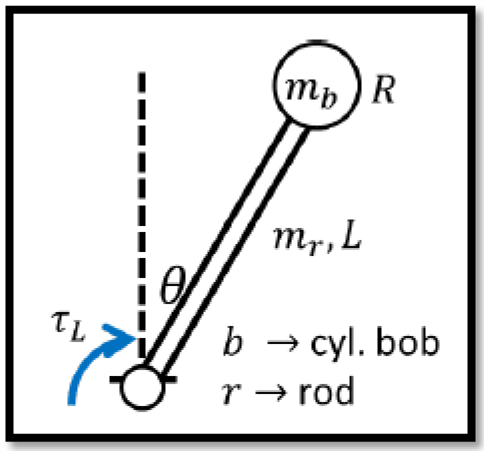

Project 3: Digital Control of an Inverted Pendulum

Applied Mechatronics (MECH 5970-019)

April 6, 2022

Walter Livingston   Matthew Covington

wbl0009   mac0139

Table of Contents

Table of Figures

Supporting Calculations

The calculations for this project were done entirely within MATLAB as to provide ease of use when updating values or testing various design changes. The code shown in Figure 1 is for the ideal system. All system constants and ideal specifications are defined and utilized to design a controller for the actual system.

IDEAL SYSTEM

```
clear; clc; close all

% System Constants
g = 9.81; % m/s/s
L_r = 101.6/1000; % m
m_r = 18.14/1000; % kg
m_b = 32/1000; % kg
r_b = 6.35/1000; % m

% Motor Constants
K_t = 25.9/1000; % N.m/A
K_A = 0.3; % A/Vcom

% Moments of Inertia
J_b = 1/2 * m_b*r_b^2; % kg.m^2
J_r = 1/3 * m_r*L_r^2; % kg.m^2
J_m = 33.5/(1000*100^2); % kg.m^2

% Second Order System Coefficients
m1 = J_r + J_b + m_b*(L_r + r_b)^2 + J_m; % kg.m^2
m2 = (m_b + m_r/2) % kg

% Given Ideal Specifications
zeta = 0.707;
ts2 = 0.5; % sec

% Natural and Damped Frequencies
w_n = 4/(zeta*ts2); % rad/s
f_n = w_n/(2*pi); % Hz
w_d = w_n*sqrt(1-zeta^2); % rad/s
f_d = w_d/(2*pi); % Hz

% Ideal System and the Derivative System
ideal_sys = tf(K_t/m1,[1, 2*zeta*w_n, w_n^2])
s = tf('s');
p_deriv = ideal_sys * s;

% Maximum change in Velocity and Voltage
max_vel = max(step(p_deriv)) % rad/s
max_dV_dt = max_vel * 180/pi * 1/70 % Vmeas/s
```

Figure 1: Ideal System Calculations in MATLAB

Continuous Domain FSF Gains

A digital controller utilizes various gains to condition an input response to an output control effort. Proportional (K_p) and derivative (K_d) gains were used to achieve this task. Based on the ideal closed loop system shown in Figure 1, K_p and K_d were calculated to make the system meet the ideal design specifications. Figure 2 shows these calculations within MATLAB. Although, before being altered by control gains, the input signal needed to be conditioned so that there was a known relationship between measured voltage and motor position as well as motor velocity. This was achieved with a position gain of 70 and a velocity gain of 5.143 as were calculated in Figure 2. For the lower sampling rate experiment, the position gain was changed to 180 after observation. This change was made due to the DC motor not receiving enough voltage to bring the pendulum upright. Given the known relationship between voltage, position, and velocity, the dynamic system gains were converted into usable gains denoted as K_theta and K_theta_dot.

CONTROLS

```
% FSF Gains for Continuous System
K_p = (m1*w_n^2 + m2*g*L_r)/K_t
k_d = 2*zeta*w_n*m1/K_t

% Position and Velocity Gains
K_pos = 350/5; % deg/Vmeas
K_vel = 10/max_dV_dt % Vcom/Vmeas/s

% Voltage Gains
K_theta = 1/((K_p * pi) / (K_t * K_A * 180))
K_theta_dot = 1/((k_d) / (K_t * K_A * K_vel))
```

Figure 2: Control Calculations in MATLAB

Amplification Decision

The National Instruments myRIO-1900 used in this project has a 12-bit Analog to Digital Converter (ADC). Based on Equation 1, the smallest increment of voltage change, from -5V to 5V, that can be detected by the myRIO is approximately 2mV. This seemed reasonable enough to use without amplifying to the full -10V to 10V range the analog terminal can read.

$$\Delta = \frac{R}{2^n}$$

Equation 1: Quantization Equation

Although not required, many solutions to achieve better performance were explored in this project, and a 1.5x amplification stage was added to the input at a time. This was achieved using an inverting op-amp. This op-amp circuit was made up of an LF412 op-amp, a 1 kilohm resistor, and a 1.5 kilohm resistor. The op-amp circuit was used in tandem with an anti-aliasing filter discussed in a moment. The schematic for the full circuit can be seen in Figure 3.

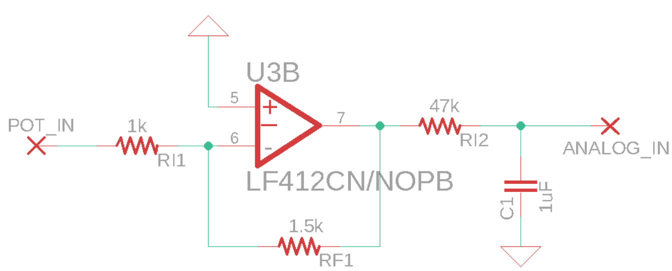

Figure 3: Anti-Aliasing Filter & Op-Amp Circuit

Digital Calculations

The pendulum was operated at two different sampling rates. These sampling rates were chosen to be 10-times and 30-times the natural frequency of the open loop system, which came out to be approximately 18Hz and 54Hz, respectively. With these sampling rates, the period may be calculated, as seen in Figure 4. The 18Hz sampling rate had a period of 55.53 ms, while the 30 Hz sampling rate had a period of 18.51 ms.

DISCRETE MATH

```
% Lower Sampling & Nyquist Frequency
w_s1 = 10*w_n; % rad/s
f_s1 = w_s1/(2*pi); % Hz
w_nyq1 = w_s1/2; % rad/s
f_nyq1 = w_nyq1/(2*pi); % Hz
T_s1 = 1/f_s1 * 1000 % ms

% Higher Sampling & Nyquist Frequency
w_s2 = 30*w_n; % rad/s
f_s2 = w_s2/(2*pi); % Hz
w_nyq2 = w_s2/2; %rad/s
f_nyq2 = w_nyq2/(2*pi); % Hz
T_s2 = 1/f_s2 * 1000 % ms
```

Figure 4: Digital Calculations in MATLAB

Anti-Aliasing Filter Design

As mentioned previously, multiple solutions were pursued to achieve the desired performance. One of these solutions was an anti-aliasing, or low-pass, filter. Initially, the myRIO was reading a lot of noise on the input signal from the Midori (potentiometer). In attempts to abate this, an anti-aliasing filter was added to the input signal coming from the Midori. The diagram for this circuit was shown previously in Figure 3. This circuit had a cutoff frequency of approximately 3.4Hz and caused a significant amount of phase shift. The reason for choosing this cutoff frequency was that frequencies around 1.8Hz were desirable, and anything higher was considered noise. Given the large phase shift though, a different design was attempted. This design was attempted without an op-amp, with a cut-off frequency of approximately 58Hz. This cutoff frequency was around the highest that could be achieved without picking up 60Hz noise from the environment. This did not improve performance in a meaningful way, so the circuit was removed from the design.

FILTERING

```
% Math for attempted Anti-Aliasing (Low Pass) Filter
% Proposed Cut-Off Frequency (Less than 60Hz noise)
f_c = 50; % Hz
% Constant Capacitor Value
C = 1e-6; % F
% Theoretical Resistor Value
R = 1/(2*pi*f_c*C) % Ohm
% Actual Resistor Value Chosen
R_a = 47e3;
% Actual Cut-Off Frequency
f_ca = 1/(2*pi*C*R_a)
% Phase shift associated with Anti-Aliasing Filter
phase_shift = -atand((2*pi*f_ca*R_a*C))
```

Figure 5: Filtering Calculations in MATLAB

Digital Controls Code

This project was completed utilizing a National Instruments myRIO. This device is programmed using the LabView programming environment. The code utilized in this project is shown in Figure 5. In this code, we have a choice of two sampling frequencies that we labeled as “10x Period (ms)“ and “30x Period (ms)”; as the Wait VI within LabView takes milliseconds as an input. These periods are calculated by the MATLAB script in Figure 4. This period value is then fed into the Wait VI as well as into the differentiation for the velocity calculations. The velocity was calculated using a feedback loop that essentially stores the last analog in value and allows the programmer to use it in the next loop iteration. The previous value was subtracted from the current value and this difference was divided by the currently used period. The velocity and position values are fed through various gains and offsets to better condition the inputs, and are then multiplied by their respective control gains; these are calculated by the MATLAB script in Figure 2. Once all the gains are applied, the values are summed and output by an analog out pin.

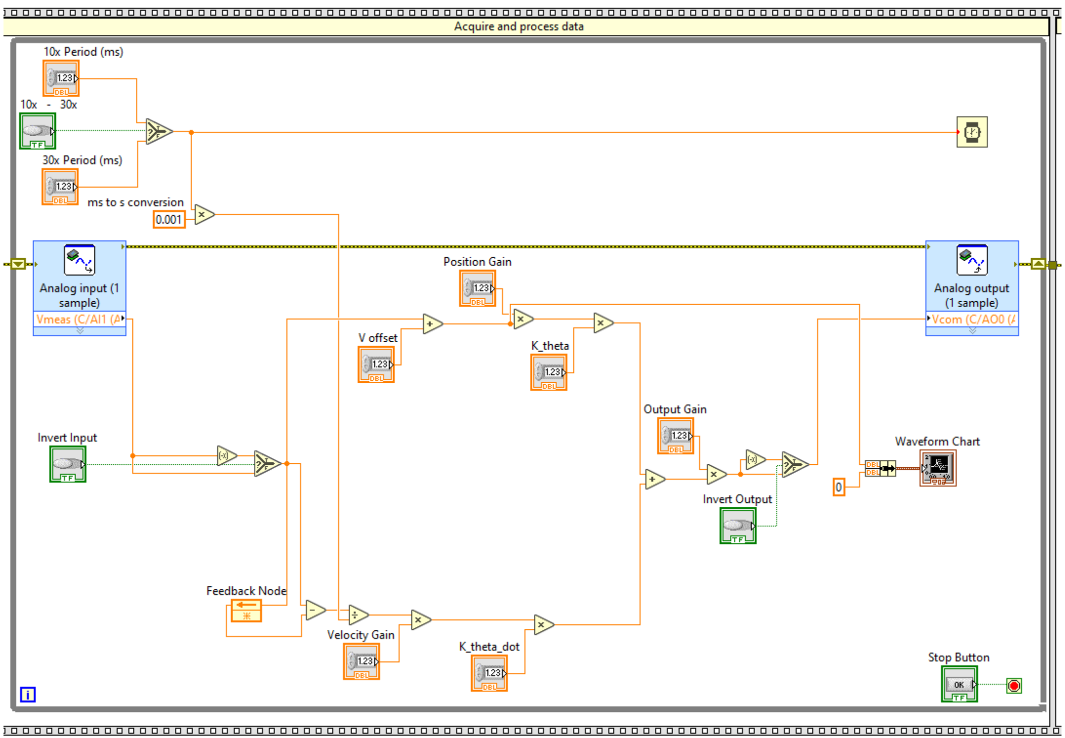

Figure 6: LabView Code

Controls Circuit Photos

The myRIO was wired to a breadboard to supply power to the potentiometer, read the output of the potentiometer, and output a control voltage to the AMC motor driver. The wiring for the myRIO and the breadboard can be seen in Figure 6 and Figure 7 respectively. The red wire, which was connected to the "5V" terminal, provided power to the analog potentiometer; the white wire, connected to the "AI (±10 v)→1+" terminal, read the position signal from the potentiometer; the blue wire, connected to the "AO→0" output positive and negative voltage to the DC motor which controlled the angular position and angular velocity of the pendulum; the black wire, connected to the "AO→AGND" terminal, was used as a ground terminal for both the potentiometer and the DC motor.

Figure 7: myRIO Wiring

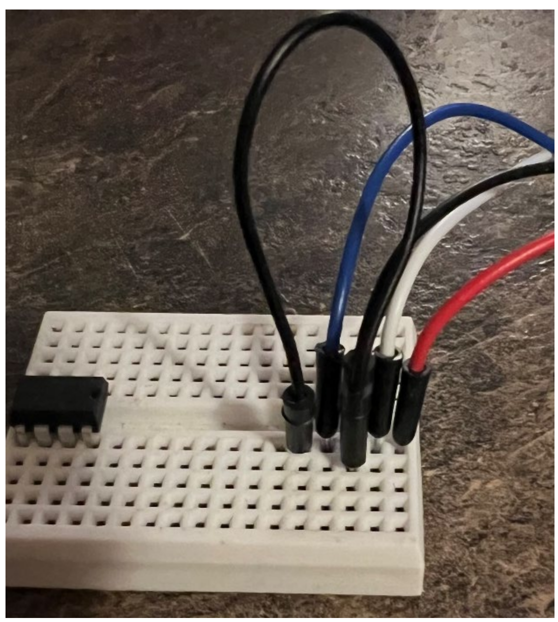

Figure 8: Breadboard Wiring

Sample Rate Evidence

18Hz Sampling

Position Signal

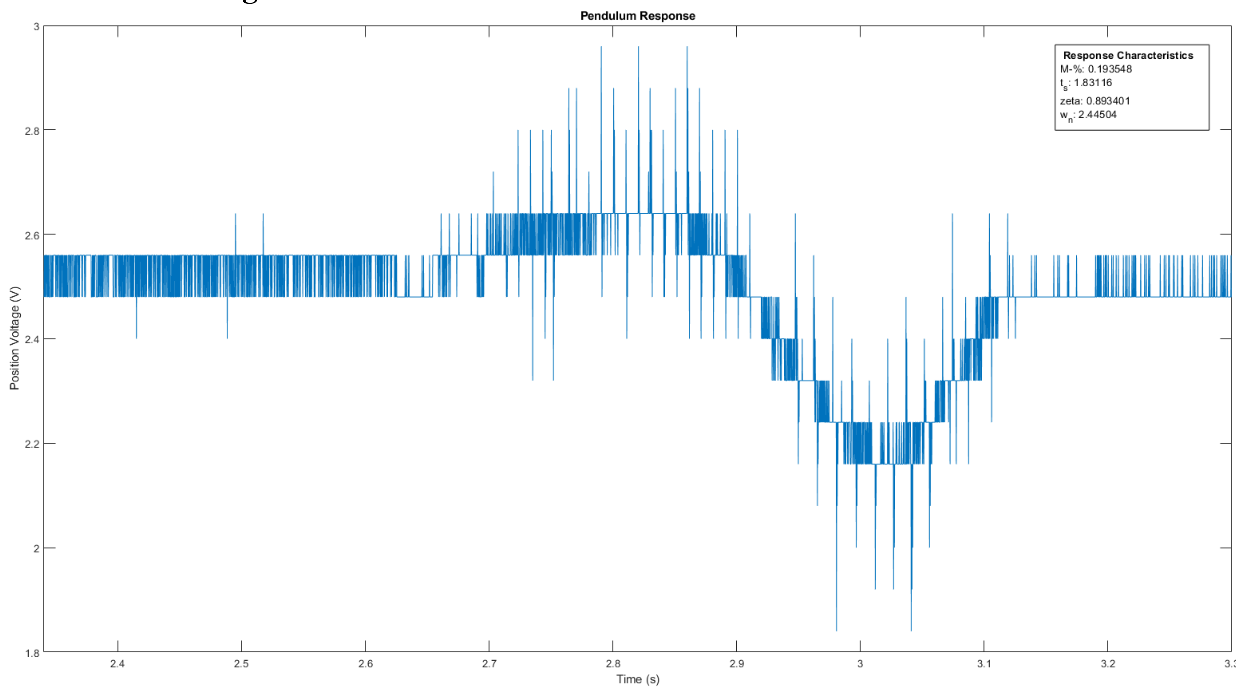

Figure 9: Low Frequency Response Plot

Loop Rate Discussion

The 18Hz loop rate corresponds to a frequency which is 10x the magnitude of the natural frequency of the inverted pendulum system as calculated in Figure 1. This frequency yielded a fair bit of jitter and provided an overall poor performance. The oscilloscope response shown in Figure 8 shows the significant jitter at this lower sampling frequency. It can be seen in Figure 8 that, at this frequency, a zeta of approximately 0.8 was achieved, which is close the required specification. This is closer than at the higher frequency, although the settling time at this lower frequency is more than 3-times larger than that at the higher frequency. At the lower frequency, the system is less like its continuous counterpart, thus making the performance overall less accurate to the specifications.

Video

Figure 10: Response Video at 18Hz

54Hz Sampling

Position Signal

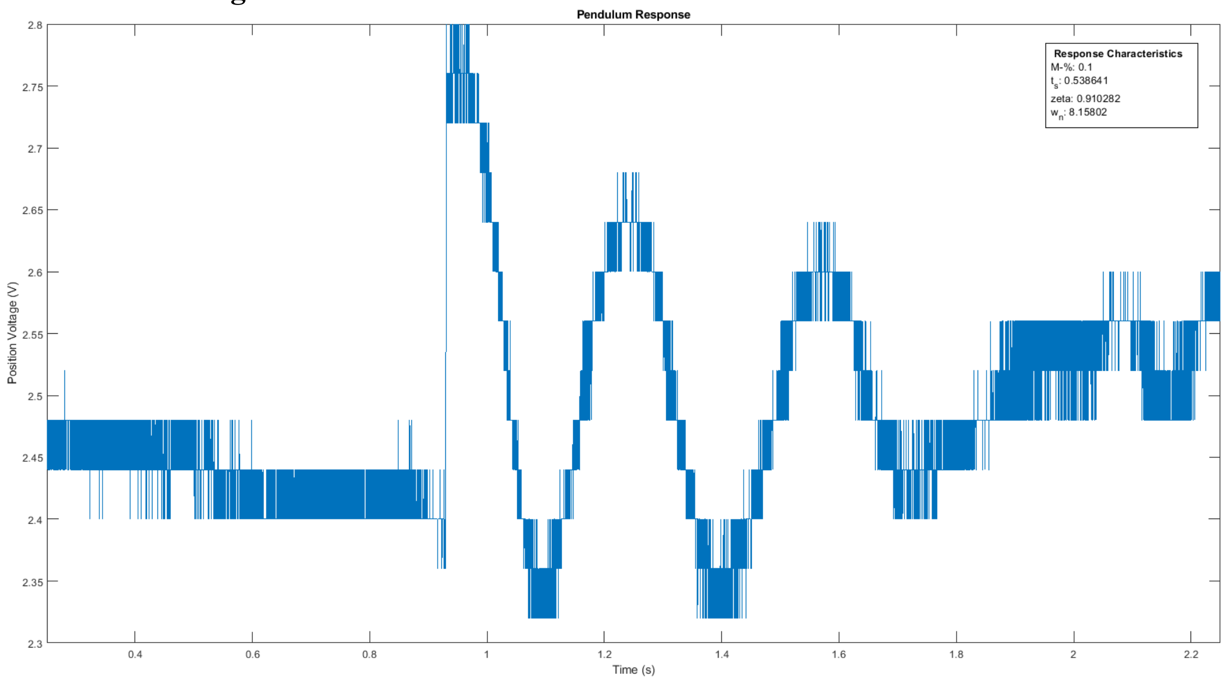

Figure 11: High Frequency Response Plot

Loop Rate Discussion

The 54Hz loop rate corresponds to a frequency which is 30x the magnitude of the natural frequency of the inverted pendulum system as calculated in Figure 1. This frequency yielded far less jitter than the lower 18Hz frequency and provided an overall more stable performance. The oscilloscope response shown in Figure 10 was not from the digital controller's best performance but displays a decent performance that almost meets all specifications required by this project. The settling time displayed is approximately 0.5s as the specifications require, yet the damping coefficient (ζ) is approximately 0.9 where the specifications required a coefficient of 0.707. There were times where the controller performed better and appeared to have a damping coefficient closer to the required specification, but due to inconsistencies with the testing apparatus, it did not always perform in the same way.

Video

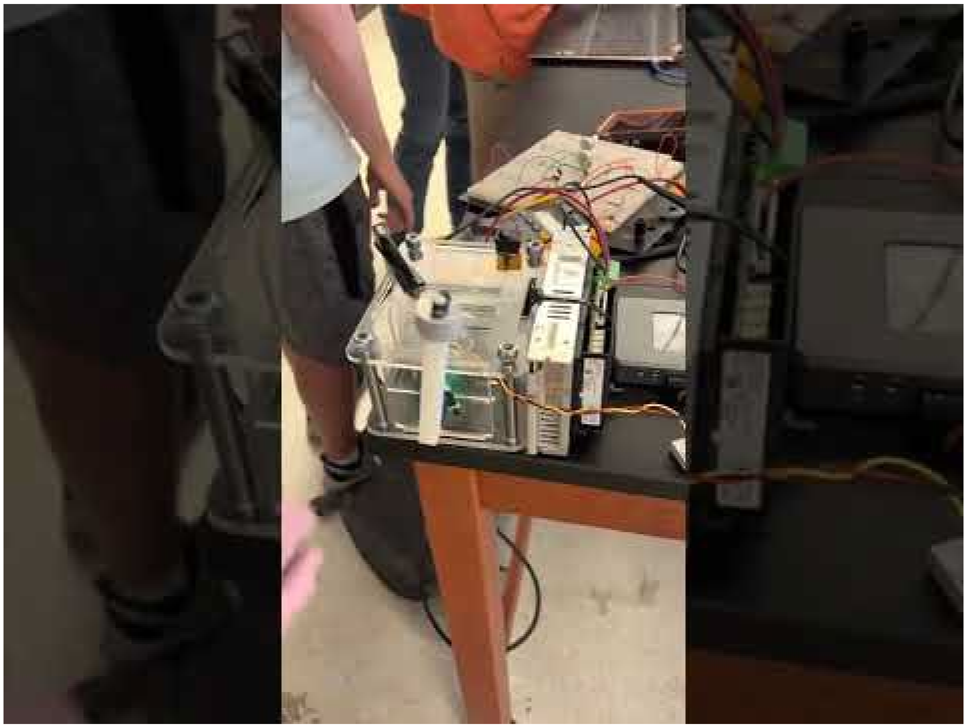

Figure 12: Response Video at 54Hz

Conclusion

Overall, the controller designed for this project was a success. The specifications were nearly achieved, and a great deal of knowledge was gained about digital controls. I believe the main issue within this project, and the largest obstacle to a perfect controller design, was inconsistency with testing equipment. Given the same code and control gains, different results were experienced, and this led to a constant retuning of control gains after every test. Given a consistent testing apparatus, and more time on the testing apparatus, a near-perfect controller could have been designed.

Another obstacle in the design of the digital controller was inaccurate analog input readings. The myRIO analog input pin utilized in this controller design is rated for ±10V. When reading a value of 5V, the myRIO only reports approximately 4.5V or lower. This led to additional tuning of voltage offsets and conditioning gains, which costed more time during the testing times.