Wind Sensor: Final Presentation Advisor: Dr. Kuh

Creighton Chan • Jerry Wu • Scott Nishihara

Summary

Project background & motivation

Goal for your project

Acoustic & Ultrasonic:

- **Block Diagrams**
- **Description, Methods & Issues**
- **Final Status & Remaining problems**

Motivation

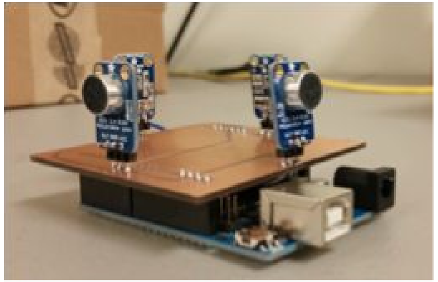
- Knowing the wind patterns (speed and direction) allows for predicting where buildings can be built so that there’s natural ventilation
- Traditional wind sensors are large, have moving parts, and are generally expensive
- We want something that is small, has no moving parts, and is inexpensive to manufacture, something that can be integrated with the weatherbox

Some examples of traditional wind sensors

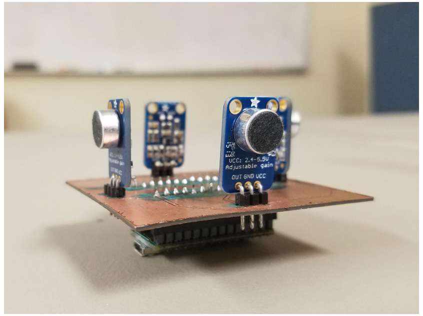
Project Overview

Objective: To build a small, static, and inexpensive, wind sensor that is able to:

- Gather accurate data in real-time on wind speeds and directions using microphones and signal processing
- Be integrated into a weatherbox design



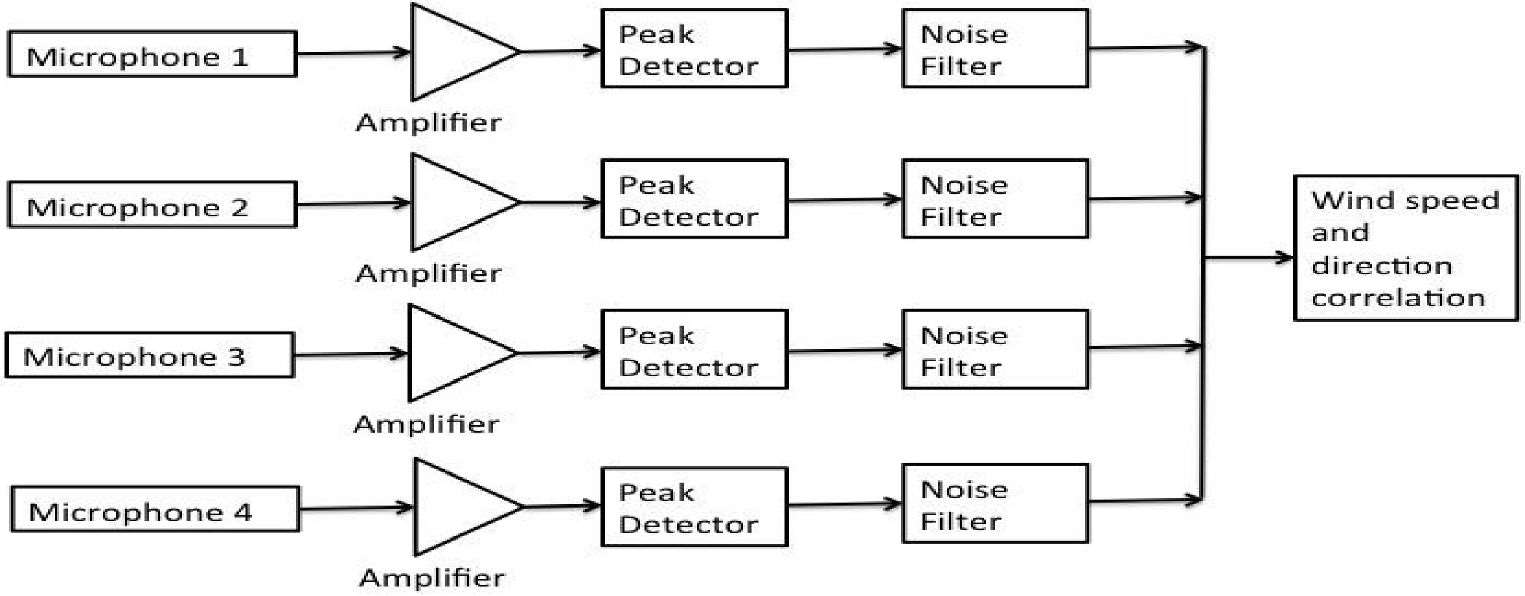
First iteration of the wind sensor using an Arduino and 4 omnidirectional microphones



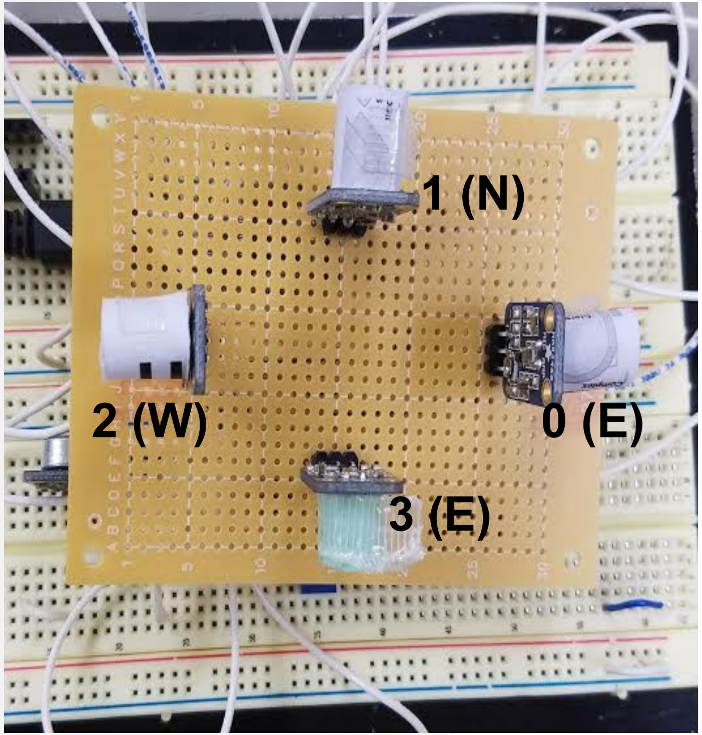
Second iteration of the wind sensor using a Teensy and 4 omnidirectional microphones

Acoustic Wind Sensor

Block Diagram

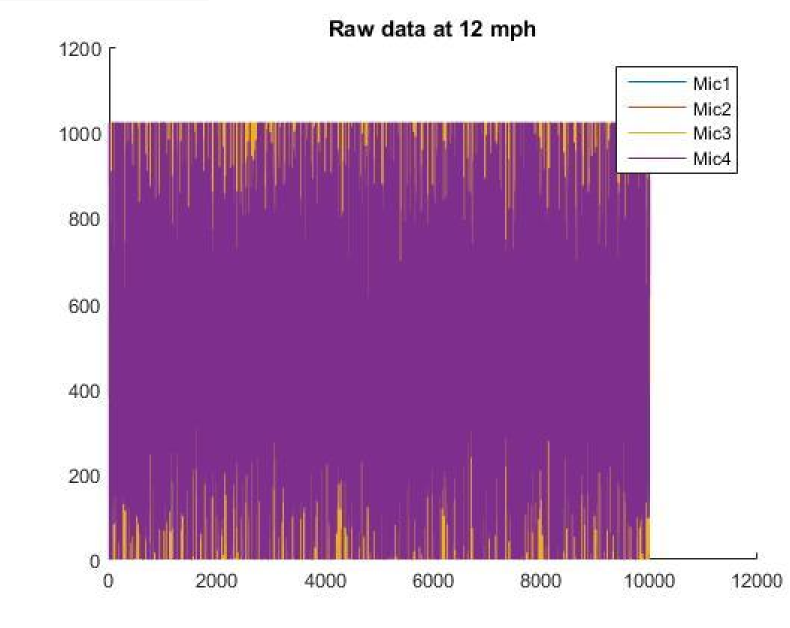


Overall Design

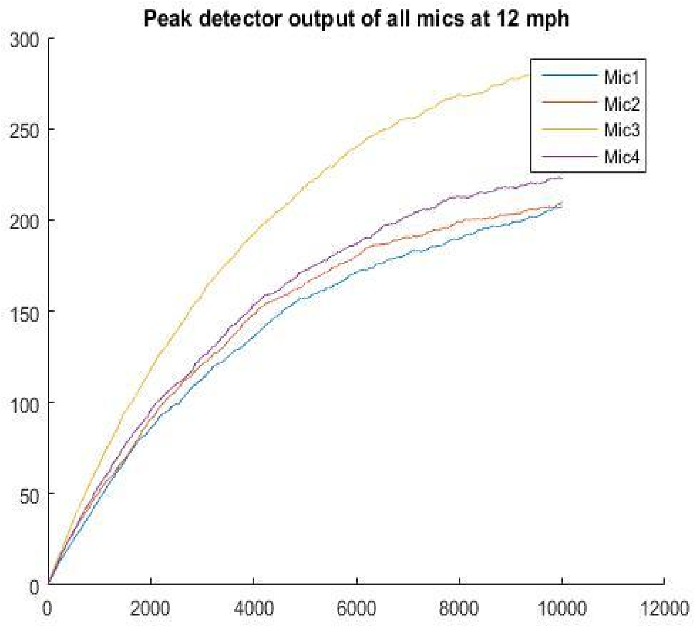


- Uses the amplitudes from the four microphones and our developed algorithms to get the wind speed and direction
- Algorithms:
  - Peak Detector
  - IIR LPF
  - FIR LPF
  - Linearization
- **Costs:**
  - 4 x $6.95 for microphone and amplifier breakout board = $27.80
  - 1 Teensy 3.2 = $19.95
  - Total cost = $47.75
  - Other: Cortex M4 ($5), RMS Chip

Methods

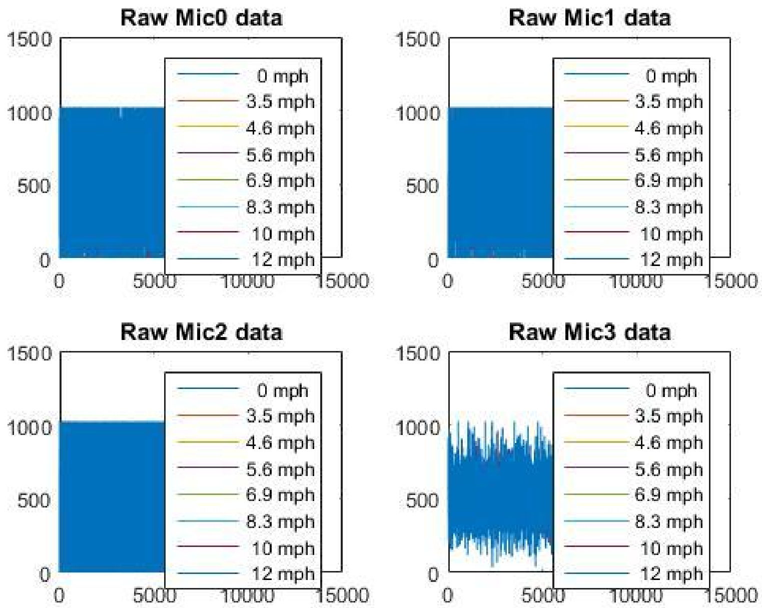


Raw data at 12 mph
Mic1
Mic2
Mic3
Mic4
1200
1000
800
600
400
200
0
0
2000
4000
6000
8000
10000
12000

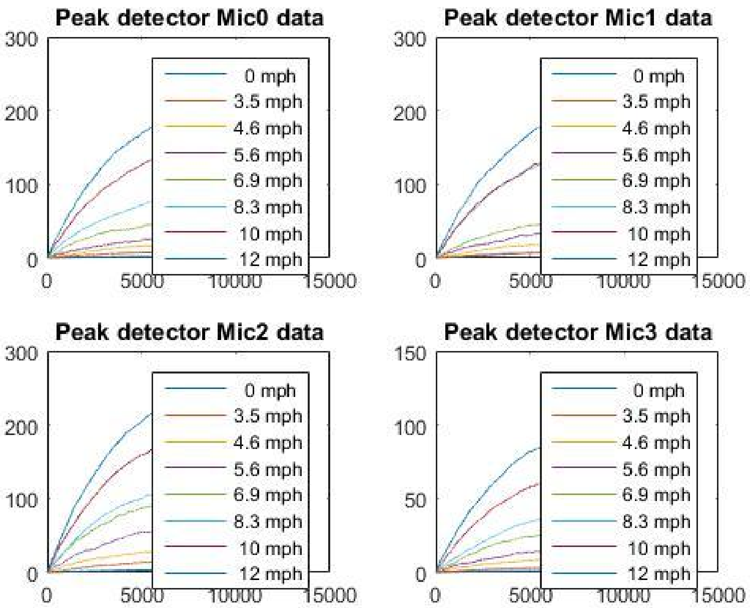


Peak detector output of all mics at 12 mph
Mic1
Mic2
Mic3
Mic4
300
250
200
150
100
50
0
0
2000
4000
6000
8000
10000
12000

Methods

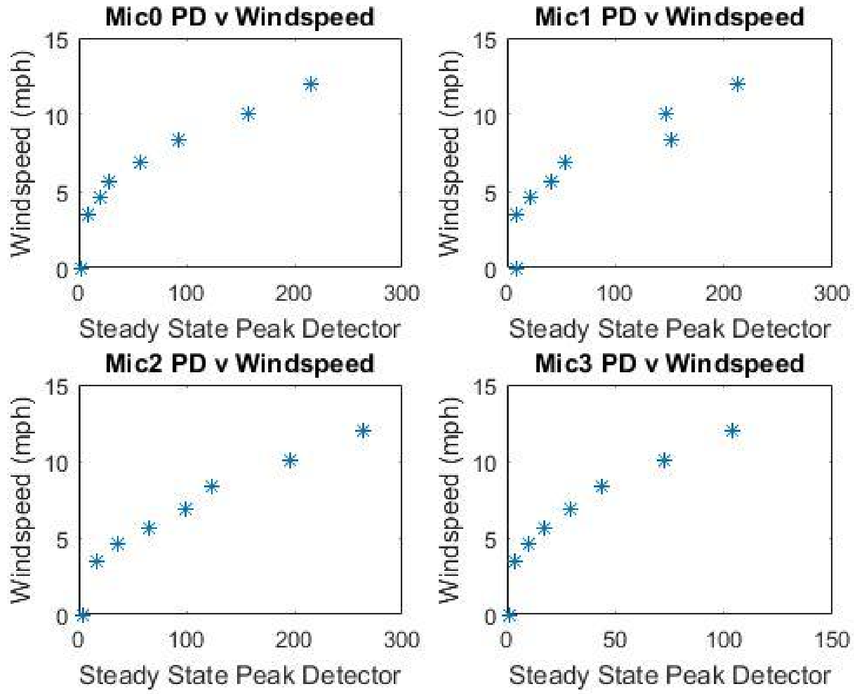


Raw Mic0 data
Raw Mic1 data
Raw Mic2 data
Raw Mic3 data
1500
1000
500
0
0
5000
10000
15000
0 mph
3.5 mph
4.6 mph
5.6 mph
6.9 mph
8.3 mph
10 mph
12 mph



Peak detector Mic0 data
Peak detector Mic1 data
Peak detector Mic2 data
Peak detector Mic3 data
300
200
100
150
50
0
0
5000
10000
15000
0 mph
3.5 mph
4.6 mph
5.6 mph
6.9 mph
8.3 mph
10 mph
12 mph

Methods



Alpha = 0.0004

Samples = 100001

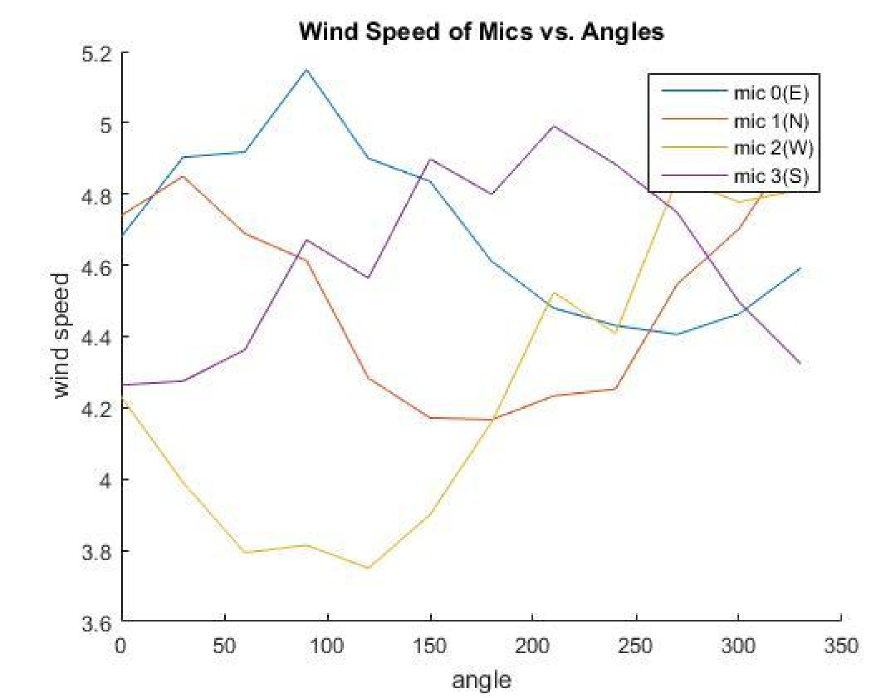
m0 = (pd0)*0.0384+4.0826;

m1 = (pd1)*0.0373+3.8906;

m2 = (pd2)*0.0339+3.3874;

m3 = (pd3)*0.0817+4.0016;

Methods



- Reference Angle: wind source to Mic 1(N)
- Mic measurements recorded at the same time
- Samples: 51/mic
- 30 degree increments between 0 and 360

Problems and Solutions

- Problem: Old PCB actually had faulty connections
  - Inconsistent results due to board
  - Solution: Soldered a perforated board for testing purposes
  - Will probably entail recalibration when housing is designed
- Assumed microphones were identical
  - Microphones did not have the same response under the same conditions
  - Solution: designed each microphone individually

Final Status & Remaining Problems

Final Status:

- Verified Andy's results with four-microphone implementation
- Began verifying algorithm in uncontrolled conditions
- Explored different filtering methods (FIR low pass filter)

Remaining Problems:

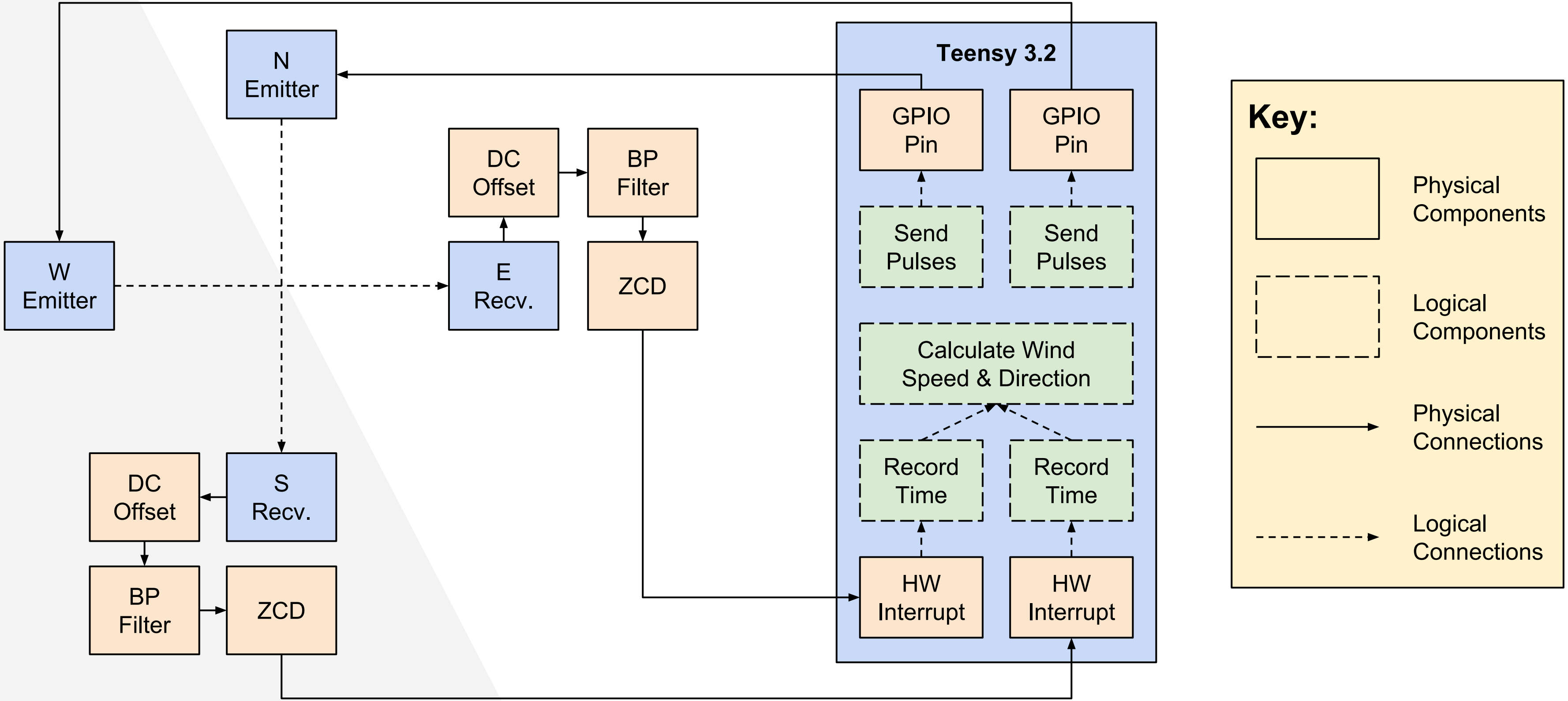
- More testing

Future Improvements

- Build housing
- Look more into frequency domain
- Compare accuracy of different filtering methods
- Power source
- Wireless transmission of data
- Implement Cortex M4 microprocessor

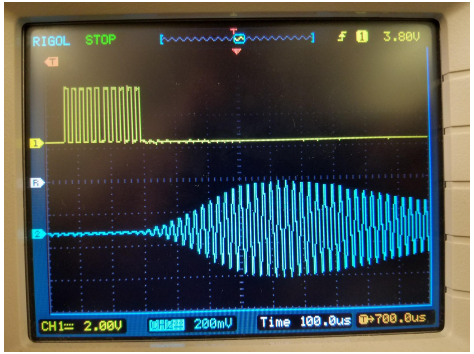
Ultrasonic Wind Sensor

Block Diagram

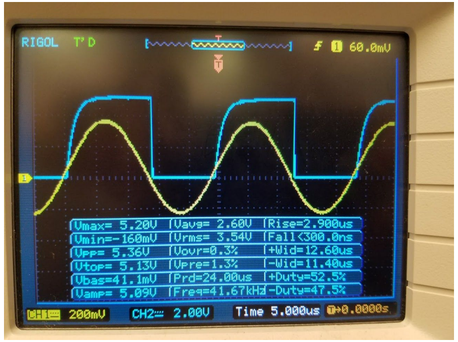


Methods

- Measuring propagation time:
  - Send pulses with the emitter, saving the emit times
  - Use the falling edge of the ZCD to determine the receive time
  - Subtract the emit time from the receive time to get the measured propagation time
  - Average measured propagation time over a large amount of samples



yellow = emitter, blue = receiver



yellow = receiver, blue = ZCD

Methods (cont.)

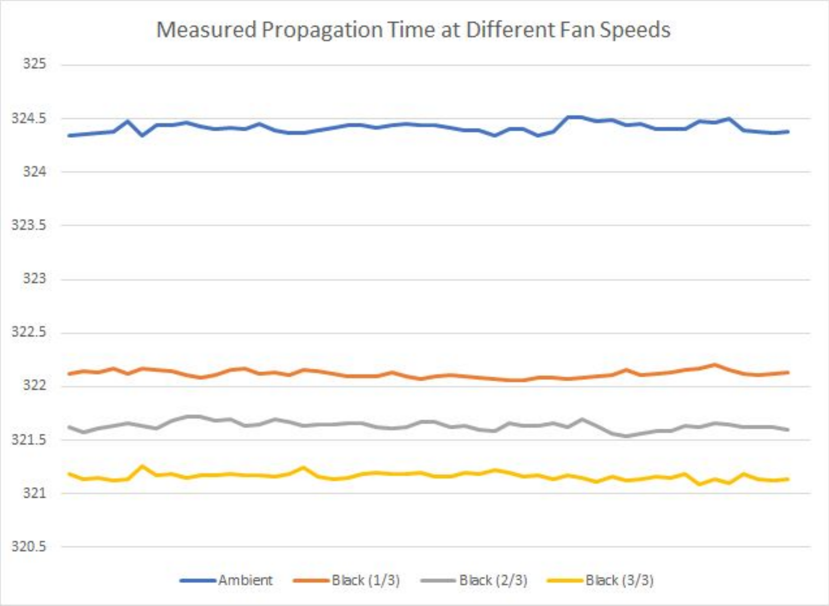
- Determining wind speed:
  - Use the difference between the measured propagation time and expected propagation time based on the speed of sound

$$V_{wind} = \frac{d}{t_{prop}} - V_{sound}$$

Issues

- Measured propagation times are significantly higher than what they should be
- At the distance and temperature we were measuring at, the propagation time should be ~ 204μs
- We believe there may be a constant time delay present, turning our previous equation into:

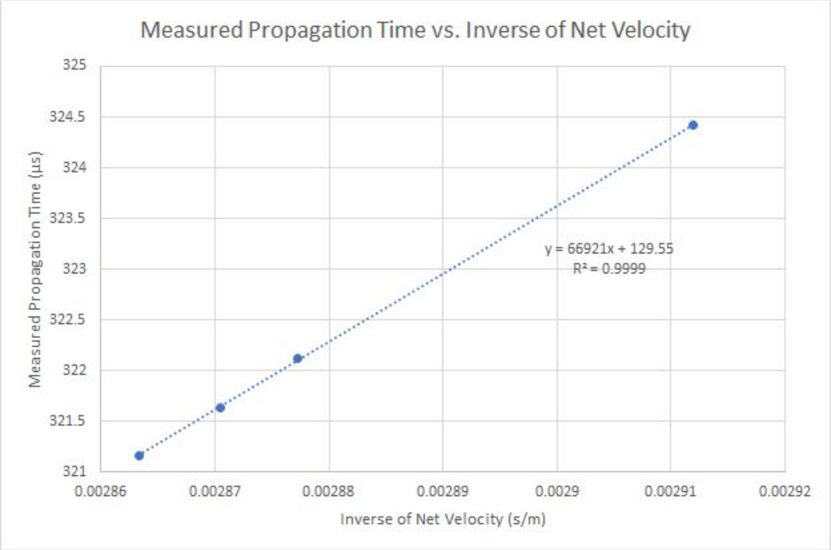
$$V_{wind} = \frac{d}{t_{meas} - T_{delay}} - V_{sound}$$



Results

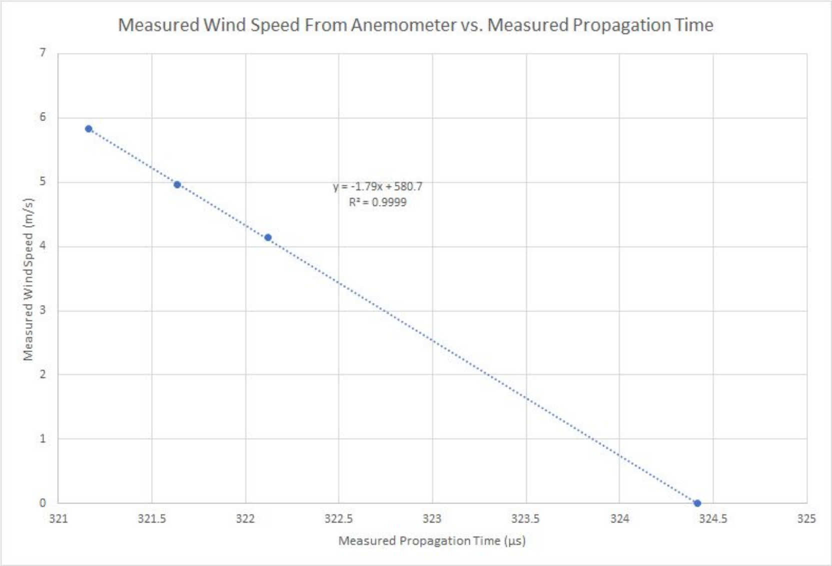
- Took wind speed measurements using the anemometer
- To find the time delay, we plotted the measured propagation time against the inverse of net velocity $(V_{sound} - V_{wind})^{-1}$
- The slope of the plot is the distance
- The y-intercept is the time delay

$$t_{meas} = \frac{d}{V_{sound} + V_{wind}} + T_{delay}$$



Results (cont.)

- We also plotted the measured wind speed against the measured propagation time to get a direct relationship between the two
- We could use this direct relationship to go straight from a measured propagation time to wind speed, but only at a certain temperature



Future Goals

Semester Goals:

- Add current findings to wind-speed algorithm and compare results with anemometer
    - Find a fan with more speed settings (high-powered computer fans--can vary wind speed by changing DC voltage)
- Add a band-pass filter to block out unwanted noise
- Implement direction calculation using two sets of emitter-receiver pairs

Stretch Goals:

- Design an aerodynamic housing and custom PCB
- Integrate into weatherbox design

The end.

Any questions?