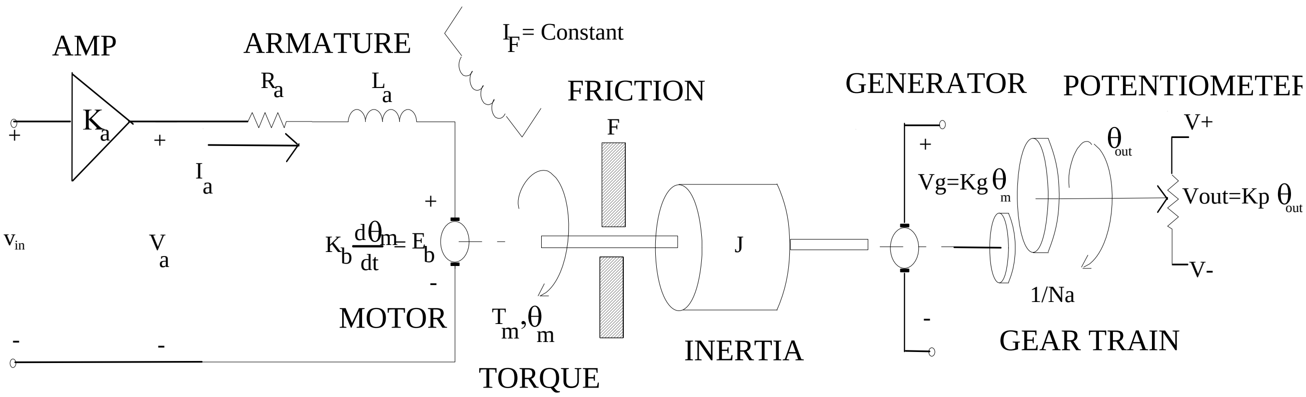
Objective: To study the effects of discretization and to design a feedback controller for the Motomatic DC Servo Motor
Prelab: (Counts as HW#10) Due Monday, February 22
Consider the following functional block diagram for our Motomatic armature-controlled DC servomotor in our Lab:



If you took EE571, you found the open-loop transfer function of the Motomatic to be:

$$\frac{V_{out}}{V_{in}} = \frac{210.9}{s^2 + 5.27s}$$

where V_{out} was the voltage off the position sensor, which was directly proportional to the shaft displacement, θ_{out}, by the relationship $V_{out} = K_p \theta_{out}$. We also discovered that the voltage from the generator (tachometer), V_g, was related to the derivative of V_{out} by the following relationship: $V_g = K_g \dot{\theta}_m = K_g \frac{\dot{\theta}_{out}}{N_a}$ where K_g is the generator constant and N_a is the gear ratio.

In EE571, we found the values of K_p =5.139 volts/rad, K_g =0.1842 volt/rad/sec, and N_a=1/9, and obtained the following continuous time state variable model for the Motomatic using $x = \begin{bmatrix} V_{out} \\ V_g \end{bmatrix}$: $\dot{x} = \begin{bmatrix} 0 & 3.0999 \\ 0 & -5.27 \end{bmatrix} x + \begin{bmatrix} 0 \\ 68.0346 \end{bmatrix} V_{in}$ with the output $y=V_{out}$

1.a) Is the continuous model stable, marginally stable, or unstable?

b) Discretize the model using $\hat{A} = e^{AT_s}$ and $\hat{B} = \int_0^{T_s} e^{At} B dt$ with T_s=10 msec.

c) Is the discrete model stable, marginally stable, or unstable? Is it controllable?

d) Using this discrete model, design a feedback **regulator**, $V_{in} = -Kx_k$, such that the Motomatic position, $y=V_{out}$, has no overshoot and "settles" in less than 0.3 sec. (please pick distinct eigenvalues so we can decouple the closed-loop system in part 2b)).

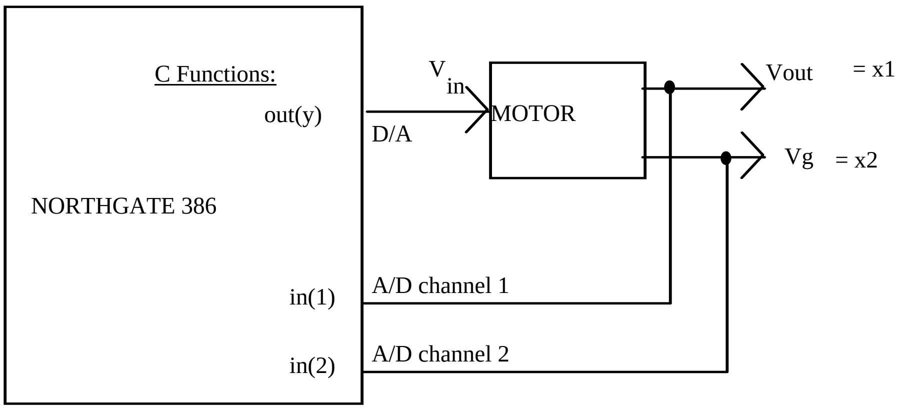
2.a) Use MATLAB's dlsim() function to simulate your regulator design in response to a 5 volt step (i.e., let $V_{in}=5-Kx_k$. What is the settling time? What is e_{ss} (i.e., the steady-state error, 5-V_{out})? (Note: If you have access to SIMULINK, you may use it instead of dlsim())

b) Recall that the settling time measures how fast the closed-loop **decoupled states** decay! Make a plot of the **decouple states** to verify the settling time (you can find the closed-loop decoupled states using the MATLAB command, z=inv(P)*x' where P is the matrix of eigenvectors of the closed-loop system, $\hat{A} - \hat{B}K$).

c) Now design a controller so that V_{out} reaches 5 volts in steady-state while keeping the specifications of part 2a) (that is, find N_x and N_u)

d) Repeat parts 1d) through 2c) until all the specifications are met.

f) Use the in(), out(), and sleep() functions from Lab #1 to write a C program to implement your controller. Let us assume that we have two channels on our A/D converter and that in(1) will sample channel 1 where $x_1 = V_{out}$ is connected and in(2) will sample channel 2 where $x_2 = V_g$ (see below)



In the Lab: (Counts as HW#11) Due Wednesday, February 24

1. Test your **regulator** design by choosing the regulator section of the menu, a sampling time of T_s=10 msec, and an input of 5 volts. Obtain a plot and compare to Prelab results.
2. Test your **controller** design by selecting the controller option and by entering the correct values of N_u and N_x. Compare your results to the Prelab.
3. Can your closed-loop system follow a ramp? Repeat part 2 but select a 5 volt ramp input. Does your system follow a ramp?
4. How robust is your design? Repeat part 2, but start with the prony brake on the flywheel. Does your system respond well to the torque disturbance?
5. Study the effects of altering the sampling time on your system by repeating part 2 using a sampling time of 40 msec and then 80 msec (do not redesign your controller).