1. Return DSP pre-lab and give some notes.
   - Must convert transfer functions to ratios of polynomials in order to find poles.
     - For example,
       \[ 1 + 8z^{-1} = 1 + \frac{8}{z} = \frac{z + 8}{z} \]
       and so this transfer function has a pole at zero.
     - Remember that a (tent) pole is where the transfer function escapes to infinity.
   - In s-domain, stability depends on real part of poles (i.e., \( \Re(s) < 0 \)).
     - The prototypical \( y(t) = Ae^{st} \) only decays to zero if \( \Re(s) < 0 \).
   - In z-domain, stability depends on magnitude of poles (i.e., \( |z| < 1 \)).
     - The prototypical \( y[k] = Az^k \) only decays to zero if \( |z| < 1 \).
   - Using the forward-rectangular rule to go from s to z can introduce instability.
     - The rule is
       \[ s = \frac{z - 1}{T} \quad \text{or} \quad z = Ts + 1. \]
       So the s-plane is scaled by \( T \) and shifted over 1.
     - Poles in the s-plane with \( \Re(s) < 0 \) may land outside of the \( |z| < 1 \) circle.
     - Decreasing \( T \) (i.e., increasing sampling rate) squeezes more of the s-plane into the z-domain stability circle.
     - If the Nyquist frequency (i.e., half of the sampling frequency) is close to a stable s-pole, the pole will most likely transform to an unstable z-pole.
       * You need frequencies above that pole to be attenuated.
       * They get aliased down to frequencies you care about.
       * The net effect can be amplification of frequencies near that pole.
     - Use stability, proximity to Nyquist frequency, and then pole distortion to determine whether an approximation is good or not.

2. Return DAQ lab reports and give some notes.
   - A decibel (dB) is a unit representing the ratio of two power gains.
   - Power is squared magnitude.
     - Actually, it’s squared absolute magnitude.
     - Compare to vector norms/inner products.
   - So the formula for dB is
     \[ \text{dB} \triangleq \frac{10 \log_{10} \text{Bel}}{\text{Ratio of Power}} = 10 \log_{10} \frac{\text{magnitude}_a}{\text{magnitude}_b} = 10 \log_{10} \left( \frac{\text{magnitude}_a}{\text{magnitude}_b} \right)^2 = 20 \log_{10} \frac{\text{magnitude}_a}{\text{magnitude}_b} \]
   - Use units, axes labels, and semilogx.
3. Linear Time-Invariant (LTI) Systems in the Time Domain

- Impulse response.
  - As with frequency domain, can break signals into sums of simpler components.
  - Instead of using one frequency as a test signal, use one time.
  - By applying an impulse at zero, it’s like applying every frequency simultaneously.
    * Recall characterization of “Black Box” from first lab.
    * An impulse is a useful one-shot probe of every frequency of a system.
  - Each impulse excites transients. If system is stable, each transient dies out.
  - Response to signal is sum of every transient response (this is the convolution sum).

- Step response.
  - An impulse is difficult to realize, but step response is easy to generate and used often.
  - In fact, to find the impulse response from a step response, simply differentiate.

- Recall that signals can be written as sums of (complex) exponentials.
- Recall that differentiating or integrating an exponential results in more exponentials.
- So step responses and impulses responses capture the same information.

4. Physics-based first-order DC motor model \(v_{\text{motor}} = K_m \omega_{\text{motor}}\) and \(v_{\text{motor}}i_{\text{motor}} = \omega_{\text{motor}}\tau_{\text{motor}}\).

5. Complete the Time Domain System Identification for a DC Servo lab

1. Find step response of a motor that is approximated by a first-order LTI system.
   - Such systems characterized by rise time. Given rise time, can find pole and vice versa.
   (i) In MATLAB, change Termination settings for DAC block.
     - Under Termination tab, check box to set 0 V stop value.
   (ii) Using dSPACE, use two plotters to plot theoretical and measured outputs.
   (iii) In dSPACE, add a simState control.
     - Wire simState to 2-option radio button.
     - Label one option Run and give value 2.
     - Label other option Stop and give value 0.
     - Set Capture Settings to automatically restart and set capture time to simulation time.
     - Restart simulation as needed by using simState control (i.e., no need to change modes).
   (iv) Save step responses (both theoretical and measured).
   - Use measured step response to estimate rise time (in MATLAB for report).

2. Allow the motor to turn one revolution to find tachometer gain (i.e., to calibrate).
   (i) In Simulink, change stop time to 2 s and step time to 1 s.
   (ii) In dSPACE, create two numeric inputs to control step Time and After Value.
   (iii) At start of each iteration, align motor.
   (iv) Change step time or after value until motor travels exactly one revolution.
     - Keep after value between 0.5 V and 4 V.
   (v) Save single revolution data.
   (vi) Calculate average angular velocity with \(2\pi/t_{\text{stop}}\) where \(t_{\text{stop}}\) is the time of one full revolution.
   (vii) Using MATLAB’s mean function, calculate average tachometer signal after the step starts.
     - Make sure you truncate the data by removing pre-step part.
     - Recall \(v_{\text{tach}} = K_{\text{tach}} \omega_{\text{motor}}\). Hence, \(\text{mean}(v_{\text{tach}}) = \frac{1}{t_{\text{stop}}} \int_0^{t_{\text{stop}}} v_{\text{tach}}(t) \, dt = \frac{1}{t_{\text{stop}}} \int_0^{t_{\text{stop}}} K_{\text{tach}} \omega(t) \, dt\)
     \[\frac{K_{\text{tach}} t_{\text{stop}}}{t_{\text{stop}}} \int_0^{t_{\text{stop}}} \omega(t) \, dt = \frac{K_{\text{tach}} t_{\text{stop}}}{2\pi},\] and so \(K_{\text{tach}} = \text{mean}(v_{\text{tach}}) \times \frac{t_{\text{stop}}}{2\pi}\).

As you complete the lab, compare motor response to expectation/theory. Other tips:

- Do work out of directory on local hard drive — use as MATLAB working directory.
- In Simulink, the hotkey for building a model is Ctrl - B.
- Start dSPACE ControlDesk before doing Simulink builds.