Shock Dyno Control System

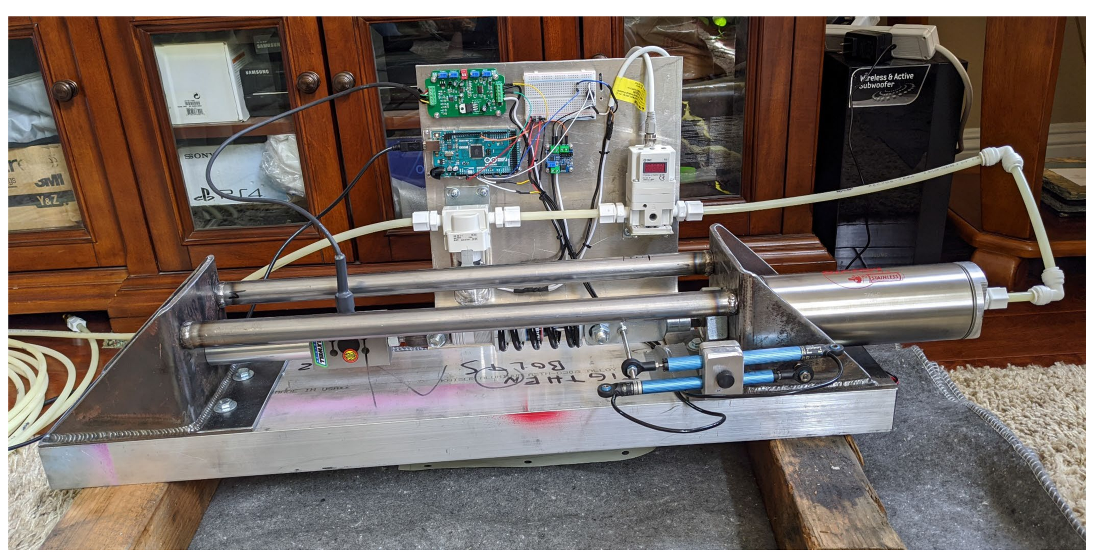
Scott Addams, Mechanical Engineering

Mentor: Dr. Nolan E Tsuchiya

Kellogg Honors College Capstone Project

Introduction/Purpose

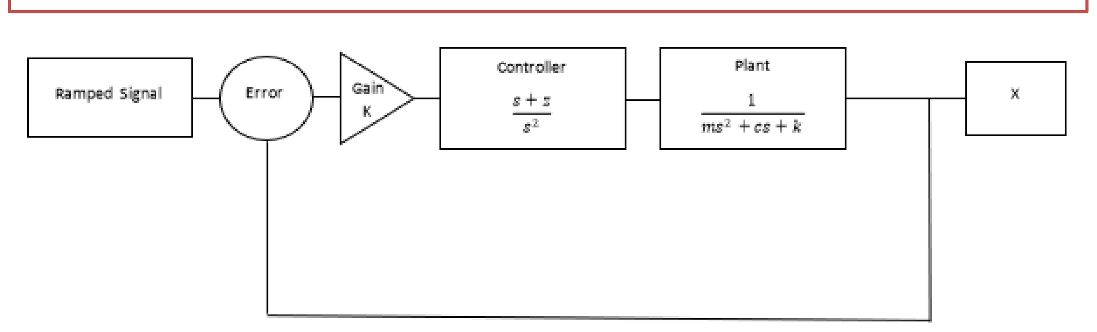
This project is part of a larger senior project in which the goal is to test various rear bike shock absorbers and collect their performance data. This data can then be used to help customers make well informed decisions on which shock absorber they would like to buy. The mechanical engineering objective of this project is to create a control structure that compresses each shock at a constant velocity to give them all fair and similar tests for comparison. The focus of this project is to design and code the control and data analyzation structure used to run the tests on the shocks. The control and data analyzation structure are to be coded within LabVIEW's graphical interface system.



Design and Simulation

Feedback Design

The control structure implements a feedback loop that follows a ramped signal. This ramped signal simulates the desired compression rate (velocity) that the user inputs into the control panel (shown later). The feedback loop takes this ramped signal and compares its simulated position with the actual position of the shock absorber, the difference between these is the error. This error is then inputted into the controller which sends a command signal to the plant in response in an attempt to reduce the error. The plant is the real world system that is being controlled and includes the shock absorber, pneumatic system, and electronics; the command signal is the signal tells the pneumatic system and electronics to apply a specific force to the shock. Following this, the distance is then measured again and the loop repeats.



Controller Design

The most challenging portion of the design was the controller design. In order to design the controller, one would need to understand the plant. Luckily the plant is a relatively straight forward system to understand, being that it is a Mass, Spring, Damper system. This means that the plant can be modeled like this:

$$\frac{1}{ms^2 + cs + k}$$

m = mass of the system
c = damping constant
K = spring constant

Once the plant is modeled, it is possible to model a controller that will properly control the system. Since it was desired to be able to track the ramped signal with no steady state error. For this to happen, the system would need to be Type 2 system meaning the controller needs to be in a form similar to $\frac{1}{s^2}$ (which is a double integral).

However, a controller designed in this way will lead to an instability in the system. This is found using the Routh-Hurwitz Stability Criterion [1]. In order to make the system more stable the numerator would need a zero (s + z), making the controller look more like this:

Controller

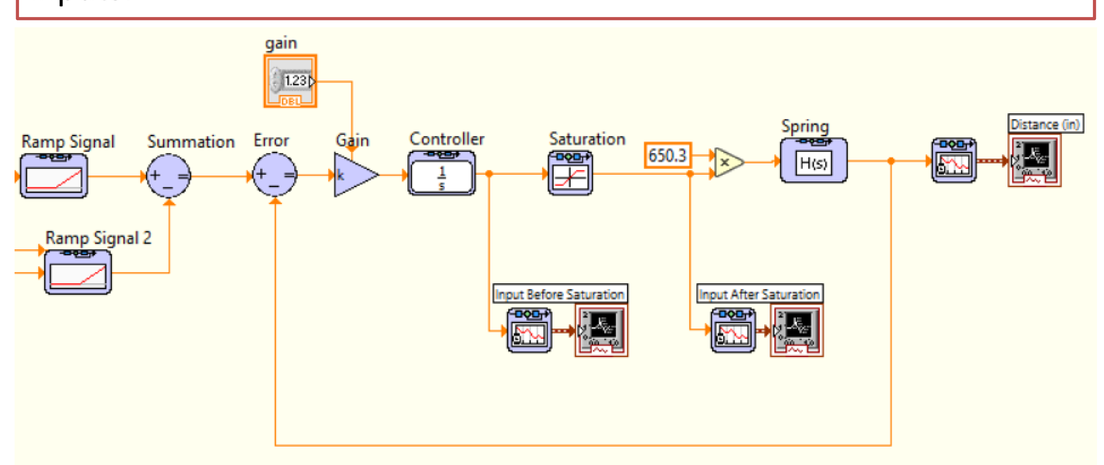
$$\frac{s+z}{s^2}$$

z = Zero (user controlled)

The zero and gain ranges of the controller are to be determined based on weight, spring constant, and damping constant range of the shock to be tested.

Simulation

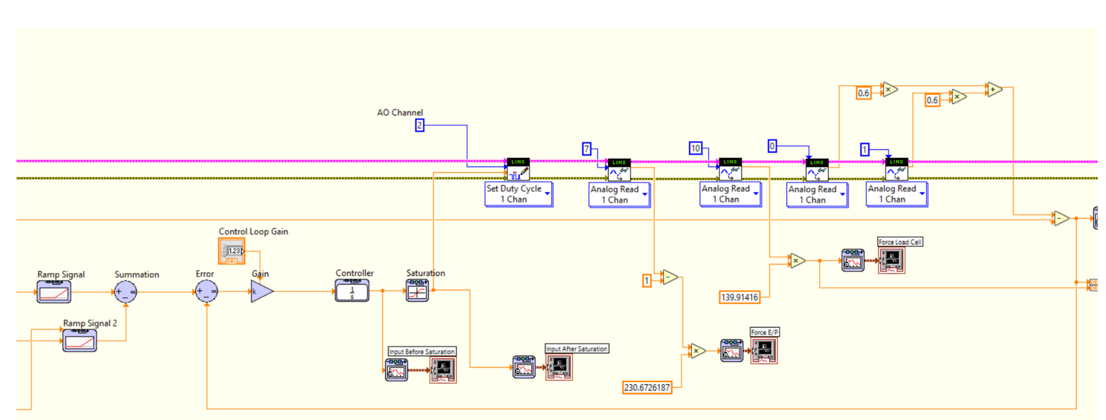
Since the feedback loop and controller now have a basic design, it is important to simulate these. This is to get a better sense of whether or not the system will go unstable when brought to the real world. To accomplish this National Instruments LabVIEW Control and Simulation module was used [2]. Using this module, it becomes easier to observe/understand how the system may react when given inputs.



Even though the simulations bring the system closer to the real world there is a high likelihood that it does not accurately model the real system. This will be expanded upon in the next section.

Real World Design and Testing

When designing the control loop for the real world system, the main component removed from the loop is the plant transfer function, since that was only a simulation of the true plant. Instead what follows the controller is the force command input into the plant and the output of the plant is the reading from the distance sensor.



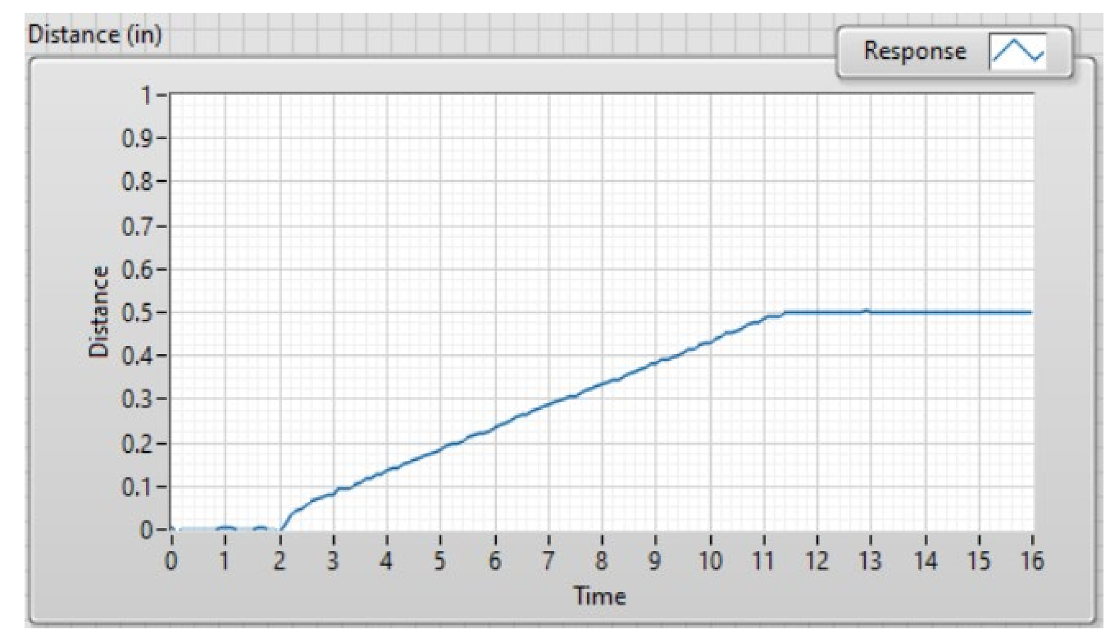
Once the system was set up to input the commanded force and receive the distance output (along with other force measurements for testing the spring) it was time to test the Shock Dyno. However, after repeated trials, it was observed that the system would either not track the ramped signal or become unstable as the controller gain is increased. The reason for this is that the controller that was designed as a double integral, so that if there's any error in the system it gets squared, meaning the correction response is much greater and much more likely to overshoot. To over come this the controller was simplified to a single integrator which in turn functioned as desired.

Controller

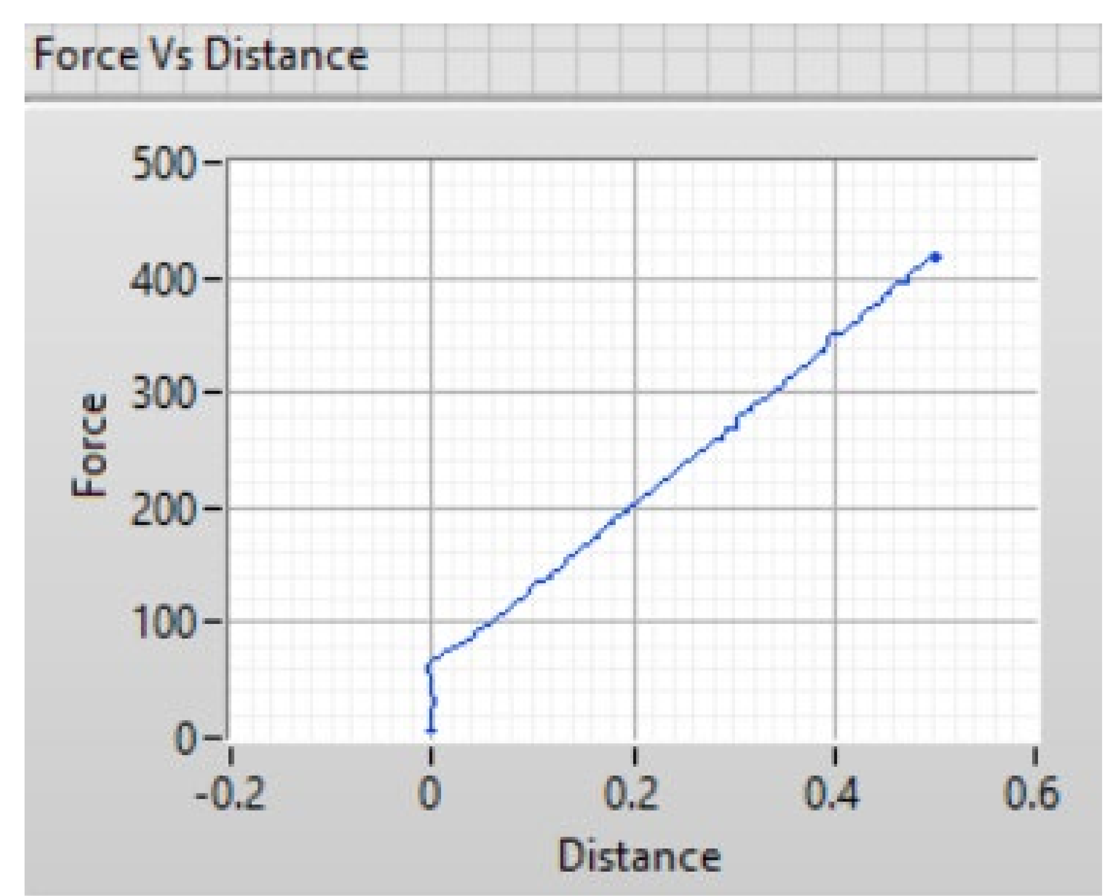
$$\frac{1}{s}$$

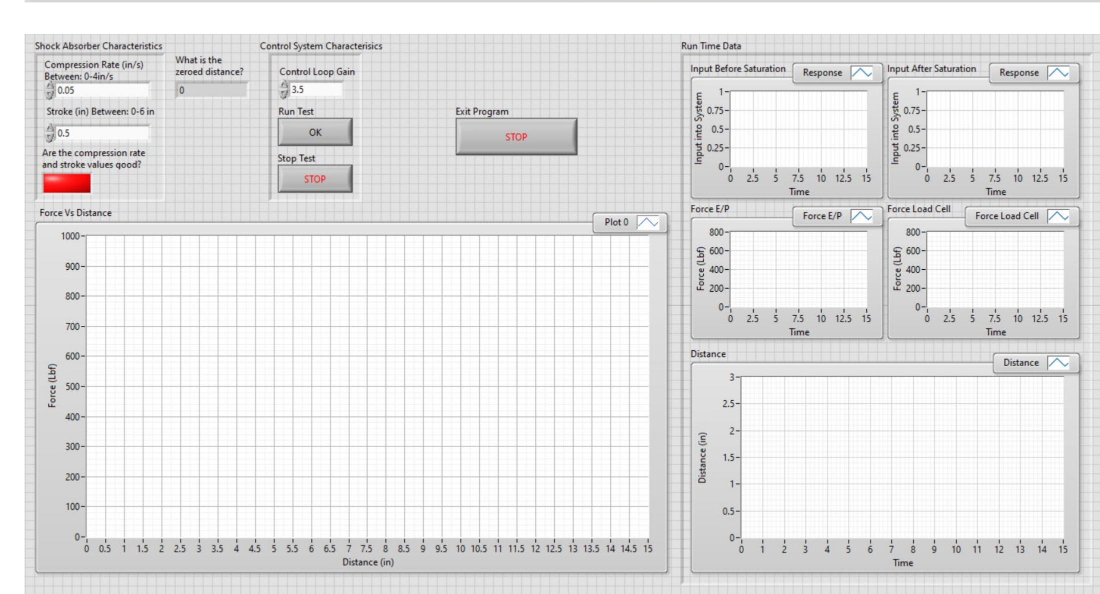
Results

After the controller was changed to a single integrator with a gain of 3.5, the system was quite easily able to follow a ramped signal (starting at the first second, **not zero,** and finishing at 11 seconds) of 0.05 inches every second taking 10 second to reach 0.5 inch of compression, shown below. This meant the system is tracking.



Since the system was able to track the ramped signal. The final test was to see if the Dyno could properly display the performance curve of a linear spring with a spring rate of 750-lbs/in. As seen in the graph below this spring has a preload of around 60 pounds and a slope of 750 lbs over one inch, confirming that the dyno is functioning as intended and desired.





Above is an image of the user interface when testing shocks.

References

[1] ECE 680 Modern Automatic Control "Routh's Stability Criterion" January 5, 2020 http://et.engr.iupui.edu/~skoskie/ECE680/Routh.pdf

[2] NIglobal "Teach Tough Concepts: Closed-Loop Control With NI LabVIEW and a DC Motor" YouTube video, 7:00, March 7, 2020, https://www.youtube.com/watch?v=7OgGejC0_3c