SentiBots: smart, centimeter sized swarmbots

Project by: Sudharshan & Isaac   Mentor: Prof Prahlad Vadekkapat

INTRODUCTION

This project aims to achieve a hardware and software optimized swarm flight platform that allows for swarm developers to have an easier time developing and using swarm platforms in real life applications.

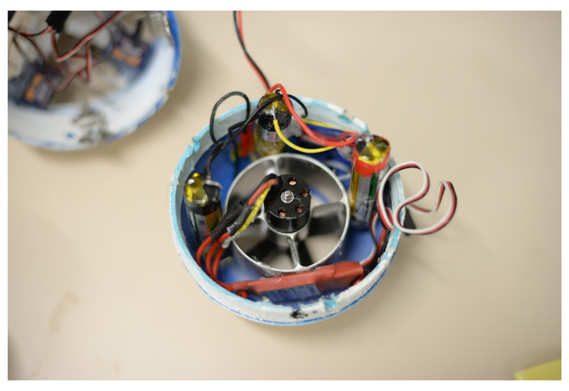
Hardware

Frame Design

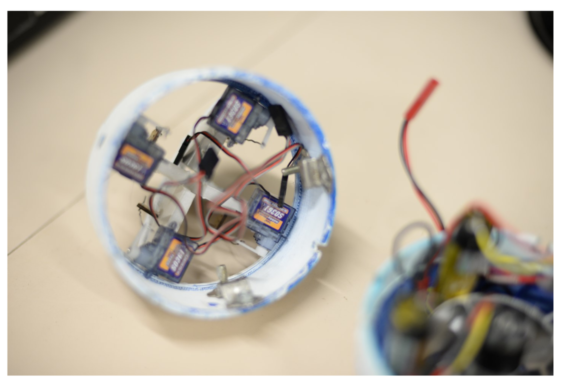
Our frame design goes against the conventional design of 4 motors by replacing it with a single high power Electronic Ducted Fan (EDF).

This lowers the complexity and chance of failure and allows for much better static thrust performance which is ideal of hovering in these types on drones.

We went for control surfaces to direct air in the same way airplane flaps do. By enclosing the whole design in a bean shaped frame, the frame contains all the essential electronics inside the frame and prevent damage to the moving components.



The bean-shaped frame is very modular, modularity is very important as users of the platform may want to add other sensors to enhance the robot's functionality. The design is split into the top half and the bottom half. The top half of the frame is dedicated to the "power" while the second half of the frame will be used to contain the "intelligence" electronics. The 2 halves are connected by a joining piece which you bolt into a captive nut. This allows for easy modifications to the frame without excessive modifications to be made to the frame itself.



Software

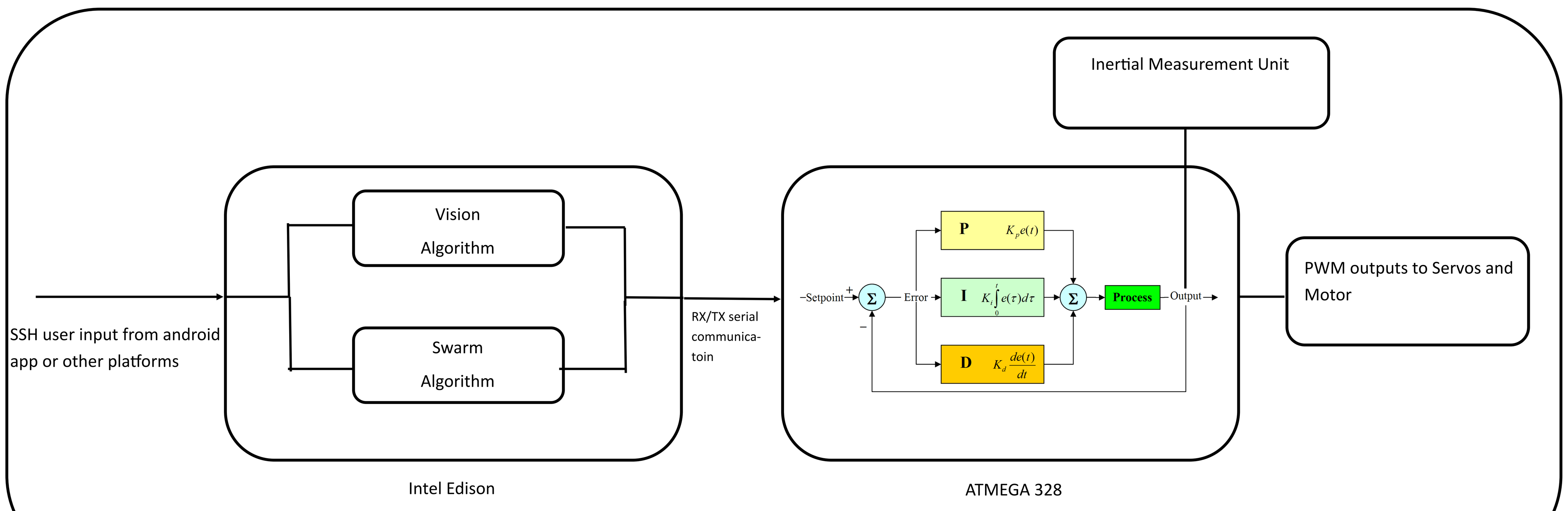
Algorithms

The bot is stabilised by a PID Loop in the Arduino. The vison system uses the mono front facing camera and feeds it to the Edison to do **S**imultaneous **L**ocalisation **A**nd **M**apping (**SLAM**) to navigate and create a 3D map . The 3D map is only shared between the swarm to minimise the amount of information needed to be communicated.

This algorithm would treat each group of bots as a swarm. So, when it has 2 or more paths to explore, the swarm would split equally if possible. Each swarm is also assigned a leader. When a swarm splits, the leader of the swarm will assign each of the new swarms a new leader. If the number of bots is lesser than the number of paths, the location of an unexplored path is marked. When a swarm meets a dead end, it would communicate to the other swarms. If there are any paths left unexplored, the leader communicate to the previous leader. If the leader has any unexplored paths, the swarm would follow the path back to the point of splitting and then follow the path to the explored path it is assigned to by the previous leader while also marking the path as exploring. Otherwise, the swarm would converge with the other swarm with the path provided by the previous leader. Upon finding the exit, all the bots would be recalled unless specified to find multiple exits.

Computer Pooling

Computer pooling model optimizes the swarm performance by allowing them to act as a supercomputer. This is done by making each bot a node in a system that forms the basis of a supercomputer which is connected via Wi-Fi. This pooled computing power therefore allows the swarm to make complex decision as a whole. This allows the swarm to think as a whole where the sensor data can be pooled to allow the swarm to survive as a whole.



Flowchart of Hierarchy of control in the SentiBot

SPECIFICATIONS:

Dimensions: 100mm x 100mm x 250mm

Flight time: 10 - 15min

Motor thrust: 450g

Payload weight: 100g

Top speed: 20km/h

Results

The results of initial testing were quite positive. The SentiBot is able to remain in constant flight even without any PID stabilization algorithms and remain in relatively oscillation free flight when there is PID stabilization. Furthermore, the drop tests were significantly positive showing that the SentiBot can survive drops from up to 3m high due to its flexible and rigid 3D printed PLA frame.

In testing, the top speed achieved from the SentiBot is also about 20km/h and flight times are about 10 minutes with normal flying.

The vision and swarm algorithm results are still inconclusive as the algorithms are still being tuned at the time of the writing of this paper.

Conclusion

The SentiBot project set out to bring a solution to swarm researchers worldwide. We aimed to provide for all the hardware and software requirements for an optimized swarm system. The results seem to indicate that we have made considerable progress towards that goal but there is further testing and tweaking to be carried out in the following years.

The results also seem to indicate that the hardware side of swarm research is just as important as the software side as the hardware optimization solved many problems with current flying swarms. We hope that this report can act as a kick starter for hardware development in the field swarm research.