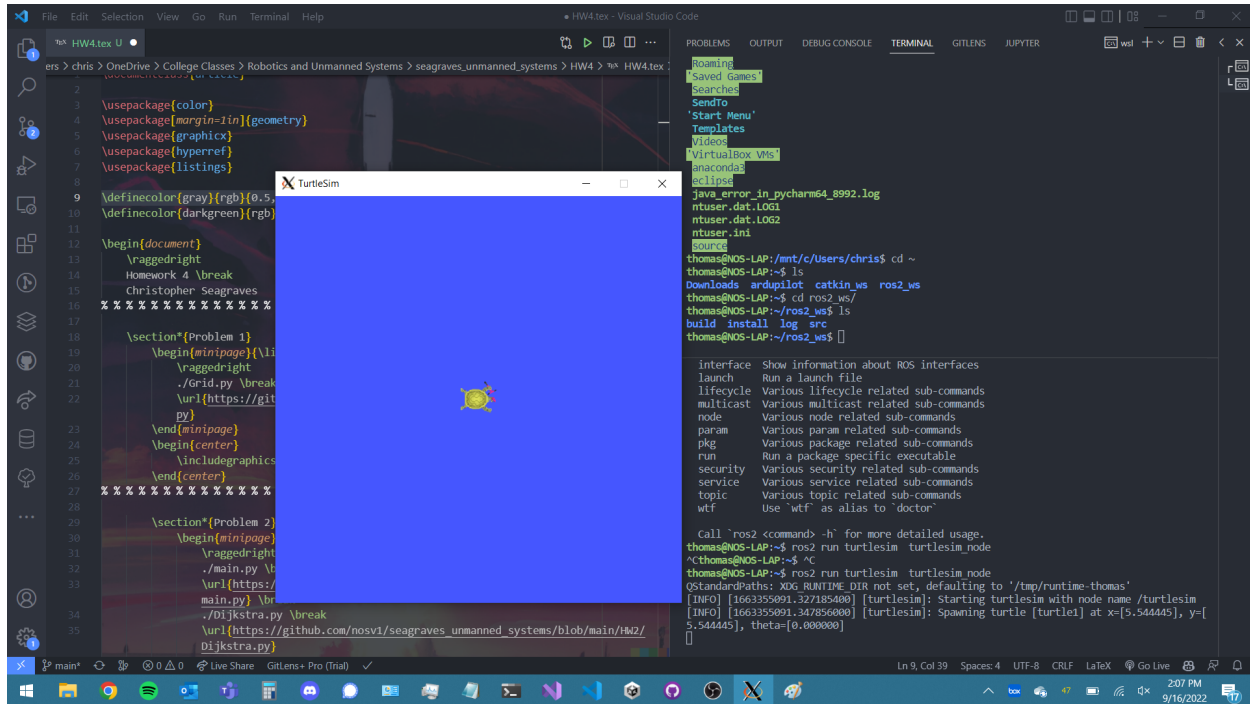


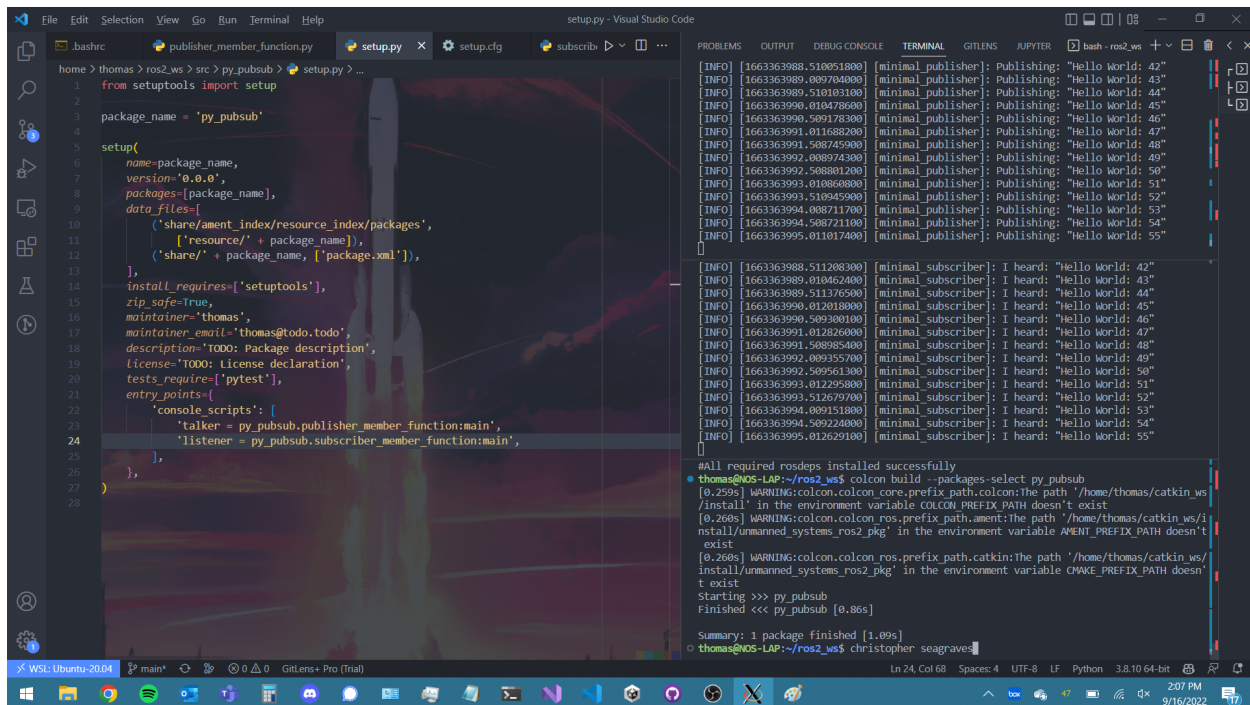
Homework 4 Christopher Seagraves

Problem 1



The screenshot shows a Visual Studio Code editor with a LaTeX document titled "HW4.tex". The document contains a preamble with package imports for color, margin, geometry, graphics, hyperref, and listings. It also includes a document structure with sections for Problem 1 and Problem 2. A TurtleSim window is open in the center, displaying a blue background with a yellow robot icon. The terminal window on the right shows the execution of ROS commands, including `ros2 run turtlesim turtlesim_node`, and the output of the `ros2 run` command, which shows the turtle's position and orientation.

Problem 2



The screenshot shows a Visual Studio Code editor with a Python script titled "setup.py". The script defines a ROS2 package named "py_pubsub" and sets up the package's metadata, including the name, version, author, and description. It also defines the package's dependencies and entry points. The terminal window on the right shows the execution of the `colcon build` command, which builds the package and its dependencies. The output of the build command shows the package's build progress and the final build status.

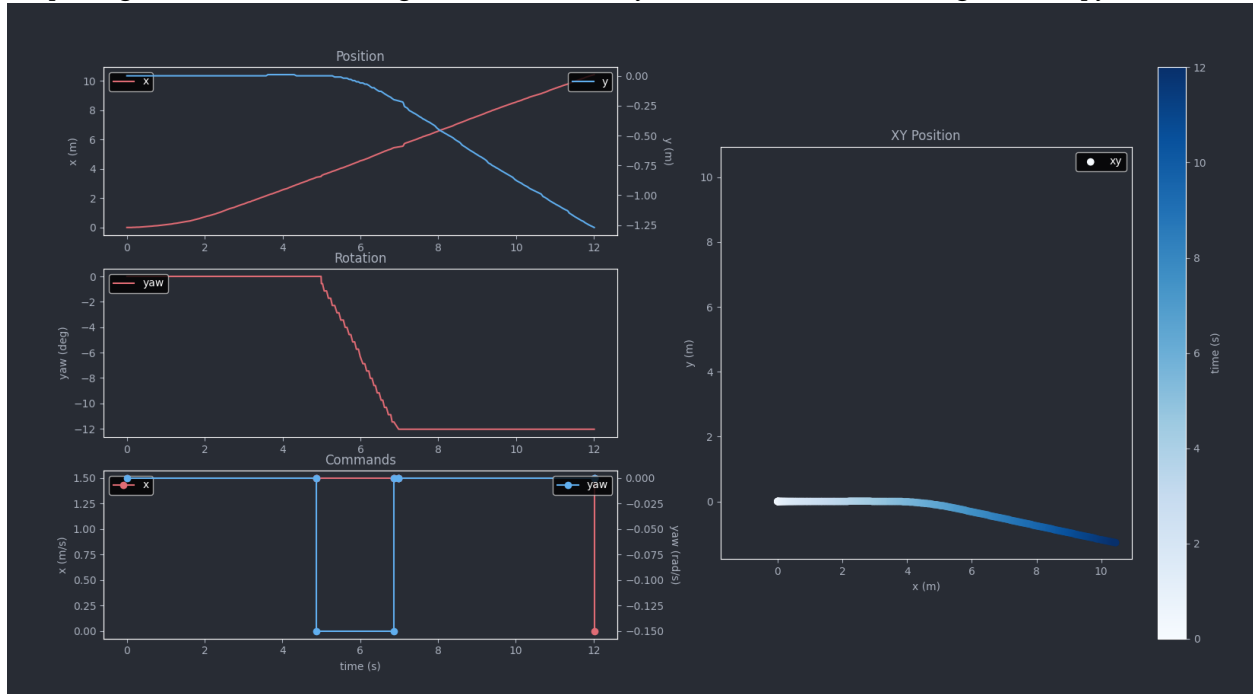
Problem 3

command_controller.py

https://github.com/nosv1/seagraves_unmanned_systems_pkg/blob/master/seagraves_unmanned_systems_pkg/command_controller.py

./LogPlotter.py

https://github.com/nosv1/seagraves_unmanned_systems/blob/main/HW4/LogPlotter.py

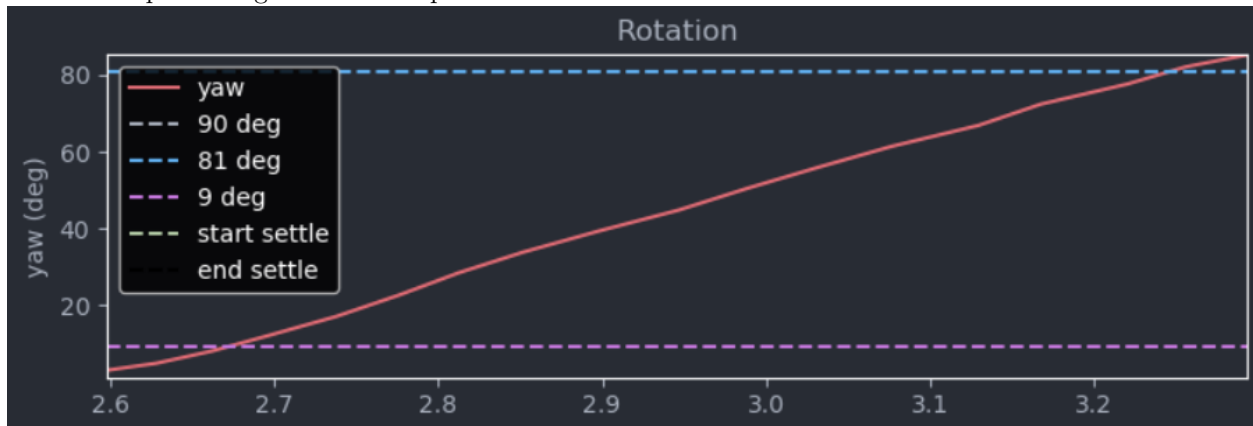


Problem 4

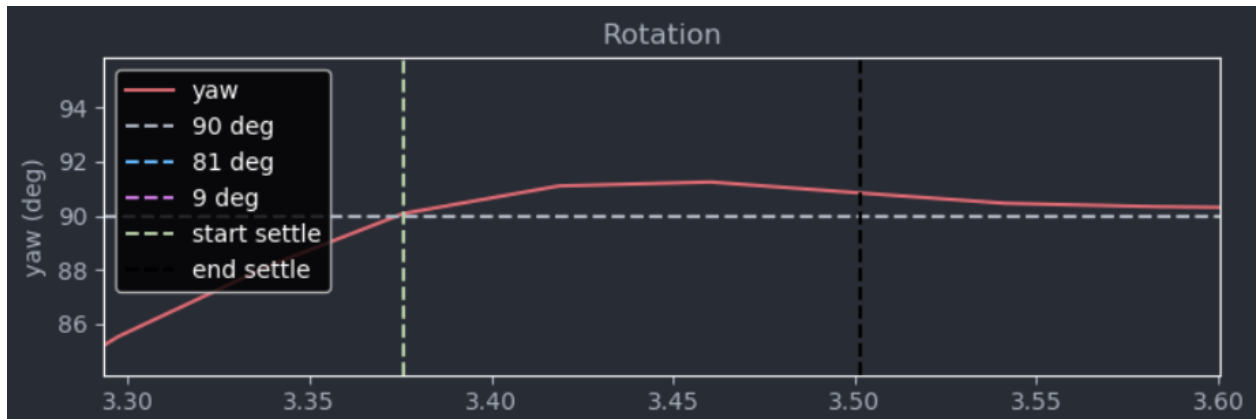
$K_p = 6.5$, $K_i = 0.0$, $K_d = 0.0$

Rise time = 0.6 seconds

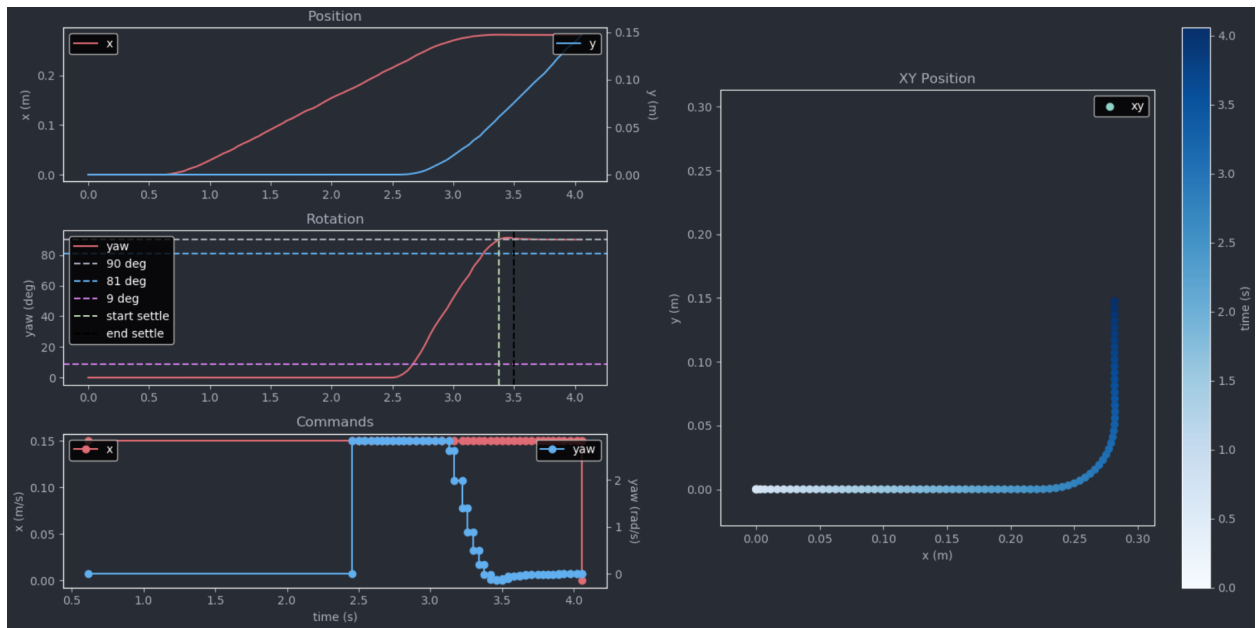
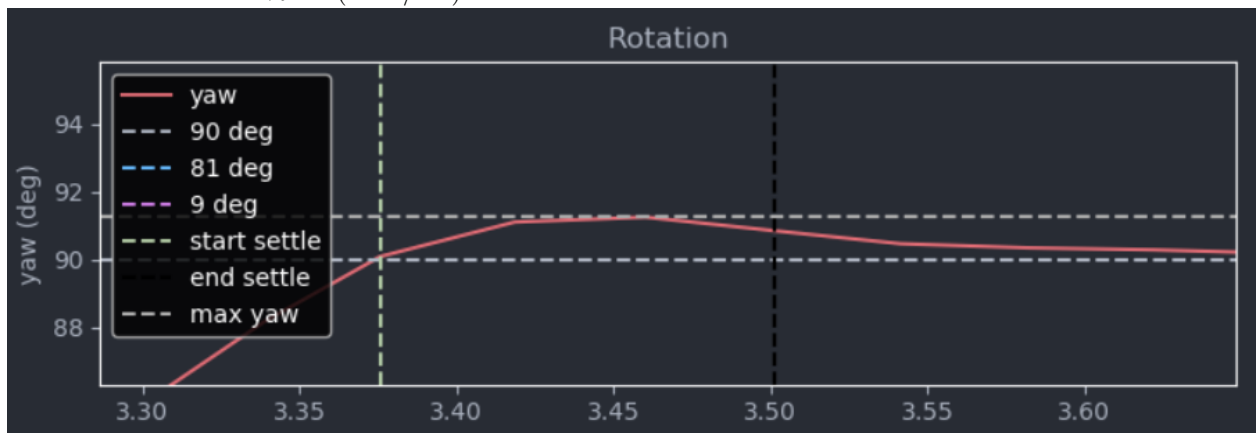
Given Burger can turn at 2.84 rad/s (162.4 deg/s), it can complete a 90 degree turn in 0.55 seconds - about .05 seconds per 10 degrees - so the optimal rise time would be about 0.44 seconds.



Settle time = 0.2 seconds



Percent overshoot = 1% $1 - (91.2 / 90)$



Problem 5

