

Robotic Mining Competition - Software Design

Liam Sapper (lsapper2020@my.fit.edu)

Faculty Advisor: Dr. Marius Silaghi (msilaghi@fit.edu)

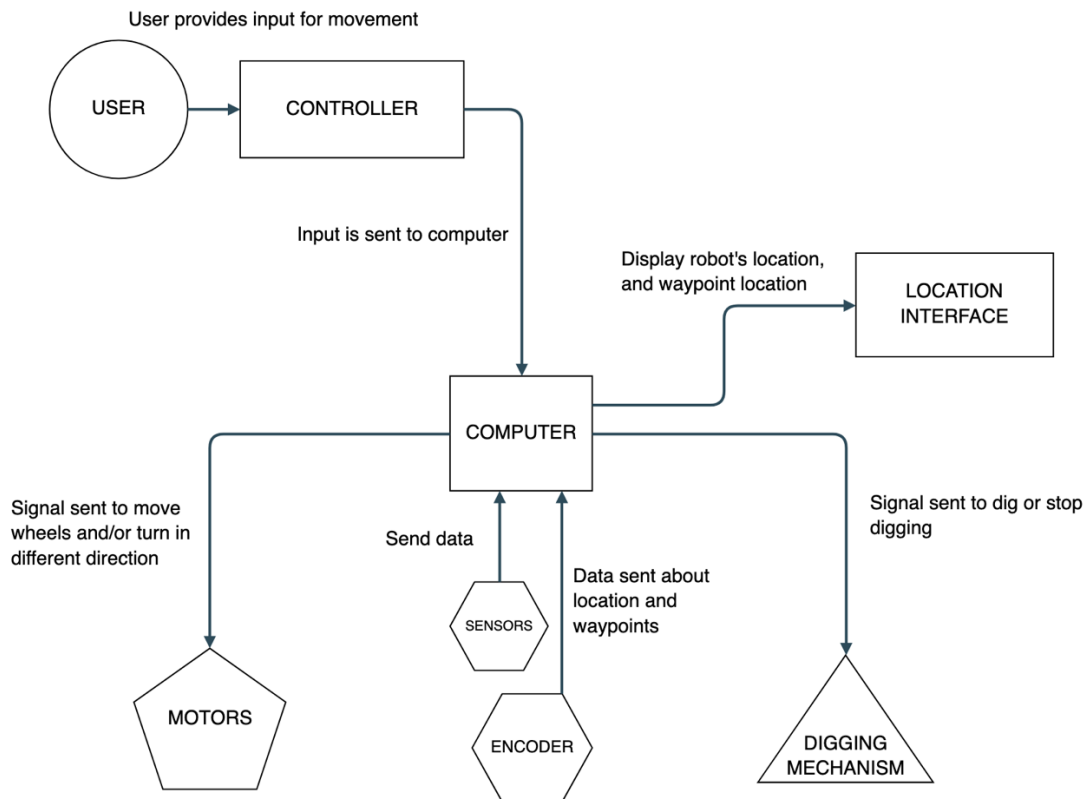
Client: Robotic Mining Competition team, NASA

Meeting Times: Wednesdays, 4:00pm - 5:00pm; Fridays, 3:00pm - 3:30pm

Programming language – Python

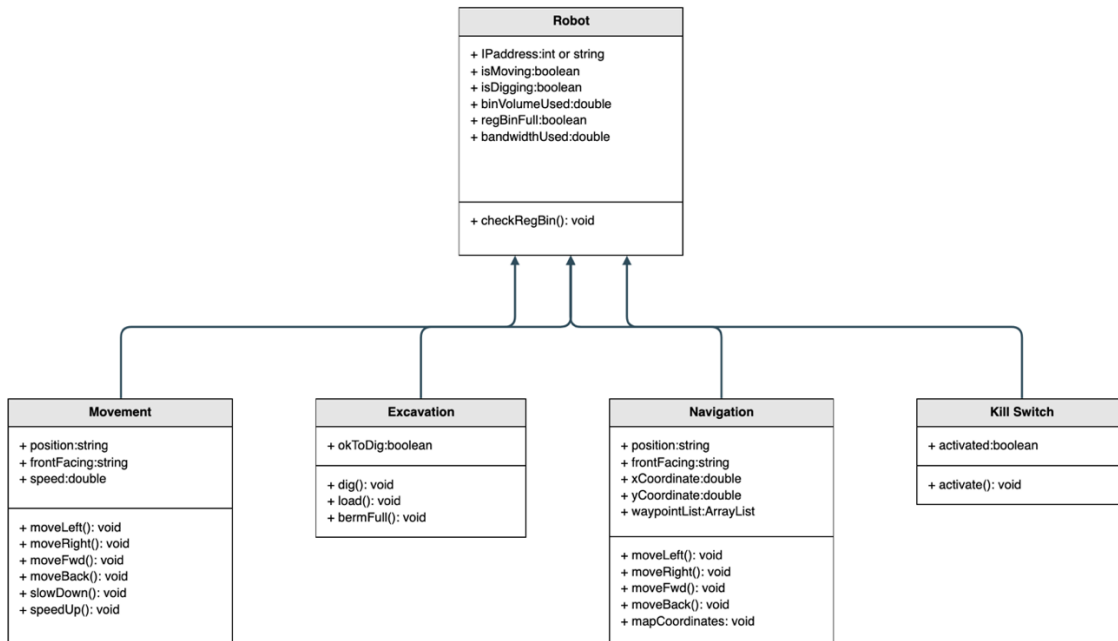
Interactions – Functions will be sent as signals to different components of the robot through a central computer. Those signals will be sent either automatically with data gathered by sensors and an encoder for navigation, or manually with input from a physical controller, with that input made by the user. Signals will be sent to the motors to go forward, backward, and/or turn. Signals will be sent to a digging mechanism to tell it to dig or stop digging. There will also be a separate graphical interface to show the location of the robot and waypoints.

RMC INTERACTION DIAGRAM



There should be a class for each functional requirement. Classes should have low coupling to remove dependency on each other. This will also reduce the number of problems faced if something breaks on the robot, and it shouldn't keep any of the other features from working if their respective parts are still functioning.

Robot Class Diagram Mockup



The only coupling that should be happening is between Movement and Navigation. However, between Movement and Navigation should only be one way; Navigation may be deterred if Movement is not functioning, but Movement should still be available if Navigation is not working. There will not necessarily be coupling between the Kill Switch code and everything else, because it would just be cutting the power to the robot. Everything else fails when there is no power.

The design for the user interface is still in progress. While the team knows it needs to display the location of the robot and waypoints, there are a few questions that still need to be answered; Will the point marking the robot's location be moving?; Will the robot's position be fixed while the waypoints are adjusted graphically based on their position relative to the robot?; How will the points and distances be kept track of? What units? The computer to be used is also undecided. The team has narrowed it down to an Arduino or Raspberry Pi.

The designs for all of these are likely to change over the course of the next year. Classes may be added, removed, or even merged, as well as certain functions. Software adjustments will be made to meet the demands of the rest of the RMC team.