Lab 2: Digital PID Speed Control of a Turntable

Group 1

Jeremy Nielsen

Tara Boyle

Scott Miller

October 28, 2019

MEC 411 Control System Analysis and Design

Department of Mechanical Engineering

Stony Brook University, College of Engineering and Applied Sciences

Contents

I Abstract

The objective of this experiment was to design and subsequently implement a proportional-integral-derivative (PID) feedback loop control system to control the rotational speed of a turntable. An analog electrical circuit was constructed to model a given block diagram. The circuit included a proportional term controlled by R_{p2}, an integral term controlled by R_i, and a derivative term controlled by R_{d2}. These system parameters were adjusted to meet the desired performance specifications. Theoretical results were simulated using MATLAB and compared to the experimental data. The experimental results in the form of waveforms were gathered using software accompanying a digital oscilloscope. The findings showed the values of $R_{p2} = 4.5\ k\Omega$, $R_i = 0.91\ k\Omega$, and $R_{d2} = 0.535\ k\Omega$ led to a percent overshoot of $4 \pm 0.00641\%$, a settling time of $T_s = 21 \pm 1$ ms, and a rise time of $T_r = 12 \pm 1$ ms for the PID controller. These values fall within the desired performance specifications which means the PID speed control system adequately regulated the rotational speed of the turntable. However, the rise time and settling time did not agree with the theoretical values. It was determined that there was time delay since both values were off by approximately the same 10 ms. Once this time delay was taken into account, the results fit the specified criteria and confirmed with the theoretical model. Overall, this experiment was performed to demonstrate an understanding of closed loop control systems.

II Introduction

In order to properly analyze and design a control system to satisfy performance specifications, the first step is to understand the underlying theory governing the system [1]. Specifically, this experiment relies on adjusting the parameters of a PID feedback loop analog circuit to control the rotational speed of a turntable to achieve a set of given performance specifications. Since a closed loop control system operates on the principle of comparing the error between the input and the output, there must exist a mathematical equation which relates the two parameters. More specifically, a PID feedback control system uses three terms - the proportional gain (K_p), integration gain (K_i), and derivative gain (K_d) - to match a desired signal based on the measured different between the input and output of the system. The input of the system in the case of this experiment is a repeated, alternating unit step function, $u(t)$. A single pulse of the unit step function can is defined as follows:

$$u(t) = \begin{cases} 0 & \text{if } t = 0 \\ 1 & \text{if } t > 0 \end{cases} \tag{1}$$

While this function is given in terms of time, it must be transformed into the Laplace domain. The Laplace transform of the input unit step function can be performed as follows:

$$U(s) = \int_0^\infty u(t) \cdot e^{-st} dt = \int_0^\infty 1 \cdot e^{-st} dt = \frac{-1}{s}[e^{-st}]\Big|_0^\infty = \frac{-1}{s}[0-1] = \frac{1}{s} \tag{2}$$

The system is simply a response to the input which the controller tries to match. The ratio of the output to the input of the system, commonly referred to as the transfer function, can be obtained from the given block diagram (Figure 1). Since this particular system compares the output rotational speed to the input supplied voltage, the transfer function can be shown as:

$$T(s) = \frac{\omega(s)}{V(s)} \tag{3}$$

Where $\omega(s)$ represents the speed of the motor and $V(s)$ represents the input voltage. The derivation of this transfer function can be performed using Mason's Gain Formula as follows:

$$T(s) = \frac{\sum_{i=1}^{k} P_i \cdot \Delta_i}{\Delta} \tag{4}$$

Where P_i is the i^{th} forward path gain, Δ_i is the determinant of the i^{th} forward path, Δ is the determinant of the system, and k is the total number of forward paths. From the given block diagram, there is only one forward gain which is observed to be:

$$P_1 = G_c(s) \cdot G(s) \tag{5}$$

There is a feedback loop which has a gain of:

$$L_1 = -k_t \cdot G_c(s) \cdot G(s) \tag{6}$$

The determinant of the system is found by subtracting the sum of all loop gains from 1. The single loop is substituted into the formula as follows:

$$\Delta = 1 - L_1 = 1 + K_t G_c(s) G(s) \tag{7}$$

The determinant of the i^{th} forward path is the determinant for the portion of the block diagram which is not touching the i^{th} forward path. Since all portions of the block diagram are touching the forward path, the determinant of the forward path is simply shown as:

$$\Delta_1 = 1 - 0 = 1 \tag{8}$$

Therefore, the closed loop transfer function is:

$$T(s) = \frac{\omega(s)}{V(s)} = \frac{\sum P_k \cdot \Delta_k}{\Delta} = \frac{G_c(s)G(s)}{1 + K_t G_c(s) G(s)} \tag{9}$$

where

$$G(s) = \frac{K_m}{(L_a s + R_a)(Js + b) + K_b K_m} \approx \frac{K_m}{R_a(J_S + b) + K_b K_m} \tag{10}$$

and

$$G_c(s) = K_p + \frac{K_i}{s} + K_d s \tag{11}$$

$G(s)$ is a negative feedback loop for the motor composed of the:

$$\text{Armature} = \frac{K_m}{L_a s + R_a}, \tag{12}$$

$$\text{Load} = \frac{1}{J_S + b}, \tag{13}$$

and

$$\text{Back EMF} = K_b. \tag{14}$$

The paramters of the DC motor, tachometer, and load are given to be:

$$\text{Torque constant}(K_m) = 16.2\,\text{OZ} - \text{IN/A}, \tag{15}$$

$$\text{DC armature resistance}(R_a) = 11.5\,\Omega, \tag{16}$$

$$\text{DC armature inductance}(L_a) = 0, \tag{17}$$

$$\text{Moment of inertia}(J) = 2.5\,\text{OZ} - \text{IN}^2, \tag{18}$$

$$b = 0, \tag{19}$$

$$\text{Back EMF constant}(K_b) = 12\,\text{V/KRPM}, \tag{20}$$

and

$$\text{Tachometer constant}(K_t) = 12\,\text{V/KRPM}. \tag{21}$$

The transfer function can be expanded by substituting values of $G(s)$ and $G_c(s)$:

$$T(s) = \frac{(K_p + \frac{K_i}{s} + K_d s)(\frac{K_m}{R_a(Js+b)+K_bK_m})}{1 + K_t(K_p + \frac{K_i}{s} + K_d s)(\frac{K_m}{R_a(Js+b)+K_bK_m})} \tag{22}$$

This can be simplified by multiplying by the denominator of $G(s)$:

$$T(s) = \frac{(K_p + \frac{K_i}{s} + K_d s)(\frac{K_m}{R_a(Js+b)+K_bK_m})}{1 + K_t(K_p + \frac{K_i}{s} + K_d s)(\frac{K_m}{R_a(Js+b)+K_bK_m})} \cdot \frac{R_a(Js+b)+K_bK_m}{R_a(Js+b)+K_bK_m} \tag{23}$$

The result of this operation is:

$$T(s) = \frac{K_m(K_p + \frac{K_i}{s} + K_d s)}{R_a(Js+b) + K_bK_m + K_tK_m(K_p + \frac{K_i}{s} + K_d s)} \tag{24}$$

To simplify the complex fractions, the transfer function can be multiplied by $\frac{s}{s}$ as follows:

$$T(s) = \frac{K_m(K_p + \frac{K_i}{s} + K_d s)}{R_a(Js+b) + K_bK_m + K_tK_m(K_p + \frac{K_i}{s} + K_d s)} \cdot \frac{s}{s} \tag{25}$$

The result of this operation is:

$$T(s) = \frac{K_m(K_p s + K_i + K_d s^2)}{R_a(Js^2+bs) + K_bK_m s + K_tK_m(K_p s + K_i + K_d s^2)} \tag{26}$$

Combining like terms yields:

$$T(s) = \frac{K_m(K_p s + K_i + K_d s^2)}{s^2(R_aJ + K_tK_mK_d) + s(R_ab + K_bK_m + K_tK_pK_m) + (K_tK_mK_i)} \tag{27}$$

This can be compared to the general form of a transfer function for a second order ordinary differential equation which can be represented as:

$$T(s) = \frac{k\omega_n^2}{s^2 + 2\zeta\omega_n^2 s + \omega_n^2} \tag{28}$$

Where ω_n is the undamped natural frequency and ζ is the damping ratio. However, the transfer function must be manipulated into the general form by isolating the s^2 term of the characteristic equation (the polynomial in the denominator). This isolation is accomplished by multiplying the entire transfer function by $\frac{1}{R_aJ+K_tK_mK_t}/\frac{1}{R_aJ+K_tK_mK_t}$ which is shown as follows:

$$T(s) = \frac{K_m(K_p s + K_i + K_d s^2)}{s^2(R_aJ + K_tK_mK_d) + s(R_ab + K_bK_m + K_tK_pK_m) + (K_tK_mK_i)} \cdot \frac{\frac{1}{R_aJ+K_tK_mK_t}}{\frac{1}{R_aJ+K_tK_mK_t}} \quad (29)$$

The result of this operation now gives the transfer function in the general form as presented in Eq. (28):

$$T(s) = \frac{K_m \frac{(K_p s+K_i+K_d s^2)}{(R_aJ+K_tK_mK_d)}}{s^2 + s\frac{(R_ab+K_bK_m+K_tK_pK_m)}{(R_aJ+K_tK_mK_d)} + \frac{(K_tK_mK_i)}{(R_aJ+K_tK_mK_d)}} \quad (30)$$

The coefficients of the characteristic equation of the general form and this system's transfer function can be equated to determine ω_n and ζ. The characteristic equation of the general form and the transfer function of this system are set equivalent as follows:

$$s^2 + s\frac{(R_ab + K_bK_m + K_tK_pK_m)}{(R_aJ + K_tK_mK_d)} + \frac{(K_tK_mK_i)}{(R_aJ + K_tK_mK_d)} = s^2 + 2\zeta\omega_n^2 s + \omega_n^2 \quad (31)$$

The value of ω_n can be determined as follows:

$$\omega_n = \sqrt{\frac{(K_tK_mK_i)}{(R_aJ + K_tK_mK_d)}} \quad (32)$$

Similarly, the value of ζ can be determined as follows:

$$\zeta = \frac{R_ab + K_bK_m + K_tK_pK_m}{2\omega_n(R_aJ + K_tK_mK_d)} \quad (33)$$

To get the output of a transfer function, the function must be multiplied by the input. In this case, the Laplace transform of the unit step input was found to be $\frac{1}{s}$ in Eq. (2). For simplicity, the transfer function in the form of Eq. (27) is used. Therefore, the output speed of the motor is expressed as:

$$\omega(s) = T(s)V(s) = \frac{K_m(K_p s + K_i + K_d s^2)}{s^2(R_aJ + K_tK_mK_d) + s(R_ab + K_bK_m + K_tK_pK_m) + (K_tK_mK_i)} \cdot \frac{1}{s} \quad (34)$$

The result of this operation is:

$$\omega(s) = \frac{K_m(K_p s + K_i + K_d s^2)}{s^3(R_a J + K_t K_m K_d) + s^2(R_a b + K_b K_m + K_t K_p K_m) + s(K_t K_m K_i)} \tag{35}$$

To isolate the numerator as a polynomial, the transfer function must be multiplied by $\frac{1/K_m}{1/K_m}$ as follows:

$$\omega(s) = \frac{K_m(K_p s + K_i + K_d s^2)}{s^3(R_a J + K_t K_m K_d) + s^2(R_a b + K_b K_m + K_t K_p K_m) + s(K_t K_m K_i)} \cdot \frac{1/K_m}{1/K_m} \tag{36}$$

The result of this operation is:

$$\omega(s) = \frac{K_p s + K_i + K_d s^2}{s^3(\frac{R_A J}{K_m} + K_t K_d) + s^2(\frac{R_A b}{K_m} + K_b + K_t K_p) + s(K_t K_i)]} \tag{37}$$

However, s can be factored out of the denominator which yields:

$$\omega(s) = \frac{K_p s + K_i + K_d s^2}{s[(\frac{R_A J}{K_m} + K_t K_d)s^2 + (\frac{R_A b}{K_m} + K_b + K_t K_p)s + (K_t K_i)} \tag{38}$$

This expression of the rotational speed of the motor as a function of the Laplace variable can be transformed into the motor speed as a function of time using the inverse Laplace transform. To avoid solving the inverse Laplace transform of this transfer function with the gains as variables, the inverse Laplace will be determined once an experimental value of K_p, K_i, and K_d are determined. However, this is a necessary step since it enables the analysis of the time response of the system by transforming the system into a function of time. From that point, the experimental values of the parameters of interest can be determined to ascertain whether the system is considered to be within specifications or not.

To determine the gains of the PID controller, it is necessary to understand how the fundamental mathematics of the system are related to the physical interpretation of components specific to this experiment. For instance, the input signal of the system is provided by the waveform generator which was used to create a square wave with an amplitude of 1V (2V peak to peak) at a frequency of 1 Hz. The waveform generator is connected to the digital oscilloscope so that the signal can be viewed from the Ultrascope Software. An input signal of 12V to the turntable leads to an unloaded rotational speed of 1000 RPM [2]. However, the input signal of the waveform generator steps the voltage down to 1V. The rotational speed of the turntable varies proportionally to voltage, so the output becomes a square wave with a magnitude of 83.3 RPM, which the PID controller tries to match by minimizing the error between the output and the input. This happens by recording the tachometer sensor signal which is related to the motor speed.

A tachometer functions by detecting the frequency of voltage pulses produced by the spinning of a magnet that is attached to the shaft of the motor [3]. The tachometer voltage gradient is given to be 0.52 V/KRPM. This means that the output signal is scaled so that 0.52V is the output for 1000 RPM. The signal goes through a non-inverting proportional operational amplifier (op-amp) to provide a gain of 12V/0.52V so that 1000 RPM now creates a 12V signal. This gain is accomplished in the circuit by adjusting a potentiometer to the proper resistance to match the gain ratio. The derivation for this resistance value relies on the fundamental operating principles of operational amplifiers and is performed follows:

$$V_{out} - I_2 R_2 - 0 - V_{in} = 0 \tag{39}$$

So the R_2 resistance that makes V_{out} equal to 12V, or the K_t value, must be determined. Therefore, the expression for the current across the potentiometer is necessary:

$$I_2 = \frac{V_{out} - V}{R_2} \tag{40}$$

The current across the resistor connected to the negative input of the op-amp is also necessary:

$$I_1 = \frac{V_{in} - 0}{R_1} \tag{41}$$

Finally, the currents are set to be equal to each other since this op-amp is assumed to behave ideally:

$$V_{out} = V_{in}(1 + \frac{R_2}{R_1}) \tag{42}$$

The potentiometer resistance, R_2, was solved to be 23.1kΩ which is connected from the negative terminal of the op-amp to its output, and a 1kΩ resistor to produce a gain ratio of 23/1, which is close to the desired gain of 12/0.52. The amplifier was non-inverting because the voltage input was connected to the non-inverting input terminal, so the output voltage maintains the same sign as the input. This op-amp in the circuit was used to convert the output of the system into the same sign as the system's input.

This signal goes to the differential amplifier where the difference between the output and the input is measured. This is important since measuring the output signal is the first step in being able to adjust it to be closer to a desired output. The PID controller measures the difference between this output and the target signal generated by the waveform generator to find the error in the output.

In terms of the analog circuit, the PID controller is comprised of three op-amps. When analyzing each of the op-amps, the assumption is made that they behave ideally according to their ideal characteristics. The proportional term of the controller is an inverting amplifier, so the output is expressed as follows::

$$V_{out} = \frac{R_{p2}}{R_{p1}} \cdot V_{in} \tag{43}$$

The integral term of the controller is an inverting integrator amplifier, so the output is expressed as follows:

$$V_{out} = \frac{1}{R_i C_i} \int V_{in} dt = \frac{1}{R_i C_i} \cdot \frac{1}{s} \tag{44}$$

The derivative term of the controller is an inverting derivative amplifier, so the output is expressed as follows:

$$V_{out} = \frac{C_d R_{d2}}{R_{d1}} \cdot \frac{dV_{in}}{dt} = \frac{C_d R_{d2}}{R_{d1}} \cdot s \tag{45}$$

However, the forward gain of the PID controller, $G_c(s)$ as defined in the block diagram (Figure 1), is given in the form of equation 11. Therefore, the expressions for the K_p, K_i, and K_d terms can be determined by equating coefficients which is shown as follows:

$$K_p + \frac{K_i}{s} + K_d s = \frac{R_{p2}}{R_{p1}} + \frac{1}{R_i C_i} \cdot \frac{1}{s} + \frac{C_d R_{d2}}{R_{d1}} \cdot s \tag{46}$$

Therefore, K_p is determined as follows:

$$K_p = \frac{R_{p2}}{R_{p1}} \tag{47}$$

Similarly, K_i is determined as follows:

$$K_i = \frac{1}{R_i C_i} \tag{48}$$

Finally, K_p is determined as follows:

$$K_d = \frac{C_d R_{d2}}{R_{d1}} \tag{49}$$

after the signal passes through the PID gain, the output passes through a push-pull follower which amplifies the current supplied to the DC motor.

Once a clear understanding of the control system is established, there must be a figure of merit to precisely define whether or not the system is successful according to the given specifications. It is important to note that each of these performance metrics must be analyzed once the inverse Laplace transform of the transform function is determine since they are related to the time response of the output. Each of these metrics must be determined by analyzing the time response of the system after it is subjected to a single pulse of the unit step input. The rise time (T_r) is the time the system takes to initially reach the desired value. The settling time (T_s) is the time the system takes to fall within a specified percent of the final value. The percent overshoot (PO) is the maximum value the system reaches as a percentage of the desired value. In the case of this experiment, the design specifications were given to be a PO of less than 10% of the input, a settling time T_s to within 2% of the final value in less than 500 ms, and a rise time T_r less than 200 ms. PO, is calculated as follows:

$$PO = 100e^{\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}}, \tag{50}$$

Manipulating to calculate the actual percentage overshoot of the signal, the expression is expressed as follows:

$$P.O. = 100(\frac{\omega_{max} - \omega_{settled}}{\omega_{settled}}), \tag{51}$$

where ω_{max} is the peak ω value and $\omega_{settled}$ is the settled output [4]. Next, the settling time is the time taken for the function value to settle within 2% of the expected value and it is calculated as follows:

$$T_s = \frac{ln(2/100)}{\omega_n\zeta}. \tag{52}$$

Rise time, T_r, is calculated as:

$$T_r = \frac{\pi - \cos^{-1}(\zeta)}{\omega_n\sqrt{1-\zeta^2}}. \tag{53}$$

These mathematical relationships give a clear, complete, and thorough understanding of the working principles of the experiment.

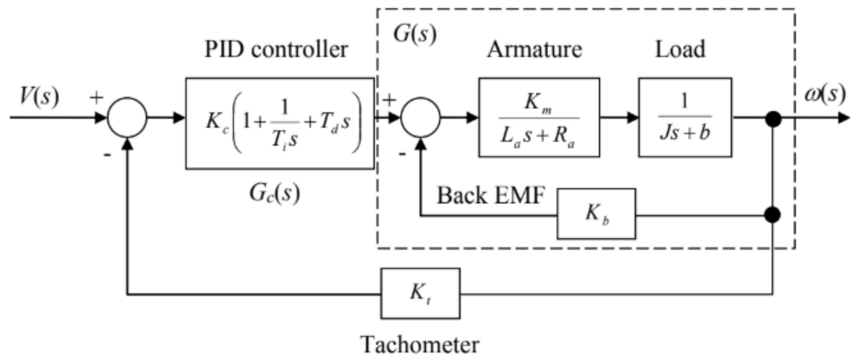

Figure 1 PID Speed Control System Block Diagram [2]

III Experimental Procedures

The experiment was performed with the following equipment: an Agilent E3630A power supply, an Agilent 33220A waveform generator, a RIGOL DS1054 digital oscilloscope, RIGOL UltraScope software, a Fairchild TIP 122 NPN darlington transistor, a Fairchild TIP 127 PNP darlington transistor, multiple operational amplifiers, a DC motor with a tachometer, a turntable, three 100 kΩ potentiometers, a 10 kΩ potentiometer, a breadboard, wires, resistors, capacitors, and diodes. Using the equipment, the turntable speed control PID system was constructed based on the circuit diagram provided (Figure 2). The potentiometer for the amplifier was set to 23.1 kΩ to normalize the output to the input. The function generator was used to create a square wave with an amplitude of 1 V (2 V peak to peak) at a frequency of 1 Hz. The system parameters of the proportional term (R_{p2}), integration time (R_i), and derivative time (R_{d2}) were controlled by adjusting the resistance of the potentiometers at each respective point in the circuit. The UltraScope software was used to record the experimental output waveforms resulting from the constructed control system. A closed loop feedback system with a proportional (P) controller was made by disconnecting the circuit at nodes C, D, E, and F. The resistance of the potentiometer, R_{p2}, was physically adjusted using a screwdriver such that the turntable speed was as close to the input as possible while maintaining system stability. All system parameters for the P controller were recorded along with the input and output waveforms using the UltraScope software. The circuit was made into a proportional-integral (PI) system by reconnecting nodes C and D. The resistances of the potentiometers, R_{p2} and R_i, were physically adjusted using a screwdriver such that the turntable speed was as close to the input as possible while maintaining system stability. All system parameters for the PI controller were recorded along with the input and output waveforms using the UltraScope software. Finally, nodes E and F were reconnected to make a proportional-integral-derivative (PID) controller. The resistances of the potentiometers, R_{p2}, R_{d2}, R_i, were physically adjusted using a screwdriver such that the turntable speed was as close to the input as possible while maintaining system stability. All system parameters for the PID controller were recorded along with the input and output waveforms using the UltraScope software. Additionally, the PID controller output needed to meet the design specifications of having a percent overshoot less than 10% settling time to within 2% of the final value in less than 500 ms, and a rise time of less than 200 ms. Once all of the criteria were satisfied, the PID variables and waveforms were recorded. The waveform data were exported as comma separated variable file formats (.CSV) which were exported to Microsoft Excel for analysis. The three sets of experimental results for a P, PI, and PID control system were compared to the theoretical results computed by simulating the respective systems using a MATLAB program.

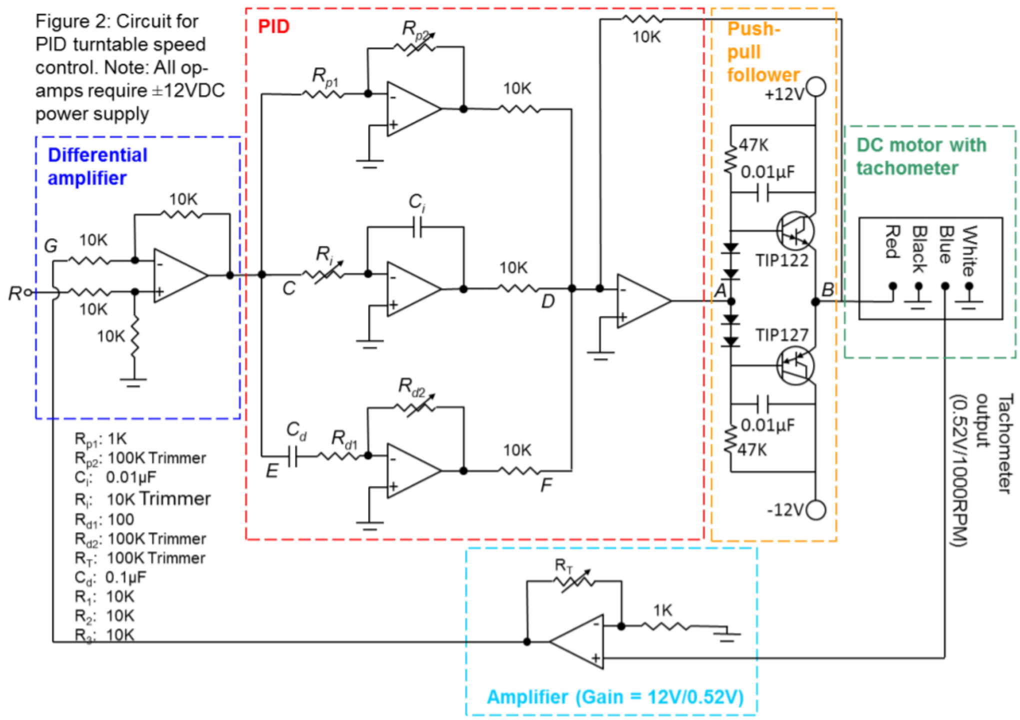

Figure 2 PID Speed Control System Circuit Diagram [2]

IV Results

As described in the Experimental Procedure (Section III), the analog circuit was initially modified to perform strictly as a proportional controller. While calibrating the P controller to match the square wave as closely as possible, the resistance of the potentiometer R_{p2} was increased until a point of marginal stability was observed at a value of 4.683 kΩ. This experimental result is recorded in Experimental Potentiometer Values (Table 1). Due to the fact that the signal experienced instability before reaching the desired square wave, there is a large gap between the desired and actual outputs as observed in experimental P controller input and output voltage versus time (Figure 3).

Similarly, the circuit was then modified to act as a PI controller. While calibrating the PI controller, the potentiometers were adjusted until a point of marginal stability was observed at an R_i value of 0.94 kΩ and an R_{p2} value of 4.2 kΩ. The output signal of the PI controller was observed to be within an acceptable range for the desired square wave signal which is shown in Experimental PI Controller Input and Output Voltage Versus Time (Figure 4). With the exception of the peak value, the theoretical value is always within an acceptable uncertainty range compared to the experimental PI voltage values. This corresponding uncertainty range is calculated in Error Analysis (Section V). However, the peak value represents an instantaneous, abnormal change considering the adjacent values are both within an acceptable range of uncertainty. The PI controller can be observed graphically to achieve the specifications of having a settling time to within 2% of the final value less than 500 ms, and a rise time is less than 200 ms (Figure 4). While the exact values for these specifications were not analyzed since they were not relevant for analysis of the full

PID controller, the success of the PI controller gives an approximated value for R_{p2} and R_i for the PID controller.

Finally, the circuit was modified to act as a full PID controller. While calibrating the PID controller, the R_{p2} and R_i parameters were kept to as close to the values used in the PI controller as possible, as the PI controller functioned within all specifications with the exception of the excessive percent overshoot. A point of marginal stability was observed while R_{p2} was equal to 4.5 $k\Omega$, R_i was 0.91 $k\Omega$, and R_{d2} was 0.535 $k\Omega$ (Table 1). The performance of the PID controller was an improvement upon that of the PI controller since there was an observed significant decrease in the percent overshoot as seen in Experimental PID Controller Input and Output Voltage Versus Time with Settling and Rise Time (Figure 5).

The experimental rise time, settling time, and percent overshoot were all determined graphically from the experimental data (Figure 5). The rise time (T_r) was observed to be 0.012 ± 0.001 seconds from the start of the waveform. The settling time (T_s) was observed to be 0.021 ± 0.001 seconds from the start of the waveform. Lastly, the peak value of the controller was determined to be 1.04 V at a time of 15 milliseconds from the start of the waveform. With a final settling point of 1 V, the percent overshoot (PO) of the output is calculated using Eq. (51), as shown below, and its uncertainty is calculated in Error Analysis (Section V):

$$P.O = 100(\frac{1.04 - 1}{1}) = 4 \pm 0.00641\%. \tag{54}$$

Theoretical simulations of the P, PI, and PID controllers were generated using MATLAB to establish theoretical expectations for the experiment and create a model for the behavior of the system. The source code of the script used to produce the simulated outputs can be found the in the Appendix (Section IX)(Figures A1-A4). The code utilized the transfer function from Equation 38 to solve for output speed of the motor ($\omega(s)$). All of the given constants are: K_m is the torque constant, K_b is the back EMF constant, R_a is the DC armature resistance, K_t is the tachometer output ratio, J is the internal moment of the motor, K_i is the integration constant, K_d is the derivative constant, and K_p is the proportional gain parameter. All of the aforementioned constant values are documented in Parameters of the DC motor, the Tachometer, and the Load (Table 2). This function of the output speed of the motor is converted to a function of time using the inverse Laplace transform, which was accomplished by the *ilaplace* command in MATLAB. The results of the *ilaplace* command are as follows:

$$\omega(t) = 1/Kt - JR_a e^{-t\frac{K_bK_m+K_pK_mK_t}{2JR_a+2K_dK_mK_t}} *$$

$$(\cos(t * \frac{\sqrt{(K_b^2K_m^2/4)-(K_bK_m^2K_tK_p/2)+(K_p^2K_m^2K_t^2/4)-(K_dK_m^2K_iK_t^2)-JK_iK_mK_tR_a}}{(JR_a+K_dK_mK_t)}) -$$

$$-\frac{\sin(t*\frac{\sqrt{(K_b^2K_m^2/4)-(K_bK_m^2K_tK_p/2)+(K_p^2K_m^2K_t^2/4)-(K_dK_m^2K_iK_t^2)-JK_iK_mK_tR_a}}{(JR_a+K_dK_mK_t)})(JR_a+K_dK_mK_t)(\frac{K_bK_m+K_pK_mK_t)}{2*J*R_a+2*K_d*K_m*K_t}-\frac{K_bK_m}{JR_a}}{\sqrt{(K_b^2*K_m^2)/4+(K_b*K_m^2*K_p*K_t)/2+(K_m^2*K_p^2*K_t^2)/4-K_d*K_i*K_m^2*K_t^2-J*K_i*R_a*K_m*K_t)}})$$

$$/(K_t * (J * R_a + K_d * K_m * K_t)).(55)$$

It should be noted that both J and K_m are converted to SI units for this calculation.

The MATLAB code simulated the behavior of the experimental P controller output within its uncertainty for the majority of the wavelength, which was calculated in the error analysis (Section V). However, the simulation failed predicted a much faster rise to the maximum value than was observed in the experiment, as observed in the MATLAB simulated and experimental P controller input and output voltage versus time (Figure 6). The MATLAB code simulated the behavior of the experimental PI controller output within the uncertainty with the exception of the abnormal experimental percent overshoot, as observed in the MATLAB simulated and experimental PI controller input and output voltage versus time graph (Figure 7). Lastly, the MATLAB code simulated the output of the PID controller correctly with the only exception being the simulation having a lower rise time, as observed in the MATLAB Simulated and Experimental PID Controller Input and output voltage Versus Time (Figure 8). From this simulation theoretical values for rise time $T_{r-theory}$, percent overshoot $P.O._{theory}$, and settling time $T_{s-theory}$ are graphically obtained for the PID controller. From the simulated data the following information was recorded: $T_{r-theory} = 0.002 \pm 0.0000294$ seconds, a peak of 1.0486 V, and $T_{s-theory} = 0.012 \pm 0.0012$ seconds. Using Eq.(26)and a settled signal of 1 V, the percent overshoot can calculated to be 4.86%±5.762%. The signal behavior of the PID controller (Table 3) shows that both the theoretical signal and the experimental signal are within the specification of our PID design parameters. These results can be compared to that of the PID performance parameters for the digital PID controller (Table 6).

Table 1 Experimental Potentiometer Values for all Experimental Controllers

	R$_{p2}[k\Omega]$	R$_i[k\Omega]$	R$_{d2}[k\Omega]$
P	4.683	0	0
PI	4.2	0.94	0
PID	4.5	0.91	.535

Table 2 Parameters of the DC motor, Tachometer, and Load [2]

Parameter	Value
K_m	16.2 OZ-IN / A
R_a	11.5 Ω
L_a	0
J	2.5 OZ-IN^2
b	0
K_b	12 V / KRPM
K_t	12 V / KRPM

Table 3 Time Response of PID Controller

	Specification Limit	Experimental Value	Theoretical Value
$T_r[ms]$	200	12	2±1± 0.0294
$T_s[ms]$	500	21	12±1± 1.2
P.O. [%]	10	4±0.00641	4.86 ± 5.762

Table 4 Lab 1 Experimental P, I, and D Terms for Each Configuration [5]

	K_c	T_i	T_d
P	2	0	0
PI	2	4.5×10^{-4}	0
PID	2	5.0×10^{-4}	5×10^{-5}

Table 5 PID Gain Values for Both Experiments

	Lab 1	Lab 2
K_p	2	4.5
K_i	4×10^3	1.0989×10^5
K_d	10×10^{-5}	5.35×10^-7

Table 6 PID Experimental Performance Parameters for Both Experiments

	Lab 1	Lab 2
$T_r[ms]$	79.9	12±1
$T_s[ms]$	89.0	21±1
PO [%]	2.13	4 ±0.00641

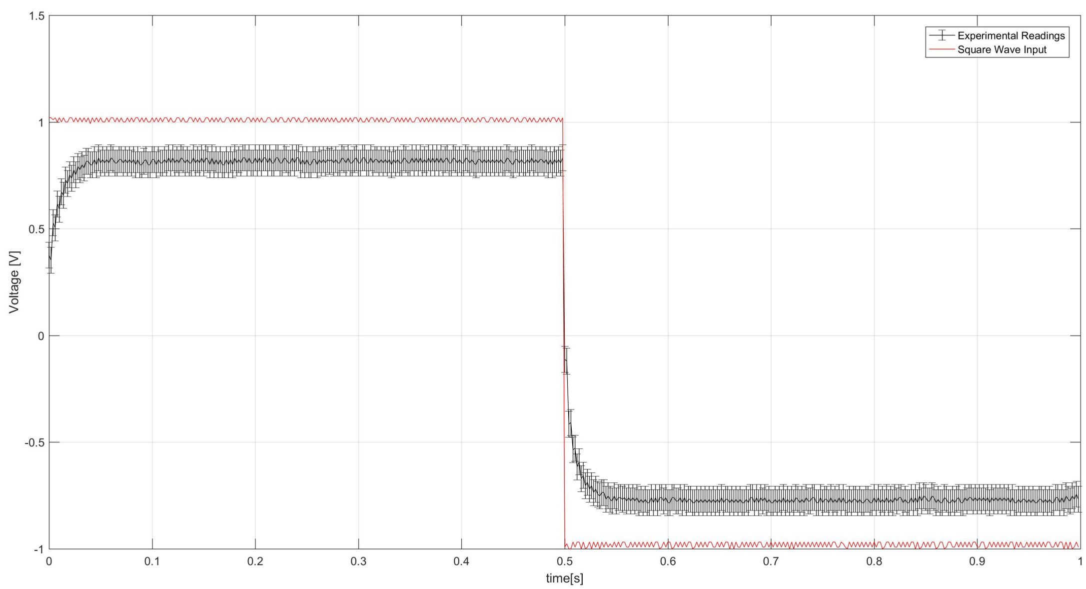

Figure 3 Experimental P Controller Input and Output Voltage Versus Time

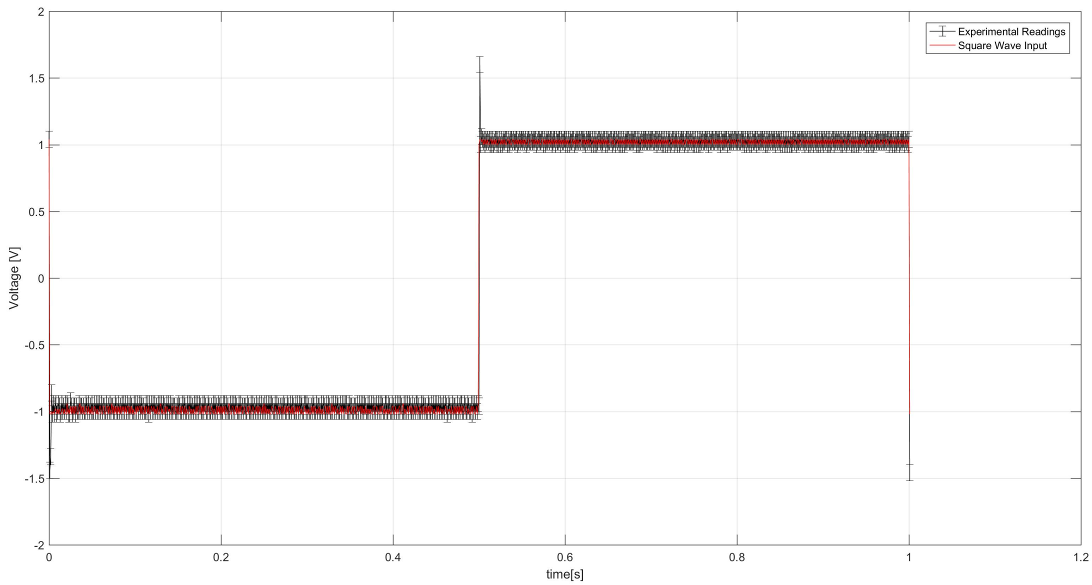

Figure 4 Experimental PI Controller Input and Output Voltage Versus Time

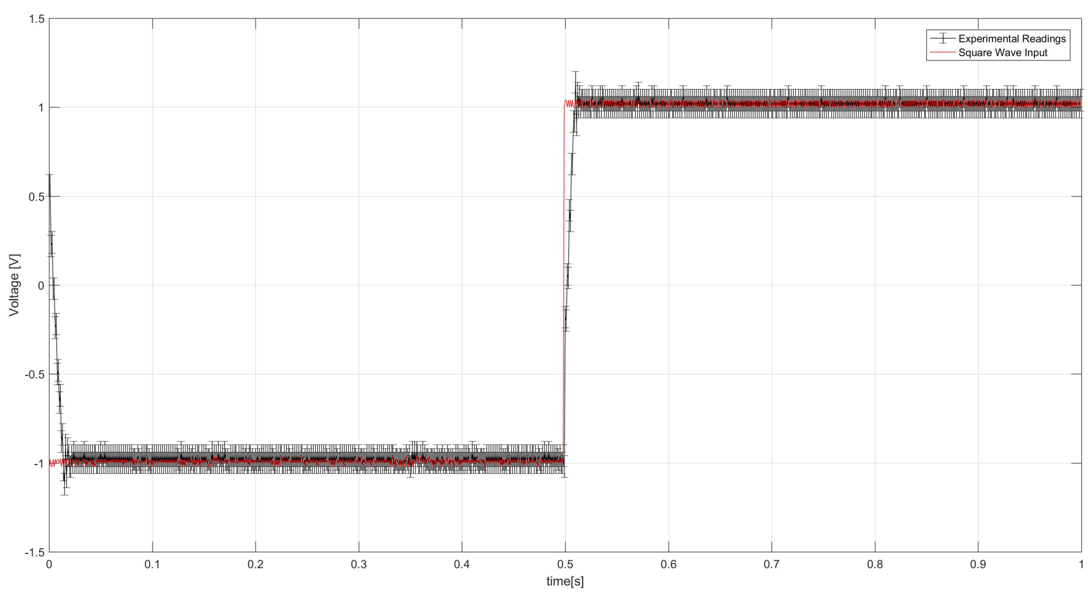

Figure 5 Experimental PID Controller Input and Output Voltage Versus Time with Settling Time and Rise Time

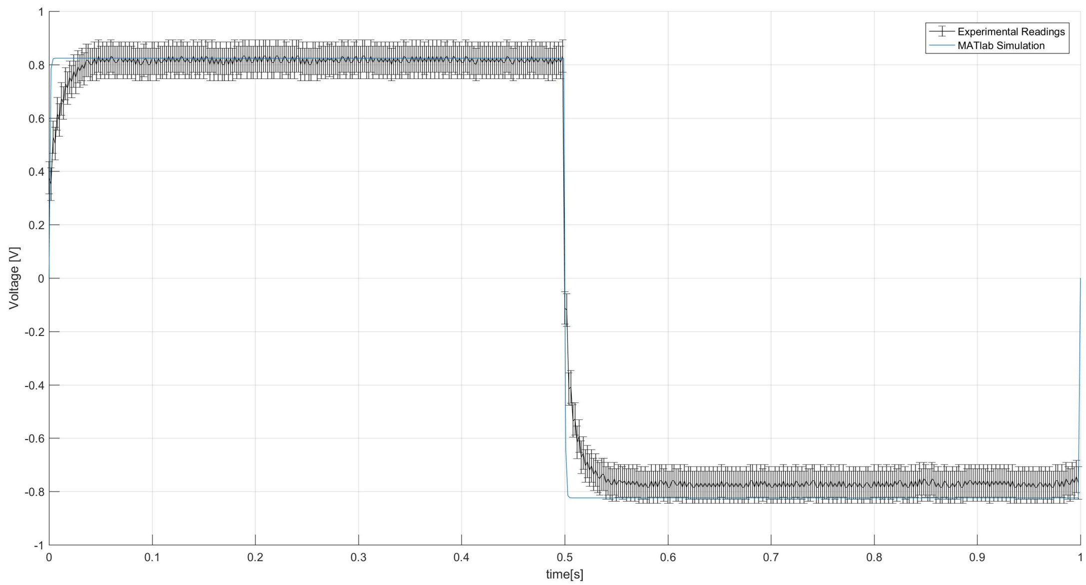

Figure 6 MATLAB Simulated and Experimental P Controller Input and Output Voltage Versus Time

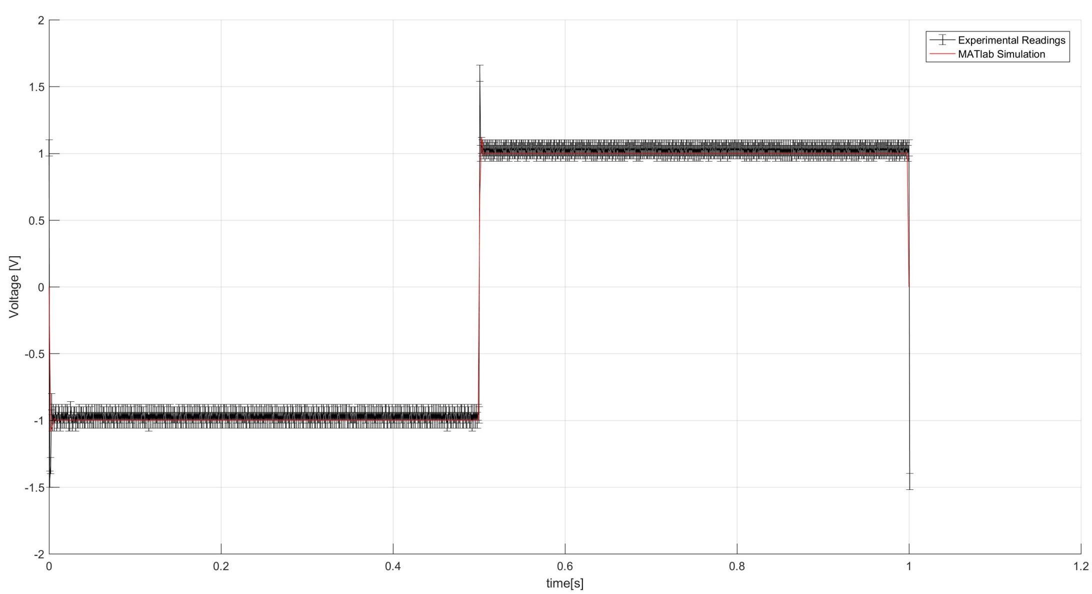

Figure 7 MATLAB Simulated and Experimental PI Controller Input and Output Voltage Versus Time

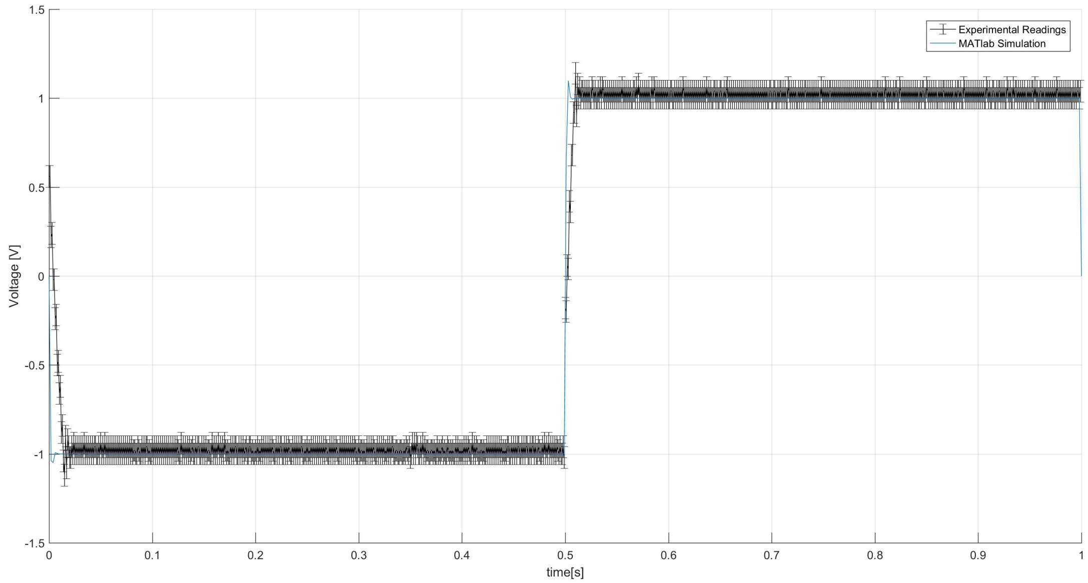

Figure 8 MATLAB Simulated and Experimental PID Controller Input and Output Voltage Versus Time

V Error Analysis

The experimental data gathered from this experiment was generated from a signal created with a waveform generator, which was used to operate an operational amplifier connected to the motor. A DC power supply was used to create the current required by the motor. While the voltage from the power supply is stepped by the circuit to match the signal of the waveform generator, errors in both are considered to have effect on the input voltage of the motor and thereby the experimental data. According to the datasheet for the waveform generator the system has a 4-digit resolution, thus an incremental uncertainty of 0.0001 V should be applied to the waveform generator signal, ΔV_{wave-i} [6]. Additionally, the same datasheet points to a 2 mV accuracy of the generator as the manufacturer's error, so ΔV_{wave-m} of 0.002 V is included in the calculation. Next, the power supply datasheet indicates a meter resolution of 10 mV, so an incremental error from the power supply is then applied such that ΔV_{pow-i} equals 0.01 V [7]. Next, the power supply also has a manufacturer accuracy of 0.5%. Once again a manufacturer error is considered such that ΔV_{pow-m} is equal to $0.005 * Vpow$ which has a value of 0.06 V, where V_{pow} is the ± 12 volts supplied by the power supply. Both of these voltage sources are connected directly in some way to the motor's power input. The waveform generator was connected through the DAQ system to the operational amplifier's positive input, which was replicated at the negative input connected to the motor. The power supply was connected to the motor when the transistor was actively drawing current from the generator to the motor. Therefore, each error is considered with equal weight in the calculation of input voltage error to the motor. Lastly, it should be noted that the oscilloscope collected data at much lower resolution than that of the DAQ system for digital circuit of the previous experiment, taking measurements at $0.02V$ increments [5]. An additional ΔV_{osc} of $\pm 0.02V$ is considered. The input voltage error, ΔV, was calculated as follows:

$$\Delta V = \sqrt{\Delta V_{wave-i}^2 + \Delta V_{wave-m}^2 + \Delta V_{pow-i}^2 + \Delta V_{pow-m}^2 + \Delta V_{osc}}. \tag{56}$$

Substituting the known values, the ΔV term is solved to be $\pm 0.0641\,V$. This uncertainty is to be considered in the experimental results of all three controller configurations. ΔV is represented in the error bars of each graph containing experimental data (Figures 3-8).

In order to assess the uncertainty of the theoretical error values described by the MATLAB simulation, the mathematical elements of the transfer function, as expressed in Eq.(33), must be identified. The natural frequency (ω_n) and the damping ratio (ζ) are described by Eq.(34) and Eq.(35), respectively. Both of those equations are derived from Eq.(27). These values are solved to be $\omega_n = 5.4083 \times 10^3 \pm \Delta\omega_n$ Hz and $\zeta = 0.1353 \pm \Delta\zeta$. The uncertainty of ω_n and ζ can be propagated with the errors and corresponding partial derivatives of K_b, K_m, and R_a. The error ΔK_b is equal to $0.1K_b$, which is equal to $\pm 1.2\frac{V}{KRPM}$. Similarly, the error ΔK_m is equal to $0.1K_m$ which is equal to $\pm 1.62\frac{oz-in}{A}$ or $\pm 1.144X10{-}2\frac{Nm}{A}$. Lastly, the error ΔR_a is equal to $0.15R_a$ which is equal to $\pm 1.725\Omega$. The partial derivative of ω_n with respect to K_m is shown as follows:

$$\frac{\partial \omega_n}{\partial K_m} = \frac{K_i K_t J R_a}{2\sqrt{\frac{(K_t K_m K_i)}{(R_a J + K_t K_m K_d)}}(K_d K_m K_t + J R_a)^2} \tag{57}$$

The partial derivative of ω_n with respect to R_a is shown as follows:

$$\frac{\partial \omega_n}{\partial R_a} = -\frac{K_i K_t J K_m}{2\sqrt{\frac{(K_t K_m K_i)}{(R_a J + K_t K_m K_d)}}(K_d K_m K_t + J R_a)^2}. \tag{58}$$

By obtaining the root sum square of the product of each partial derivative and each of the corresponding uncertainties, the total error in ω_n can be determined as follows:

$$\Delta\omega_n = \sqrt{(\Delta K_m \cdot \frac{\partial \omega_n}{\partial K_m})^2 + (\Delta K_m \cdot \frac{\partial \omega_n}{\partial R_a})^2} \tag{59}$$

Numerically, $\Delta\omega_n$ is solved to be $\pm 487.4281\, Hz$. Similarly, the partial derivatives of ζ in terms of ω_n, K_b, R_a, and K_m must be determined. The partial derivative of ζ with respect to ω_n is shown as follows:

$$\frac{\partial \zeta}{\partial \omega_n} = -\frac{K_b K_m + K_t K_p K_m}{2\omega_n^2 (R_a J + K_t K_m K_d)}, \tag{60}$$

The partial derivative of ζ with respect to K_b is shown as follows:

$$\frac{\partial \zeta}{\partial K_b} = \frac{K_m}{2\omega_n (R_a J + K_t K_m K_d)}, \tag{61}$$

The partial derivative of ζ with respect to R_a is shown as follows:

$$\frac{\partial\zeta}{\partial R_a} = \frac{J(K_bK_m + K_tK_pK_m)}{2\omega_n(R_aJ + K_tK_mK_d)^2}, \tag{62}$$

The partial derivative of ζ with respect to K_m is shown as follows:

$$\frac{\partial\zeta}{\partial K_m} = \frac{R_aJ(K_b + K_tK_p)}{2\omega_n(R_aJ + K_tK_mK_d)^2}. \tag{63}$$

By obtaining the root sum square of the product of each partial derivative and each of their respective errors, the total error in ζ can be obtained as follows:

$$\Delta\zeta = \sqrt{(\Delta\omega_n \cdot \frac{\partial\zeta}{\partial\omega_n})^2 + (\Delta K_b \cdot \frac{\partial\zeta}{\partial K_b})^2 + (\Delta R_a \cdot \frac{\partial\zeta}{\partial R_a})^2 + (\Delta K_m \cdot \frac{\partial\zeta}{\partial K_m})^2} \tag{64}$$

Numerically, $\Delta\zeta$ is calculated as ±0.0274. Rise time, T_r, is defined as:

$$T_r = \frac{\pi - \cos^{-1}(\zeta)}{\omega_n\sqrt{1-\zeta^2}}. \tag{65}$$

The partial derivative of T_r with respect to ω_n and ζ must be determined. The partial derivative of T_r with respect to ω_n is shown as follows:

$$\frac{\partial T_r}{\partial\omega_n} = -\frac{\pi - \cos^{-1}(\zeta)}{\omega_n^2\sqrt{1-\zeta^2}} \tag{66}$$

$$\frac{\partial T_r}{\partial\zeta} = \frac{1}{\omega_n\left(1-\zeta^2\right)} + \frac{\zeta\left(\pi - \cos^{-1}(\zeta)\right)}{\omega_n\left(1-\zeta^2\right)^{3/2}}, \tag{67}$$

respectively. By obtaining the root sum square of the product between each partial derivative and each of there respective errors, the total error in T_r can be obtained:

$$\Delta T_r = \sqrt{(-\frac{\pi - \cos^{-1}(\zeta)}{\omega_n^2\sqrt{1-\zeta^2}}\Delta\omega_n)^2 + ((\frac{1}{\omega_n\left(1-\zeta^2\right)} + \frac{\zeta\left(\pi - \cos^{-1}(\zeta)\right)}{\omega_n\left(1-\zeta^2\right)^{3/2}})\Delta\zeta)^2}. \tag{68}$$

Numerically, the error in the rise time equates to $\pm2.94 \times 10^{-5}$ seconds.

The percent overshoot formula, $P.O.$, is defined as follows [1]:

$$P.O. = 100e^{\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}}. \tag{69}$$

The only component with a known error in this formula is ζ. Therefore, the partial derivative in terms of ζ must be obtained,

$$\frac{\partial P.O.}{\partial \zeta} = \frac{100\pi e^{-\frac{\pi\zeta}{\sqrt{1-\zeta^2}}}}{(1-\zeta^2)^{3/2}}. \tag{70}$$

By multiplying the error in the damping ratio, $\Delta\zeta$ with the partial derivative of $P.O.$ in terms of ζ the error in the theoretical percent overshoot can be obtained.

$$\Delta P.O. = \frac{100\pi e^{-\frac{\pi\zeta}{\sqrt{1-\zeta^2}}}}{(1-\zeta^2)^{3/2}}\Delta\zeta \tag{71}$$

The error, $\Delta P.O.$, solves to be $\pm 5.7620\%$.

Next, the settling time is the time when the value settles to 2% of the expected value and it is represented as follows [1]:

$$T_s = \frac{ln(2/100)}{\omega_n\zeta}. \tag{72}$$

As done previously the partial derivatives are taken for ω_n and ζ respectively:

$$\frac{\partial T_s}{\partial \omega_n} = -\frac{ln(2/100)}{\omega_n^2\zeta} \tag{73}$$

and

$$\frac{\partial T_s}{\partial \zeta} = -\frac{ln(2/100)}{\omega_n\zeta^2} \tag{74}$$

The root sum square of the product between each error and its respective partial derivative is then taken,

$$\Delta T_s = \sqrt{(-\frac{\Delta\omega_n ln(2/100)}{\omega_n^2\zeta})^2 + (-\frac{\Delta\zeta ln(2/100)}{\omega_n\zeta^2})^2}. \tag{75}$$

Numerically, the error in the settling time solves to be ± 0.0012 seconds. This method of numerical error analysis is crucial for determining the validity of the results by establishing uncertainty ranges which the values can fall within. Otherwise, there is no way to tell if a discrepancy between theoretical and experimental values is valid or not.

All experimental time values corresponding to recorded voltage values for the PID controller are recorded at 1 ms increments. As a result, the experimental rise and settling time are attributed an uncertainty of 0.001 ms of incremental error. Additionally, the experimental percent overshoot is calculated to the peak recorded voltage, which is subjected to the aforementioned experimental voltage error present in (Figures 2-8) The experimental percent overshoot error can thereby be calculated using equation 51 replacing the $\omega_{settled} - \omega_{max}$ term with the voltage error $\pm 0.0641\,V$. The resulting uncertainty for experimental percent overshoot is 0.00641 V.

Lastly, while the calculated K_p, K_i, and K_d were dependent on the recorded resistances of the potentiometers R_{p2}, R_i, and R_{d2}, the resolution on the multi-meter reading was so low relative to the recorded values that it was considered negligible. Therefore, the uncertainty of these values is not considered.

VI Discussion

The MATLAB simulation for the PID controller implemented in this experimented replicated the percent overshoot of the experimental results, determined as 4 milliseconds, within the uncertainty which was determined as $4.86 \pm 5.762\%$. However, both the rise time and settling time of the experimental data are found outside of the theoretical uncertainties. While the experimental values for rise time and settling time were 12 ± 1 and 21 ± 1 ms respectively, the theoretical values were 2 ± 0.0294 and 12 ± 1.2 ms respectively. The expected rise time, settling time, and peak time are all between 9 and 11 millisecond from their experimental counterparts. Therefore, the collected PID experimental data was observed with possible time delay in data acquisition. Data was observed specifically at the point where the square wave, supplied by the waveform generator, shifted from positive to negative. As suspected, the tachometer voltage did not start changing until after the square wave shifted from positive to negative, signifying some time delay in the reaction of the motor. This can be observed from the PID controller's experimental data (Figure 5), This data starts at the beginning of the negative half of the square wave, but the motor output remains positive for multiple data points. Based on the uniform discrepancy between the theoretical and experimental rise, peak, and settling time of roughly 10 milliseconds, a time delay of the same value was placed in the a corrected simulation of the PID controller (Figure 9). This correction resulted in a simulated settling and rise time of 12 ± 0.0294 and 22 ± 1.2 ms respectively, both within uncertainty of the experimental values of 12 ± 1 and 21 ± 1 ms.

To compare The PID parameters from digital and analog control circuits, both are considered in terms K_p, K_i, and K_d as shown in Lab 1 Experimental Proportional, Integral, and Derivative Time Terns for all Experimental Controllers (Table 4). For the analog circuit for the PID controller these values are found using Eq(47), Eq(48), and Eq(49) respectively using the resistance and capacitance values found on (Table 1). As a result, K_p was found to be 4.5, K_i was found to be 1.0989×10^5 Hz, and K_d was found to be 5.35×10^{-7} seconds for the analog control circuit. To convert the digital control PID controller parameters to K_p, K_i, and K_d the following equations were used:

$$K_p = K_c, \tag{76}$$

$$K_i = \frac{K_c}{T_i}, \tag{77}$$

and

$$K_d = K_c T_d. \tag{78}$$

By plugging in the values time terms received from the PID controller with the digital controls (Table 4) the results were as follows; K_p was found to be 2, K_i was found to be 4×10^3 Hz, and K_d was found to be 10×10^{-5} seconds for the digital control circuit. Based on these values the K_i for this experiment is pretty extreme, and is what resulted in the very high uncertainty of percent overshoot. Further examination reveals the K_d term to be insignificant in comparison to that of the digital control circuit, as seen in the PID K values table (Table 5). Compared to the digitally controlled PID controller, this PID controller had such a small K_d it was essentially a PI controller; This is why a higher level of control was reached for the PID in the last lab (Figure 6). In terms of the user in both PID controller experiments, this experiment was much harder to configure than the previous one. In order to check the values of a given parameter, a potentiometer had to be completely removed from the circuit to measure its resistance using a multimeter. However, with the digitally controlled PID controller it was possible to change and observe parameters in real time while still knowing the relevant values.

VII Conclusions

- The experimental results showed that a PID turntable speed control system with analog controls can be created with accurate results.
- The relevant results were found to be the Experimental Potentiometer Values for all Experimental Controllers (Table 1) and the Time Response of the PID Controller (Table 3).
- The experimental results did not agree with the theoretical values within acceptable ranges of uncertainty.
- The discrepancy between the theoretical and experimental results is attributed to a time delay, since all the values were off by a similar time value. When this offset is taken into account, then the model agrees with the specifications and theoretical results.
- In future investigations, it is recommended to use equipment with a resolution finer than 2% of the desired final value, since the resolution of oscilloscope used to obtain these results made the settling time difficult to reasonably determine.

VIII References

[1] Machtay, Noah Donald, Ph.D., P.E. MEC 411 Control System Analysis and Design. Fall 2019. SUNY Stony Brook University Department of Mechanical Engineering College of Engineering and Applied Sciences

[2] Machtay, Noah Donald, Ph.D., P.E. "MEC 411 PID Speed Control of a Turntable." SUNY Stony Brook University Department of Mechanical Engineering College of Engineering and Applied Sciences

[3] "What Is a Tachometer?" *AZoSensors.com*, 8 Aug. 2019, www.azosensors.com/article.aspx?ArticleID=310.

[4] Gene F. Franklin,J. David Powell, and Abbas Emami-Naeini, "Feedback Control of Dynamic Systems", 6th ed., Prentice Hall, 2010.

[5] Jeremy Nielsen, Tara Boyle, Scott Miller,"Lab 1: Digital PID Speed Control of a Turntable"

[6] "Keysight 33220A 20 MHz Function/Arbitrary Waveform Generator Data Sheet," *Keysight Technologies*, http://www.testequipmenthq.com/datasheets/Agilent-E3630A-Datasheet.pdf.

[7] "Agilent E36XXA Series Non-Programmable DC Power Supplies Data Sheet," *Agilent Technologies*, https://literature.cdn.keysight.com/litweb/pdf/5988-8544EN.pdf?id=187648.

IX Appendix

```
clear
clc
%Km=16.2
Ra=11.5
%J=2.5
Kb=12
Kt=12
Km=16.2*0.02835*9.81*0.0254; %%Converted to Nm/A
J=2.5*0.02835*9.81*0.0254^2; %%Converted to Nm^2
b=0


s=sym('s')
G=Km/(Ra*(J*s+b)+Kb*Km)

Rp2=4.683e+03
Ri=.91e+03
Rd2=535

Rp1=1000
Ci=0.01*10^-6
Rd1=100
Cd=0.1*10^-6


```

```
Kp=Rp2/Rp1
Ki=1/(Ri*Ci)
Kd=Rd2*Cd/Rd1
Gc=Kp%+Ki/s+Kd*s
w=Gc*G/(s*(1+Kt*Gc*G))
wt=ilaplace(w)
WT=@(t) double(subs(wt,'t',t))
t=linspace(0,1,500);
d=0
D=0
for i=1:length(t)
    if (t(i)>=0 & t(i)<.5)|(t(i)>=1 & t(i)<1.5)
        if d==1
            D=D+.5
        end
        d=0;
        V(i)=WT(t(i)-D);
    else
        if d==0
            D=D+.5
        end
        d=1;
        V(i)=-WT(t(i)-D);
    end
end
figure(1)
```

```
grid on
%plot(t,V);%plots simulation
hold on
S = stepinfo(V,t)

%the rest is taras plot of our real data, you can run all together to
%compare of just highlight up to hear for the simulation only.


%
% PID= xlsread('Lab1.xlsx','C23:C1215');
% Swave=xlsread('Lab1.xlsx','B23:B1215');
% time=xlsread('Lab1.xlsx','A23:A1215');
PID=xlsread('MEC411lab2P','MEC411lab2P','C100:C599');
Swave=xlsread('MEC411lab2P','MEC411lab2P','B100:B599');
time=xlsread('MEC411lab2P','MEC411lab2P','A100:A599')*2*10^-3;
time=time-time(1)
% PID= xlsread('Lab1.xlsx','C119:C267');
% Swave=xlsread('Lab1.xlsx','B119:B267');
% time=xlsread('Lab1.xlsx','A119:A267');
difference=abs((Swave-PID));
S = stepinfo(PID,time)
dPID(length(PID),1)=0
dPID=dPID+(sqrt((0.005*12)^2+.01^2+0.002^2+0.0001^2))


 errorbar(time,PID,dPID,'k');
```

```
hold on
plot(t,12*V);%multiplied by 12 to convert from KRPM to V
legend('Experimental Readings','MATlab Simulation')
 xlabel('time[s]')
ylabel('Voltage [V]')



figure(2)
grid on
 errorbar(time,PID,dPID,'k');
hold on
plot(time,Swave,'r')
 hold on
xlabel('time[s]')
ylabel('Voltage [V]')
```

Figure A1 MATLAB code for Experimental and Theoretical P Controller Plots

```
clear
clc
%Km=16.2
Ra=11.5
%J=2.5
Kb=12
Kt=12
Km=16.2*0.02835*9.81*0.0254; %%Converted to Nm/A
J=2.5*0.02835*9.81*0.0254^2; %%Converted to Nm^2
b=0


s=sym('s')
G=Km/(Ra*(J*s+b)+Kb*Km)

Rp2=4.2e+03
Ri=.94e+03
Rd2=535

Rp1=1000
Ci=0.01*10^-6
Rd1=100
Cd=0.1*10^-6



Kp=Rp2/Rp1
```

```
Ki=1/(Ri*Ci)
Kd=Rd2*Cd/Rd1
Gc=Kp+Ki/s%+Kd*s
w=Gc*G/(s*(1+Kt*Gc*G))
wt=ilaplace(w)
WT=@(t) double(subs(wt,'t',t))
t=linspace(0,1,500);
d=0
D=0
for i=1:length(t)
    if (t(i)>=0 & t(i)<.5)|(t(i)>=1 & t(i)<1.5)
        if d==1
            D=D+.5
        end
        d=0;
        V(i)=-WT(t(i)-D);
    else
        if d==0
            D=D+.5
        end
        d=1;
        V(i)=WT(t(i)-D);
    end
end
figure(1)
grid on
%plot(t,V);%plots simulation
```

```
hold on
S = stepinfo(V,t)

%the rest is taras plot of our real data, you can run all together to
%compare of just highlight up to hear for the simulation only.


%
% PID= xlsread('Lab1.xlsx','C23:C1215');
% Swave=xlsread('Lab1.xlsx','B23:B1215');
% time=xlsread('Lab1.xlsx','A23:A1215');
PID=xlsread('MEC411lab2PI','MEC411lab2PI','C103:C1104');
Swave=xlsread('MEC411lab2PI','MEC411lab2PI','B103:B1104');
time=xlsread('MEC411lab2PI','MEC411lab2PI','A103:A1104')*10^-3;
time=time-time(1)
% PID= xlsread('Lab1.xlsx','C119:C267');
% Swave=xlsread('Lab1.xlsx','B119:B267');
% time=xlsread('Lab1.xlsx','A119:A267');
difference=abs((Swave-PID));
S = stepinfo(PID,time)
dPID(length(PID),1)=0
dPID=dPID+(sqrt((0.005*12)^2+.01^2+0.002^2+0.0001^2))


 errorbar(time,PID,dPID,'k');
 hold on
 plot(t,12*V,'r');%multiplied by 12 to convert from KRPM to V
```

```
legend('Experimental Readings','MATlab Simulation')
 xlabel('time[s]')
ylabel('Voltage [V]')



figure(2)
grid on
 errorbar(time,PID,dPID,'k');
hold on
plot(time,Swave,'r')
 hold on
xlabel('time[s]')
ylabel('Voltage [V]')
```

Figure A2 MATLAB code for Experimental and Theoretical PI Controller Plots

```
clear
clc
%Km=16.2
Ra=11.5
%J=2.5
Kb=12
Kt=12
Km=16.2*0.02835*9.81*0.0254; %%Converted to Nm/A
J=2.5*0.02835*9.81*0.0254^2; %%Converted to Nm^2
b=0


s=sym('s')
G=Km/(Ra*(J*s+b)+Kb*Km)

Rp2=4.5e+03
Ri=.91e+03
Rd2=535

Rp1=1000
Ci=0.01*10^-6
Rd1=100
Cd=0.1*10^-6



Kp=Rp2/Rp1
```

```
Ki=1/(Ri*Ci)
Kd=Rd2*Cd/Rd1
Gc=Kp+Ki/s+Kd*s
w=Gc*G/(s*(1+Kt*Gc*G))
wt=ilaplace(w)
WT=@(t) double(subs(wt,'t',t))
t=linspace(0,1,500);
d=0
D=0
for i=1:length(t)
    if (t(i)>=0 & t(i)<.5)|(t(i)>=1 & t(i)<1.5)
        if d==1
            D=D+.5
        end
        d=0;
        V(i)=-WT(t(i)-D);
    else
        if d==0
            D=D+.5
        end
        d=1;
        V(i)=WT(t(i)-D);
    end
end
figure(1)
grid on
%plot(t,V);%plots simulation
```

```
hold on
S = stepinfo(V,t)

%the rest is taras plot of our real data, you can run all together to
%compare of just highlight up to hear for the simulation only.


%
% PID= xlsread('Lab1.xlsx','C23:C1215');
% Swave=xlsread('Lab1.xlsx','B23:B1215');
% time=xlsread('Lab1.xlsx','A23:A1215');
PID=xlsread('MEC411lab2PID','MEC411lab2PID11','C104:C1103');
Swave=xlsread('MEC411lab2PID','MEC411lab2PID11','B104:B1103');
time=xlsread('MEC411lab2PID','MEC411lab2PID11','A104:A1103')*10^-3;
time=time-time(1)
% PID= xlsread('Lab1.xlsx','C119:C267');
% Swave=xlsread('Lab1.xlsx','B119:B267');
% time=xlsread('Lab1.xlsx','A119:A267');
difference=abs((Swave-PID));
S = stepinfo(PID,time)
dPID(length(PID),1)=0
dPID=dPID+(sqrt((0.005*12)^2+.01^2+0.002^2+0.0001^2+0.02^2))


 errorbar(time,PID,dPID,'k');
 hold on
 plot(t,12*V);%multiplied by 12 to convert from KRPM to V
```

```
legend('Experimental Readings','MATlab Simulation')
 xlabel('time[s]')
ylabel('Voltage [V]')



figure(2)
grid on
 errorbar(time,PID,dPID,'k');
hold on
plot(time,Swave,'r')
 hold on
xlabel('time[s]')
ylabel('Voltage [V]')
```

Figure A3 MATLAB code for Experimental and Theoretical PID Controller Plots

```
clear
clc
%Km=16.2
Ra=11.5
%J=2.5
Kb=12
Kt=12
Km=16.2*0.02835*9.81*0.0254; %%Converted to Nm/A
J=2.5*0.02835*9.81*0.0254^2; %%Converted to Nm^2
b=0


s=sym('s')
G=Km/((Ra)*(J*s+b)+Kb*Km)

Rp2=4.5e+03
Ri=.91e+03
Rd2=535

Rp1=1000
Ci=0.01*10^-6
Rd1=100
Cd=0.1*10^-6



Kp=Rp2/Rp1
```

```
Ki=1/(Ri*Ci)
Kd=Rd2*Cd/Rd1
Gc=Kp+Ki/s+Kd*s
w=Gc*G/(s*(1+Kt*Gc*G))
wt=ilaplace(w)
WT=@(t) double(subs(wt,'t',t))
t=linspace(0,1,500);
d=0
D=0
for i=1:length(t)
    if (t(i)>=0 & t(i)<.5)|(t(i)>=1 & t(i)<1.5)
        if d==1
            D=D+.5
        end
        d=0;
        V(i)=-WT(t(i)-D);
    else
        if d==0
            D=D+.5
        end
        d=1;
        V(i)=WT(t(i)-D);
    end
end
figure(1)
grid on
%plot(t,V);%plots simulation
```

```
hold on
S = stepinfo(V,t)

%the rest is taras plot of our real data, you can run all together to
%compare of just highlight up to hear for the simulation only.



%
% PID= xlsread('Lab1.xlsx','C23:C1215');
% Swave=xlsread('Lab1.xlsx','B23:B1215');
% time=xlsread('Lab1.xlsx','A23:A1215');
PID=xlsread('MEC411lab2PID','MEC411lab2PID11','C104:C1103');
Swave=xlsread('MEC411lab2PID','MEC411lab2PID11','B104:B1103');
time=xlsread('MEC411lab2PID','MEC411lab2PID11','A104:A1103')*10^-3;
time=time-time(1)
% PID= xlsread('Lab1.xlsx','C119:C267');
% Swave=xlsread('Lab1.xlsx','B119:B267');
% time=xlsread('Lab1.xlsx','A119:A267');
difference=abs((Swave-PID));
S = stepinfo(PID,time)
dPID(length(PID),1)=0
dPID=dPID+(sqrt((0.005*12)^2+.01^2+0.002^2+0.0001^2+0.02^2))


 errorbar(time,PID,dPID,'k');
 hold on
 plot(t+0.01,12*V);%multiplied by 12 to convert from KRPM to V
```

```
legend('Experimental Readings','MATlab Simulation')
 xlabel('time[s]')
ylabel('Voltage [V]')
```

Figure A4 MATLAB Code for Corrected Controller Plots

Comment Summary

Page 4

1. difference

Page 10

2. Vin

Page 12

3. Please provide derivations.
4. To make the analysis complete, please also provide the derivation for differential amplifier. And explain the purpose of the push-pull follower.

Page 21

5. Percentage of overshoot is not dependent on time.