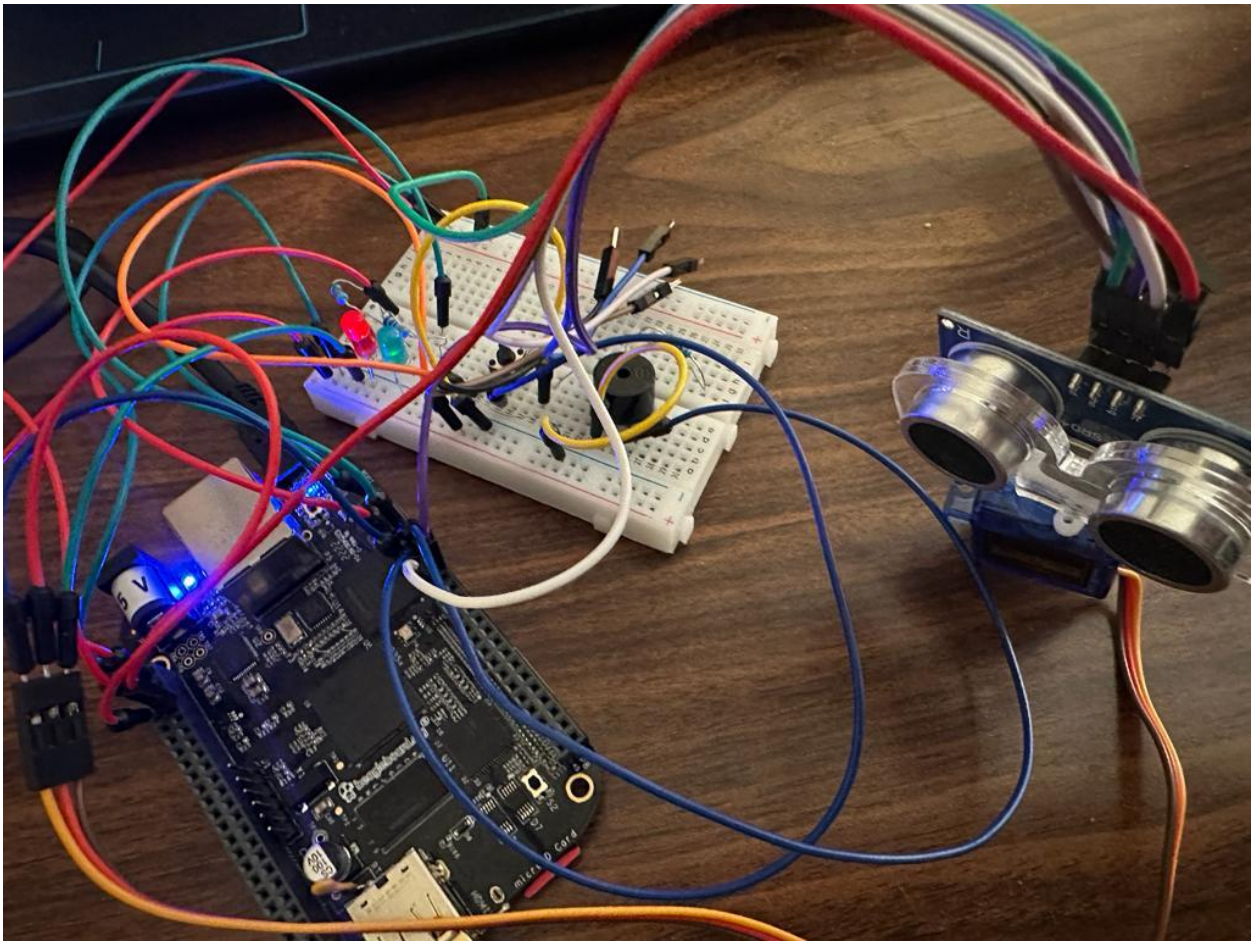
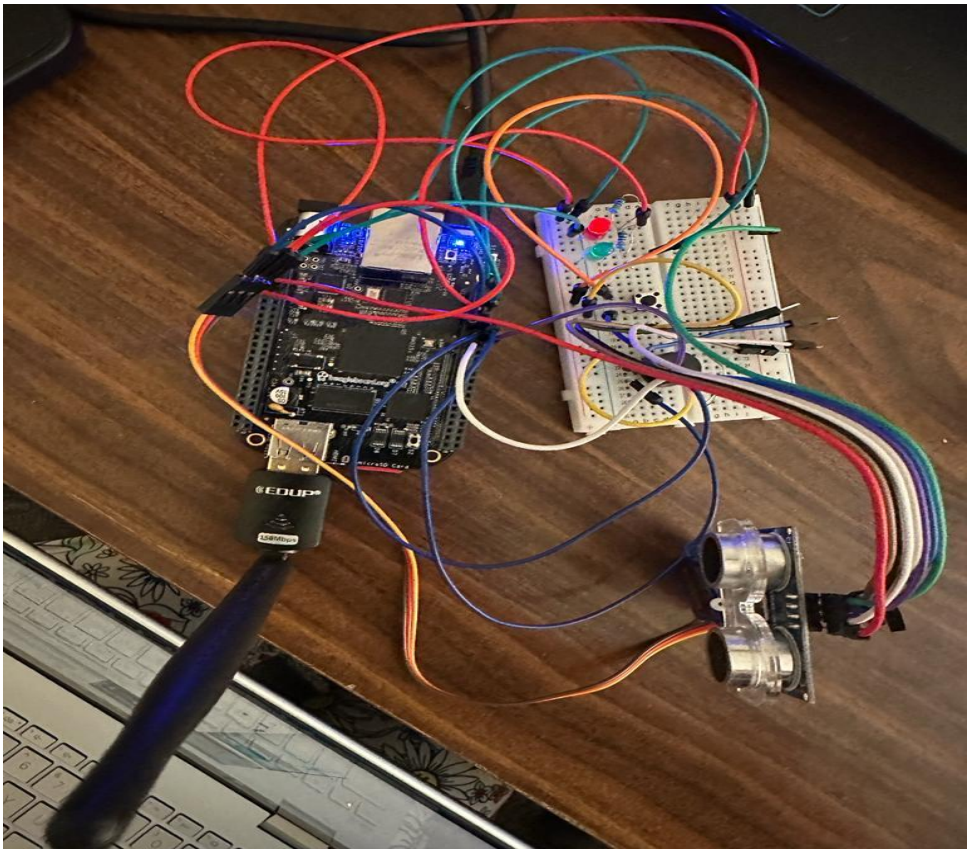
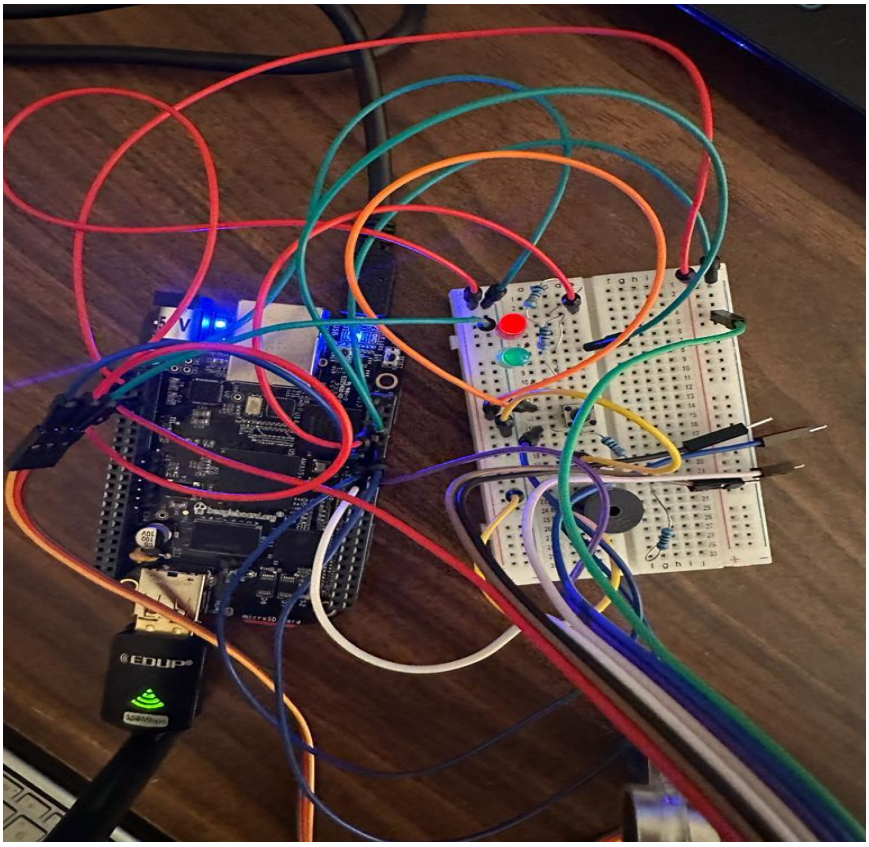
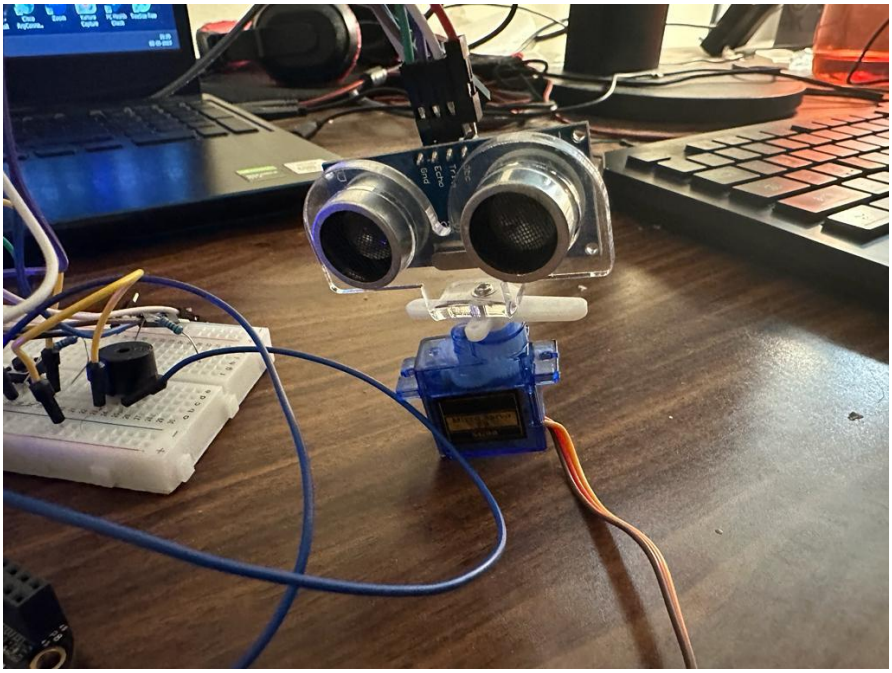
Demonstration

The following images show the setup for the working system of the Object Proximity Detection System.

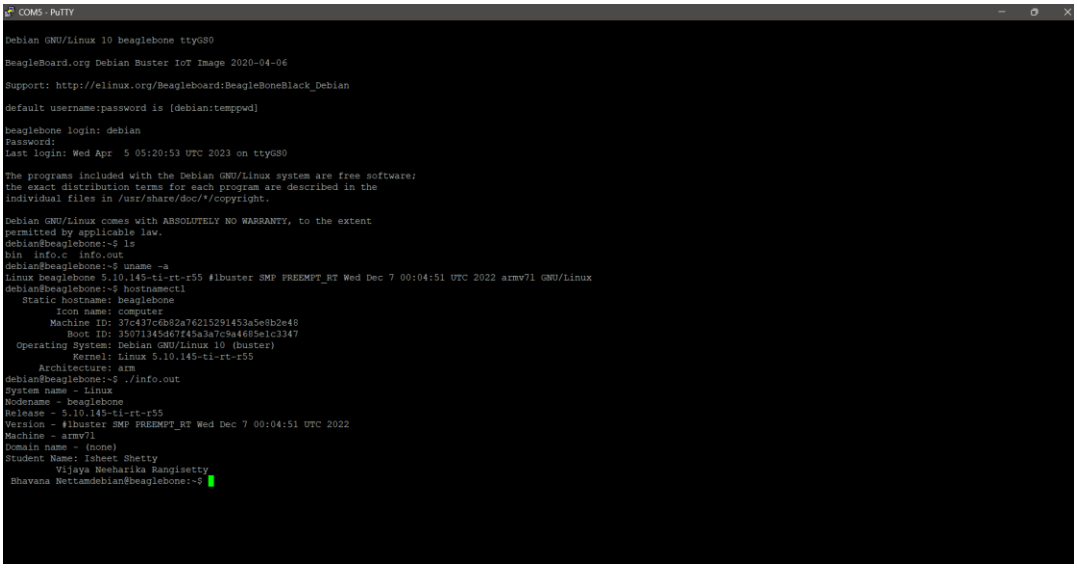



EDUP

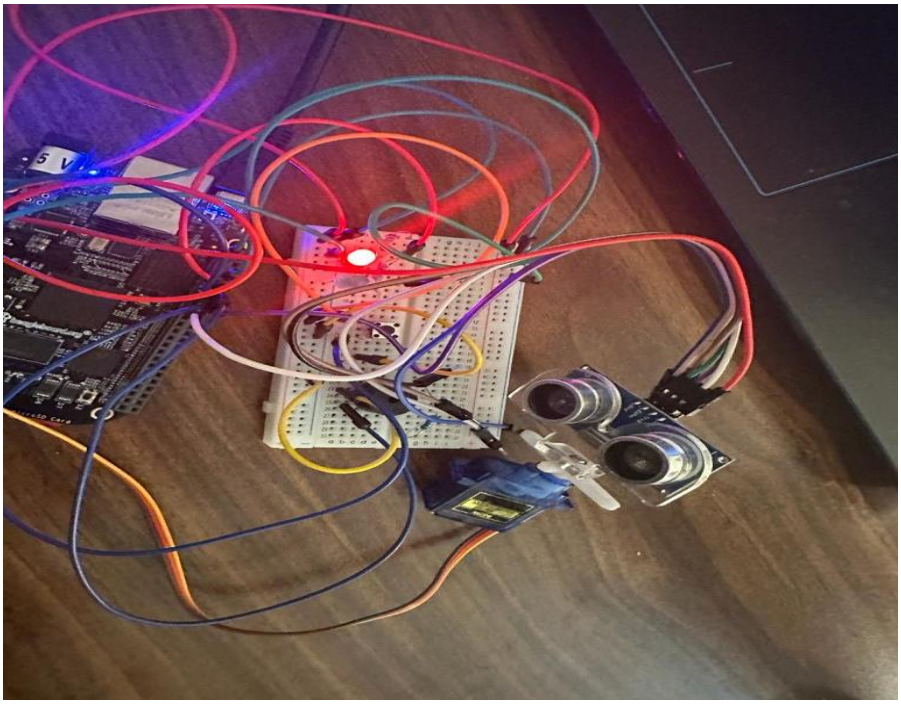

EDUP



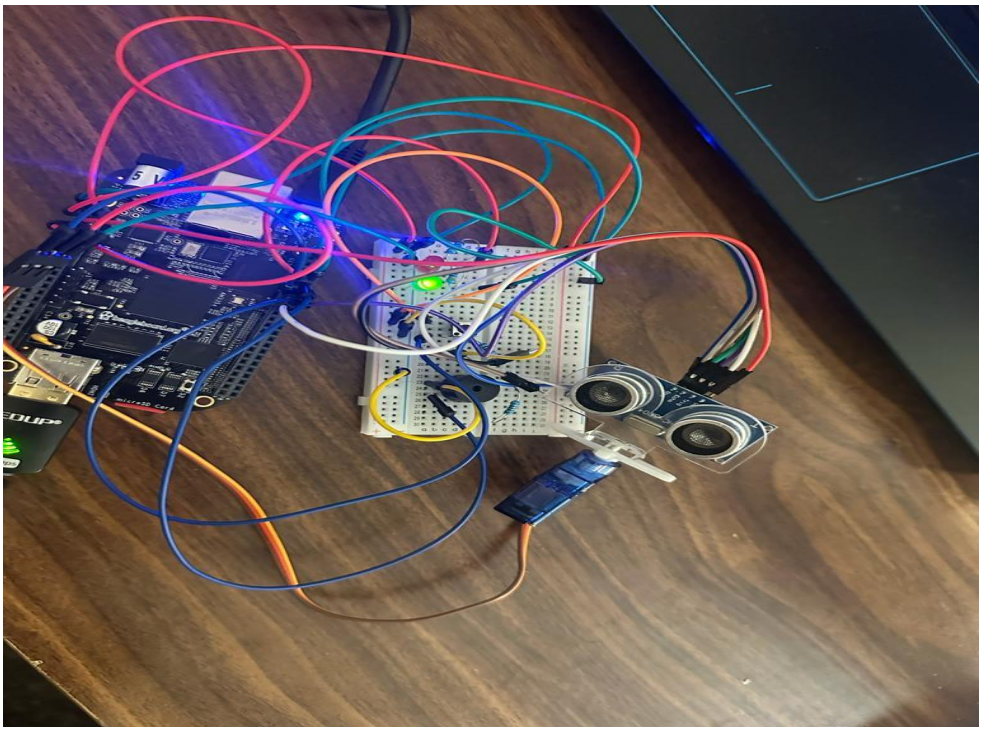
Url to the working video:
https://drive.google.com/file/d/1fhrQoCWaErns7PuNUYxjt1QObyAbE2yA/view?usp=share_link



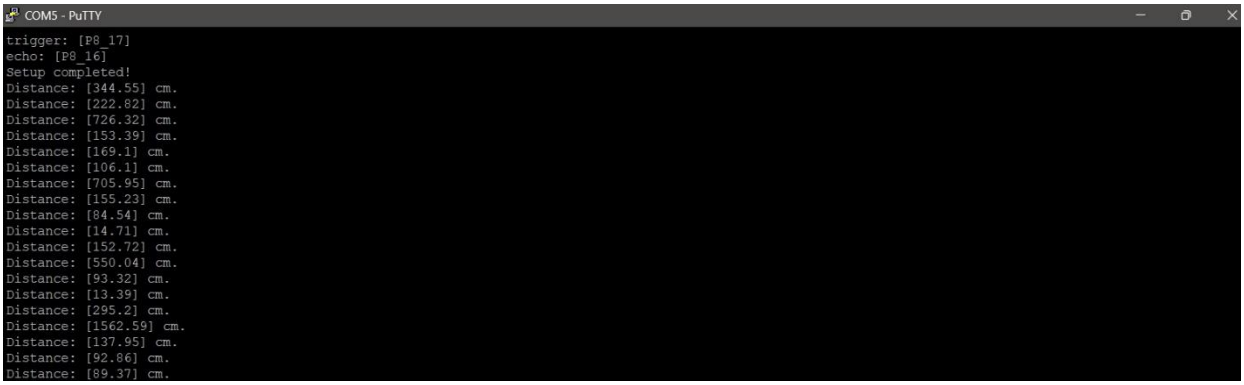
The kernel used was an RT kernel as shown in the above image



When an object is near 30 cm red led glows and buzzer beeps.



When an object is farther than 30 cm green led glows and buzzer doesn’t beep.



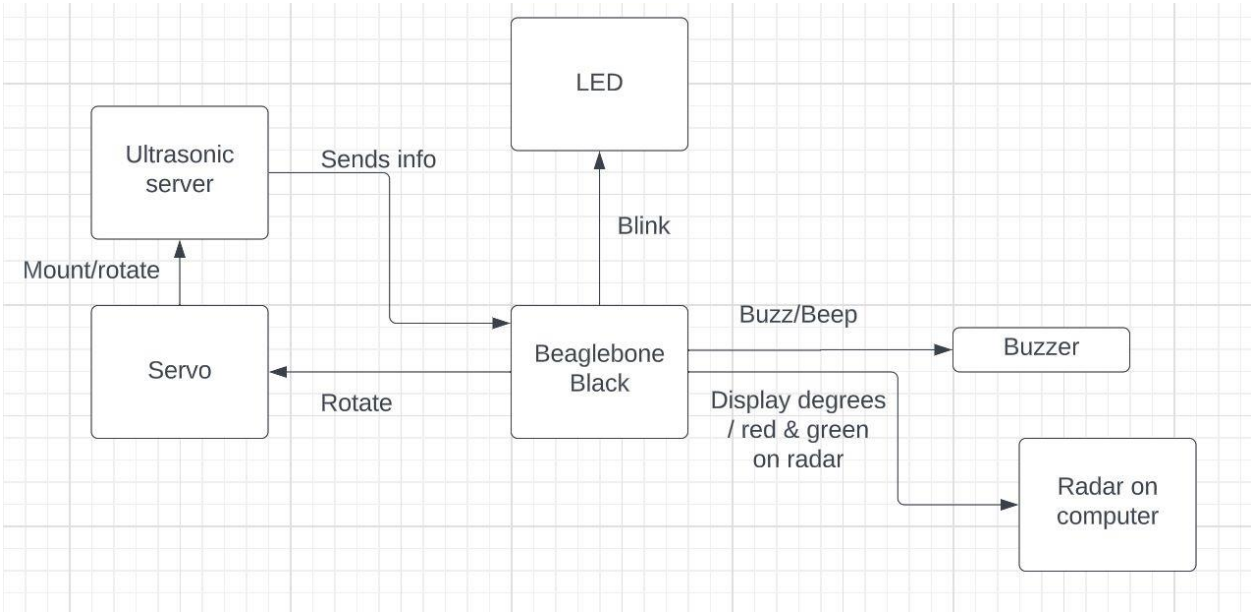
Output in the kernel shows the distance at each second.

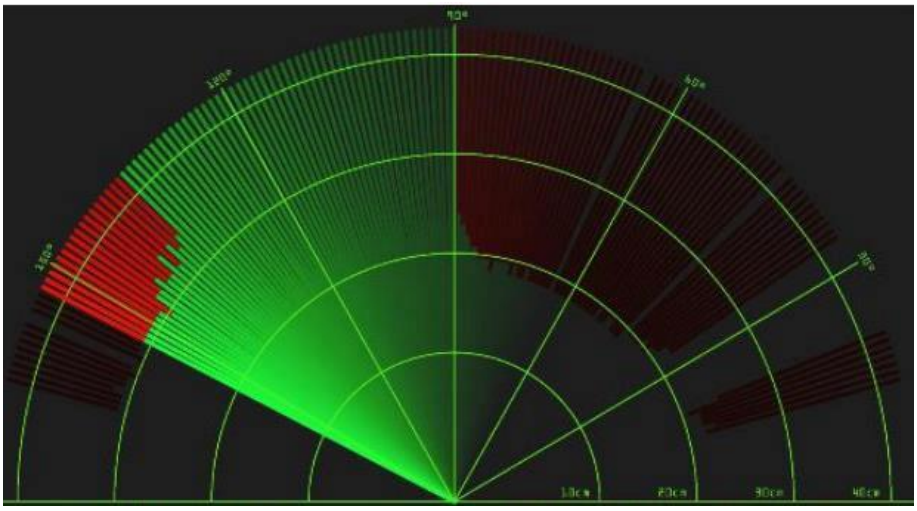
What was planned?

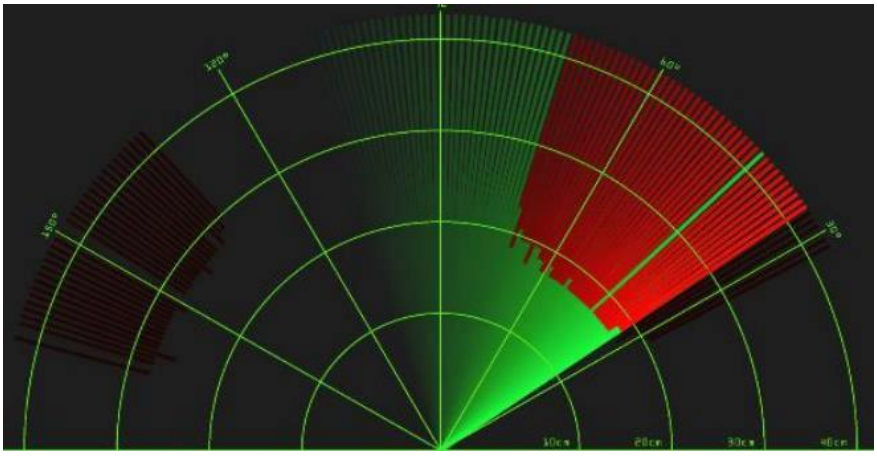
The following diagram architecture shows the planned architecture which included a displaying degree with a radar on the computer but due to time constraints and the beagle bone OS being corrupted a few times and the ultrasonic sensor PRU working to take a longer time than expected caused the graph radar to not be implemented. This can be considered as future work along with the working of the ultrasonic sensor with the working of the ultrasonic sensor most customized and user friendly not just for the user but also for the developer.

The connection of the ultrasonic sensor can be made much easier in C as compared to the roots taken as there can also be an alternative sensor that can be considered for this project, for example. PIR sensor.

The execution of the project in C was very problematic with lots of additional systems required which can be considered as a topic for future reference.

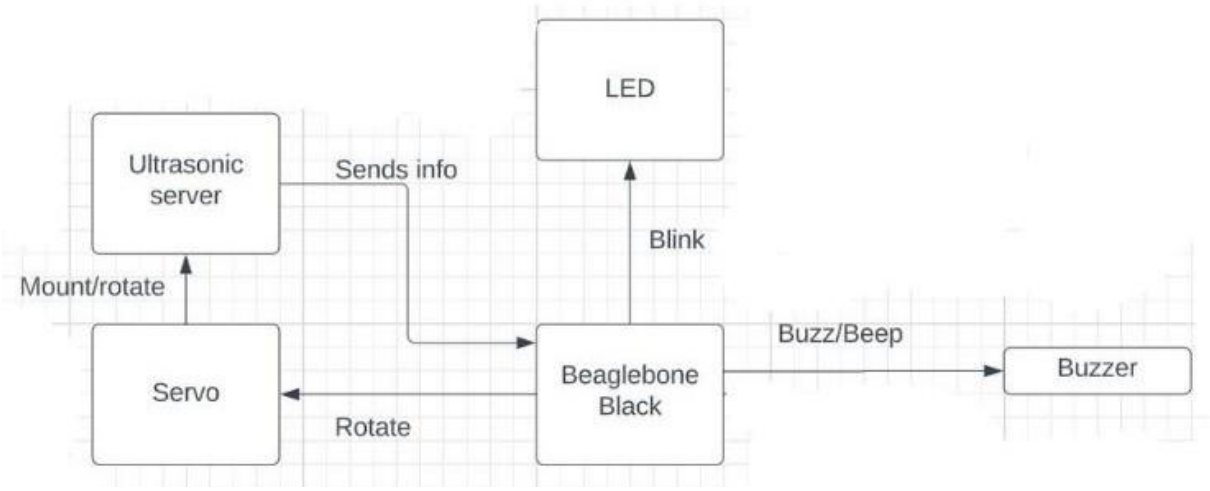


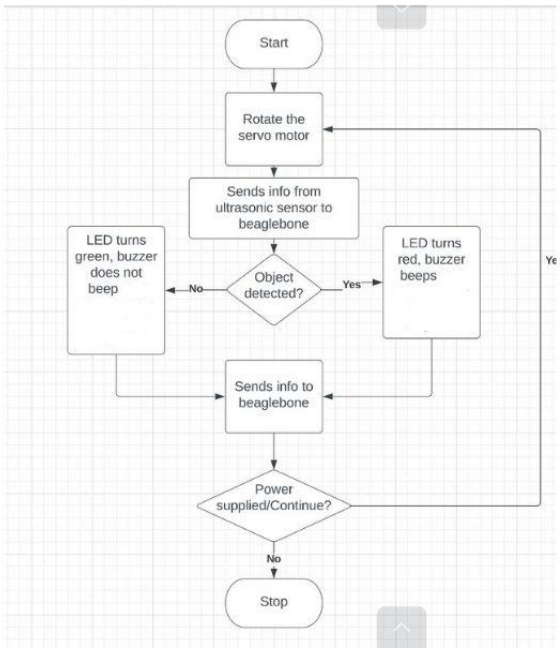




What was accomplished?

Apart from the graph all the required components were working perfectly and also an additional button could be added to the project which controls the servo motors' start and stop. This excellent accomplishment was not just executed in C but also had a Python implementation which took more execution time and was very slow this helped to distinguish between real-time fast working programs in C vs Python.





Finally, we successfully implemented almost 80-90% of the project not just in C but also in Python and we have added images and videos which show the working of the project.