MEEN 432 –Automotive Engineering

Fall 2026

Instructor: Dr. Arnold Muyshondt

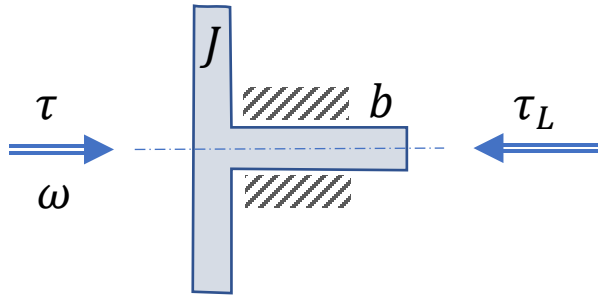
Acknowledgement: Most of the material for this class was developed by Dr. Swami Gopalswamy

Lecture 3a: Using Matlab Simulink for Simulation of Dynamic Systems

- Demo of Simple Example

Simple Shaft revisited

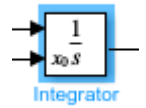
- Consider the simple mechanical shaft:
  - $J\,\dot{\omega} = \tau - \tau_L - b\,\omega$
- τ, τ_L are inputs to the component
- ω is a state, and an output of the component
- J, b are parameters of the component



- Also recall the analytical solution to the evolution of the speed of the shaft, for a step in applied torque of τ_0, assuming load torque is zero, starting from an initial angular velocity ω_0
  - $\omega(t) = \frac{\tau_0}{b}\left(1 - e^{-\frac{b}{J}t}\right) + \omega_0 e^{-\frac{b}{J}t}$ (how do you get this?)

General Approach (1/2)

- Express the equations of motion as a "first order" ODE, with state derivatives on LHS:
  - i.e., of the form: $\dot{x}(t) = f(x(t), u(t))$.
  - For the example this becomes: $\dot{\omega} = \frac{1}{J}(\tau - \tau_L - b\,\omega)$
  - Note:
    - x could be a vector
    - When higher-order derivatives are present, they need to be reduced to first order by introducing additional state variables
- Use Matlab primitives (either block diagram or Matlab Function) to calculate the RHS
- Use the "integrator block" to integrate the RHS



General Approach (2/2)

- Avoid “hard coding” parameter values whenever possible – assign variable names
- Use a script file to “initialize” the parameter variables in the Matlab “workspace”
- Save all dynamic variables of interest using “To Workspace” blocks in Simulink
- “Simulate” the system with appropriate test signals that reflect the scenarios of relevance
- Use a script file to “plot” all dynamic variables of interest
  - You can use “scopes” to look at dynamic variables during development for debugging purposes