AE 4610
Georgia Institute of Technology
Lab Final: 1D Reaction Control System Rotator
Fall 2024
Group A05 – B

Name	Contribution
Brian Goldblatt	Controller Evaluation
Jonathan Bargsten	Controller Design, Simulation
Prashant Panta	Proof-reading, Formatting, Edited Implementation
Ethan Heyns	Implementation and System Dynamics Modeling
Mohammed S. Al-Mahrouqi	Introduction, Conclusion, Edited Implementation

Introduction	**10 points**
- Motivation for the topic [4 pts] - Brief background with references about the topic [4 pts] - Brief description of experimental set up [2 pts]	
System Dynamics Modeling	**10 points**
- Study of the system being controlled and justification of modeling [4 pts] - Mathematical model of the open loop system [1 pt] - Assumptions and approximations [1 pt] - System identification (include a brief description of the experiments conducted and data collected) [4 pts]	
Controller Design	**10 points**
- Controller design specifications [2.5 pts] - Type of controller used and a block diagram of the controller [2.5 pts] - Controller design steps and results [5 pts]	
Simulation	**15 points**
- Simulink diagrams or other simulation software used [5 pts] - Controller tuning, how changes affect the response [5 pts] - Show some results of your gain tuning iterations - Response plots, check if specifications are met [5 pts]	
Hardware Implementation and Controller Evaluation	**40 points**
- Table of components, specifications if necessary [5 pts] - Pictures of hardware set-up, wiring schematics, etc. [5 pts] - Design/Fabrication/Manufacturing and sensor/actuator testing and calibration [10 pts] [20 pts for either of the two below] - Final controller tuning and results (including response plots) and proposed future extensions. - If controller evaluation is unsuccessful, explain in detail how and why it failed. Include plots/figures to support explanation. Provide feasible solution (in detail) to overcome the problems and pending future work.	
Conclusion	**10 points**
- Summary of the experiments [4 pts] - Key findings from the results [6 pts] - **NO** new additional results/findings/calculations [**-1 pt for this**]	
References	**5 points**
- Proper reference citation within the report [3 pts] - Relevant reference [2 pts]	

Individual Contributions	
- List out each group member, and what contributions they made to the project. - Specifics required! Simply listing the section or task would not be accepted. [**-5 pts if not followed**] E.g.: Person A: Developed the equations of motion, performed sensor and actuator calibration, machined the base plate and other components, identified controller specifications, wrote the system ID section of the report. Person B: Designed the experimental setup on CAD, performed the system ID experiments to identify parameters X and Y, developed Simulink model and performed simulations, wrote the controller design section of the report. Person C: ….. Person D: …..	

Note:

- Write the report as if the reader has NO prior knowledge about the topic or control system.
- Extra materials such as code, derivation and images should be included in the Appendix.
- Report should look professionally written, i.e., no formatting error, no hand drawn plots, incomplete equations.
- Minimum of 5 pages, excluding cover page, plots, tables, appendix.
- Use AIAA manuscript template. **[-5 pts if not followed]**

1. Introduction

Reaction Control Systems (RCS) are broadly infamous with their utility to stabilize a body in motion by countering the forces imposed by external factors till reaching steady state. Contemporary applications involve cinematic tripod, reaction wheel, and gyroscopes. The recovery mechanism varies from utterly-passive to controller-operated. The experiment's motivation is to mimic an actual RCS application, similar to those embedded in launch vehicles and other space probes once in orbit, intended to properly orient the vehicle toward a desired trajectory. Practical application of RCS systems include the re-entry specific thrusters used for the Gemini III through the Gemini XII detailed by the Smithsonian website's collection objects [1]. These 25 pound thrusters stabilized the spacecraft attitude using cold, pressurized nitrogen gas. For the Gemini VIII, the RCS system was required for an emergency during the docking procedure, though "premature use of the RCS, however, required the mission to be terminated early" [2]. This emphasizes the importance of precise and accurate RCS controllers to improve safety and mission performance. Moreover, there are limited solutions to produce controllable propulsive forces in space due to scarce resources and compressed cold gas thrusters are deemed as one of the most practically reliable ones. Cervone [3] states that cold gas systems involve multiple thrusters oriented in different directions based on the target maneuvers.

The experiment explores a one Degree-of-Freedom (DoF) RCS with two thrusters to stabilize a rotating platform at a desired angle from the initial orientation after experiencing external forces. The system employs compressed air as a thrust force and values to induce and regulate rotational motion about the z-axis. The controller is responsible for determining the appropriate valve state, whether closed or open, and each valve induces either clockwise or counterclockwise rotation. Feedback is acquired by tracking the measured error via the resistance change of a potentiometer, placed at the center of the rotating axis. The system dynamics concentrates on deploying a Simulink PD controller to the one-axis rotating system, allowing to iterate with minimal number of variables, thus reducing the complexity of tuning the system. Lastly, a ± 5 degree tolerance is targeted while accounting for the valve control precision which may cause some deviation from the preferred orientation angle.

2. System Dynamics Modeling

The equations of motion used to describe the RCS can be broken down into two categories: the thrust equation for gas being expelled from a nozzle and the dynamics equations used to describe angular motion. First of all, equations 1-4 were used to find the thrust of the nozzles. Equation 1 solves for the choked mass flow rate of a compressible fluid using the coefficient of discharge (C_d), orifice area (A), the specific heat ratio (γ), density (ρ) which can be solved from ideal gas show in equation 2, and upstream pressure (P). Coefficient of discharge is geometry dependent and 0.8 was selected based on the length of the nozzle being more similar to the geometry of a short pipe.

$$\dot{m} = C_d A \sqrt{\gamma \rho P \frac{2}{\gamma + 1}^{\frac{\gamma+1}{\gamma-1}}} \tag{1}$$

$$\rho = \frac{P}{RT} \tag{2}$$

Since the upstream pressure is $100\ psi$ which is more than double the ambient pressure of 14.7 psi, the nozzle is choked. With a straight orifice that is choked, the exit is assumed to be at mach 1. This means the exit velocity can be calculated by solving for the speed of sound as shown in equation 3. Then using both mass flow and velocity thrust can be calculated as shown in equation 4.

$$u = \sqrt{\gamma RT} \tag{3}$$

$$T = \dot{m}u \tag{4}$$

The thrust can be varied by changing the exit area of the nozzle or the upstream pressure of the gas. A $0.06''$ diameter nozzle was selected since when doing basic hand calculations with $100\ psi$ of pressure the resulting force was $0.82\ N$. Taking this force and solving for the angular position resulted in roughly $200°$ after 1 second. This may seem fast but this does not account for friction and, the air pressure can be easily varied using a pressure regulator which means the system's responsiveness can be tuned once it is tested.

The dynamics of the system are modeled using equations 5-7. The torque (τ) is solved using the thrust (T) and the radius of where the nozzles are located (r). The moment of inertia (I) of the rotator is solved using that of a disk. Lastly, the angular acceleration (α) is calculated using the torque and moment of inertia.

$$\tau = Tr \tag{5}$$

$$I = \frac{1}{2}mr^2 \tag{6}$$

$$\alpha = \frac{\tau}{I} \tag{7}$$

While there are other factors that impact the dynamics of the system and some unknowns (for example: exact mass, friction, actual thrust, etc.) impacting these equations, these equations still serve as a valid model of the system and the unknowns can be better determined with test data. Through experimentation data will be collected of the angular position to understand the system response and the signals from the controller to command the valves. With this data the functionality of the system can be understood and it the controller can be tuned. The experiment to collect this data involves starting the rotator at some angular position and telling it to move to another position. Then observing the response of the valves in order to reach that position while optimizing the controller for overshoot, error margin, rise time, and settling time through multiple experiment iterations.

3. Controller Design

Designing the controller involved developing signals to open and close solenoid valves based upon the error and controller gains. Using binary states, a value of one represents the open state and a value of zero represents the closed state. Furthermore, one valve controls counterclockwise (positive) rotation, while the other controls clockwise (negative) rotation. Depending on the sign of the initial error at each time step, positive error will send a one to the clockwise actuator while negative error will send a one to the counterclockwise actuator. This is then modified for a proportional-differential (PD) controller by subtracting the actual position scaled by the proportional gain and subtracting the derivative of the actual position scaled by the derivative gain.

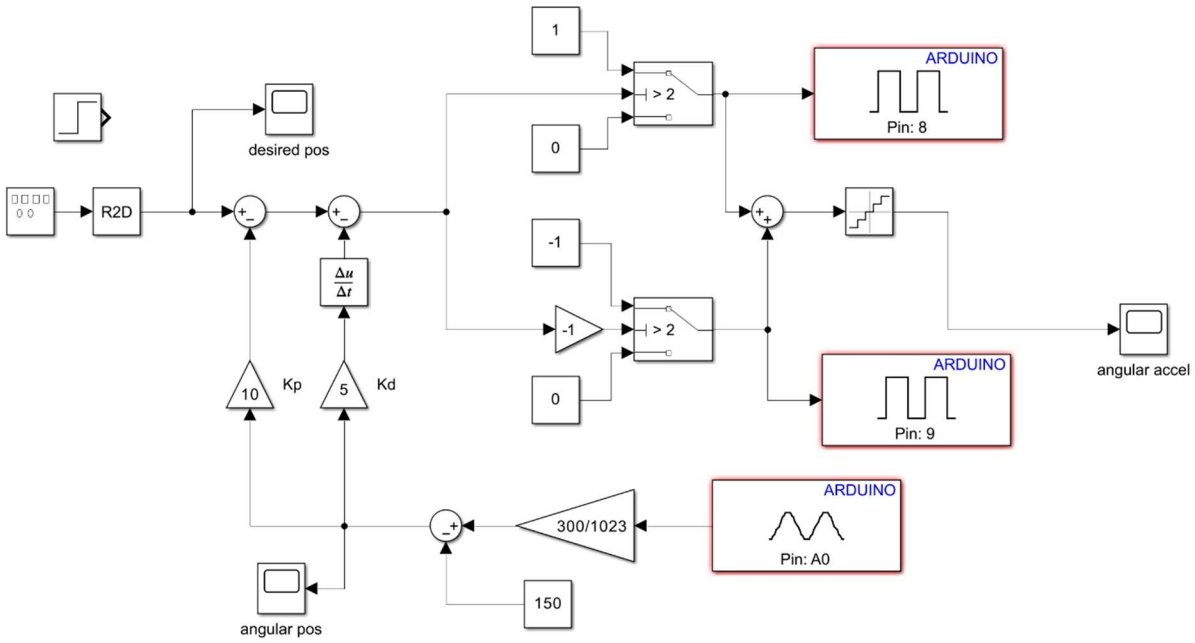

Figure 1: PD controller block diagram with arbitrary Kp and Kd values

Points of note not discernible in Figure 1 include the input which was set to a step input for the experiment itself, the scaling factor of 300/1023 to convert the potentiometer output to degrees, and the translation constant of 150 to center the 300-degree range of the potentiometer around zero. Additionally, pin 8 and pin 9 correspond to the solenoid valves.

For the controller specifications, we determined that it must have a rise time of under $5\ s$, a settling time of under $3\ s$, a steady state error band of $\pm\ 5°$, and a maximum overshoot of 20%. These were at least the starting points for the experiment as multiple precision bands from $\pm\ 2°$ to $\pm\ 15°$ were tested.

Due to the nature of the system (variable air pressure, variable weight, friction, etc.), the Kp and Kd values could only be estimated and tested manually, but $Kp\ =\ 10$ and $Kd\ =\ 0.5$ was used as a starting point.

4. Simulation

To develop the simulation, a Simulink model was used to calculate the updated angular positions by converting on/off signals to the compressed air valves into angular acceleration. Additionally, simulated friction and efficiency loss were considered by scaling the calculated angular acceleration down by a percentage of the angular velocity. These parameters of efficiency and angular acceleration scalar had to be approximated from trial and error to match the experimental data later. Figure 2 below shows the template used for the simulations.

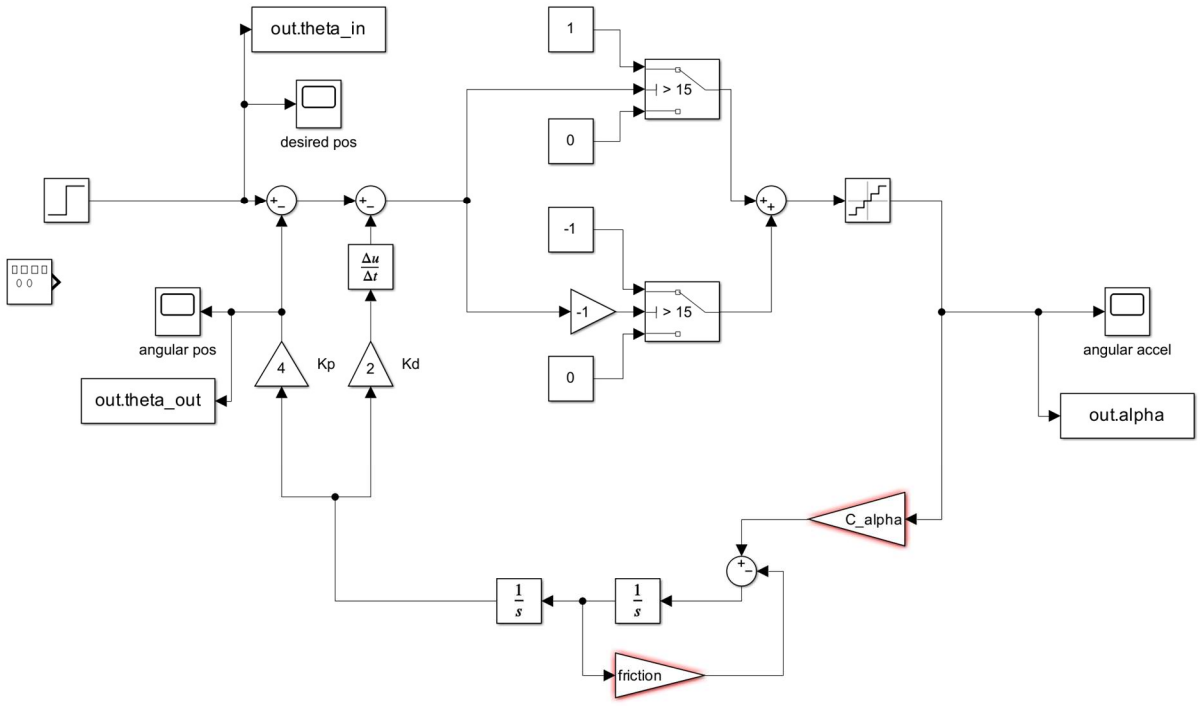

Figure 2. Simulink simulation block diagram

Initial simulations yielded promising results, but due to the uncertainty of friction and the alpha scalar, the simulation results were more effective at providing verification of the functionality of the controller. However, the results are still worth comparing and seeing how varying the gain affected the response. Testing some values for friction and C_alpha, the most reasonable results came from 30% friction and a C_alpha of 40 with results below. The following three simulated results use a $\pm 5°$ margin as well.

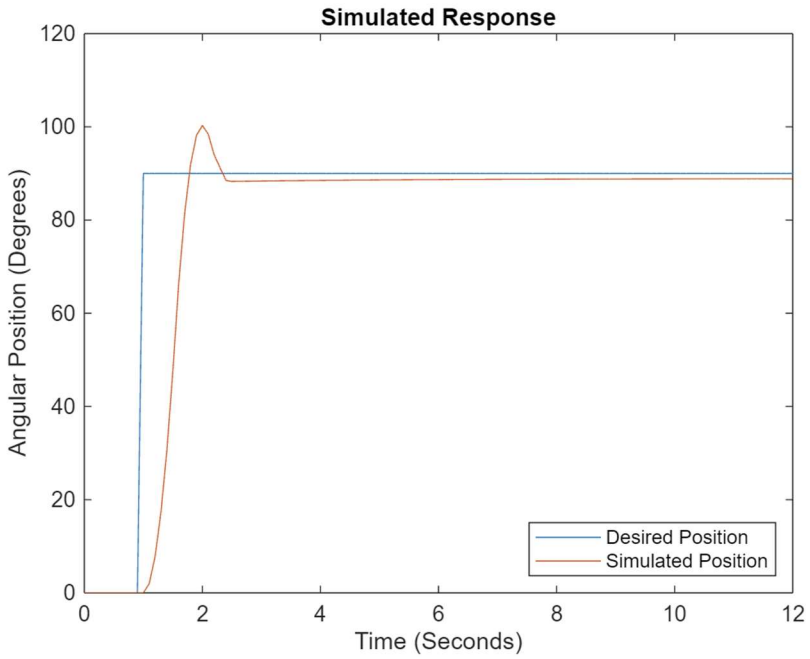

Figure 3. Simulated results for $Kp = 10, Kd = 1.7$.

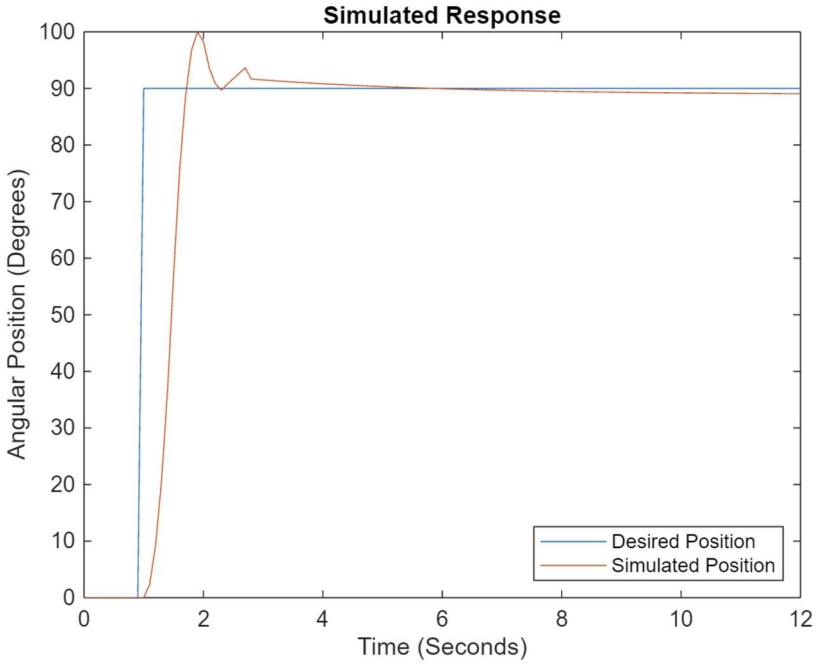

Figure 4. Simulated results for $Kp = 12,\ Kd = 2.1$.

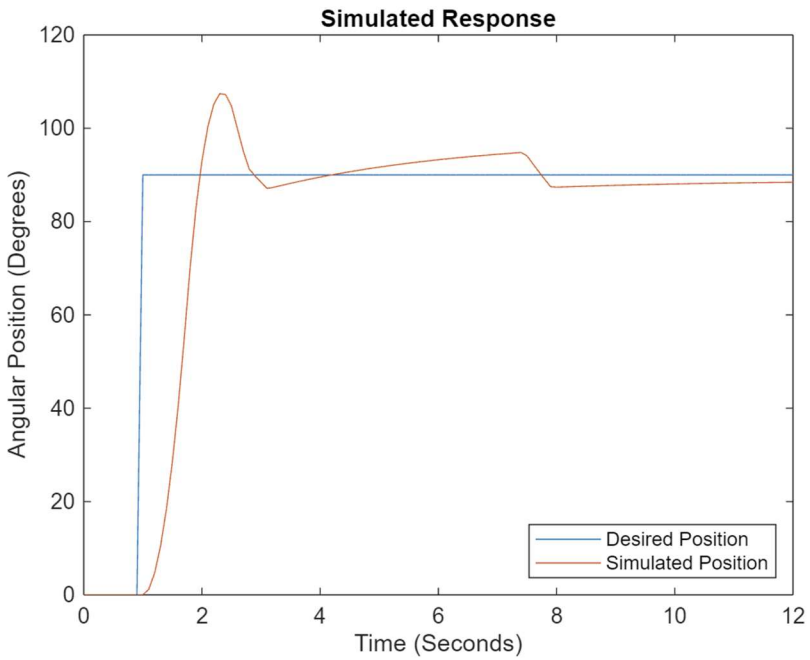

Figure 5. Simulated results for $Kp = 6,\ Kd = 1.2$.

Ultimately the simulations behave as expected when utilizing educated approximations and worked even better under idealized conditions. However, error, particularly in C_alpha, and interesting interactions from having a range of degrees where there is no effective error, resulted in data unintuitive to the typical trends from changing Kp and Kd. While the effects on overshoot, rise time, and settling time made sense, there are jumps from the influence of Kd. When the angular position enters the region of zero error, the derivative may not necessarily be zero leading to "sliding" motion from the derivative which will only be corrected once the position exits the region of zero error for the simulated valve to open again. This is most noticeable in figure 5. Expanding further on these simulations, the table below outlines their performance in relation to the design specifications. The first two meet all specifications with the third failing to meet three of four but acts as an example for the "sliding" and more significant changes in gain.

Table I. Simulations performance

	$Kp = 10$ $Kd = 1.7$	$Kp = 12$ $Kd = 2.1$	$Kp = 6$ $Kd = 1.2$	Specifications
Rise Time	$0.4794\ s$	$0.4433\ s$	$0.6019\ s$	$< 3\ s$
Settling Time	$1.3212\ s$	$2.9654\ s$	$6.7296\ s$	$< 5\ s$
Error Band	$\pm 5°$	$\pm 5°$	$\pm 5°$	$\pm 5°$
Max Overshoot	12.8691%	12.2043%	21.4474%	$< 20\%$

Attempting to simulate data obtained from the actual experiment in Figure 11, a C_alpha of 80 and a friction of 85% obtained similar results. When testing these values, relatively small changes had drastic effects on the results, so if more accurate values could be found, the integrity

of the overall simulation would be drastically improved. However, the same gain values of $Kp = 10,\ Kd = 0.5$ were used, providing greater validity to the below results.

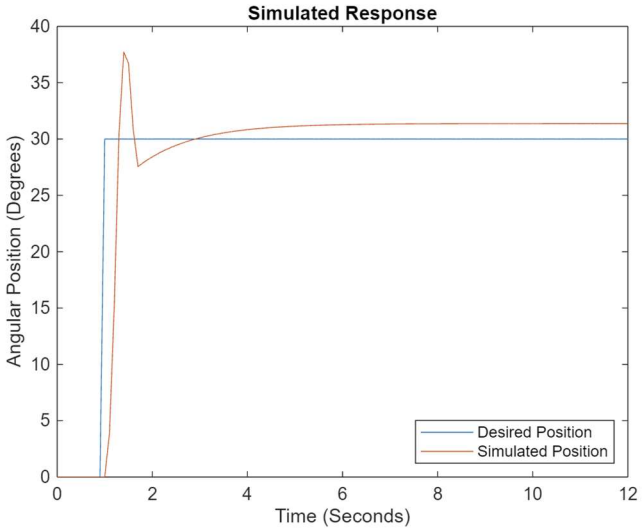

Figure 6. Simulation mimicking iteration 1 in controller implementation, $Kp = 10, Kd = 5$.

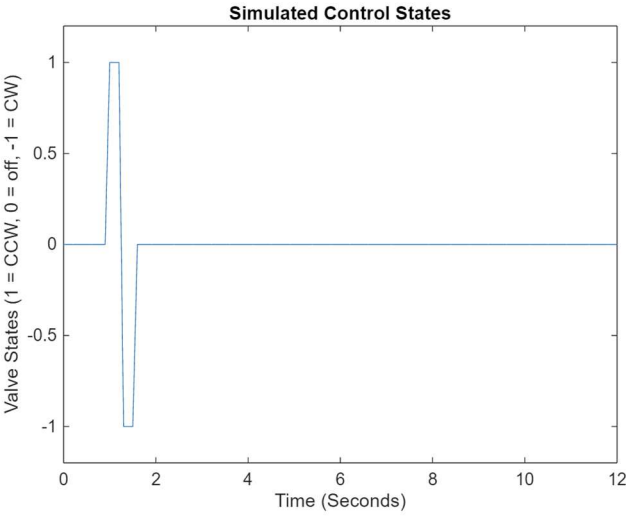

Figure 7. Simulated inputs sent to solenoids representing angular acceleration inputs, for the same simulation represented by figure 6.

It is interesting to note the similarities of these simulated results to the later experimental results as the simulated control states are effectively the same, discounting an extraneous initial input in the experimental data. It also has similar values for rise time, settling time, and overshoot. The simulated values are a rise time of $0.2048\ s$, a settling time of $2.8240\ s$, and an overshoot of 20.2332%. Of these values, the overshoot is only slightly above specification and was a decent starting point which was iterated from.

5. Hardware Implementation and Controller Evaluation

The Reaction Control System hardware can be broken down into 3 key subsystems: the rotator, the pneumatics, and the electronics. The rotator is constructed from two pieces of wood and a ball bearing rotator placed in-between. The wooden top piece was laser cut to allow for the desired placement of components, and its dimensions are illustrated in Figure 8. The pneumatics system is composed of two solenoid valves connected via plastic tubing to an air compressor with a large tank volume set by a pressure regulator. Lastly, the electronics utilizes an Arduino which commands two relays with $5V$ which pilot the two solenoid valves with a $12V$ power supply. The Arduino also reads the signal from a potentiometer which is mounted to the center of the rotator and secured to both pieces of wood with adhesive. The potentiometer was calibrated by going from the minimum and maximum rotations and reading their values in Simulink, and then those values were scaled to correspond to the correct angular position of the rotator when at the minimum and maximum positions. The potentiometer has $0° \leftrightarrow 300\ °$ of rotation and read $0 \leftrightarrow 1023$ on Simulink prior to calibration. Once calibrated, the angular position was verified by rotating the potentiometer from zero and using a protractor to confirm the angle.

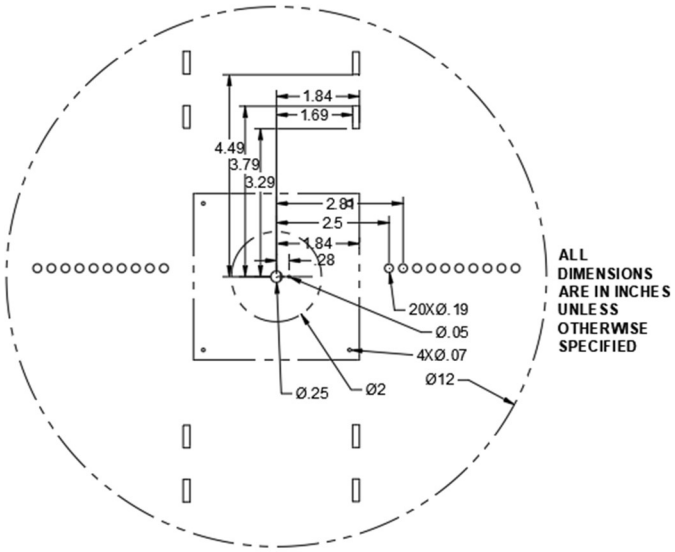

Figure 8. Dimensions of Wooden Top Piece

Table II. Component specification list

Component	Specifications
Solenoid Valves	$12V$, Normally closed, ¼" NPT connections
Plastic Tubing	¼" OD Tubing
Fittings	¼" Push to Connect to ¼" NPT
Nozzles	0.060" diameter orifice
Rotator Bearing	$3.75" \; x \; 3.75"$
Arduino Uno Rev3 SMD	$5V, 3.3V$, 14 Digital IO Pins ($20 \; mA$), 6 Analog Input Pins
Potentiometer	$10k \; Ohm, 300°$
Relays	$30V, 10A, DC$
Power Supply	$12V \; DC, 5A$
Laser Cut Wood Mounting Plate	$1 \; kg, 12 \; in \; diameter, ¼" \; thick$

Figure 9 shows the initial concept for the wiring and plumbing setup on the rotator. We ended up switching to a potentiometer instead of an IMU for simplicity. The wiring diagram does not show all the wires, particularly those going to the Arduino, but it gives a general concept of how the Arduino is used to pilot relays which control the valves with a higher voltage supply.

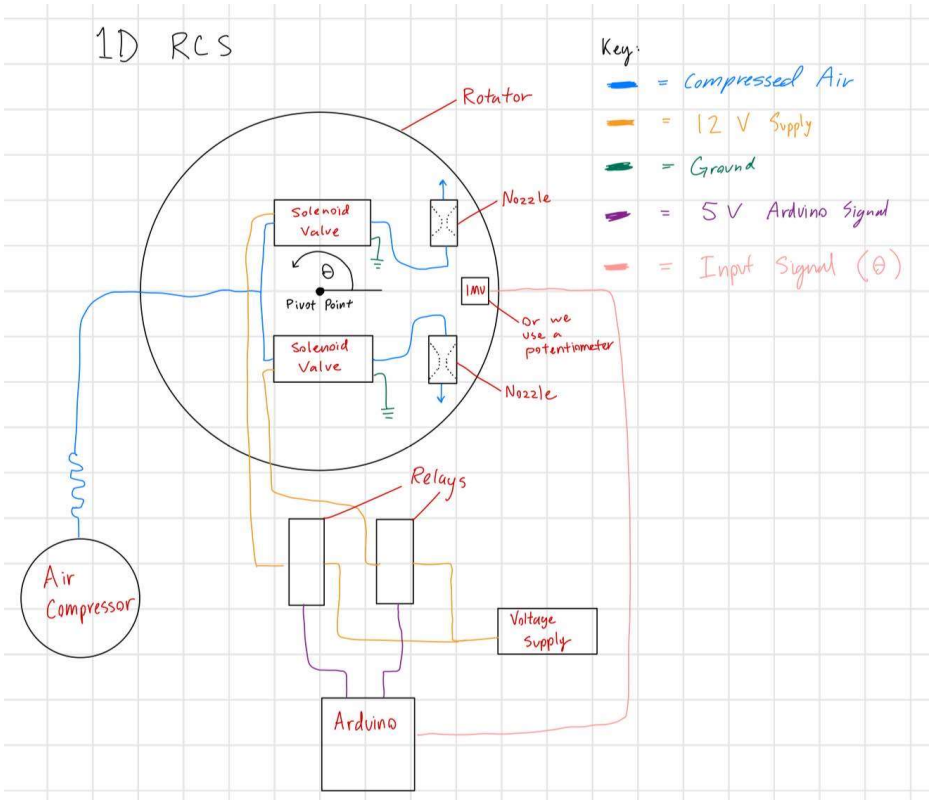

Figure 9. Basic wiring diagram of RCS setup ideation

The following figure shows the final construction of the system. There were a few issues with the electronics and last minute changes made during troubleshooting so the wiring is somewhat

chaotic, but functionally the system works. The main issue with the final assembly was the high friction at certain angular positions due to the way the top piece of wood was connected to the rotator and to the potentiometer. The friction overpowered the thrusters at some places even at 100 psi air pressure. The potentiometer may not have been perfectly centered which caused high friction. At some angles the friction was tolerable, and the thrusters had complete control over the angular position, and this is where data was collected.

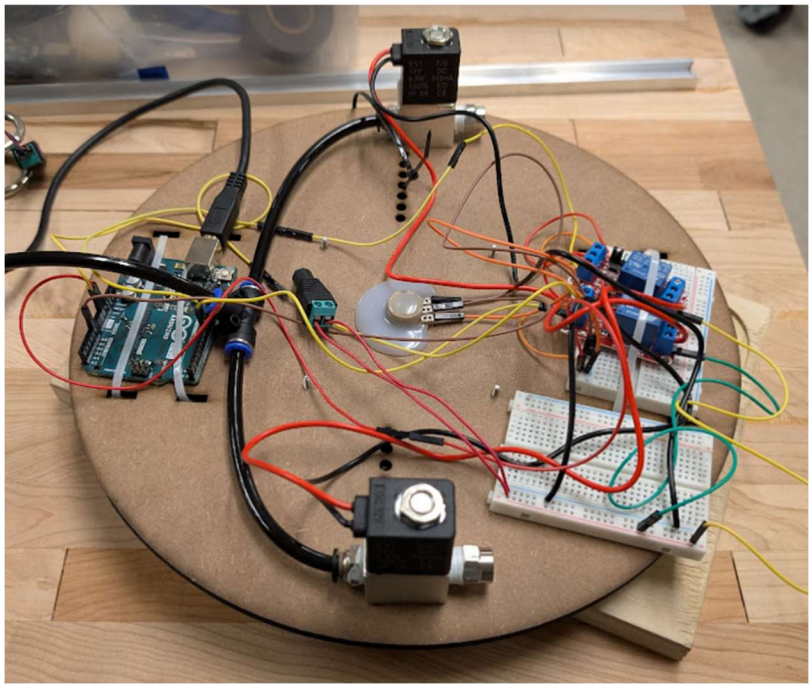

Figure 10. Fully assembled RCS rotator

Onto the results of the carried-out experiment, there were numerous iterations done with this setup and small tweaks that were made in order to come to the best results of the system. We have recorded three examples showing the results of the system at different points in the refining and gain tuning process, which will be explored below.

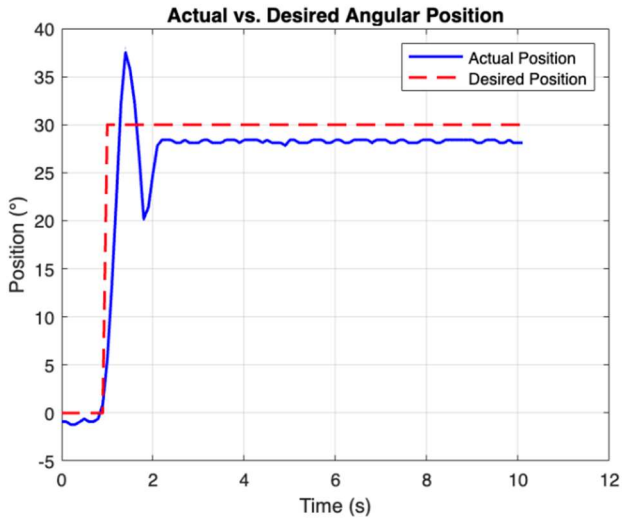

Figure 11. Iteration 1 moving from $0° \; to \; 30°, \; Kd = 0.5$.

For the first iteration in the process of tuning our Simulink model to better fit reality and deal with unideal things like voltage drops, friction, and inaccuracies in measurement equipment, it was attempted to bring the angular position from $0° \; to \; 30°$. This was done experimentally with a $Kd = 0.5$. There is a large amount of overshoot before settling with a 3° range of 30°. This is likely in part due to the methodology of controlling when to stop and start each valve still being fine-tuned in Simulink, and friction may be causing the disk to get slightly stuck in certain angles of the movement. The active states can be seen below. It should be noted that for the first second of the displayed data, the -1 or clockwise valve is turned on, but there is no compressed air connected to the system so it does not move anywhere. Once the air is connected and the tray starts rotating, the other (1) valve turns on, until there is some overshoot seen, in which case it turns back on the -1 valve until the desired position within 3° of error is reached.

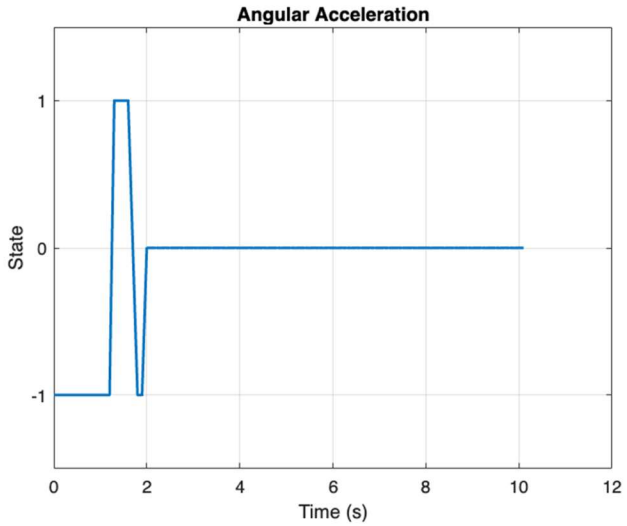

Figure 12. Iteration 1 Active States.

In the second iteration seen below, with further testing, the margin of error was opened up to 15° in order to home in on the proper Simulink model. Again, looking at the active state view

below, the first 3 seconds where -1 is activated is while the model sees it is at $50°$, but has no air connected to it so it is unable to move. In this case, once the air is connected, the -1 valve moves until it reaches within $15°$ of $0°$ and then shuts off, with no correction from the other valve needed. This was likely due to hitting some rough patch on the rotating part of the tray, and high friction being experienced. In some cases like this one, the model did not deem it necessary to activate any valves other than the initial movement to reach the desired state.

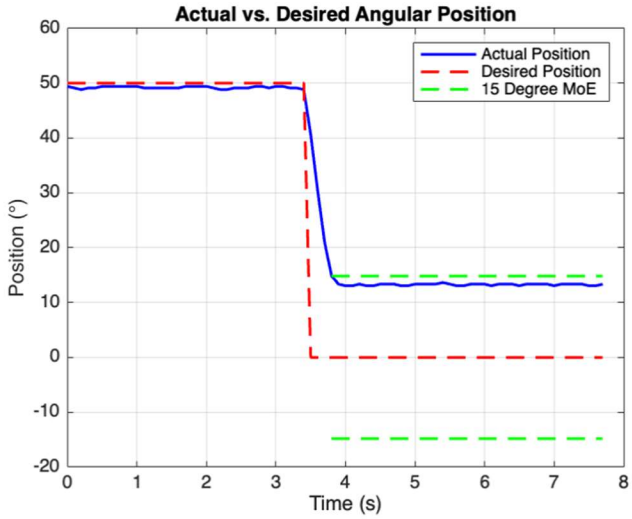

Figure 13. Iteration 2 moving from $50° \rightarrow 0°$ with $15°$ margin of error. $Kd = 0.5$.

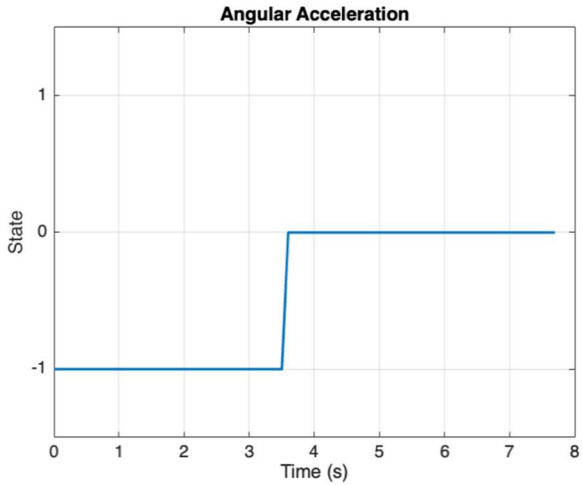

Figure 14. Iteration 2 active states.

For Iteration 3, we targeted within ± 5 degrees from zero. The rotator starts at $50°$ because there were issues with friction from $50°$ to $90°$; however, the start position should not impact the effectiveness of the controller. As shown in the acceleration figure, the RCS starts preemptively slowing down before reaching the desired position due to the derivative component of the PD controller. The derivative gain was set such that $Kd = 0.2$ which resulted in no overshoot. After reaching the target position the rotator was manually moved by hand to observe the response. As shown at the end of the plot, the position is moved $10° - 20°$ away from center repeatedly and

each time it returns to within 5° from center showing that the controller is operating as expected. (This same behavior can be seen in the video submitted with this report).

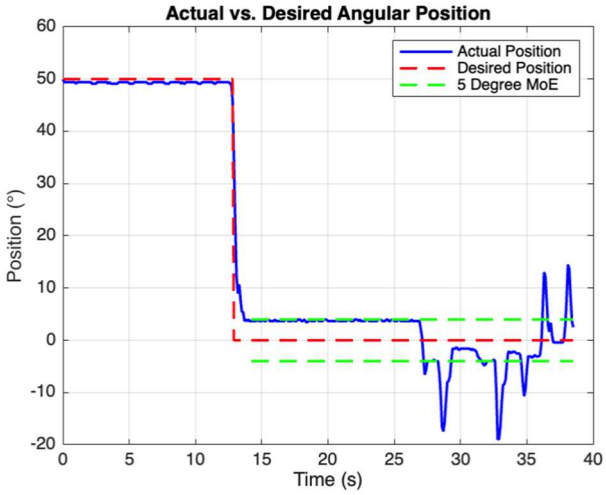

Figure 15. Iteration 3 moving from $50° \rightarrow 0°$. $Kd = 0.2$.

The angular acceleration plot of Iteration 3 shows which valve is open, and thus which direction thrust and angular acceleration is being produced. At 13 seconds the compressed air valve is open supplying pressure to both solenoid valves which allows the rotator to start moving since the valve corresponding to -1 angular acceleration is open. Once approaching the desired position the valves swap and the +1 valve opens to slow down the rotator. The valve state quickly goes back and forth, and then both valves close since the angular position is within 5° from the desired position. The later spikes in angular acceleration are from when the rotator is manually moved by hand and they show how the states change to re-enter the desired position range.

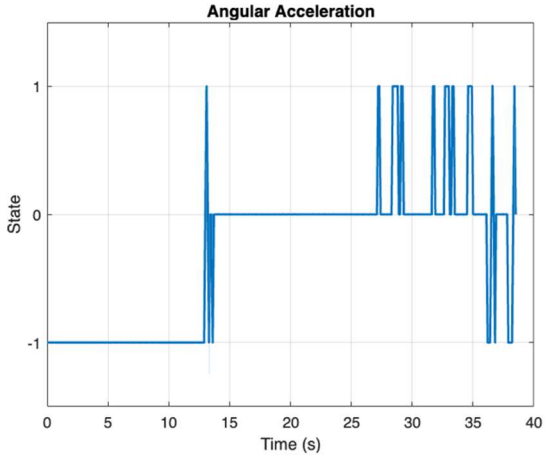

Figure 16. Iteration 3 Active States.

From these results and comparing back to simulation results, it can be deemed that the experiment was a success. Some feasible future extensions of this experiment might include upgrading our hardware to limit the inconveniences of large friction spikes in rotation, wires

getting in the way, voltage drops experienced, and other faulty parts. Once the fundamentals are covered and upgraded, it could be experimented with adding a second DoF with a whole different system added onto the initial experiment.

6. Conclusion

The experiment is intended to apply control theory fundamentals by theorizing, fabricating, modeling, and testing a simple one-DoF RCS. The angular motion due to nozzle thrust correlates the control inputs to the system response in order to orient the rotator to the desired angle. The valve state, whether on or off, merely governs the rotational motion direction, clockwise or counterclockwise, and the duration of compressed air release. Error margin, K_d , and K_p are fed to the PD controller to tune the response. The hardware has caused unforeseen issues while fine-tuning the controller because of its inherent limitations such as friction, voltage spikes, and Simulink bugs. Despite some challenges and rigorous troubleshooting, three test cases are implemented aiming at achieving reasonable response metrics – low rise time, settling time, overshoot, and steady state error.

The initial case is performed with well-educated inputs inferred from the Simulink model for the broad intention of narrowing down the experimental setup deviation and error margin. Afterwards, trial and error approach is pursued because of the high system uncertainty. To summarize the experiment findings, the controller has been proven to be effective by meeting all the design criteria. The controller behavior, along with friction, is interestingly efficient to reach the desired angle within acceptable error margins with virtually minimal to no overshoot. Additionally, the response when manually disturbing the rotator demonstrates the capability of the controller to dynamically respond to stimuli. All in all, the Reaction Control System was able to successfully control orientation utilizing a PD controller demonstrating similar dynamics to the RCS used on spacecraft and launch vehicles.

7. References

[1] “Rocket engine, Liquid Fuel, gemini reentry control system (RCS),” *National Air and Space Museum* Available: https://airandspace.si.edu/collection-objects/rocket-engine-liquid-fuel-gemini-reentry-control-system-rcs/nasm_A19740229000.

[2] “Rocket engine, Liquid Fuel, reentry control system (RCS), Gemini VIII,” *National Air and Space Museum* Available: https://airandspace.si.edu/collection-objects/rocket-engine-liquid-fuel-reentry-control-system-rcs-gemini-viii/nasm_A19721289001.

[3] Cervone, A., “18 - Chemical and cold gas propulsion systems,” *Next Generation CubeSats and SmallSats*, Elsevier, 2023, pp. 421-445 https://doi.org/10.1016/B978-0-12-824541-5.00021-2

8. Individual Contributions

Brian Goldblatt: Assisted with wiring and Arduino setup, Arduino connection to the computer. Worked on integration between Simulink model and hardware. Captured data and worked on debugging inputs/outputs. Processed data in Matlab after completion and worked on the Controller Evaluation section of report.

Jonathan Bargsten: Worked on the equations of motion and developed the Simulink controller and simulation. Ran simulations and found appropriate starting gains and processed data in MATLAB. Assisted in running the experiment by holding wires and adjusting the controller and apparatus base for the project to run properly despite the extreme issues with high-friction regions. Wrote the entirety of the controller design and simulation sections. Contributed to the introduction to include the examples of RCS systems. Recorded the final video that was submitted. Formatted and proofread the report throughout its progress.

Prashant Panta: Worked out the hardware component of the project. Strategically screwed the rotating piece with the wood on both top and bottom. Carried out experiments. Help document the experiment with media support (video recorded). Also, helped troubleshoot the issues that arrived during the experiment. Proofread the entire lab report, check the formatting, and finally edited wherever required.

Ethan Heyns: Wrote the equations of motion and did initial sizing for the nozzles and rotator dimensions. Did the wiring for the Arduino, solenoid valves, relays, and potentiometer. Plumbed the pneumatics system and supplied the air compressor. Wrote parts of the System Dynamics Modeling, Hardware Implementation and Controller Evaluation sections.

Mohammed S. Al-Mahrouqi: Figured out the Arduino package for Simulink and ensured its compatibility with the allocated board. Dimensioned the wooden rotating piece, manufactured it at AMS, and assembled the hardware together using screws, nails, hot glue, playdoh, and zip ties. ensured proper functionality of the assembly prior to control model deployment. Wrote Introduction, Conclusion & edited Implementation.