EE443L Lab 6: DC Motor Trajectory Tracking

Introduction

Many applications require a motor to accurately track a position profile that varies smoothly over time rather than a sharp transition from one fixed value to another. This type of control objective is referred to as trajectory tracking and can serve two purposes. The first application of trajectory tracking is for tasks such as robotic welding, robotic spray painting, or radar target tracking where a motor driven system needs to be precisely positioned for each instant of time. The second usage is for transitioning a motor driven system such as a large antenna from one fixed position to another without the sudden motions and excessive control action often required for following a step transition. This lab investigates the use of proportional and integral (PI) position control along with proportional (P) velocity control for DC motor trajectory tracking as shown in figure 1. Making use of both position and velocity information in the control system gives the controller more information about the task and should improve tracking accuracy.

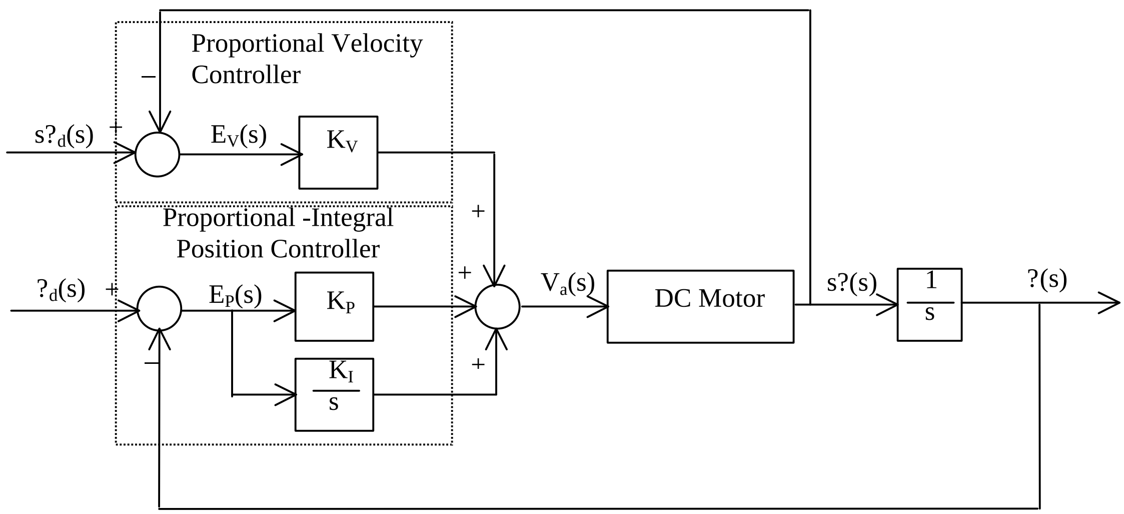

Figure 1: Proportional-Integral Position plus Proportional Velocity Control of a DC Motor

Procedure

1. Trajectory generation is the first step in tracking control and many techniques exist for connecting points with a smooth curve. One of the simplest is to use sines and cosines for desired position $q_d(t)$ and velocity $\dot{q}_d(t)$ profiles as shown

$$
\begin{aligned}
q_d(t) &= q_0 + \left(\frac{q_f - q_0}{2}\right)\left(1 - \cos\left(\frac{pt}{t_f}\right)\right) \\
\dot{q}_d(t) &= \left(\frac{q_f - q_0}{2}\right)\left(\frac{p}{t_f}\right)\sin\left(\frac{pt}{t_f}\right)
\end{aligned}
\tag{1}
$$

where q_0, q_f are the initial motor position and desired final motor position, respectively, and t_f is the final (stop) time of the trajectory assuming the start time is zero. This method of trajectory generation has been implemented in both simulation with the related simulink and matlab files lab6.mdl and lab6.m, respectively, as well as the LabVIEW VI lab6.vi where zero initial motor angle, i.e., $q_0 = 0$, is assumed. These files can be found in the network directory N:\EE443L\Lab6\. Run lab6.m in matlab to load the simulation constants and view the control system transfer functions, then run lab6.mdl in simulink to simulate the DC motor control of figure 1 and view the trajectories generated. Note that $q_d(t)$ transitions smoothly from 0 to q_f and $\dot{q}_d(t)$ begins and ends at zero allowing the motor to be started and stopped at rest.

2. As mentioned above the matlab and simulink files provided simulate the Yaskawa DC motor under the PI position and P velocity control scheme of figure 1. Change the motor parameters in lab6.m to those you have previously determined for your motor and run the m-file to see the system transfer functions $sq(s)/sq_d(s)$ and $q(s)/q_d(s)$ as well as their pole-zero diagrams. Then run lab6.mdl in simulink to view the total control system's performance. Vary the controller gains K_V, K_P, and K_I until a desirable pole-zero diagram and system response are achieved. Print this response and note the corresponding controller gains.

3. Download the LabVIEW VI lab6.vi (and its associated sub VIs) that already contains all motor input/output functions (speed in, position in, and PWM out) and the trajectory generator of equation 1. Look carefully through lab6.vi to make sure you understand its components and layout and then run it to see the implemented trajectory generator in action noting the PWM value remains fixed.

4. Add the PI position and P velocity control algorithms shown in figure 1 to lab6.vi to implement the control approach on the DC motor. Implement the integral as simple Euler integration shown in equation 2 noting that shift registers will need to be used to initialize and compute the sum. Use the online help for more information on shift registers and use gains K_V, K_P, and K_I determined in the simulation. Make any necessary adjustments to the gains to further enhance performance and print the final VI and motor position and velocity responses. Comment on any differences between simulated and actual responses, problems, or unforeseen difficulties.

$$\int_{t=0}^{t} e(t)dt \approx \sum_{i=0}^{n} e[i]\Delta t \qquad (2)$$

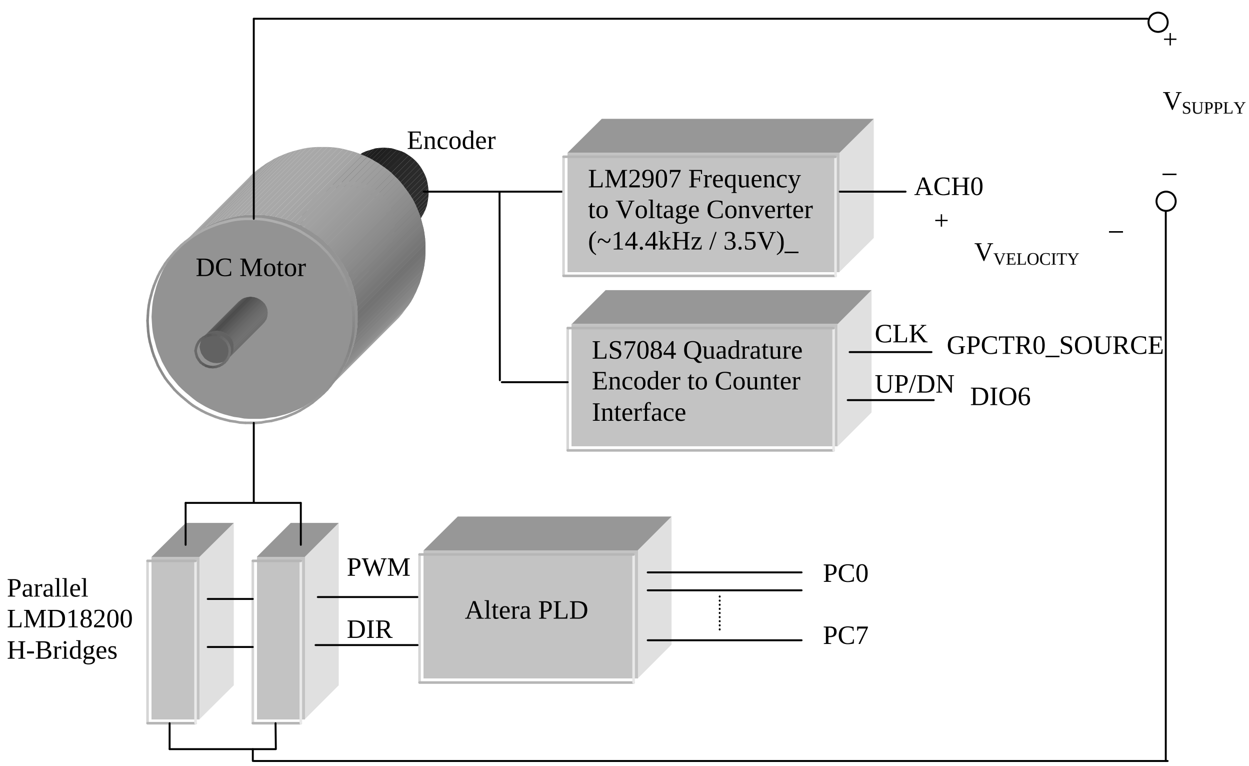

Figure 2: DC Motor Experiment Setup