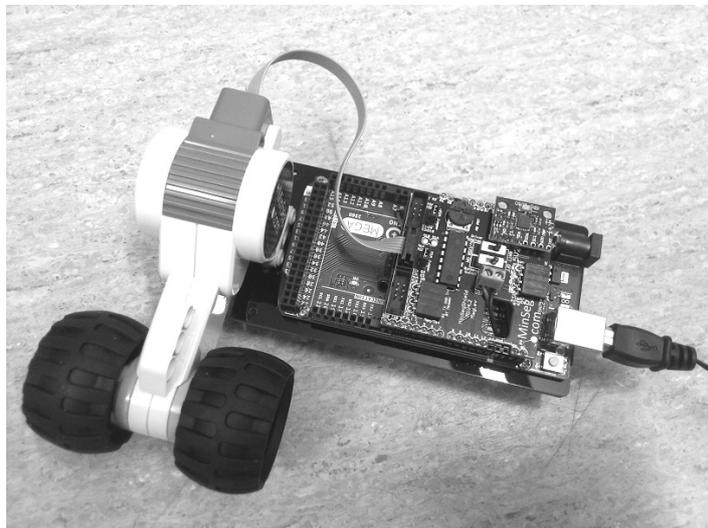


DC-motor modelling and parameter identification

This version: 2024-04-16



Name: _____

P-number: _____

Date: _____

Passed: _____

Chapter 1

Introduction

The purpose of this lab is to give an introduction to a simple dynamical system, models, signals, and introduce the MinSeg robot and associated software. By performing experiments on a small DC-motor, some of its physical parameters will be identified (also called estimated), and a first-order differential equation describing the dynamical behavior of the motor will be developed. As an alternative approach, a model is also developed using a simple step-response experiment.

DC-motors are a central part of many products and mechatronic designs, and having a model of a motor is important for dimensioning, simulations, development of controllers, performance analysis etc.



Figure 1.1: Robotic hand using 15 DC-motors to position 5 fingers independently. Having good models of the motors is crucial when developing a high-precision control system for the hand.

1.1 Hardware set-up

The lab is based on three main hardware components.

To begin with, we have a standard desktop computer. This computer is used to automatically develop and deploy code using MATLAB and SIMULINK models.

To supply power to the DC-motor and perform measurements of motor angles, we use a board with an Arduino micro-controller which runs the auto-generated code. It also communicates with the desktop computer and thus allows us to look at measured signals in real-time.

The motor we experiment with is a simple DC-motor with two wheels attached. The motor is normally part of a LEGO Mindstorms kit.

The Arduino together with the motor and wheels is called the MinSeg.

1.2 Trouble shooting

Complaints about COM port or connection when downloading to board Try again. If it still fails, disconnect USB-cable and connect it again. If it still fails, follow the instructions in the file `fixcomport.pdf`.

The motor does not run Have you reinstalled the motor jumper after current measurements? Make sure the motor cable is firmly attached.

The motor runs but no measurements Make sure the motor cable is firmly attached.

Chapter 2

Preparation

The questions below, and all questions throughout the document marked **Preparation** must be done before attending the lab. Note that there are additional preparations in Chapter 3. Written solutions to all preparations should be available upon request from the lab assistant.

The scheduled time spent with the laboratory equipment is only a small part of the complete lab, as a major part is spent on the theoretical material during preparations. When the lab starts, it is assumed you have done all preparations, understood what they are used for in the lab (when applicable), and have a clear idea of the tasks that will be performed during the lab.

Remember that the course material includes an English-Swedish dictionary in case you need it.

Preparation 1 *Read Section 2.1 and 2.6 in the course book by Ljung & Glad.*

Preparation 2 *Verify (by differentiation and checking, not by solving from scratch!) that*

$$T\dot{y}(t) = -y(t) + Ku(t) \quad (2.1)$$

has the solution

$$y(t) = K(1 - e^{-t/T})c \quad (2.2)$$

when we have an input switching from $u(t) = 0$ to $u(t) = c$ at $t = 0$ (i.e., a step-response of amplitude c), and initial condition $y(0) = 0$.

Preparation 3 *The constant K in the form (2.1) is called the **static gain** of the system and describes how much the system amplifies constant inputs in **steady-state** (i.e., when derivatives are 0 and $y(t)$ has converged to its stationary value). Given the differential equation with input $u(t) = c$ and solution as above, show that the output $y(t)$ converges*

to Kc when $t \rightarrow \infty$. Do this using two different strategies. One approach where you simply analyze the solution, and another approach where you study the differential equation and exploit the fact that you know the value of $\dot{y}(t)$ in steady-state.

Preparation 4 The constant T in the first-order differential equation in the form (2.1) is called the **time-constant** and describes how fast the system reacts to changes on the input. Given $y(t) = K(1 - e^{-t/T})c$, what value will $y(t)$ have when $t = T$?

Preparation 5 The time-constant is more generally defined as the time it takes before the output reaches a certain percentage of its final value, when the input has made a step change. To be consistent with the definition in the first-order system above, what percentage is this?

Preparation 6 Sketch the function $y(t) = K(1 - e^{-t/T})c$ for $K = 3$, $T = 10$ and $c = 2$. Particularly specify the value attained when $t = 10$.

Preparation 7 An oven placed in a room with temperature 0° is described by the differential equation $T\dot{y}(t) = -y(t) + Ku(t)$ where $y(t)$ is the oven temperature and $u(t)$ is the supplied power. In an experiment performed to identify T and K , the oven was turned on with $u(t) = 1000W$ at $t = 4$ (not at $t = 0!$), and the temperature was measured. Based on the results seen in Figure 2.1, what is the time-constant T and static gain K of the oven? **Hint: What percentage of the final temperature should have been reached T minutes after the oven was turned on?**

Preparation 8 Write $a\dot{y}(t) + by(t) = cu(t)$ in the standard form $T\dot{y}(t) = -y(t) + Ku(t)$, i.e., express the time-constant T and gain K in terms of a , b , and c .

Preparation 9 Read the complete lab-pm. There are more questions in the pm which you are supposed to complete as preparation.

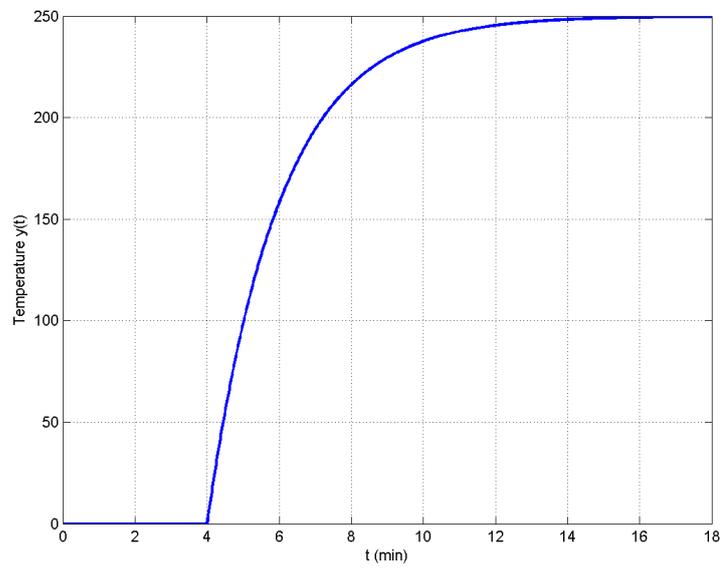


Figure 2.1: Step-response experiment of an oven. At $t = 4$ minutes the input power is changed from 0W to 1000W (a step), and the oven temperature is recorded.

Chapter 3

The lab

As explained above, the lab is primarily experimentation and data collection, using theoretical results and strategies derived during your preparation.

Items labeled **Preparation** are questions you must solve before attending the lab.

Items labeled **Task** are performed when attending the lab and you have access to the hardware.

There are two ways to keep notes during the lab (taking structured and complete notes during the lab is mandatory).

1. Print this document and take notes by pen and paper.
2. Take notes in this document electronically. To take electronic notes, you can
 - Use your own computer, but make sure your pdf reader actually handles the text forms we use in this document, and that you can save the pdf with your data. Test this now if you intend to use it. If you don't see any text form after Task 1 that you can fill in, your pdf reader does not support it.
 - Use the lab computer. This document will be available among the lab files.

3.1 DC-motor modeling and identification

Consider a standard DC-motor model as depicted in Figure 3.1.

The electrical and mechanical differential equations governing the dynamics are

$$u_A(t) = L \frac{di(t)}{dt} + Ri(t) + k_v \omega(t) \quad (3.1)$$

$$k_a i(t) = J \dot{\omega}(t) + f \omega(t) \quad (3.2)$$

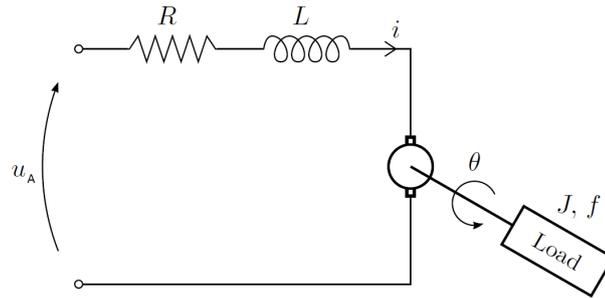


Figure 3.1: DC-motor. When a voltage $u_A(t)$ is applied on the motor, a current $i(t)$ develops and accelerates the motor. As the angular velocity $\omega(t) = \dot{\theta}(t)$ increases, the friction $f\omega(t)$ in motor and drive shafts, and back-emf (electromotiv force) $k_v\omega(t)$, reduces the acceleration $\dot{\omega}(t) = \ddot{\theta}(t)$. The back-emf voltage can be seen as the voltage you would obtain if you would use the motor as a generator instead.

where $u_A(t)$ is the voltage applied to the motor, $i(t)$ is the current in the motor, and $\omega(t) = \dot{\theta}(t)$ is the angular velocity of the motor shaft. The product $k_a i(t)$ is the torque generated by the motor, and $f\omega(t)$ is the velocity dependent friction in the motor. The parameters in the model are

- L: Motor inductance, assumed to be 0.
- R: Motor armature resistance, unknown and will be measured directly.
- k_v : Back-emf constant. Unknown and will be identified through steady-state analysis.
- k_a : Torque constant. Unknown and will be identified through steady-state analysis. Related to k_v .
- f : Friction coefficient. Unknown and will be identified through steady-state analysis.
- J: Motor inertia. Unknown and will be identified through dynamic analysis.

For a perfect motor with no energy losses, it holds that $k_a = k_v$. However, in practice this does not hold, and we know that

$$k_a \approx 0.65 k_v. \quad (3.3)$$

Our task in this lab is to perform experiments to identify R, k_v (and thus k_a), f and J, to obtain a dynamical model which relates the applied voltage $u_A(t)$ to the angular velocity $\omega(t)$. This model will be used in later labs.

3.1.1 Rotary encoder

The motor is equipped with a simple encoder¹ to measure rotation of the motor shaft. The encoder uses $n = 12$ holes with a so called quadrature design, meaning it has a resolution of 48 pulse changes per revolution. The motor shaft is connected via a gear with transmission ratio 15, meaning the encoder will deliver $48 \cdot 15 = 720$ pulse changes per motor shaft revolution, leading to a maximal theoretical accuracy of 0.5° .



Figure 3.2: Encoder in motor

Preparation 10 *One full revolution of a wheel will give a measurement from the encoder of 720. However, we wish to work in standard units of radians, and would thus like to obtain the value 2π . Which scaling gain should the measurement be multiplied with to accomplish this?*

Task 1 (Setup) *Double-click the icon **Copy minseg files** on the Desktop. Template files will be copied to a directory locally on the computer, and a link to this directory will be created on the Desktop. Open this directory and go to the directory **DCmotor**. Double-click on `template1.slx` to open with MATLAB. This SIMULINK scheme graphically describes the code that will be generated and downloaded to the Arduino micro-controller.*

¹A small light-source shines at the right-most cog in the figure, and by detecting when the light is blocked or let through the holes, rotation is detected

Some important features are marked in Figure 3.3

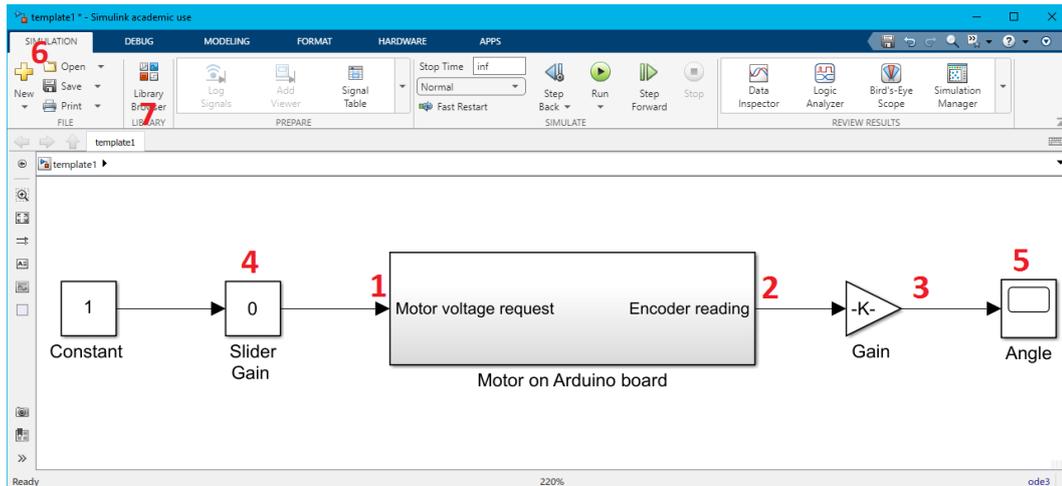


Figure 3.3: Template SIMULINK model

1. The value here will be sent by the Arduino micro-controller to the motor driver, and is our control signal $u(t)$. Due to voltage losses internally on the board in the motor driver chip, the voltage $u_A(t)$ which actually is applied to the DC-motor is lower than $u(t)$, and in this lab we will concentrate on a model from $u_A(t)$ to angular velocity $\dot{\theta}(t)$ of the wheel.
2. The Arduino micro-controller counts pulses on the encoder, and this value is used for computing the angle of the wheel.
3. We convert the number of pulses to angle $\theta(t)$ in radians. This is done in the *Gain* block using the value in Preparation 10.
4. We can also send information about parameters from MATLAB to the Arduino micro-controller in real-time. We will use a slider gain to change the requested motor voltage $u(t)$ while the code is running. We will be able to multiply the constant 1 with values between 0 and 4.5, i.e., requesting up to 4.5V.
5. When running, the Arduino micro-controller can send data to MATLAB and SIMULINK. Here, we send the scaled encoder measurements to a *scope* to plot them in real-time.
6. Make sure which tab you are at. When working with the model, the first tab is often used.
7. The different building blocks can be found in the library.

3.1.2 Identification of motor resistance R

The resistance R of the motor is obtained by measuring the resistance over the motor using a multimeter. However, we do not have direct access to the motor connection. Instead, we perform this measurement on the Arduino board. The resistance of the motor can be obtained by measuring the resistance between the two points labeled **M1A** and **M1B** in front of the motor cable, next to the current jumper, as illustrated in Figure 3.4 and Figure 3.5 (some boards look slightly different). These two points are in connection with the motor, and are placed on the board to simplify motor measurements.

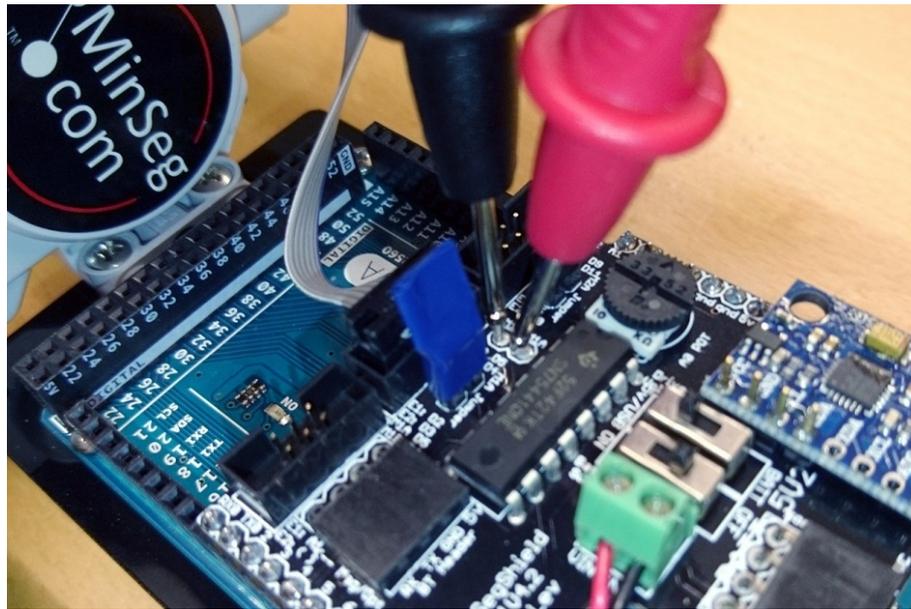


Figure 3.4: Motor resistance and voltage measurements are done between the points labeled **M1A** and **M1B** next to the motor connection (some boards look slightly different).

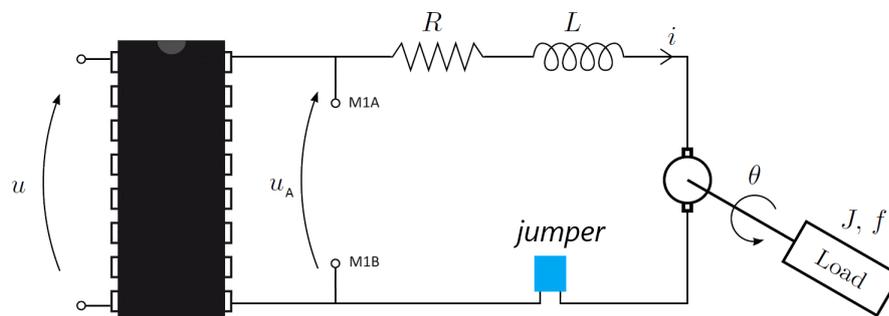


Figure 3.5: Schematic picture of the measurement points **M1A** and **M1B** and the relationship between the requested voltage $u(t)$ and the applied voltage $u_A(t)$. A large voltage loss occurs in the motor driver chip (actual pin configuration might not be correct, picture is only for illustration)

Task 2 (Resistance measurement) *Make sure the Arduino board is unpowered (USB cable disconnected). Put the multimeter in resistance mode (turn knob to Ω), and measure the resistance. Note that it takes the multimeter a couple of seconds to stabilize on a final value. You typically see a value between 3Ω and 6Ω . If you obtain a measurement of 0Ω or a very high value, check that the gray motor cable is firmly snapped into place in the motor, and that the current jumper is attached (right in front of motor cable, see Figure 3.6 where it has been removed to allow for current measurements).*

3.1.3 Steady-state measurements for identification of k_a , k_v and f

In a batch of experiments, we will study steady-state currents and angular velocities, for different constant applied motor voltages. For a constant applied motor voltage $u_A(t) = u_{ss}$, we have the corresponding steady-state current and applied voltage

$$i_{ss} = \lim_{t \rightarrow \infty} i(t) \quad (3.4)$$

$$\omega_{ss} = \lim_{t \rightarrow \infty} \omega(t) \quad (3.5)$$

Preparation 11 *Based on (3.1), which equation relating voltage, current and angular velocity holds at steady-state, and how can this relationship be used to estimate k_v by measuring these steady-state values of voltages, currents and angular velocities? You can assume R is known by now.*

To use the result in Preparation 11, we must measure the voltage $u_A(t)$ over the motor and the resulting angular velocity $\omega(t)$ and current $i(t)$ going through the motor. In the following paragraphs we explain how this is done. The actual measurements are done later!

Measuring motor voltage

The applied voltage $u_A(t)$ is measured over the same positions as the resistance measurement, see Figure 3.4 and Figure 3.5. To do this, we thus put the multimeter in DC-voltage mode (turn knob to $V=$), and measure over **M1A** and **M1B**.

Measuring motor current

To measure current, we must open the electrical circuit (voltages are measured in parallel over objects, while currents are measured in series). The board is prepared for

such a measurement. On the board just in front of the motor connection, there is a jumper which can be removed (pull it straight up) to open the electrical circuit. To measure the current, we will put the multimeter in current mode (**mA**) and close the circuit with the multimeter, as illustrated in Figure 3.6.

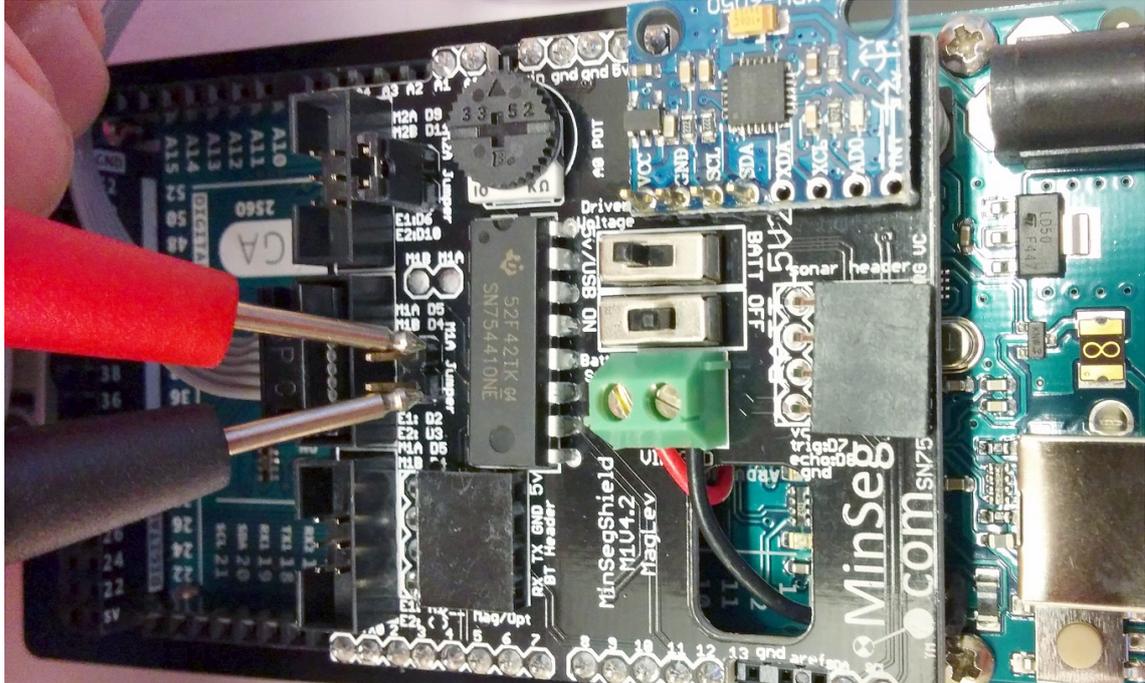


Figure 3.6: By removing the jumper next to the motor connection, the circuit is opened and we can **measure current** over the two pins. Be careful not to bend the pins by applying excessive force.

Measuring angular velocity

The encoder only gives us the angle, so we have to estimate the angular velocity by software. Given two measurements of the angle $\theta(t)$ and $\theta(t - T_s)$ where T_s is the sampling time (in our case fixed to 0.025s), the most basic estimate of the angular velocity is the Euler backwards approximation

$$\omega(t) \approx \frac{\theta(t) - \theta(t - T_s)}{T_s} \quad (3.6)$$

Unfortunately, the estimate obtained from this simple computation will not behave well when the code runs in external mode. The problem is that the Arduino computer is slow, and when it has to communicate with MATLAB, it will not be able to finish computations in time sometimes, which means that the actual sampling time will be different from the fixed value 0.025s that is used in the computation. As an effect, the derivative estimate will look noisy. We will counteract this by continuously taking the average of the last 20 derivative estimates (a low-pass filter called an FIR filter)

Open the SIMULINK component library (either by writing `simulink` in the MATLAB command prompt, or through the menu in your SIMULINK model **Simulation/Library browser**).

Task 3 (Code for velocity computation) Update your SIMULINK model to incorporate the derivative computations as in Figure 3.7. The block **Discrete Derivative** (implements the Euler approximation) and **Discrete FIR filter** are found under **Simulink/Discrete**. A so called plot **scope** can be found under **Sinks**, or you can copy the scope already available in the model. Note that the text under blocks is completely arbitrary, and you can change this. Double-click the FIR filter block and change the code in the field coefficients to `repmat(0.05,1,20)` (everything in boxed bold text!). This code will create a vector of length 20 with all elements equal to 0.05, which leads to a filter which takes the average of the 20 last values. Save your model!

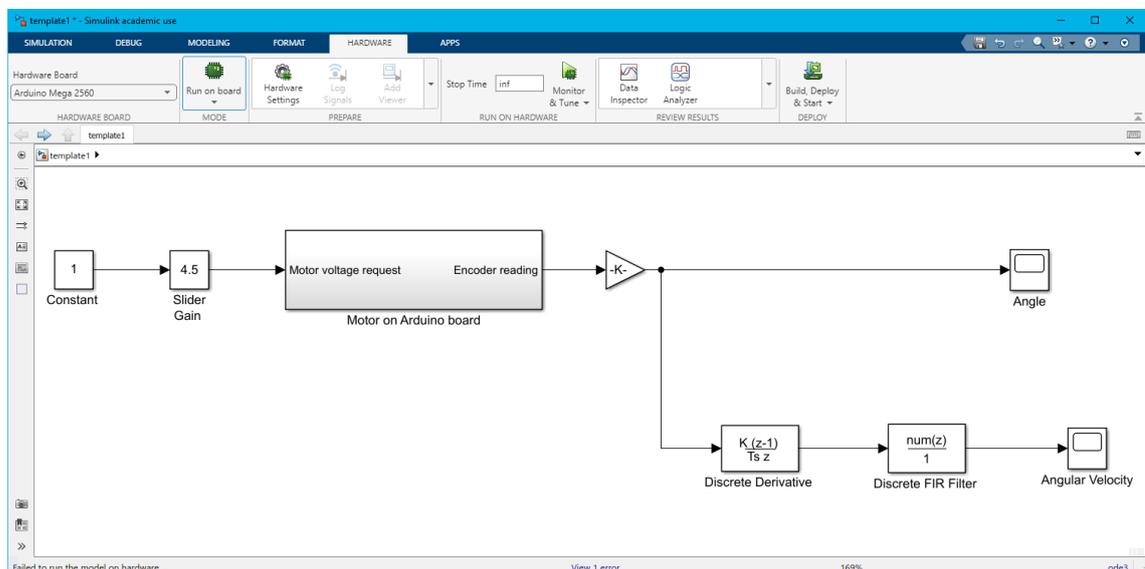


Figure 3.7: Updated SIMULINK model with derivative computation.

3.1.4 Experiments

Let us now perform the experiments. We will run the motor with different requested voltages $u(t) = 2.5V$, $u(t) = 3.5V$ and $u(t) = 4.5V$ (set in the middle edit box in the slider gain while running), and record the resulting steady-state applied motor voltages u_{ss} , angular velocities ω_{ss} , and currents i_{ss} . The obtained data will then be used to estimate k_v and f . During the following tasks, the table below will be filled out.

	Task 4		Task 5	Task 6	Task 9
Requested $u(t)$	u_{ss}	ω_{ss}	i_{ss}	k_v	f
2.5V					
3.5V					
4.5V					

Task 4 (Measurements of u_{ss} and ω_{ss}) Attach the USB-cable and download your code to the Arduino by pressing the **Monitor & Tune** button.

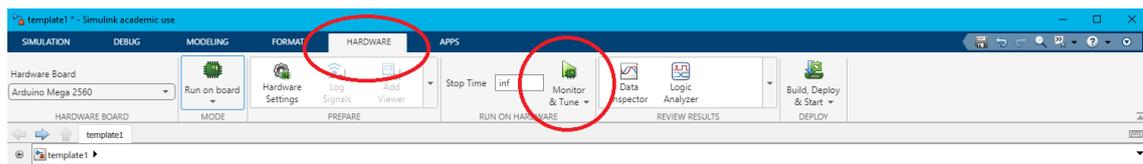


Figure 3.8: Change to the Hardware tab!

Record the steady-state motor voltage u_{ss} (multimeter) and angular velocity ω_{ss} (plot scope) with the three different requested voltages. To set the requested motor voltage in the Slider gain precisely, use the edit box. In the angular velocity plot scope figure, to see the signal better you will have to click the cog-wheel in the top left and set Axes scaling to Auto.

You should see applied voltages slightly below the requested voltages, and angular velocities up to 7 rad/s or so.

Task 5 (Measurements of i_{ss}) Remove the current jumper and repeat the 3 experiments now measuring the current i_{ss} as illustrated in Figure 3.6, and insert in the table.

You should see currents in the range 20mA to 50mA. If you see currents above 100mA you are probably measuring the wrong thing or you have the multimeter in incorrect mode.

Task 6 (Compute k_v estimates) Once you have all the measurements, you can compute estimates of the back-emf constant k_v using your result in Preparation 11. Compute estimates of k_v based on the three experiments and insert in the table (remember that i_{ss} is in the unit mA!)

Task 7 (Compute final k_v estimate) *To decide on a final value of k_v one can for instance compute the average value of the three estimates. However, a more robust approach which will guard us against a failed experiment is to take the median (i.e, middle value) of the three estimates of k_v . What is your final estimate of k_v ?*

Task 8 (Compute k_a) *With k_v available, what is the value of the torque constant k_a ?*

With all measurements available, and the estimated value of k_a , we are ready to use (3.2) to estimate the dynamic friction coefficient f .

Preparation 12 *How can f be computed from steady-state measurements i_{ss} and ω_{ss} if k_a is known, using the model (3.2)*

Task 9 (Compute f) *Use the model to compute estimates of f from the three experiments, and complete the table. Use the middle value as an estimate of f .*

3.1.5 Dynamic analysis for identification of J

Through static experiments (i.e., analysis of steady-state signals), we have managed to identify 3 out of the 4 parameters. The parameter J requires us to study the dynamic behaviour. The fact that we need dynamic analysis to estimate J is natural as it only enters our equations multiplied with $\dot{\omega}(t)$, which is 0 in a steady-state analysis.

The constant J makes it harder to accelerate the motor. The large J is (heavier wheels), the longer time it will take for the motor to reach steady-state velocity. Hence, our analysis will be based on estimating the time-constant of the system from $u_A(t)$ to $\omega(t)$.

By combining (3.1) and (3.2) we arrive at the differential equation

$$RJ\dot{\omega}(t) = -(Rf + k_v k_a)\omega(t) + k_a u_A(t) \quad (3.7)$$

Preparation 13 *Prove (3.7) from (3.1) and (3.2)*

Preparation 14 *Write the differential-equation (3.7) as a standard first-order system $T\dot{\omega}(t) + \omega(t) = Ku_A(t)$, i.e., derive the time-constant and static gain of the system (3.7), in terms of the parameters R, J, f, k_v and k_a .*

Preparation 15 *If you know the time-constant of the system, and all parameters except J, how can you compute J?*

In our final experiment, we study the transient behaviour of the velocity during a step-change in applied voltage. Open the SIMULINK model `template2` which has been prepared such that it will generate a series of steps in requested voltage $u(t)$ from 0V to 4.5V, which will lead to a series of steps in applied voltage $u_A(t)$ (with an amplitude you have recorded in your table). The (approximate) angular velocity is sent to MATLAB and recorded in a variable `data` in the MATLAB workspace.

Task 10 (Collect step-response data) *Compile, download and start the model through the Monitor & tune button. You should see and hear a sequence of steps being performed on the motor. Let it perform a couple of those, and then stop the code with the stop button (preferably when no voltage is applied). If it is stopped and the motors keep spinning, you can carefully disconnect the gray cable going into the motor, but note that it is attached with a small hook that has to be pressed on when re-inserting it (it should click)*

Task 11 (Study step-response data) *Plot the step-responses in MATLAB by running*

```
plot(data.time,data.signals.values(:))
```

Study the plot of the steps (zoom in on a single step!). What is the steady-state value of $\omega(t)$ in the steps (this should be consistent with previous experimental results in your table)? What was the applied steady-state voltage u_{ss} on the motor when requesting 4.5V according to earlier experiments?. Based on this, what is the experimentally derived static gain K from applied voltage $u_A(t)$ to angular velocity $\omega(t)$?

Task 12 *Based on the plot of the step-response of the motor, what is the experimentally derived time-constant T ?*

Task 13 *Based on the experimentally derived time-constant and previously derived motor parameters, what is the value of J ?*

Task 14 *To summarize, what does the model look like (with numerical values) if you write it as a standard first-order model in the form used in preparation 2 and 7.*

Task 15 *If you managed to supply $u_A(t) = 5V$ to the motor, what angular velocity would you expect it to end up at?*

Comments on the model and the lab

A lot of effort has been spent on experimentally identifying the physical parameters J , R , f , k_v and k_a . However, in the end, these 5 parameters are used in a first-order model which just as well can be defined directly from the experimentally obtained time-constant and static gain. In many situations, this modeling approach is just as useful, as it completely describes the input-output behaviour of the system.

A simple step-response experiment, and we have a sufficiently precise model for simulations, designing controllers, and analyze performance. Performing step-responses on systems to quickly derive models is extremely common practice, and often leads to sufficiently good models to be used for control.

The first-order model derived in this lab is used for analysis in forthcoming labs, and is part of a complete model of the MinSeg, required for the development of a balance controller.

3.2 Summary and reflections

Summarize and reflect on concepts in this lab.

Question	Answer
1. The MinSeg DC-motor is	a model a system a signal
2. The differential equation describing the MinSeg DC-motor is	a model a system a signal
3. The time-constant describes	how fast the system reacts to changing inputs which constant value the output will converge to
4. The static gain K describes	the ratio between the output and input amplitudes in steady-state the final value of the output
5. The final amplitude of the output depends on	the time-constant the input amplitude the static gain
6. A first-order linear differential equation is	uniquely defined by time-constant and static gain is not uniquely defined by time-constant and static gain

Most unclear to me is still: