Inverted Pendulum Experiment

Introduction
This lab experiment consists of two experimental procedures, each with sub parts. Experiment 1 is used to determine the system parameters needed to implement a controller. Part A finds the hardware gains in each direction of motion. Part B requires calculation of system parameters such as the inertia, and experimental verification of the calculations.

Experiment 2 then implements a controller. Part A tests the system without the controller activated. Part B then activates the controller and compares the stability to Part A. Part C then has a series of increasing step responses to determine the controller’s ability to track the desired output. Finally Part D has an increasing frequency input into the system, and a Bode plot is created to determine the system’s frequency response.

Hardware
Figure 1 shows the Inverted Pendulum Experiment. It consists of a pendulum rod which supports the sliding balance rod. The balance rod is driven by a belt and pulley which in turn is driven by a drive shaft connected to a DC servo motor below the pendulum rod. There are two series of weights included that affect the physical plant: brass counter weights connected underneath the pivot plate with adjustable height and weight and brass “donut” weights attached to both ends of the balance rod.

Figure 1: Model 505 ECP Inverted Pendulum Apparatus
Safety
Read Appendix B on the course website prior to conducting this lab. For this lab experiment in particular:

- Be careful on the step where students are asked to physically turning the equipment upside-down. Make sure the device is not on the edge of the table after it is inverted.
- Make sure the pendulum, when released, will not hit anyone or anything.
- Be sure to stay clear of the mechanism when turning on the controller or implementing an input. Selecting “Implement Algorithm” immediately implements the specified controller; if there is an instability or large control signal, the plant may react violently. If the system appears stable after implementing the controller, first displace the disk with a light, non sharp object (e.g. a plastic ruler) to verify stability prior to touching plant.

Hardware/Software Equipment Check
Before starting the lab, make sure that the equipment is working correctly using the following steps:

Step 1: Enter the ECP program (Note, there is an ECP USR Program as well. Do not use this program). You should see the Background Screen. Gently move the sliding and pendulum rods back and forth. You should observe some following errors and changes in encoder counts. The Control Loop Status should indicate “OPEN” and the Controller Status should indicate “OK”.

Step 2: Make sure that you can rotate the rods freely. Now press the black "ON" button to turn on the power to the Control Box. You should notice the green power indicator LED lit, but the motor should remain in a disabled state. Do not touch the mechanism whenever power is applied to the Control Box since there is a potential for uncontrolled motion of the rods unless the controller has been safety checked.
Experiment 1: System Identification

In this initial stage, the physical parameters needed to implement a controller such as inertia and damping are calculated and found experimentally.

### Table 1-1. Mass Property Values

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$l_0$</td>
<td>0.330 (m)</td>
<td>Length of pendulum rod from pivot to the sliding rod T section</td>
</tr>
<tr>
<td>$m_1$</td>
<td>TBD† (kg)</td>
<td>Mass of the complete sliding rod including all attached elements.</td>
</tr>
<tr>
<td>$m_{1o}$</td>
<td>0.103 (kg)</td>
<td>Mass of the sliding rod with belt, belt clamps, and rubber end guards (but without sliding rod brass &quot;donut&quot; weights)</td>
</tr>
<tr>
<td>$m_{w1}$</td>
<td>0.110 (kg)</td>
<td>Combined mass of both of the sliding rod brass &quot;donut&quot; weights (=0 if not used)</td>
</tr>
<tr>
<td>$m_2$</td>
<td>TBD (kg)</td>
<td>Mass of the complete assembly minus $m_1$</td>
</tr>
<tr>
<td>$m_{w2}$</td>
<td>1.000 (kg)</td>
<td>Mass of brass balance weight</td>
</tr>
<tr>
<td></td>
<td>(±2 if only one weight used)</td>
<td></td>
</tr>
<tr>
<td>$m_{2o}$</td>
<td>0.785 (kg)</td>
<td>Mass of the complete moving assembly minus $m_1$ and $m_{w2}$</td>
</tr>
<tr>
<td>$l_{co}$</td>
<td>0.071 (m)</td>
<td>Position of c.g. of the complete pendulum assembly with the sliding rod and balance weight removed</td>
</tr>
<tr>
<td>$J_0^*$</td>
<td>0.0246 (kgs$^2$)</td>
<td>[$J_{0o}$-$m_1l_o^2$] evaluated when $m_{w2}$=0. (Note: this value is known to be a bit low, just because it was measured WITHOUT the threaded rod inserted at the bottom!)</td>
</tr>
</tbody>
</table>

†To be determined

From the definitions in the table we have:

\[
\begin{align*}
m_1 &= m_{1o} + m_{w1} \quad \text{(Equation 1-1)} \\
m_2 &= m_{2o} + m_{w2} \quad \text{(Equation 1-2)}
\end{align*}
\]

The parameter

\[
l_{w2} = \frac{\Delta}{2} \quad \text{signed distance from pivot to c.g. of balance mass (m_{w2})}
\]

is changeable by the user and is readily measured. That is, referring to Figure 1-1,

\[
l_{w2} = -\frac{(t+b+h_b)}{2} \quad \text{(Equation 1-3)}
\]
The three remaining parameters – \( J^* \), \( J_{\infty} \), and \( l_c \) – are derived from the above as follows:

\[
J_{\infty} = J^*_0 + m_1 l_c^2 + m_2 (l_0 l_2)^2 \quad \text{ (Equation 1-4)}
\]

\[
l_c = \frac{m_0 l_0 l_2 + m_2 l_c}{m_2} \quad \text{ (Equation 1-5)}
\]

\[
J^* = [J_{\infty} - m_1 l_c^2] \quad \text{ (Equation 1-6)}
\]

**Experiment 1a.**
In this part, follow the procedure steps to determine \( k_a \) and \( k_x \).

**Procedure**

1. Turn OFF the control box so that there is no possibility that the system will accidentally move. Position the pendulum rod to the right and the sliding rod to the far right at the limit of travel. Enter the Executive program.

2. Select Zero Position in the Utility menu. You should see the position of Encoder 1 and Encoder 2 change to be approximately zero.
3. Hold on to the pendulum rod and push the sliding rod from one end until it hits the opposite end of its travel (hits the right limit switch). Now record the number of counts moved by encoder 2 on the Background Screen. With a ruler, measure the distance in meters that the sliding rod has traveled from one limit switch to the other. The ratio of the two is the value of \( k_x \).

4a. Now move the pendulum rod in the anti-clockwise direction from the 20 degree mark to the other 20 degree mark. Record the number of counts moved by encoder 1 on the Background Screen. Measure the angle of rotation in radians, the ratio of the two is the value of \( k_a \).

4b. Write a MATLAB script to solve for \( k_a \), \( k_x \), clearly labeling the counts at each limit switch, the distance the rod moves, converting from degrees to radians, and the counts at the 2 extreme values of encoder 1. Make sure to comment each line with correct units and a small explanation.

For this section, the final report is expected to include:

A diagram identifying the control elements and signals in the Torsion Experiment including:

- Sensor:
- Actuator:
- Controller:
- Reference Input:
- Actuator Output:
- System Output:

Calculations showing how you found the following values, along with units for EVERY quantity found.

- \( k_a \)
- \( k_x \)

A short explanation of why the two parameters above are important for modeling the dynamics of the system in the steps that follow.

**Experiment 1b:**

Follow the procedure steps to confirm the value of \( J^* \) from which \( J^r \) and \( J_{0c} \) are obtained via Equations 1-1 through 1-6.

**Procedure:**

1. Remove the threaded rod attached to brass balance weights from the apparatus. Use a rubber band to restrain the sliding rod in its center of travel as shown in Figure 1-2.
2. **Again, POWER OFF the control box.** This allows the encoder signals to pass to the control card but precludes accidental driving of the motor. Very carefully position the entire pendulum mechanism up-side down on two coplanar flat surfaces such that the pendulum rod is free to rotate as a non-inverted (regular) pendulum. (E.g. two tables side-by-side with approx. 8 in. gap between.)

3. Set up the ECP data acquisition. With the controller powered up, enter the Control Algorithm box via the Set-up menu and set $T_s = 0.00442$. Also be sure to set all gain values to zero, and to select continuous time Go to Set up Data Acquisition in the Data menu and select encoder #1 as data to acquire and specify data sampling every 5 servo cycles (i.e. every 5 $T_s$'s). Select OK to exit. With the pendulum hanging freely under gravity, select Zero Position from the Utility menu to zero the encoder positions.

4. Set up a zero input. Check again that the pendulum rod can freely rotate in this position. Now select Trajectory in the Command menu. Enter the Step dialog box and click on ser-up. Choosing Open loop Step, input a step size of 0 (zero), a duration of 10000 ms and repetition of 1. Exit to the Background Screen by consecutively selecting OK. This puts the controller in a mode for acquiring 20 sec of data on command but without driving the actuator. This procedure may be repeated and the duration adjusted to vary the data acquisition period.

5. Make sure to read this step completely before running the test. Select Execute from the Command menu. Manually displace the pendulum rod approximately 20 deg. from vertical and let go. You should notice that the "non-inverted" pendulum rod will oscillate and slowly attenuate while the encoder data is collected to record this response. Select OK after data is uploaded.

6. Export the Encoder 1 Data and plot in MATLAB.

7a. Measure the period of oscillation, $T$, in seconds by taking the time for completion of several cycles divided by the number of cycles. Convert this to the natural frequency of the system. Use the Data Cursor feature in MATLAB. Confirm that its value for
the factory default setting of the balance weight (all the way down the lead screw but just clearing the pocket hole) is approximately 1.25 seconds\(^1\).

7b. Use the following classic linearized "non-inverted" equation of motion to derive an expression for the inertia \(J\) in terms of the measured \(T\). Hint, use the small angle approximation to simplify.

\[
\sum T = J \alpha
\]  
(Equation 1-8)

In your report, show your derivation of this expression.

8. Calculate the values of \(m\) and \(l_{cg}\) for the test case and hence obtain \(J\) from your derived expression. (\textit{Note that }\(J\) \textit{here is the same as }\(J_0\) \textit{of this manual's notation}) Now determine \(J_0^*\) according to \(J_0^* = J - m_1 l_0^2\) and hence verify the value shown in Table 1-1.

9. Carefully re-orient the mechanism to the regular "inverted pendulum" position (up-side up) and turn on the control box. This concludes the confirmation procedure for \(J^*\).

The final report is expected to include:

One (1) MATLAB Plot of the non-inverted pendulum response, with two (2) Data Cursor Points on each plot, along with titles, labels and legends if necessary that clearly show which plot corresponds to which situation.

Calculations showing how you found the following values, along with units for EVERY quantity found.

- Period of oscillation, \(T\)
- Natural frequency, \(\omega_n\)
- Mass of the assembly, \(m\) (Figure out using values in Table 6.1-1)
- Center of gravity (c.g.), \(l_{cg}\)
- Moment of inertia of the assembly \(J\)
- \(J_0^*\)
- Percent error between the table value and the experimentally determined value for \(J_0^*\)

For all the questions highlighted, the questions should be copied and pasted into your lab report and explicitly answered immediately thereafter.

\(^1\)For more accurate measurement, you may use tabular data via Export Raw Data in the Data menu to export data to a text editor where precise numerical values may be read.
**Experiment 2: System stabilization and control**

We will study the stabilization and control of the unstable pendulum system with a close-loop control system.

**Experiment 2a. Hardware set up:**
Set up the pendulum in its original inverted configuration. Insert and fasten (with a nut) the counter-weight rod into the base plate of the pendulum so that it appears to be 14 cm long. Mount and fasten (by tightening against each other) two counter weights to the rod at the free end of the rod. Also, the sliding rod with its two donut weights should be mounted in the system. Make sure it can be driven by the motor by manually turning the knob back and forth. Also, make sure the belt is not too loose and tighten it if so.

Adjust the pendulum and the sliding rod position to reach an equilibrium position upright. Displace the pendulum gently in both directions to see if it can return to its original upright position. If so, record the range of angles.

**Can the pendulum return to its upright position? If yes, estimated the range of angles for which this is possible.**

You may estimate the angle by reading off the counts of Encoder #1 displayed on the monitor and then convert to degrees using the gain measurements you did in the previous experiments.

**Experiment 2b. Controller set up and stabilization:**
Position the pendulum at the upright equilibrium position and then reset it to zero position via the utility menu. Power up the controller. Enter the Control Algorithm box via the Setup menu. Set the sampling period to be $T_s = 0.00884$. Select Discrete Time and then select General Form. Click Implement Algorithm and then Ok to close the menu window.

Gently push away the pendulum from the equilibrium position back and forth. You should see the motor actuates the sliding rod accordingly to stabilize the system so that you can displace the pendulum to a large angle while it can return to its upright position upon release. If you don’t get this stabilizing response, ask course instructor or TA to reset the controller for you. With this stabilization controller at work, displace the pendulum gently in both directions to exam its ability to return to it upright position.

**Can the pendulum return to its upright position when displaced away by its maximum measurable angles of ± 20°? If no, what is the range of angles?**
Experiment 2c. Step response:

With the pendulum in its upright equilibrium position and with the controller engaged and sensor positions zeroed, enter a step input via the **Command** menu and **Trajectory Configuration**. Select **Step** and then click **Setup**. Select **Closed Loop Step** and input a step size of 500 counts, a duration of 5000 ms and 1 repetition. Exit to the Background Screen by consecutively selecting **Ok**. This will command the system with a 500 counts step move forward and back with a 5.0 seconds dwell. **You may use a different step size you see fit and state it in your report.** NOTE: a step size of 500 will exceed the limits of the system, so you will have to investigate values that work for you that are smaller than this but still give large enough motions that the system dynamics are clearly evident in the plots. You must document the size of command you are using in your lab report.

Execute the controlled motion via **Command** and **Execute**. You should see a stable pendulum response to the step input you specified.

Record and plot the response via **Data** menu and select **Set up Data Acquisition**. Select encoder #1 and commanded position to collect data at every sampling period (every $T_s$). Zero the sensor position and re-run the experiment to collect and plot the data. Plot encoder #1 (pendulum swing) and commanded positions on one axis and encoder #2 position (rod displacement) on the other.

Next, conduct a sequence of experiments starting with a step size of 100 counts and an increment of 100 counts (or 50 counts) until the step size reaches its maximum value for stable operation. For each run, record the pendulum swing position (encoder #1) at steady-state as a function of the commanded position in an excel file for later plotting; if the plot oscillates up and down, estimate an average value for it. Plot the results. **Based on your results, briefly describe the system’s ability to respond to the command as the size of the command increases.**

**The final report is expected to include:**
MATLAB Plots illustrating the responses above, along with titles, labels and legends if necessary that clearly show which plot corresponds to which situation.
- Plot(s) of the step response, with encoder 1 and 2 on different axes, along with the commanded position
- Series of plots starting with a step input of 100 and increasing by 50 until the maximum
- Plot of pendulum swing vs. size of step command

Calculations showing how you found the following values, along with units for EVERY quantity found.
- Percentage position error at steady-state for the step response between encoder #1 and commanded position. Hint: use the MATLAB Data cursor.

For all the questions highlighted, the questions should be copied and pasted into your lab report and explicitly answered immediately thereafter.

**Experiment 2d: Frequency response:**
Conduct a series of experiments with sinusoidal inputs. Enter a sinusoidal input similar to what you did for the step input above. Select an amplitude of 300 counts and 6 cycles. You may use a different amplitude value you see fit and state it in your report.

Conduct a sequence of experiments starting with a frequency of 0.25 Hz and then increase by 0.25 Hz in subsequent runs (or a smaller increment). In each run, observe how the system follows the command (encoder #1 vs. commanded position), which may be characterized by estimating the steady-state amplitude (in ratio to command) and phase lag (in deg) of the response from your plot. Record this result in an excel file for each frequency of the input for later plotting. Terminate the sequence at the frequency at which the system can no longer follow the command, which can be easily seen from the plotted command and response curves.

Your goal is to produce a Bode plot, e.g. a top plot with log10(frequency) on the x-axis and 20*log10(magnitude ratio) on the y-axis; and a bottom plot aligned with the top plot again with log10(frequency) on the x-axis and phase drop on the y-axis. If you do not remember details about Bode plotting, review your Vibrations and Modeling of Dynamic Systems notes.

Based on your results, briefly describe the system tracking behavior as the frequency increases.

**The final report is expected to include:**
Two (2) MATLAB Plots, along with titles, labels and legends if necessary that clearly show which plot corresponds to which situation.
- Plot of response amplitude (ratio) as function of frequency.
- Plot of response phase lag (deg) as function of frequency

Calculations showing how you found the following values, along with units for EVERY quantity found.
- The frequency at which the system loses tracking

For all the questions highlighted, the questions should be copied and pasted into your lab report and explicitly answered immediately thereafter.