The original design problem which our system aimed to solve was that our project partner had a woodshop which would accumulate a large amount of dust and dirt on the floor. Cleaning this manually took time, and he felt the task could be better completed by a robotic vacuum, similar to a Roomba or other off the shelf model. The issue that our project partner had with these preexisting vacuums was that they were not designed to navigate an uneven floor such as he had in his shop, and they would become stuck on cords and small ledges.

Therefore he commissioned the creation of a robotic vacuum which would have a more rugged design, capable of climbing over whatever debris might be present while still vacuuming. He also required the robot be able to navigate to a charging base, which would also need to be built, where the robot would be able to recharge its batteries. This would allow the system to theoretically only need to stop when it needed its dust bin to be emptied, which would also have a sensor attached which would read the level of dirt present.

The entire project consisted of a total of ten students which would be split up among three teams at the beginning of the course. Ours was the embedded team, which would become responsible for driving cleaning motors, capturing data from various sensors, and communicating that information with a high level processor which would itself be responsible for navigating the robot.

The initial phase of the project involved a large amount of research by each one of our team members. The knowledge we gained would be required as we moved towards design, as a mistake made in the design process could have cascading effects leading to the project being impossible to complete. Each one of our team members focused on a separate area of study, those being serial communication, sensor use and control, and motor driving systems. We then began the actual design process of producing schematics and preliminary code layout.

The next phase of design was designing and simulating circuits in parallel to writing code for testing with a stand in processor. This phase was especially important and was possible due to earlier design decisions where we chose to use a processor common on microcontrollers we already owned. This allowed us to test individual components and blocks of code before having to do final integration.

The last phase was the ordering of the final PCb and the integration of each of the team members' design blocks. PCBs were soldered and then tested with each member’s contributions individually and then together so a final system demonstration could be performed. Some difficulties arose in the programming process of the final system, but these issues were quickly resolved and the final demonstration was completed.

As a team we learned several valuable lessons about project management and the technical process. The most important lessons are the value of keeping each team member informed of what the others are doing and staying up to date with project deadlines. Following these two rules helps avoid the situation of being set behind or having incompatible designs.