Matthew Harter
Travis Stanger

ENGR 454 – Project Milestone 8

Due: 5/12/2021

Process:

To place poles with LQR, we simply had to define the Q and R matrices in Jupyter Lab with the Octave Kernel and then use the lqr() function as shown to the right. We tried several different values for Q and R , and found that those shown here seem to work the best. It also might be good to note that these gain values are somewhat similar to those found when we "hand-placed" the poles (similar ballpark values), but the gain affecting angular velocity is a bit lower as is the gain affecting position.

LQR Pole Placement & Gain Calculations

```
% Place Poles with LQR

A = [[0, 1];[-M*g*l/J, -0.8]];
B = [0; l/J];

Q = [[0.8 0];[0 0.01]];
R = [10];

G_lqr = lqr(A, B, Q, R)
```

```
G_lqr =

   0.136183   0.050262
```

Results:

The values we found using the lqr() function were tested by plugging them into the Arduino code from hand-placing the poles. We found that the system responded and settled more smoothly and quickly, coming down faster than when testing with the gains found with hand-placing the poles—within about 2 to 2.5 seconds (see included video in the folder).