**Fall 2024**

**CSE-5342 - Embedded Systems II**

**THE UNIVERSITY OF TEXAS AT ARLINGTON**

Shape

Description automatically generated with medium confidence

**Project report**

**Submitted toward the partial completion of the requirements for CSE 5342**

**By,**

**GROUP 8:**

**Kaven Yarbrough**

**Andrew Espinoza**

**Luis Contreras**

**Under the guidance of**

**Dr. Habeeb Olufowobi**

# Abstract

Our project was a static weather station, with the main feature being an ultrasonic anemometer. Supporting components included a temperature and humidity sensor, wireless communication via the Esp Now Wi-Fi protocol, as well as a lcd display to show the data in an easy-to-read format.  
 The theory pertaining to the ultrasonic anemometer is that since ultrasonic waves are just pressure waves through air, when the air itself is moving the waves will travel faster or slower depending on the direction of the movement of the air. If we apply two pairs of these transducers rotated 90 degrees from each other, given any wind from a direction we can calculate the incoming angle and magnitude of the wind.

# System Design and Architecture

A diagram of a computer system

Description automatically generatedFigure 1. Data flow diagram of the system modules

The high-level overview of our system revolves around having the sensors being isolated from the display (physically). We have the anemometer circuit interact with the TM4C where it will get the incoming signals and calculate the wind speed and direction. The temperature and humidity sensor will interact with the master Esp via i2c to get the measurements. The TM4C will send its data via uart to the master Esp where it will be transmitted to the slave Esp and then form at and display to the lcd screen.

# Implementation

The anemometer module consists of two basic circuits to condition the output signal into something readable for the TM4C.

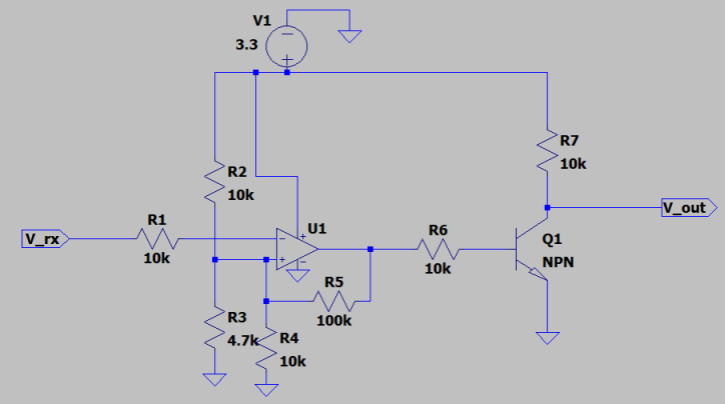


Figure 2. Receiver Circuit Diagram

Above depicts the circuit used to condition the raw analog output of the ultrasonic transducer into a square wave. It is broken up into two parts, the pre-amplification stage, and the switching stage. In the pre-amplification stage the input is a can be represented by:

We apply a gain of 10 to get a 2 V peak to peak sin wave of the same frequency, as well as a DC offset to create the positive only signal. This is then passed to an NPN transistor switch to square off the signal more and provide faster rising edges that we can pass to the GPI of the TM4C.



Figure 3. Final Conditioned signal (In Purple)

For the software side of the anemometer module, it consists of a simple main loop that calls two symmetrical functions that return the phase delay between the transmit and receive signals. Each function is just a loop that is repeated a set number of times.

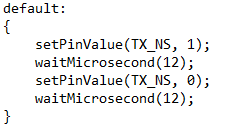


Figure 4. Default case for transmitting 40Khz square wave

This code snippet demonstrates how we drove the transmitter lead, using looped on/off pin driving. Where the delay in between the toggling gives a 50% duty cycle signal with frequency of 40 kHz.

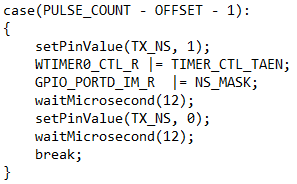


Figure 5. Start of capture

Here in this snippet of code, it shows the iteration into the pulse that we start to measure the phase shift between transmit and receive. It starts with the enabling of a one-shot timer and the GPIO interrupt for the receive pin.

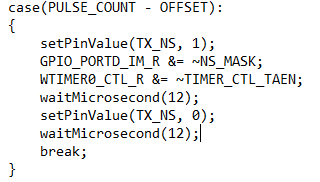


Figure 6. End of capture

In the next iteration of the loop, we will disable the timer as well as the GPIO interrupts to prevent false triggers on rising edges we do not want to read.

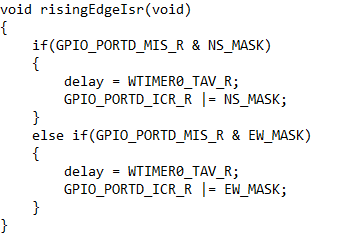


Figure 7. GPIO ISR to handle the desired rising edge

This function is what is executed on a rising edge event on the receive pin, this is the GPIO interrupt service routine that is enabled and disabled for only a single period of the transmission (at its most stable peak for better signal stability). This ISR will save off the current timer value to a global variable for use later.

Wireless communication module: in this project the TM4C would communicate with the esp32 via UART

Sending wind speed and direction. The information sent from the anemometer would be sent to another esp32 with the ESP-NOW, a wireless communication protocol on the data-link layer, which reduces the Five layers of the OSI model to only one. For this implementation They were configured in a master and slave configuration. To go into further details the master configuration is responsible for initiating the communication and managing the communication sessions.

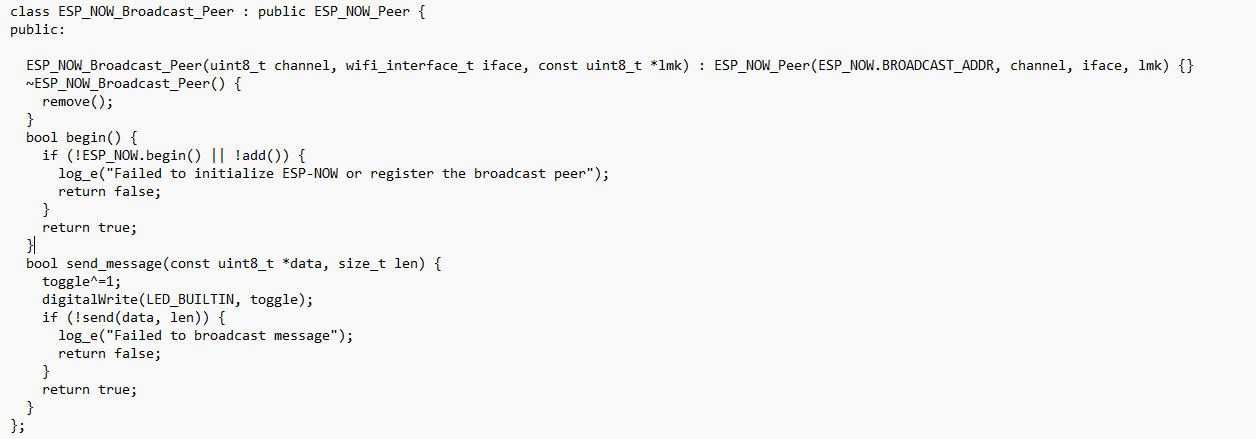


Figure 8. ESP\_NOW\_BROADCAST\_Peer

The code snippet demonstrates the construct initializes an object of the ESP\_NOW\_BROADCAST\_Peer class. It calls the constructor of the base class (ESP\_NOW\_Peer) with specific arguments channel, iface ,and lmk.

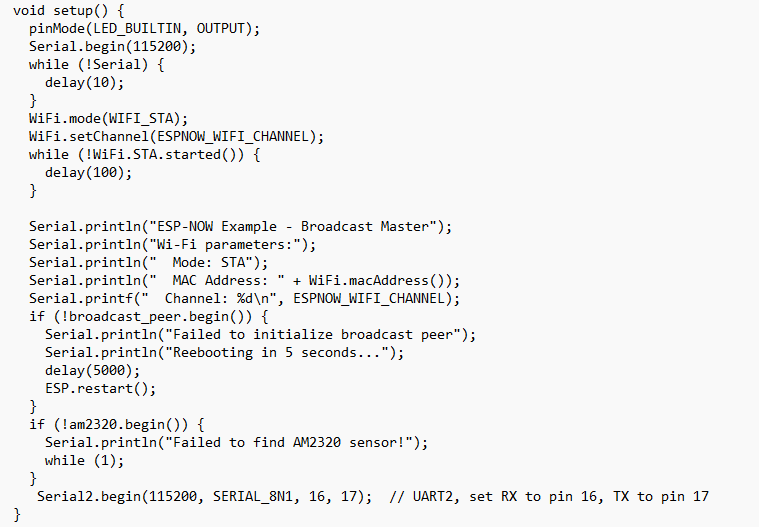


Figure 9. EspMasterSetup

This code snippet configures the devices for ESP-NOW communication, sets up the sensor initializes the AM2320 for reading and temperature and humidity via the use of I2C and finally set up the uart communication.

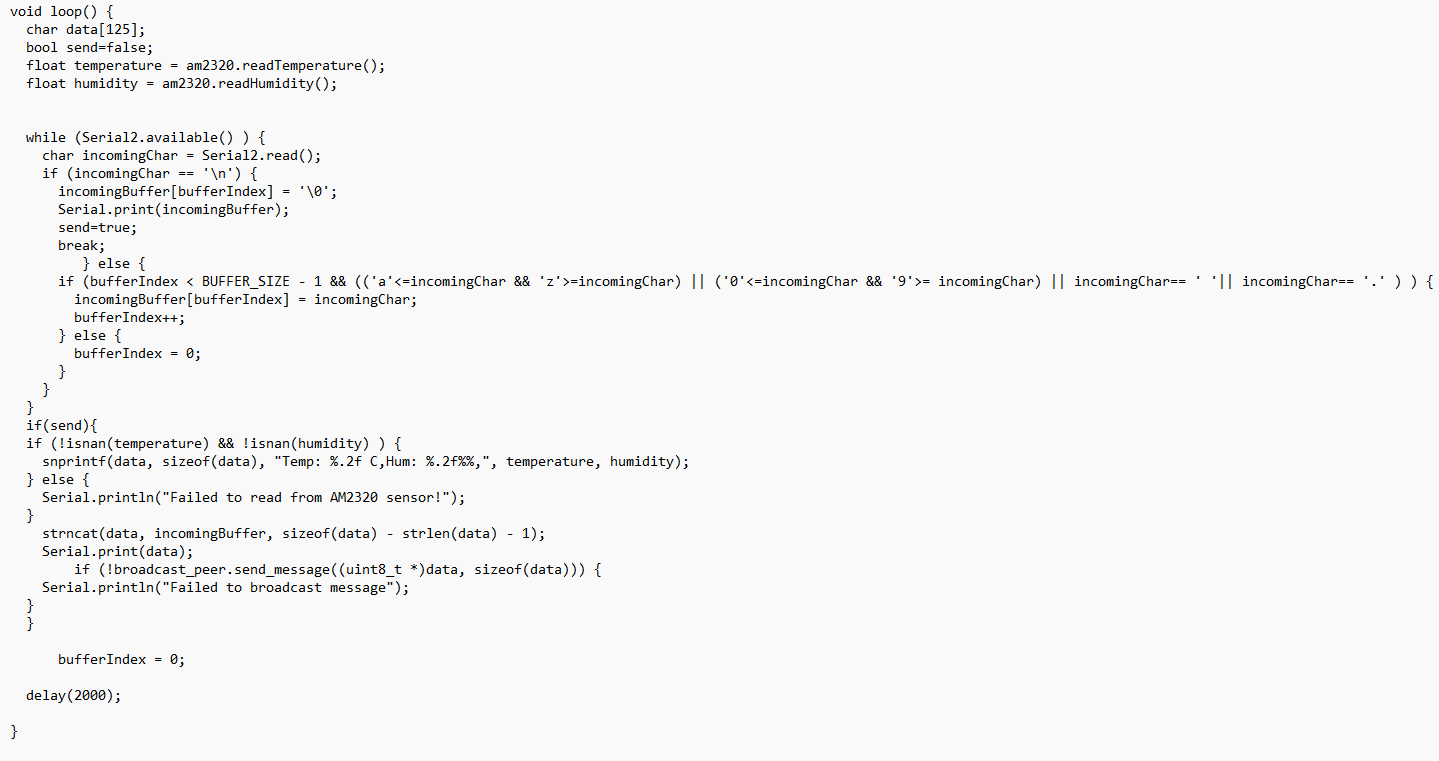


Figure 10. ESPMasterMain

The code snippet implantation of the FIFO buffer on receives of the UART signal extract only the want information Onces a null charter is received set a flag to broadcast the information in the buffer and check the tempters humidity sensor for valid readings.

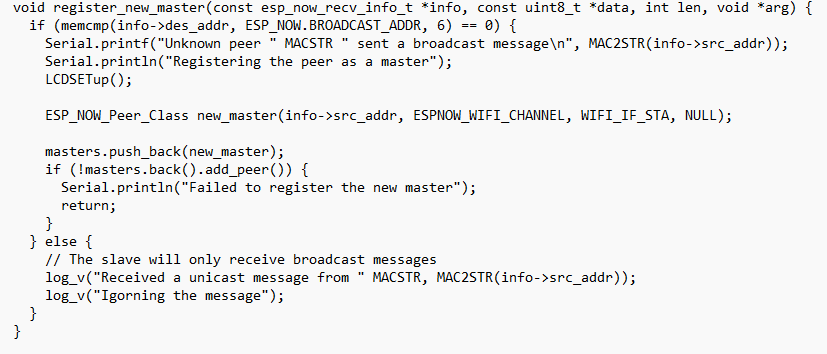


Figure 11. ESpSlaveRegsterMaster

This code snippet is part of a system that listens for incoming broadcast messages form unknow peers, identifies them, and registers them as “master” in the ESP-NOW network. It helps in dynamically discovering new masters' devices and integrating them into the communication network, while ignoring unicast messages that do not require action.

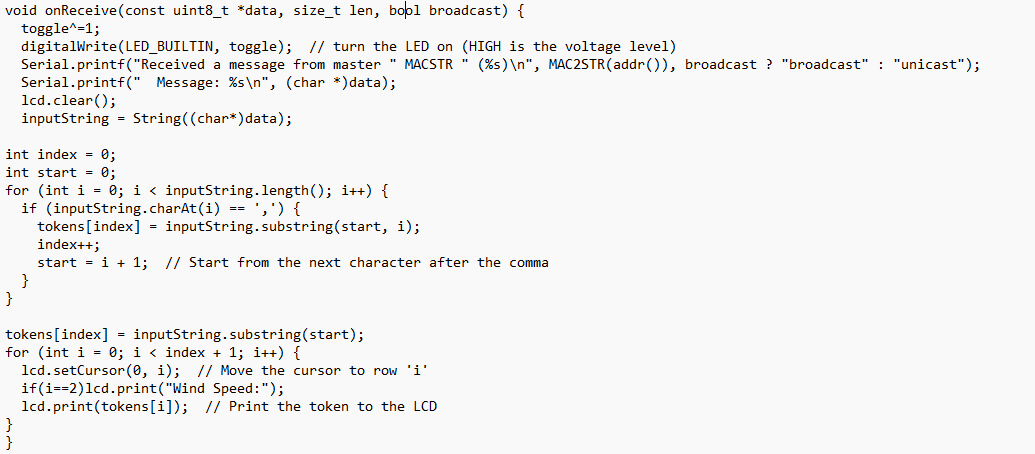


Figure 12. ESpSlaveOnRecive

This code snippet demonstrates a method in the class ESP\_NOW\_BROADCAST\_Peer that gets called when a messaged is received from a peer. The received message is parsed based on the format to be printed out onto the LCD display

# Results and Testing

Using an inverting op amp and a bjt the incoming 200mV pk-pk receive was amplified and then converted to a more digital looking signal. When blowing on the path of the transducers this amplified signal experienced a phase shift either left or right. We created an interrupt to detect this digital signal and measure the time from edge to edge. This measured time is directly proportional to the wind speed experienced by the system.

A screen shot of a graph

Description automatically generated

Figure 13. No wind

A screen shot of a graph

Description automatically generated

Figure 14. Blowing north to south (phase shift left)

A screen shot of a graph

Description automatically generated

Figure 15. Blowing south to north (phase shift right)

The system was able to accurately measure the ambient humidity and temperature of the room that it was sitting in. It also changes temperature and humidity if someone grabs the sensor and holds onto it warming it up.

A screen with white text

Description automatically generated

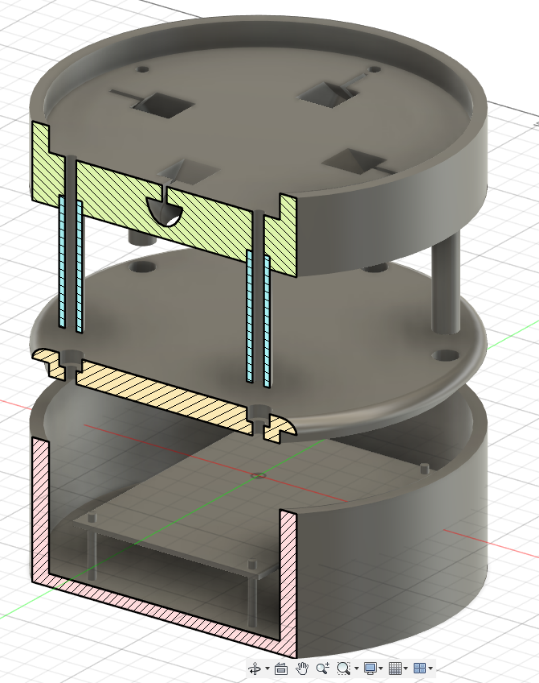
Figure 16. Ambient temperature and humidity of the room

# Conclusions and Future Work

There are changes that should be made with this project to fix its functionality and expand on that functionality. The first big change should be in the way that the C code is implemented so that it can receive anemometry data, calculate the wind speed and direction and transmit that data to the ESP-32 via UART. The next big change would be to make both ends of the system battery powered to eliminate the need for a computer. To eliminate the overhead of the ESP-32’s RF transceivers could be implemented so that we do not use three microcontrollers. The static weather machine is very expandable as well, we could implement many sensors such as rain, light and air pressure. The design of the anemometer as well as the LCD screen display could be changed to be more protected from the elements.

In conclusion, the static weather detection machine is a robust and expandable project that does not have the limitations of a traditional anemometer with moving parts. By using ultrasonic transducers, the weather machine was able to detect wind movement in all of the cardinal directions using the delay in the time-of-flight and phase shift of the received packets. Using a humidity and temperature sensor alongside the anemometer added important information to the system to better detect the environment that the machine is in. It is important that the LCD system was created and implemented wirelessly so that the user can tell the conditions of the weather outside of their immediate area. This approach to a weather detection system increases measurement reliability while also increasing the scalability which is important for expanding and integrating the project into other systems.

# Completed Build

 A round object with wires and wires on a table

Description automatically generated

# Other Deliverables

* Videos of our weather machine
  + <https://youtu.be/CqcqXPiUqNM>
* PowerPoint presentation
  + 