.html.

[12] 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.

[13] Weiser, A. W., Orchan, Y., Nathan, R., Charter, M., Weiss, A. J., & Toledo, S. (2016). Characterizing the Accuracy of a Self-Synchronized Reverse-GPS Wildlife Localization System. 2016 15th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN). doi:10.1109/ipsn.2016.7460662

[14] B. Porat and B. Friedlander, “Fractionally-Spaced Signal Reconstruction Based on the Maximum Likelihood”, Proc. IEEE Workshop on Higher-Order Statistics, July 1997.

[15] A. Smith, H. Balakrishnan, M. Goraczko, and N. Priyantha. Tracking moving devices with the Cricket location system. In Proceedings of MobiSys, pages 190–202, 2004.

[16] 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.

[17] Wescott, Tim. “Sampling: What Nyquist Didn’t Say, and What to Do About It.” 20 June 2016, www.wescottdesign.com/articles/Sampling/sampling.pdf.

[18] Wolff, Christian. “Radar Basics.” *Radar Basics - In-Phase & Quadrature Procedure*, www.radartutorial.eu/10.processing/sp06.en.html.

[19] National Instruments. “What Is I/Q Data?” *What Is I/Q Data?*, 30 Mar. 2016, www.ni.com/tutorial/4805/en/.

Appendix A: Relevant Links and Tutorials

External antenna module for the CC1310:

http://www.ti.com/tool/CC-ANTENNA-DK2

In Phase and Quadrature Helpful Tutorial:

http://whiteboard.ping.se/SDR/IQ

Analog to Digital Conversion (Plan B):

MCP3008: https://www.adafruit.com/product/856

ADC Connection to Raspberry Pi 3:
https://learn.adafruit.com/reading-a-analog-in-and-controlling-audio-volume-with-the-raspberry-pi/overview

MCP3008(external ADC) datasheet:

https://cdn-shop.adafruit.com/datasheets/MCP3008.pdf