Ander’s Slides:

Our control design project faces some major obstacles. We are dealing with an autonomous non-linear system that will dynamically change while it runs. The advantages of our design are that it is a controllable system that we will control in discrete time.

For our purposes, we will take the non-linear system and linearize it so we can use a state variable description. The advantage of using a state variable approach is that it will allow us to place poles in our system instead of relying of pole-zero cancellations.

Based on the flow diagram of a basic discrete time state variable feedback system we can obtain the state variable equations. We will have to model the system’s plant, modify any initial conditions, place poles in the system, and allow the system to reach a stable balancing state.

{Pause}

LabVIEW’s built-in examples have given us an idea of how to tackle this particular problem. This example simulates an inverted pendulum cart in a LabVIEW environment. However, some of the major control strategies for a state variable feedback system are incorporated. We are aiming to have a finished product that follows this example.

1. The controls simulation can take vectors A, B, C, D that are particularly tuned to plant.
2. Given the state vectors, LabVIEW can compute the system poles
3. This LabVIEW simulation can also calculate the system poles based on the desired input poles that the user can enter.
4. We should also be able to set any initial conditions for our system, be able to force a change on the pole or apply an external force to our system, and be able to apply a prefilter to our design for error reduction.
5. Given these user inputs, the LabVIEW algorithm will them output a linearized plant model and plot it the real-time changes in a plot for ease-of-use.

{Pause}

The front panel GUI in LabVIEW will allow the user to see the real-time simulation of the model and the real-time output of the linear plant model. The user can also apply a virtual external force to the simulated model and observe how the system attempts to retain a balanced state and observe the real-time numerical changes.

Other features that we would like to have in our front-panel include adding poles to the system and to allow easy access to the state variables so that we can modify them as we test.