Tips for Self-Erecting Inverted Pendulum

Written by Justin Hsia
(last updated 11/30/2009)

- At the start of any hardware run, all encoders are reset to 0. For the balancing labs, you held the pendulum vertical to set $\theta = 0$ to be the up position. In this lab, you start with the pendulum down, so that will be set as your new $\theta = 0$ position.

- Similarly, wherever you position the cart at the start will be set as the new $x = 0$ position. To allow the most room for movement, start the cart as close to the middle of the track as possible.

- Please keep in mind that you have a limited amount of track.

- Don't forget your units! Unlike previous labs, what units are we using for distance?

- If your block diagram starts to get too messy, you can always clean things up by using subsystems (i.e. the observer).

- Try your best to characterize the desired behavior with system characteristics/state variables. For example, I want the cart to be moving _____ when the pendulum is swinging _____ and not near vertical. You can translate this to conditions on desired x, current θ, and current $\dot{\theta}$.

- The balancing controllers do both angle and position control – they try to set $\delta\theta = 0$ and $\delta x = 0$. (I use the delta notations here as a reminder that these might not be the exact θ and x that you read from the hardware) If you are running a balancing controller and the motor spins freely (stripped gear, lifted off the track, etc.), the controller may stop working because the error in position may overpower any error in pendulum angle.

- The counts-to-radians conversion is not exact. Before this wasn't an issue because we just wanted to settle at $\theta = 0$, which was where the simulation started. In this case, we are balancing at $\theta = \pm 180°$, so it does make a difference since the balancing controller will not work properly if it is trying to balance the pendulum at a significantly non-vertical angle. It might be worth your time to tweak this conversion factor manually.