Effects of Actuator Saturation

- **Saturation** occurs in a motion control system when any of its components reach their maximum capacity.
  - The electro-hydraulic valves used in this course accept input of ±10 volts to open their ports completely and develop maximum flow rates.
  - DC electric motors have similar voltage limitations.
  - Aerodynamic stall of a wing or flap occurs when the angle of attack is too large.

- **Saturation** of components is often undesirable
  - Components are operating at maximum capacity
  - System behaves as an open-loop system, because the control signal does not vary with the feedback.

- **Saturation** often occurs in a control loop when a step position command is given due to the large error when the command is initiated.

- Saturation is usually not accounted for when the proportional control gain is determined. The model below is used to study how big an effect saturation will have on system performance.

- **With saturation,** the system will be slower than the system without saturation.
Use of Pre-Filters to Soften the Step Input

- The model below employs a normalized 2\textsuperscript{nd} order pre-filter to make the input to the control loop rise more slowly and smoothly than the step command.

- The net effect of the pre-filter is to:
  - \textit{Lessen} the position error at any time, and hence avoid saturation
  - \textit{Slow down} the system response

- Note the original design gain is used in the control loop so it functions as designed.

Use of a Motion Trajectory

- A \textit{motion trajectory} is another method of supplying a softer and smoother input than the step function. As with the pre-filter, the system response is slower.

- In the model below, a \textit{signal builder} is used to generate a simple acceleration profile which is then integrated twice to generate the position command.

- Similar results are obtained as with the pre-filter; however the trajectory is calculated directly to produce a desirable motion input for the system.
Use of a Motion Trajectory with Velocity and Acceleration Feedforward

- Using a *motion trajectory* as described above allows access to not only the *position* commands, but also the corresponding *velocities* and *accelerations*.

- Velocity and acceleration values can be *fed forward* into the control loop to *anticipate* the ensuing position commands.

- If carefully designed, the result is a system that *follows* the *command trajectory* with *little error without saturating* the system.