Project
ME 5245 Mechatronic Systems

PART B

Authors: Joshua Yoo, Tirth Patel

1 Objective

The objective of Part B is to compare the physical motor response with the simulated response in terms of motor speed and position at different Pulse Width Module (PWM) duty cycles. By running open-loop hardware tests at various speeds, the motor gain and time constant are identified and then used to tune the Simscape model parameters. This section evaluates how accurately the simulation reproduces the hardware behavior for different PWM duty cycles, based on the measured angular velocity and position. We aim to derive the experimental Transfer Function $G(s)$ of the motor system. Furthermore, we calculate the physical parameters of the motor (rotor inertia, damping coefficient, and torque constants) to populate the simscape model for accurate simulation.

2 Open-loop Hardware Tests

As we already know, the hardware setup consists of an Arduino Uno R3 interfacing with an L293D H-Bridge driver to power the DC motor. The motor is powered by a 5-AA battery pack. Ideally, this provides approximately 7.5V, though actual voltage was monitored.

2.1 Experimental Setup and Data Acquisition Methodology

Below is the modified Simulink model for motor data acquisition from Part A as shown in Fig. 1, There is an arduino encoder block where input pins are set according to the circuit. The readings from this block are connected to the motor gain and then it is filtered using the discrete filter block whose values are determined after some brute force experiments. This filter actually removes the steady state noise from the readings to give us a better looking plot. The rest of the model is the same as Part A.

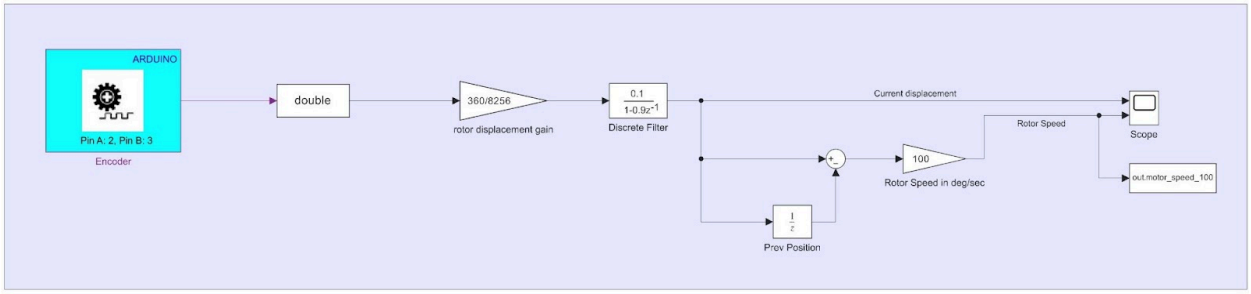

Figure 1: Simulink Model for Encoder Data Acquisition

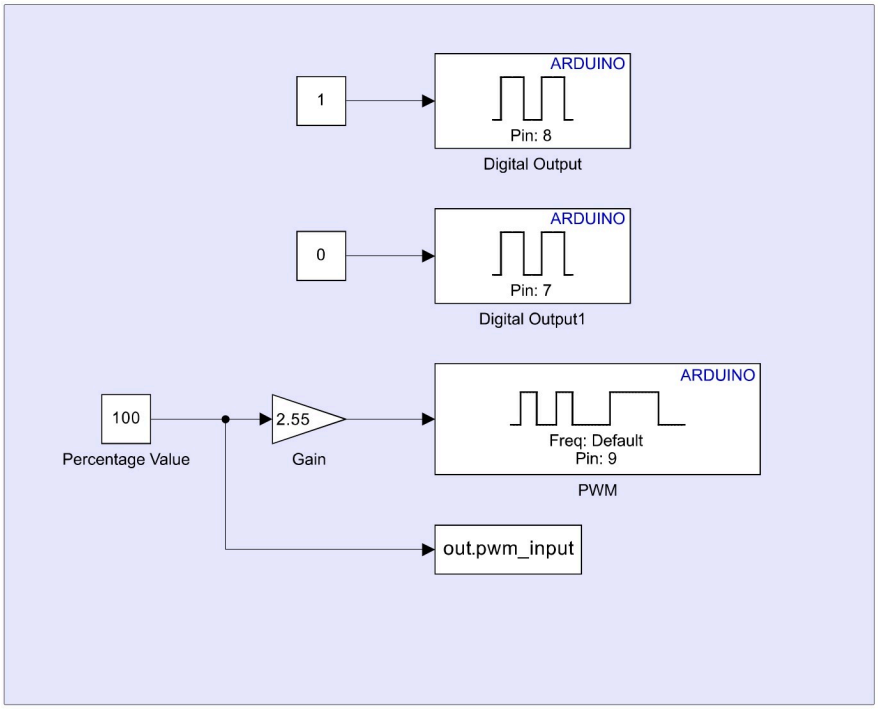

Figure 2: Simulink model for giving Motor Commands from Arduino to Encoder

Input: Step voltage inputs were applied by setting specific PWM values (0-100%) on the Arduino.

Output: The motor speed was calculated by reading the quadrature encoder counts over fixed time intervals.

Scaling: Voltage was calculated as: $V_{in} = \frac{\%PWM}{100} \times V_{battery}$

For 100% PWM cycle, input voltage: $V_{in} = 7.5V$

Output speed was converted from encoder counts to radians per second (rad/s).

2.2 Experimental Results

Step response tests were conducted at five different PWM duty cycles to observe the motor's transient and steady-state behavior.

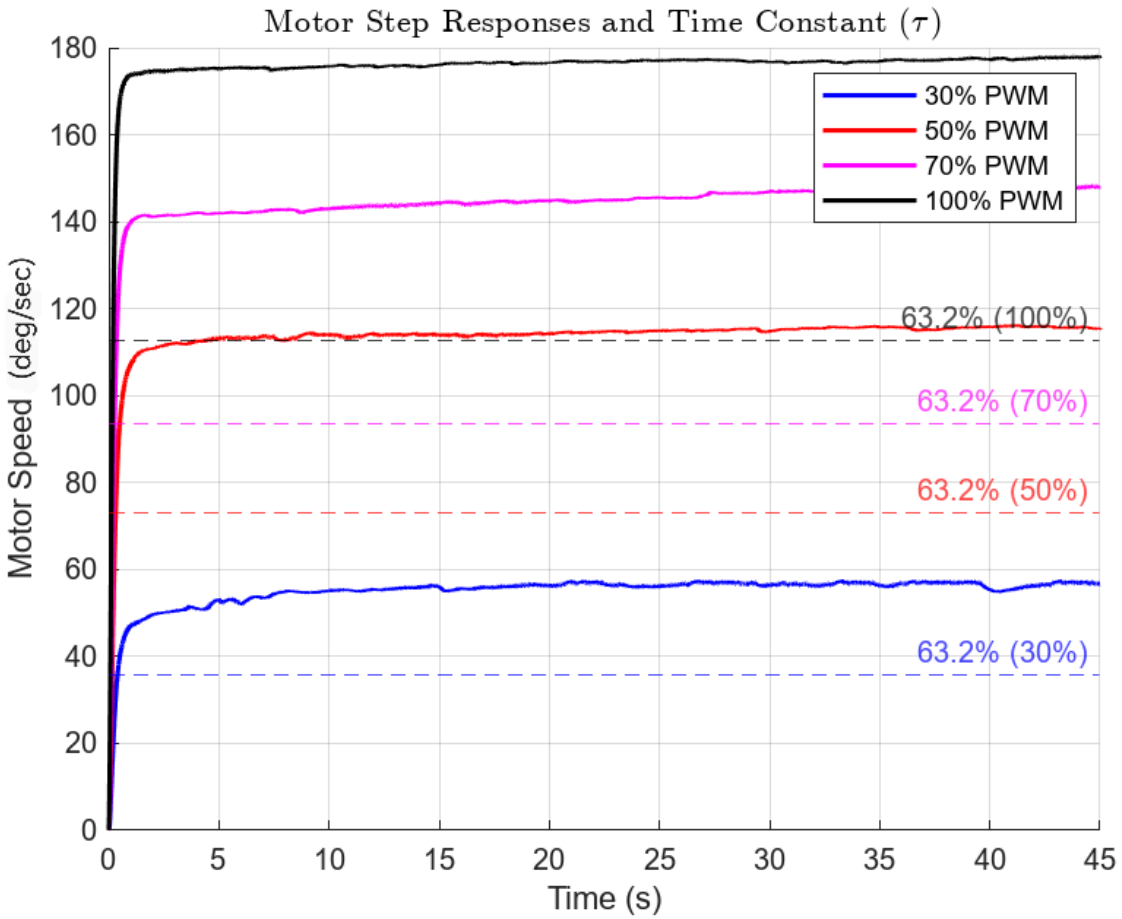

Figure 3: Motor Speed vs. Time Plot for 5 different PWMs.

As seen in Fig 3, the motor behaves as a linear first-order system. Higher PWM duty cycles result in higher steady-state velocities (ω_{ss}).

2.2.1 Calculation of DC Gain (K_{exp})

The DC Gain K_{exp} represents the ratio of the output speed to the input voltage in steady state.

$$K_{exp} = \frac{\Delta\omega_{ss}}{\Delta V_{in}}$$

The steady-state velocities were plotted against the applied voltages, and a linear regression (using MATLAB's `polyfit`) was performed to determine the average gain.

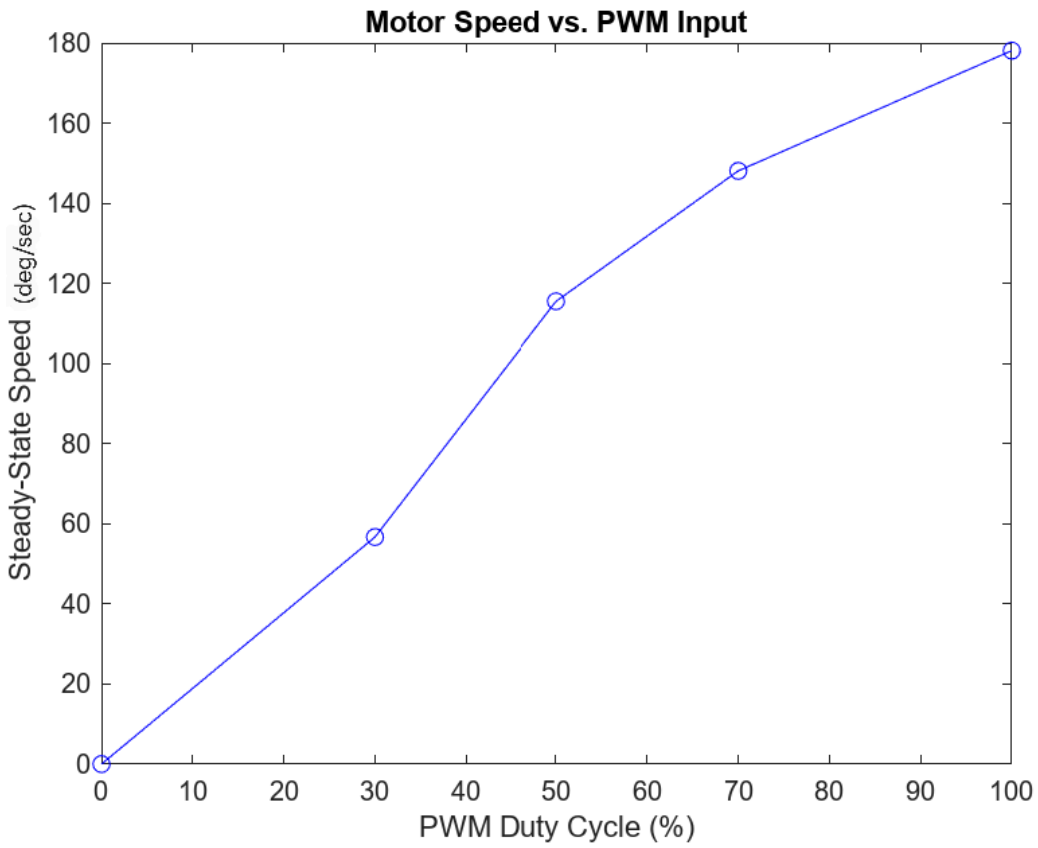

Figure 4: Plot for Steady State Speed vs. Input Voltage

Result: Based on the slope of the linear fit in Fig. 4, the experimental DC Gain is:

$$K_{exp} = slope \times \frac{\pi}{180} \times \frac{100}{V_{in}}$$

$$= 1.8508 \times \frac{\pi}{180} \times \frac{100}{7.5}$$

$$K_{exp} = 0.430 \ \frac{rad/s}{V} \quad (1)$$

2.2.2 Calculation of Time Constant (τ)

The time constant τ represents the time required for the system to reach 63.2% of its final steady-state velocity. This value was calculated for each of the five PWM trails and averaged.

PWM Cycle	**63% Steady State Speed (deg/s)**	**Time to 63.2% (t_{63})**
30% PWM	35.83	0.4000

50% PWM	73.01	0.3200
70% PWM	93.60	0.2700
100% PWM	112.58	0.2000
Average Time Constant (τ):		**0.2975**

Table 1: Average Time constant calculation at different PWMs

Therefore, the average experimental Time constant is:

$$\tau = 0.2975\ s$$

2.2.3 Estimated Transfer Function

Based on the experimental results, the first order transfer function relating input voltage $V(s)$ to output speed $\omega(s)$ is:

$$G(s) = \frac{\omega(s)}{V(s)} = \frac{K_{exp}}{\tau s + 1}$$

Substituting the measured values:

$$G(s) = \frac{0.430}{0.2975s + 1} \quad (2)$$

2.2.4 Physical Parameter Identification

To accurately model the DC motor in Simcape, we must derive the physical parameters ($J,\ b,\ R,\ k_t,\ k_b$) from the experimental transfer function derived above.

Known System Parameters:

- Armature Resistance: $R = 5\Omega$
- Gear Ratio: $N = 172$
- Output Load Inertial: $J_{load} = 0.00976\ kg \cdot m^2$

2.2.4.1 Rotor Intertia

The simscape DC Motor block requires the inertia of the *rotor*, not the output shaft. We reflect the load inertia back to the motor shaft:

$$J_{rotor} = \frac{J_{load}}{N^2} = \frac{0.00976}{172^2} \approx 3.3 \times 10^{-7} kg \cdot m^2$$

2.2.4.2 Torque Constant (k_t) and Back-EMF Constant (k_b)

Using the relationship between the mechanical time constant and the motor parameters, we solve for k_t. Note that K_{rotor} is the gain at the motor shaft ($K_{exp} \times N$):

The motor Transfer function equation is given as:

$$G(s) = \frac{k_t}{RJs + Rb + k_t k_b}$$

$$= \frac{\frac{k_t}{Rb + k_t k_b}}{\frac{RJ}{Rb + k_t k_b} + 1}$$

Eq 1 (Gain): $K_{exp} = \frac{k_t}{Rb + k_t k_b}$

Eq 2 (Time): $\tau = \frac{RJ_{rotor}}{Rb + k_t k_b}$

If we divide Equation 2 by Equation 1, the denominator terms $(Rb + k_t k_b)$ cancels out.

$$\frac{\tau}{K_{exp}} = \frac{(\frac{RJ_{rotor}}{Rb + k_t k_b})}{(\frac{k_t}{Rb + k_t k_b})} = \frac{RJ_{rotor}}{k_t}$$

Rearranging this gives the Torque constant directly:

$$k_t = \frac{R \cdot J_{rotor} \cdot K_{rotor}}{\tau}$$

$$k_t = \frac{5 \cdot (3.3 \times 10^{-7}) \cdot (0.430 \cdot 172)}{0.2975}$$

Calculated Torque Constant: $k_t = 0.00410\ N \cdot m/A$

Based on the assumption $k_t \approx k_b$ in SI units: Calculated Back-EMF constant:

$k_b = 0.00410\ V \cdot s/rad$

2.2.4.3 Viscous Friction (b)

Using the experimental DC gain equation, we solve for the internal rotor damping b:

$$b = \frac{1}{R}\left(\frac{k_t}{K_{rotor}} - k_t^2\right)$$

Calculated Viscous Friction: $b = 0.0000077\ N \cdot m \cdot s/rad$

The open-loop hardware tests successfully characterized the motor dynamics. The resulting transfer function and physical parameters will be used to configure the simscape DC motor block for the simulations in the next section.

3 Simscape H-bridge Motor Model

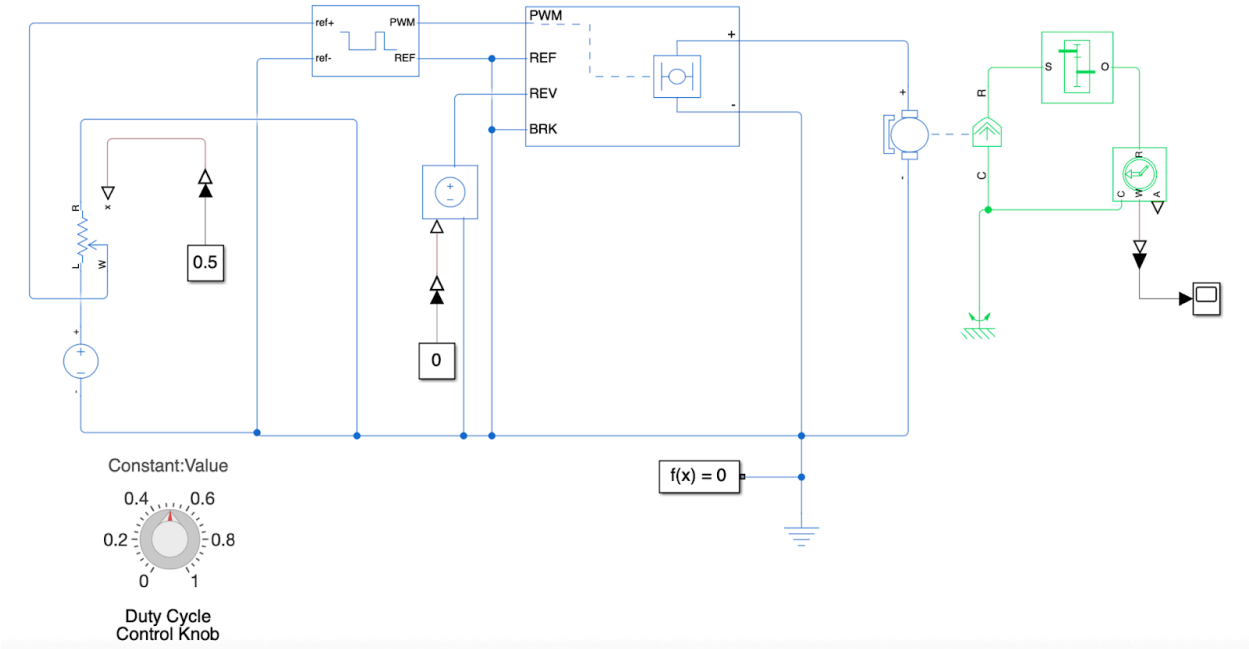

Fig 5: Simscape Model of H-bridge PWM Control w/ Motor

The Simscape model is constructed to closely replicate the hardware and electrical components of the physical system. A variable-resistance potentiometer is used to modulate the control voltage into the PWM block, which in turn scales the duty cycle according to the input voltage. The H-bridge receives the PWM signal as well as a control signal equivalent to the Arduino enable/direction pin, providing controllability over both motor speed and, consequently, motor position.

As mentioned previously, motor parameters such as the back-EMF constant, rotor inertia, rotor resistance, and rotor damping were identified through physical testing of the system. The motor is modeled mechanically in series with a gearbox to capture the transformative effect of the gear

ratio, where the rotor spins approximately 172 times for each revolution at the output shaft. Motor position and velocity are extracted from the model using a rotational motion sensor, and the resulting performance is shown in the plots below.

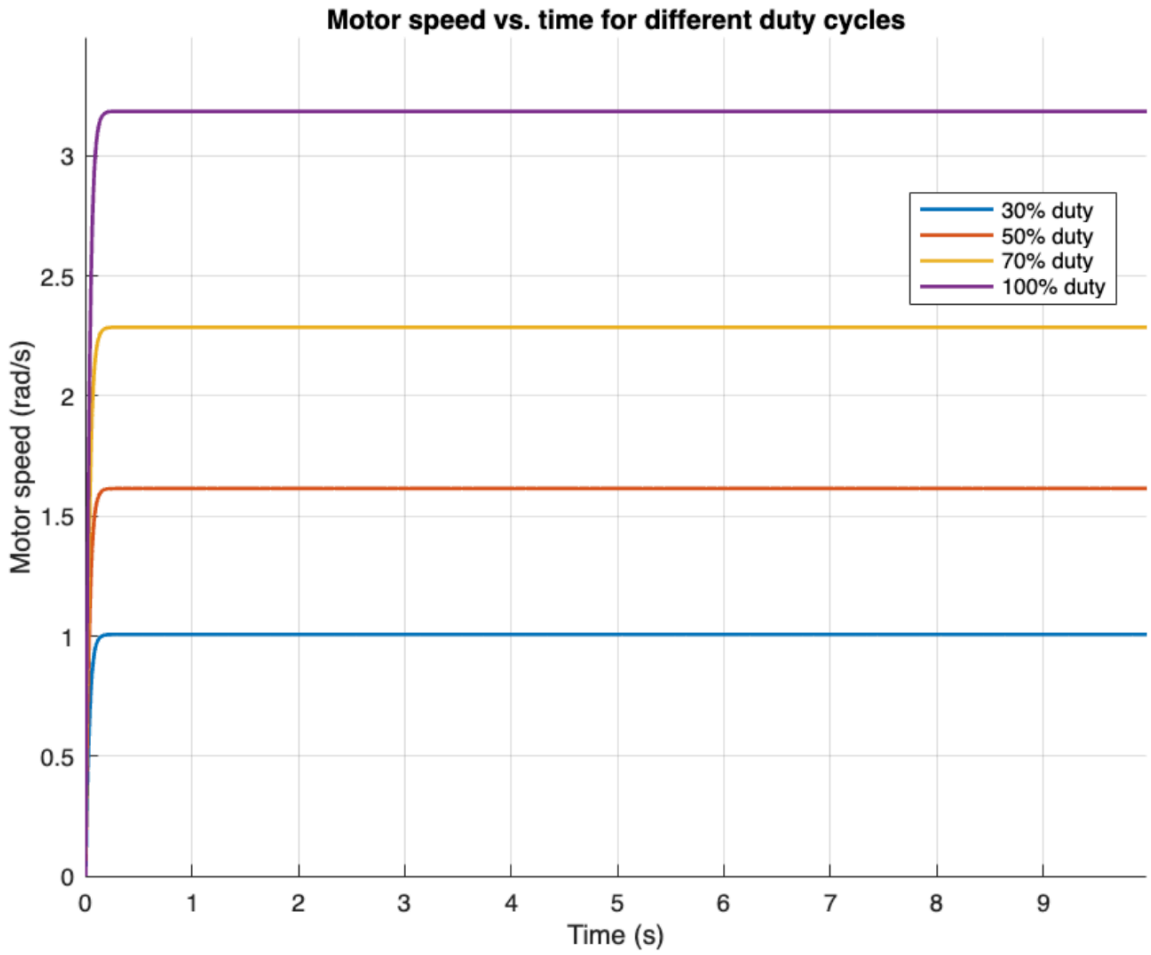

Fig 6: Angular Velocity (rad/s) vs. Time (s) of Motor at Different PWM Duty Cycle

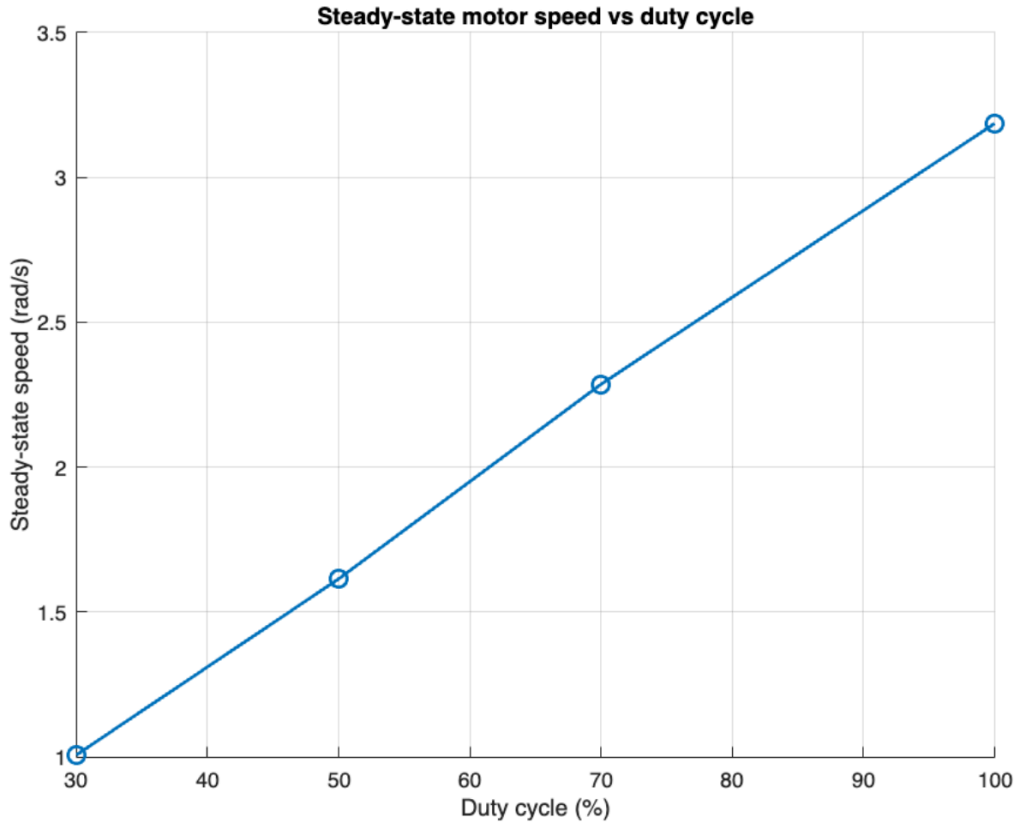

Fig 7: Relationship of Simulated Steady-state speed at different PWM Duty Cycles

From the plots above, the simulated motor behavior is consistent with hardware results but the responses appear "cleaner" due to the absence of electrical noise from surrounding equipment and wiring in the testing environment. In particular, the angular velocity response to a step in supply voltage aligns more closely with the expected first-order behavior predicted by the motor's transfer function, with smoother transients and less measurement variability than in the physical tests.

4 Comparisons

Aside from noise in the hardware testing, both models of hardware testing and simulation provided near identical results in speed and position output. Comparisons of extracted variables from both plots can be seen below:

Tested Variables	**Hardware**	**Simulation**
Time Constant	0.2975 s	0.0304 s

Gain	0.430	0.424
$\omega_{no-load}$ @ 100% Duty Cycle	3.14 rad/s	3.1854 rad/s
$\omega_{no-load}$ @ 70% Duty Cycle	2.58 rad/s	2.2862 rad/s
$\omega_{no-load}$ @ 50% Duty Cycle	2.015 rad/s	1.6148 rad/s
$\omega_{no-load}$ @ 30% Duty Cycle	0.9931 rad/s	1.0074 rad/s

Table 2: Comparison of Hardware/Simulated Motor Testing

The most notable difference between the two tests is the time the motor takes to reach steady-state speed. The physical motor requires significantly more time to saturate, which can be attributed to several factors. In the hardware system, friction from the brushes and internal surfaces of the motor creates additional resistive torque that slows the acceleration. Similarly, the gearbox introduces friction between meshing gears and modifies the effective inertia seen at the load.

In the model, it was assumed that the torque constant and back-EMF constant are numerically equal, and that the back-EMF constant at the motor shaft could be obtained by scaling with the gear ratio. Any error in estimating the torque constant or in accounting for the gear ratio when deriving the back-EMF constant directly affects the effective damping coefficient in the transfer function. If the damping is underestimated, the simulated motor will exhibit a smaller time constant and therefore a faster response than the physical motor.

The steady-state speeds at different duty cycles agree within a small fractional error between the hardware tests and the Simscape model, and the simulated no-load speeds scale almost perfectly linearly with the duty cycle, with a linear regression fit of 99.79%. This suggests that, over the tested range, the simulated motor–gearbox–load system behaves as an essentially linear first-order system.

By contrast, the physical system exhibits mild nonlinearities due to effects such as Coulomb and viscous friction, current limiting and voltage sag from the AA battery supply, and general measurement and environmental noise, all of which slightly distort the ideal response. In the linear simulated model, however, the step responses at different duty cycles share the same time constant, because tau depends only on the motor's electrical and mechanical parameters (armature resistance, torque and back-EMF constants, and equivalent inertia and damping) and is independent of input amplitude. Changing the duty cycle therefore only scales the steady-state speed without altering the underlying transient time scale.

5 Conclusion

In Part B of the project, hardware and simulated testing were able to be successfully compared with one another and observe the similarities, and key differences with both models. The hardware tests and Simscape simulations show close agreement in steady-state speeds across duty cycles, with the simulated no-load speeds scaling almost perfectly linearly with duty cycle and matching the physical data within a small fractional error. This indicates that, despite additional friction, supply limitations, and noise in the real system, the Simscape model captures the dominant first-order motor–gearbox–load behavior observed in hardware. The insights gained from comparing the hardware data and the Simscape model, together with the validated open-loop Simscape implementation, provide a foundation for Part C, where the motor will be analyzed under closed-loop control in simulation.

Appendix

[1] MATLAB script for Transfer Function Calculation (Hardware):
Part_B_Task_1_TF_calculation.pdf