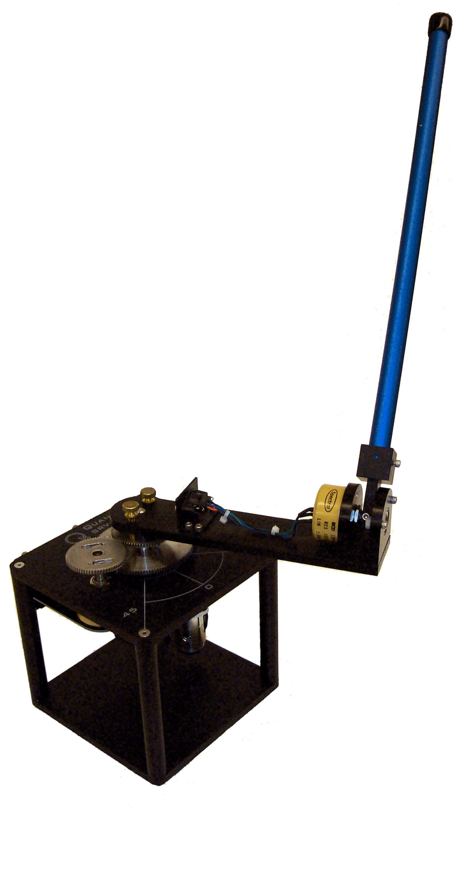
Rotary Motion Servo Plant: SRV02

Rotary Experiment #08: Self Erecting Inverted Pendulum Control

Self Erecting

Inverted Pendulum Control using QUARC



Instructor Manual

Table of Contents

1. Introduction

In this laboratory the Quanser SRV02 rotary plant and a pendulum module are used. The objective of this laboratory is to design and implement a complete control system that will swing the pendulum up from its vertical downward position and balance it in the vertical upward position. In implementing such a control system the following topics will be covered:

- Modeling the dynamics of the single inverted pendulum using Euler-Lagrange equations.
- Obtaining a linear state-space representation of the system.
- Designing an energy based swing up controller.
- Designing a state-feedback control system that balances the pendulum at its vertical upward position using LQR.
- Implementing the controllers on the Quanser SRV02 + SIP (Single Inverted Pendulum) plant and evaluating its performance.

Regarding Gray Boxes:

Gray boxes present in the **instructor manual** are not intended for the students as they provide solutions to the pre-lab assignments and contain typical experimental results from the laboratory procedure.

2. Prerequisites

In order to successfully carry out this laboratory, the user should be familiar with the following:

- Data-acquisition, amplifier, and the main components of the SRV02 (e.g. actuator, sensors), as described in References [1], [4], and [5], respectively.
- Wiring and operating procedure of the SRV02 + SIP plant with the amplifier discussed in Reference [9].
- Laboratory described in Reference [6] in order to be familiar using QUARC with the SRV02.
- Designing a PV position control for the SRV02 as dictated in Reference [8].

3. Overview of Files

Table 1 below lists and describes the various files supplied with the SRV02 + SIP Control laboratory.

File Name	***Description***
18 – Rotary Pendulum User Manual.pdf	This manual describes the hardware of the Rotary Pendulum and explains how to setup and wire the system for the experiments.
19 – Inverted Pendulum Control – Student Manual.pdf	This laboratory guide contains pre-lab and in-lab exercises demonstrating how to design and implement a controller on the Quanser SRV02 + SIP plant using QUARC.
setup_srv02_exp08_sip.m	The main MATLAB script that sets the SRV02 motor and sensor parameters, the SRV02 configuration-dependent model parameters, and the SIP sensor parameters. **Run this file only to setup the laboratory.**
config_srv02.m	Returns the configuration-based SRV02 model specifications *Rm*, *kt*, *km*, *Kg*, *eta_g*, *Beq*, *Jeq*, and *eta_m*, the sensor calibration constants K_POT, K_ENC, and K_TACH, and the amplifier limits VMAX_AMP and IMAX_AMP.
config_sp.m	Sets the model parameters of the Quanser single inverted pendulum module depending on the pendulum length and type specified. The pendulum length and type are set in setup_srv02_exp08_sip.m file.
d_model_param.m	Calculates the SRV02 model parameters *K* and *tau* based on the device specifications *Rm*, *kt*, *km*, *Kg*, *eta_g*, *Beq*, *Jeq*, and *eta_m*.
calc_conversion_constants.m	Returns various conversions factors.
q_sip.mdl	Simulink file that implements the balance controller only. The swing up has to be done manually and this controller only kicks in when the pendulum is at its vertical upward position and balances it.
q_sesip.mdl	Simulink file that implements the complete energy based swing up and state-feedback balance controller and the switching algorithm.
19 – Inverted Pendulum Control – Instructor Manual.pdf	Same as the student version except the gray boxes are no longer shaded to reveal the solution to the pre-lab and in-lab exercises.
calculate_qr.m	This file sets the Q and R weighting matrices in addition to the reference energy for swing up control when CONTROL_TYPE has been set to AUTO in setup_srv02_exp08_sip.m.
SRV02 Energy-Based Swing-Up.mws	Maple worksheet used to develop the energy based swing up controller. Waterloo Maple 9, or a later release, is required to open, modify, and execute this file.
SRV02 Energy-Based Swing-Up.html	HTML presentation of the above Maple Worksheet. It allows users to view the content of the Maple file without having Maple 9 installed. No modifications to the equations can be performed when in this format.

File Name	Description
SRV02+SIP Equations.mws	Maple worksheet used to develop the equations of motion for an inverted pendulum mounted on a rotary plant. Waterloo Maple 9, or a later release, is required to open, modify, and execute this file.
SRV02+SIP Equations.html	HTML presentation of the above Maple Worksheet. It allows users to view the content of the Maple file without having Maple 9 installed. No modifications to the equations can be performed when in this format.

Table 1: Files supplied with the SRV02 + SIP Control experiment.

4. Pre-Lab Assignments

4.1. Modeling the Rotary Single Inverted Pendulum

The kinematics of the rotary single inverted pendulum system is depicted in Figure 1. The various lengths, masses, and moments of inertia associated with the rotary arm and the pendulum link of the system are shown as well as the coordinate systems used to derive the kinematics of the system when using classical mechanics to derive the dynamics of the system.

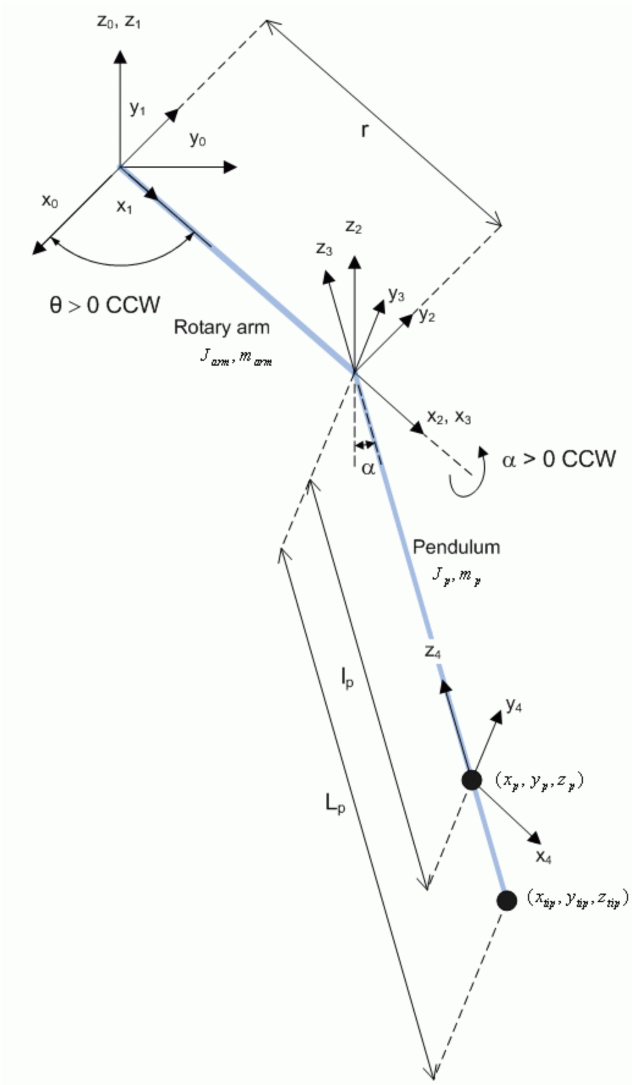


Figure 1: Kinematics of a single inverted pendulum system

4.1.1. Introduction to Euler-Lagrange

Instead of using the classical mechanics, the *Lagrangian* method will be used to find the equations of motion of the system. The classical method is often used for more complicated systems such as robot manipulators with multiple joints.

More specifically, the equations that describe the motions of the rotary arm and the pendulum with respect to the servo motor voltage, i.e. the dynamics, will be obtained using the *Euler-Lagrange* equation:

$$\left(\frac{\partial^2}{\partial t\, \partial qd_i} L\right) - \left(\frac{\partial}{\partial q_i} L\right) = Q_i \qquad [1]$$

The variables q_i are called *generalized coordinates*. For the gantry, we can define this coordinate as

$$q = [\theta(t), \alpha(t)], \qquad [2]$$

where, as shown in Figure 1, θ(t) is the rotary arm angle and α(t) is the pendulum angle. The corresponding velocities are therefore

$$qd = \left[\frac{d}{dt}\theta(t), \frac{d}{dt}\alpha(t)\right]. \qquad [3]$$

With the generalized coordinates now defined, the Euler-Lagrange equation becomes

$$\left(\frac{\partial^2}{\partial t\, \partial qd_1} L\right) - \left(\frac{\partial}{\partial \theta} L\right) = Q_1 \qquad [4]$$

and

$$\left(\frac{\partial^2}{\partial t\, \partial qd_2} L\right) - \left(\frac{\partial}{\partial \alpha} L\right) = Q_2. \qquad [5]$$

The *Lagrangian* of the system is described

$$L = T - V, \qquad [6]$$

where *T* is the total kinetic energy of the system and *V* is the total potential energy of the system. Thus the Lagrangian is the difference between a system's kinetic and potential energies.

The generalized forces Q_i are used to describe the non-conservative forces applied to a system with respect to the generalized coordinates. In this case, the generalized force acting on the rotary arm is

$$Q_1 = \tau_m - B_{arm}\left(\frac{d}{dt}\theta(t)\right) \qquad [7]$$

and acting on the pendulum is

$$Q_2 = B_p\left(\frac{d}{dt}\alpha(t)\right). \qquad [8]$$

The torque applied at the load gear, τ_m, is generated by the servo motor as described by the equation

$$\tau_m = \frac{\eta_g K_g \eta_m K_t \left(R_m I_m - K_g K_m\left(\frac{d}{dt}\theta(t)\right)\right)}{R_m}. \qquad [9]$$

See Reference [5] for a description of the corresponding SRV02 parameters (e.g. such as the back-emf constant, K_m). Our control variable is the input servo motor current, I_m. Opposing the applied torque is the viscous friction torque, or viscous damping, corresponding to the term B_{arm}. Since the pendulum is not actuated, the only force acting on the link is the damping. The viscous damping coefficient of the pendulum is denoted by B_p.

Keep in mind that the Euler-Lagrange equations is a systematic method of finding the equations of motion, i.e. EOMs, of a system. Once the kinetic and potential energy are obtained and the Lagrangian is found, then the task is to compute various derivatives to get the EOMs. For the rotary single inverted pendulum system, the nonlinear equations of motions generated by the Euler-Lagrange formula are

$$2\, m_p \cos(\alpha(t)) \left(\frac{d}{dt}\alpha(t)\right) l_p^2 \left(\frac{d}{dt}\theta(t)\right) \sin(\alpha(t)) - m_p \sin(\alpha(t)) \left(\frac{d}{dt}\alpha(t)\right)^2 l_p\, r$$
$$+ (m_p r^2 + m_p l_p^2 - m_p l_p^2 \cos(\alpha(t))^2 + J_{arm}) \left(\frac{d^2}{dt^2}\theta(t)\right) + m_p \cos(\alpha(t)) \left(\frac{d^2}{dt^2}\alpha(t)\right) l_p\, r = \tau_m - B_{arm}\left(\frac{d}{dt}\theta(t)\right) \quad [10]$$

and

$$-m_p \cos(\alpha(t))\, l_p^2 \left(\frac{d}{dt}\theta(t)\right)^2 \sin(\alpha(t)) + m_p \cos(\alpha(t))\, l_p \left(\frac{d^2}{dt^2}\theta(t)\right) r + (J_p + m_p l_p^2)\left(\frac{d^2}{dt^2}\alpha(t)\right)$$
$$+ m_p\, g \sin(\alpha(t))\, l_p = B_p \left(\frac{d}{dt}\alpha(t)\right) \quad [11]$$

.

However in order to find the kinetic and potential energies of the single inverted pendulum, it is beneficial to first describe the pendulum center of mass in terms of a three-dimensional Cartesian coordinate. The translational kinetic energy and potential energy of the pendulum can then be defined using these Cartesian equations.

4.1.2. Forward Kinematics

Forward kinematics involves describing a point on a rigid body in terms of Cartesian coordinate system. It does not involve masses or forces. In the case of the single inverted pendulum, we want a set of equations that describe the *XYZ* position of the pendulum center of mass from the rotary arm angle and the pendulum angle. Thus the following equations are needed

$$x_p = f(\theta, \alpha), \quad [12]$$

$$y_p = g(\theta, \alpha), \quad [13]$$

and

$$z_p = h(\theta, \alpha). \quad [14]$$

These can be found manually using trigonometry and Figure 1. However, a more systematic procedure is to use rotational and translational matrices. This method is favored, for instance, when finding the kinematics of robot manipulators with more than one link.

4.1.2.1. Transformation Matrices

Given the coordinate system O_0, the rotational matrices that describe a CCW rotation of angle γ about the X_0, Y_0, and Z_0 axes are

$$R_{X0,\gamma} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\gamma) & -\sin(\gamma) & 0 \\ 0 & \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad [15]$$

$$R_{Y0,\gamma} = \begin{bmatrix} \cos(\gamma) & 0 & \sin(\gamma) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\gamma) & 0 & \cos(\gamma) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad [16]$$

and

$$R_{Z0,\gamma} = \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad [17]$$

The matrices that describe a translation along the X_0, Y_0, and Z_0 axes, respectively, by a fixed length of L are

$$T_{X0,L} = \begin{bmatrix} 1 & 0 & 0 & L \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad [18]$$

$$T_{Y0,L} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & L \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad [19]$$

and

$$T_{Z0,L} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & L \\ 0 & 0 & 0 & 1 \end{bmatrix} . \qquad [20]$$

Consider the transformation from the coordinate system O_0 to O_2 shown in Figure 2.

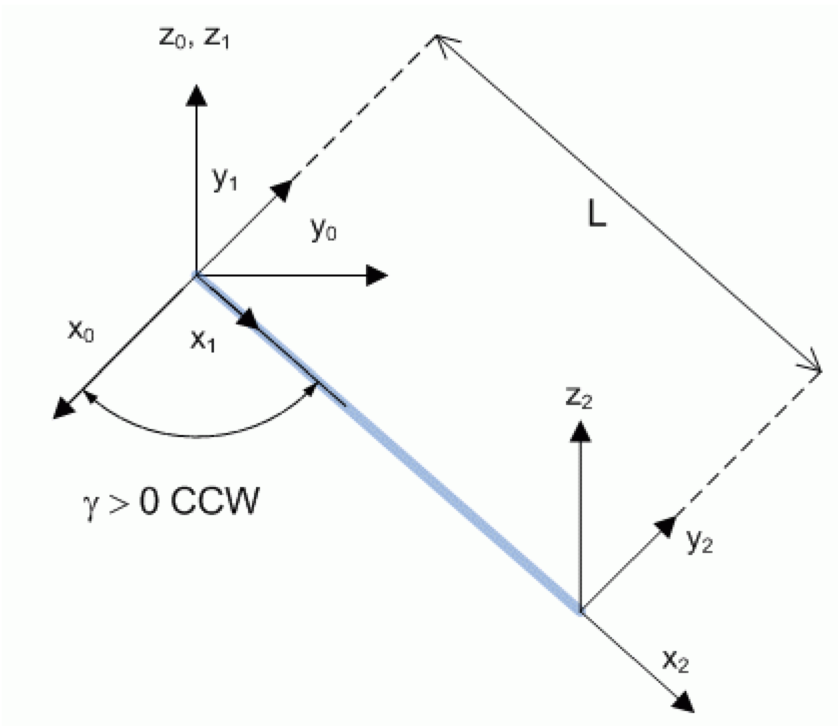


Figure 2: Kinematics example

The transformation from O_0 to O_1 is a rotation about the Z_0 axis by angle γ. Then, a translation of length of L along the X_1 axis is required to go from O_1 to O_2 system. Thus the full transformation is

$$T_{0,2} = R_{Z0,\gamma}\, T_{X1,L} . \qquad [21]$$

Performing the matrix multiplication

$$T_{0,2} = \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & L \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad [22]$$

equals

$$T_{0,2} = \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 & \cos(\gamma)L \\ \sin(\gamma) & \cos(\gamma) & 0 & \sin(\gamma)L \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} . \qquad [23]$$

The Cartesian coordinates of the O_2 system are taken from the matrix elements $T_{0,2}[1,4]$, $T_{0,2}[2,4]$, and $T_{0,2}[3,4]$ to get:

$$x_2 = \cos(\gamma)L , \qquad [24]$$

$$y_2 = \sin(\gamma)L , \qquad [25]$$

and

$$z_2 = 0 . \qquad [26]$$

This set of equations describes the tip of the pendulum in the XYZ space with respect to the base coordinate system O_0. As the beam rotates CCW from 0 degrees to 90 degrees, the distance along the X_0 axis decreases from L down 0 while the distance along the Y_0 axis increases positively from 0 to L.

4.1.2.2. Kinematics of the Inverted Pendulum

Go through these exercises to find the (x_p, y_p, z_p) equations:

1. Find the matrix $T_{0,2}$ that describes the transformation from the base coordinate system O_0, which is on the rotary servo load gear, to the coordinate system O_2, which is at the tip of the rotary arm. Make sure the Cartesian coordinates (x_2,y_2,z_2) makes sense before going on to the next exercise.

Solution:

The transformation matrix

$$T_{0,2} = R_{Z0,\theta}\, T_{X1,r} \qquad \text{[s1]}$$

expresses the (x_2, y_2, z_2) coordinates in terms of the base O_0 system. The rotational matrix describing a CCW rotation of θ about the Z_0 axis is

$$R_{Z0,\theta} = \begin{bmatrix} \cos(\theta(t)) & -\sin(\theta(t)) & 0 & 0 \\ \sin(\theta(t)) & \cos(\theta(t)) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \qquad \text{[s2]}$$

and the translation of r along the rotary arm, coordinate system X_1, is described

$$T_{X1,r} = \begin{bmatrix} 1 & 0 & 0 & r \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \qquad \text{[s3]}$$

After performing the matrix multiplication in [s1], the resulting transformation matrix is

$$T_{0,2} = \begin{bmatrix} \cos(\theta(t)) & -\sin(\theta(t)) & 0 & \cos(\theta(t))\, r \\ \sin(\theta(t)) & \cos(\theta(t)) & 0 & \sin(\theta(t))\, r \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \qquad \text{[s4]}$$

0 1 2

2. Find the transformation matrix $T_{2,4}$ that describes the transition from the rotary arm tip down to the pendulum center of mass, i.e. from the O_2, to the O_4 system.

Solution:
The transformation matrix

$$T_{2,4} = R_{X2,\alpha}\, T_{Z3,lp} \qquad [s5]$$

expresses the (x_p, y_p, z_p) coordinates with respect to arm tip, i.e. the O_2 system. The rotational matrix describing a CCW rotation of α about the pendulum pivot is

$$R_{X2,\alpha} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha(t)) & -\sin(\alpha(t)) & 0 \\ 0 & \sin(\alpha(t)) & \cos(\alpha(t)) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \qquad [s6]$$

From that point, the transformation from the pendulum pivot to its center of mass is a translation along Z_3 by a distance of l_p in the negative direction,

$$T_{Z3,lp} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -l_p \\ 0 & 0 & 0 & 1 \end{bmatrix}. \qquad [s7]$$

The resulting transformation matrix is

$$T_{2,4} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha(t)) & -\sin(\alpha(t)) & \sin(\alpha(t))\, l_p \\ 0 & \sin(\alpha(t)) & \cos(\alpha(t)) & -\cos(\alpha(t))\, l_p \\ 0 & 0 & 0 & 1 \end{bmatrix}.. \qquad [s8]$$

0	1	2

3. Finally, perform the calculation

$$T_{0,4} = T_{0,2}\, T_{2,4} \qquad [27]$$

and list the resulting (x_p, y_p, z_p) equations.

Solution:
The final transformation matrix is defined

$$T_{0,4} = \begin{bmatrix} \cos(\theta(t)) & -\sin(\theta(t))\cos(\alpha(t)) & \sin(\theta(t))\sin(\alpha(t)) & -\sin(\theta(t))\sin(\alpha(t))\,l_p + \cos(\theta(t))\,r \\ \sin(\theta(t)) & \cos(\theta(t))\cos(\alpha(t)) & -\cos(\theta(t))\sin(\alpha(t)) & \cos(\theta(t))\sin(\alpha(t))\,l_p + \sin(\theta(t))\,r \\ 0 & \sin(\alpha(t)) & \cos(\alpha(t)) & -\cos(\alpha(t))\,l_p \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{[s9]}$$

and the resulting equations describing the pendulum center of mass point (x_p, y_p, z_p) relative to the base O_0 system are

$$x_p = -\sin(\theta(t))\sin(\alpha(t))\,l_p + \cos(\theta(t))\,r \quad \text{[s10]}$$

$$y_p = \cos(\theta(t))\sin(\alpha(t))\,l_p + \sin(\theta(t))\,r \quad \text{[s11]}$$

and

$$z_p = -\cos(\alpha(t))\,l_p \, . \quad \text{[s12]}$$

0	1	2

4. Take the time derivative to find the translational velocity of the pendulum CM.

Solution:
Using the chain-rule and taking the time-derivative of equations [s10], [s11], and [s12] gives

$$xd_p = -\cos(\theta(t))\left(\frac{d}{dt}\theta(t)\right)\sin(\alpha(t))\,l_p - \sin(\theta(t))\cos(\alpha(t))\left(\frac{d}{dt}\alpha(t)\right)l_p - \sin(\theta(t))\left(\frac{d}{dt}\theta(t)\right)r \quad \text{[s13]}$$

$$yd_p = -\sin(\theta(t))\left(\frac{d}{dt}\theta(t)\right)\sin(\alpha(t))\,l_p + \cos(\theta(t))\cos(\alpha(t))\left(\frac{d}{dt}\alpha(t)\right)l_p + \cos(\theta(t))\left(\frac{d}{dt}\theta(t)\right)r \quad \text{[s14]}$$

and

$$zd_p = \sin(\alpha(t))\left(\frac{d}{dt}\alpha(t)\right)l_p \, . \quad \text{[s15]}$$

0	1	2

4.1.3. Finding the Lagrangian of the System

In this section, the Lagrangian of the inverted pendulum system is derived. As discussed, the Lagrange is the difference between a system's kinetic and potential energy. Thus the kinetic and potential energy of the SIP are first computed in sections 4.1.3.1 and 4.1.3.2, respectively. Then, in Section 4.1.3.3, the Lagrange is found.

4.1.3.1. Kinetic Energy

Generally speaking, there are two types of kinetic energies in a mechanical structure: rotational and translational. The *rotational* kinetic energy of an object with a moment of inertia of *J* spinning about an axis at an angular rate of ω, is

$$T_r = \frac{J\,\omega^2}{2} \qquad [28]$$

and the *translational* kinetic energy of an object of mass *m* moving at a linear velocity of *v* is

$$T_t = \frac{m\,v^2}{2} \qquad [29]$$

.

The two rotary objects – the arm and the pendulum link – each contribute to rotational kinetic energy of the SIP while the translational energy is caused by the pendulum only. Go through these exercises to find the total kinetic energy of the SIP:

1. Find the total rotational energy of the SIP, T_r.

Solution:
Using the same nomenclature for the moment of inertia parameters as given in Figure 1 (and the user manual), the rotational energy of the arm is

$$T_{r,\,arm} = \frac{1}{2} J_{arm} \left(\frac{d}{dt} \theta(t) \right)^2 \qquad [s16]$$

and the rotational energy of the pendulum is

$$T_{r,\,p} = \frac{1}{2} J_p \left(\frac{d}{dt} \alpha(t) \right)^2 \qquad [s17]$$

.

The total rotational energy of the SIP is therefore:

$$T_r = \frac{1}{2} J_{arm} \left(\frac{d}{dt} \theta(t) \right)^2 + \frac{1}{2} J_p \left(\frac{d}{dt} \alpha(t) \right)^2 \qquad [s18]$$

.

0	1	2

2. Find the translational energy of the SIP, T_t.

Solution:
The linear velocity of the pendulum CM is the magnitude of the translational velocity vector

$$v = \sqrt{xd_p^2 + yd_p^2 + zd_p^2} \qquad [s19]$$

Substituting this into the kinetic energy formula in [29] we obtain the equation

$$T_t = \frac{1}{2} m_p (xd_p^2 + yd_p^2 + zd_p^2) \qquad [s20]$$

Evaluating the expression with the velocity elements, we obtain the translational SIP kinetic energy

$$T_t = \frac{1}{2} m_p \Bigg(\left(-\cos(\theta(t)) \left(\frac{d}{dt}\theta(t)\right) \sin(\alpha(t))\, l_p - \sin(\theta(t))\cos(\alpha(t)) \left(\frac{d}{dt}\alpha(t)\right) l_p - \sin(\theta(t)) \left(\frac{d}{dt}\theta(t)\right) r \right)^2$$
$$+ \left(-\sin(\theta(t)) \left(\frac{d}{dt}\theta(t)\right) \sin(\alpha(t))\, l_p + \cos(\theta(t))\cos(\alpha(t)) \left(\frac{d}{dt}\alpha(t)\right) l_p + \cos(\theta(t)) \left(\frac{d}{dt}\theta(t)\right) r \right)^2 \qquad [s21]$$
$$+ \sin(\alpha(t))^2 \left(\frac{d}{dt}\alpha(t)\right)^2 l_p^2 \Bigg)$$

0	1	2

3. Give the total kinetic energy, T.

Solution:
The total kinetic energy is the sum of the rotational and translational kinetic energies,

$$T = T_r + T_t \qquad [s22]$$

Substituting the total rotational and translational energies found above gives the total SIP kinetic energy

$$T = \frac{1}{2} J_{arm} \left(\frac{d}{dt}\theta(t)\right)^2 + \frac{1}{2} J_p \left(\frac{d}{dt}\alpha(t)\right)^2 + \frac{1}{2} m_p \Bigg($$
$$\left(-\cos(\theta(t)) \left(\frac{d}{dt}\theta(t)\right) \sin(\alpha(t))\, l_p - \sin(\theta(t))\cos(\alpha(t)) \left(\frac{d}{dt}\alpha(t)\right) l_p - \sin(\theta(t)) \left(\frac{d}{dt}\theta(t)\right) r \right)^2 \qquad [s23]$$
$$+ \left(-\sin(\theta(t)) \left(\frac{d}{dt}\theta(t)\right) \sin(\alpha(t))\, l_p + \cos(\theta(t))\cos(\alpha(t)) \left(\frac{d}{dt}\alpha(t)\right) l_p + \cos(\theta(t)) \left(\frac{d}{dt}\theta(t)\right) r \right)^2$$
$$+ \sin(\alpha(t))^2 \left(\frac{d}{dt}\alpha(t)\right)^2 l_p^2 \Bigg)$$

0	1	2

4.1.3.2. Potential Energy

In the section, the potential energy of the SIP is to be found. The basic formula to compute the gravitational potential energy of an object with mass *m* is

$$V = m\,g\,dh\,, \qquad [30]$$

where *g* is the gravitational acceleration constant and *dh* is the change in the object height, e.g. if it's moved from h_1 to h_2 then $dh = h_2 - h_1$.

1. Using kinematics derived, find the gravitational potential energy of the rotary SIP system. Make sure the energy is in terms of the pendulum angle.

Solution:

The potential energy of the SIP based on the kinematics is

$$V = m_p\,g\,z_p\,. \qquad [s24]$$

Substituting the z_p coordinate from [s12] into the above equation

$$V = -m_p\,g\cos(\alpha(t))\,l_p\,. \qquad [s25]$$

0	1	2

4.1.3.3. Lagrangian

Now the Lagrangian of the SIP system can be found using Equation [6].

1. Compute the Lagrangian, *L*, of the single inverted pendulum system.

Solution:

To get the Lagrange, substitute the total kinetic energy found in [s23] and the total potential energy in [s25] into Equation [6]

$$
\begin{aligned}
L = {} & \frac{1}{2} J_{arm} \left(\frac{d}{dt}\theta(t)\right)^2 + \frac{1}{2} J_p \left(\frac{d}{dt}\alpha(t)\right)^2 + \frac{1}{2} m_p \Bigg(\\
& \left(-\cos(\theta(t))\left(\frac{d}{dt}\theta(t)\right)\sin(\alpha(t))\,l_p - \sin(\theta(t))\cos(\alpha(t))\left(\frac{d}{dt}\alpha(t)\right)l_p - \sin(\theta(t))\left(\frac{d}{dt}\theta(t)\right)r\right)^2 \\
& + \left(-\sin(\theta(t))\left(\frac{d}{dt}\theta(t)\right)\sin(\alpha(t))\,l_p + \cos(\theta(t))\cos(\alpha(t))\left(\frac{d}{dt}\alpha(t)\right)l_p + \cos(\theta(t))\left(\frac{d}{dt}\theta(t)\right)r\right)^2 \\
& + \sin(\alpha(t))^2\left(\frac{d}{dt}\alpha(t)\right)^2 l_p^2\Bigg) + m_p\,g\cos(\alpha(t))\,l_p
\end{aligned}
\qquad \text{[s26]}
$$

0	1	2

4.1.4. Nonlinear Equations of Motion

Using the Euler-Lagrange formula in [1], the equations representing the motions of the rotary arm and the pendulum of the SIP system can be obtained.

4.1.4.1. Obtaining the Nonlinear Equations of Motion

Go through these exercises to find the original SIP EOMs:

1. Show how to obtain the nonlinear EOM given in [10] using the Euler-Lagrange equation. Make sure the derivative calculations are shown.

Solution:

Substituting the Lagrange obtained in [s26] and the generalized forced coordinate in [7] into the Euler-Lagrange Equation [4] will setup the equation. Then, the derivatives must be computed. The derivative of the Lagrangian with respect to the rotary arm angle is

$$\begin{aligned}\frac{\partial}{\partial \theta} L = {} & \frac{1}{2} m_p \big(2 \left(-\cos(\theta) \sin(\alpha) l_p - \sin(\theta) r\right) \left(\sin(\theta) \sin(\alpha) l_p - \cos(\theta) r\right) \\ & + 2 \left(-\sin(\theta) \sin(\alpha) l_p + \cos(\theta) r\right) \left(-\cos(\theta) \sin(\alpha) l_p - \sin(\theta) r\right)\big) \left(\frac{d}{dt} \theta(t)\right)^2 + \frac{1}{2} m_p \\ & \left(-2 \sin(\theta) \cos(\alpha) l_p \left(\sin(\theta) \sin(\alpha) l_p - \cos(\theta) r\right) - 2 \left(-\sin(\theta) \sin(\alpha) l_p + \cos(\theta) r\right) \sin(\theta) \cos(\alpha) l_p\right) \\ & \left(\frac{d}{dt} \alpha(t)\right) \left(\frac{d}{dt} \theta(t)\right)\end{aligned} \quad [s27]$$

The derivative of L with respect to the rate of the arm angle is

$$\begin{aligned}\mathrm{Diff}\left(L, \frac{d}{dt} \theta(t)\right) = {} & J_{arm} \left(\frac{d}{dt} \theta(t)\right) + \frac{1}{2} m_p \Big(2 \\ & \left(-\cos(\theta) \left(\frac{d}{dt} \theta(t)\right) \sin(\alpha) l_p - \sin(\theta) \cos(\alpha) \left(\frac{d}{dt} \alpha(t)\right) l_p - \sin(\theta) \left(\frac{d}{dt} \theta(t)\right) r\right) \\ & \left(-\cos(\theta) \sin(\alpha) l_p - \sin(\theta) r\right) + 2 \\ & \left(-\sin(\theta) \left(\frac{d}{dt} \theta(t)\right) \sin(\alpha) l_p + \cos(\theta) \cos(\alpha) \left(\frac{d}{dt} \alpha(t)\right) l_p + \cos(\theta) \left(\frac{d}{dt} \theta(t)\right) r\right) \\ & \left(-\sin(\theta) \sin(\alpha) l_p + \cos(\theta) r\right)\Big)\end{aligned} \quad [s28]$$

Then, taking the time derivative of this gives

$$\begin{aligned}\frac{\partial}{\partial t} \mathrm{Diff}\left(L, \frac{d}{dt} \theta(t)\right) = {} & \\ & \left(J_{arm} + \frac{1}{2} m_p \left(2 \left(-\cos(\theta) \sin(\alpha) l_p - \sin(\theta) r\right)^2 + 2 \left(-\sin(\theta) \sin(\alpha) l_p + \cos(\theta) r\right)^2\right)\right) \left(\frac{d^2}{dt^2} \theta(t)\right) + \frac{1}{2} m_p \\ & \left(-2 \left(-\cos(\theta) \sin(\alpha) l_p - \sin(\theta) r\right) \sin(\theta) \cos(\alpha) l_p + 2 \left(-\sin(\theta) \sin(\alpha) l_p + \cos(\theta) r\right) \cos(\theta) \cos(\alpha) l_p\right) \\ & \left(\frac{d^2}{dt^2} \alpha(t)\right)\end{aligned} \quad [s29]$$

Inserting [s27] and [s29] into Equation [4] and doing the computation results in the nonlinear EOM shown in [10].

0 | 1 | 2

2. Show how to get the nonlinear EOM given in [11] using the Euler-Lagrange equation.

Solution:

The derivative of the Lagrangian with respect to the pendulum angle is

$$\frac{\partial}{\partial \alpha} L = \frac{1}{2} m_p$$
$$(-2(-\sin(\theta)\sin(\alpha) l_p + \cos(\theta) r)\sin(\theta)\cos(\alpha) l_p - 2(-\cos(\theta)\sin(\alpha) l_p - \sin(\theta) r)\cos(\theta)\cos(\alpha) l_p)$$
$$\left(\frac{d}{dt}\theta(t)\right)^2 + \frac{1}{2} m_p$$
$$(2(-\cos(\theta)\sin(\alpha) l_p - \sin(\theta) r)\sin(\theta)\sin(\alpha) l_p - 2(-\sin(\theta)\sin(\alpha) l_p + \cos(\theta) r)\cos(\theta)\sin(\alpha) l_p) \quad \text{[s30]}$$
$$\left(\frac{d}{dt}\alpha(t)\right)\left(\frac{d}{dt}\theta(t)\right)$$
$$+ \frac{1}{2} m_p (-2\sin(\theta)^2 \cos(\alpha) l_p^2 \sin(\alpha) + 2\sin(\alpha) l_p^2 \cos(\alpha) - 2\cos(\theta)^2 \cos(\alpha) l_p^2 \sin(\alpha))\left(\frac{d}{dt}\alpha(t)\right)^2$$
$$- m_p g \sin(\alpha) l_p$$

The derivative of L with respect to the pendulum angle velocity equals

$$\mathrm{Diff}\left(L, \frac{d}{dt}\alpha(t)\right) = J_p \left(\frac{d}{dt}\alpha(t)\right) + \frac{1}{2} m_p \Bigg($$
$$-2\left(-\cos(\theta)\left(\frac{d}{dt}\theta(t)\right)\sin(\alpha) l_p - \sin(\theta)\cos(\alpha)\left(\frac{d}{dt}\alpha(t)\right) l_p - \sin(\theta)\left(\frac{d}{dt}\theta(t)\right) r\right)\sin(\theta)\cos(\alpha) l_p$$
$$+ 2\left(-\sin(\theta)\left(\frac{d}{dt}\theta(t)\right)\sin(\alpha) l_p + \cos(\theta)\cos(\alpha)\left(\frac{d}{dt}\alpha(t)\right) l_p + \cos(\theta)\left(\frac{d}{dt}\theta(t)\right) r\right)\cos(\theta)\cos(\alpha) l_p \quad \text{[s31]}$$
$$+ 2\sin(\alpha)^2 \left(\frac{d}{dt}\alpha(t)\right) l_p^2\Bigg)$$

Which results in

$$\frac{\partial}{\partial t}\mathrm{Diff}\left(L, \frac{d}{dt}\alpha(t)\right) = \frac{1}{2} m_p$$
$$(-2(-\cos(\theta)\sin(\alpha) l_p - \sin(\theta) r)\sin(\theta)\cos(\alpha) l_p + 2(-\sin(\theta)\sin(\alpha) l_p + \cos(\theta) r)\cos(\theta)\cos(\alpha) l_p) \quad \text{[s32]}$$
$$\left(\frac{d^2}{dt^2}\theta(t)\right) + \left(J_p + \frac{1}{2} m_p (2\sin(\theta)^2 \cos(\alpha)^2 l_p^2 + 2\cos(\theta)^2 \cos(\alpha)^2 l_p^2 + 2\sin(\alpha)^2 l_p^2)\right)\left(\frac{d^2}{dt^2}\alpha(t)\right)$$

after taking the time derivative.

Substituting the computed derivatives [s30] and [s32] into Equation [5] and performing the computation gives the EOM shown in [11]. 0 | 1 | 2

4.1.4.2. Euler-Lagrange Matrix Form

The Euler-Lagrange equations can also be written in matrix form as

$$D(q(t))\left(\frac{d^2}{dt^2} q(t)\right) + C\left(q(t), \frac{d}{dt} q(t)\right)\left(\frac{d}{dt} q(t)\right) + g(q(t)) = \tau \quad [31]$$

For a system with two coordinates such as the SIP, the *mass* or *inertia* matrix can be written as

$$D(q(t)) = \begin{bmatrix} d_{11} & d_{12} \\ d_{12} & d_{22} \end{bmatrix} \quad [32]$$

and the *damping* matrix as

$$C\left(q(t), \frac{d}{dt}q(t)\right) = \begin{bmatrix} c_{11} & c_{12} \\ c_{12} & c_{22} \end{bmatrix} \quad [33]$$

.

The vector

$$g(q(t)) = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} \quad [34]$$

includes gravitational forces and

$$\tau = \begin{bmatrix} \tau_m - B_{arm}\, qd_1 \\ B_p\, qd_2 \end{bmatrix} \quad [35]$$

is called the generalized force vector, which for the SIP consists of the load gear torque and the nonconservative damping torques $B_{arm}qd1$ and B_pqd2. Recall the generalized coordinates *q* and *qd* defined in [2] and [3].

Go through these exercises to determine all Euler-Lagrange matrix elements for the SIP:

1. From the nonlinear EOMs, enter the inertial matrix elements of the SIP in the following table. Express the elements in terms of the generalized coordinate defined earlier in [2].

Inertial Parameter	***Expression***
$d_{11}(q)$	$d_{11}(q_2) = -m_p l_p^2 \cos(q_2)^2 + J_{arm} + m_p r^2 + m_p l_p^2$
$d_{12}(q)$	$d_{12}(q_2) = m_p \cos(q_2)\, l_p\, r$
$d_{21}(q)$	$d_{21}(q_2) = m_p \cos(q_2)\, l_p\, r$
$d_{22}(q)$	$d_{22} = J_p + m_p l_p^2$

0	1	2

Table 2: Euler-Lagrange nonlinear inertial parameters.

2. Enter the damping terms of the SIP in Table 2, above. These expressions can be functions of the generalized coordinates and their corresponding time-derivatives. Make sure the expressions are given in terms of the generalized coordinates defined in [2] and [3].

Inertial Parameter	***Expression***
c_{11}(q,qd)	$c_{11}(q_2, qd_2) = 2\, m_p \cos(q_2)\, qd_2\, l_p^2 \sin(q_2)$
c_{12}(q,qd)	$c_{12}(q_2, qd_1) = -m_p \sin(q_2)\, qd_2\, l_p\, r$
c_{21}(q,qd)	$c_{21}(q_2, qd_1) = -m_p \cos(q_2)\, l_p^2\, qd_1 \sin(q_2)$
c_{22}(q,qd)	$c_{22} = 0$

0	1	2

Table 3: Euler-Lagrange damping parameters.

3. Complete Table 3, above, with the torques that are generated by gravity.

Inertial Parameter	***Expression***
g_1(q)	$g_1 = 0$
g_2(q)	$g_2(q_2) = m_p\, g \sin(q_2)\, l_p$

0	1	2

Table 4: Euler-Lagrange gravitational torque parameters.

4.1.5. Linear State-Space Model

The linear state-space representation of the SIP system is to be found. The nonlinear equations of the system must be first be linearized and then solved for the acceleration terms. This is done in Section 4.1.5.1. Thus the Euler-Lagrange matrix elements found above must be linearized to obtain the equation

$$\mathrm{D}_l(\mathrm{q}(t))\left(\frac{d^2}{dt^2}\mathrm{q}(t)\right) + C_l\left(\mathrm{q}(t), \frac{d}{dt}\mathrm{q}(t)\right)\left(\frac{d}{dt}\mathrm{q}(t)\right) + g_l(\mathrm{q}(t)) = \tau \qquad [36]$$

where D_l(q) is the linear inertia matrix, C_l(q,qd) is the linear damping matrix, and g_l(q) is the linearized gravitational torque vector. The applied torque is generated by a DC motor, therefore the generalized force vector, τ, is already linear.

Solving for the angular accelerations, the equation becomes

$$\frac{d^2}{dt^2}\mathrm{q}(t) = -\mathrm{D}_l(\mathrm{q}(t))^{[-1]}\left(C_l\left(\mathrm{q}(t), \frac{d}{dt}\mathrm{q}(t)\right)\left(\frac{d}{dt}\mathrm{q}(t)\right) + g_l(\mathrm{q}(t)) - \tau\right), \qquad [37]$$

where $D_l(q)^{-1}$ is the matrix inverse of D(q). This results in the linear EOMs in terms of the generalized coordinates.

State *x* is defined and introduced in Section 4.1.5.2 in order to get the state equations and thereafter, in

Section 4.1.5.3, the state-space matrices are derived.

4.1.5.1. Linear Equations of Motion

Consider a nonlinear function *f(z)* that maps the two-element vector

$$z = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \qquad [38]$$

to a scalar value, f(z): $z \in \Re^2 \rightarrow \Re^1$. The linear approximation of f(z) when linearized about the operating point

$$z_0 = \begin{bmatrix} a \\ b \end{bmatrix} . \qquad [39]$$

is

$$f_{lin}(z) = f(z_0) + \left(\frac{\partial}{\partial z_1} f(z)\right)\Bigg|_{z = z_0} (z_1 - a) + \left(\frac{\partial}{\partial z_2} f(z)\right)\Bigg|_{z = z_0} (z_2 - b) . \qquad [40]$$

Obtain the linear EOMs by linearizing all the parameter in the Euler-Lagrange matrices:

1. Linearize the inertial terms in the D(q) matrix in Table 2 above and enter them in the table below.

Inertial Parameter	***Expression***
$d_{11,l}(q)$	$d_{11,l} = J_{arm} + m_p r^2$
$d_{12,l}(q)$	$d_{12,l} = m_p l_p r$
$d_{21,l}(q)$	$d_{21,l} = m_p l_p r$
$d_{22,l}(q)$	$d_{22,l} = J_p + m_p l_p^2$

0	1	2

Table 5: Linearized Euler-Lagrange nonlinear inertial parameters.

2. Linearize the damping terms found in Table 3 and enter the expressions below.

Inertial Parameter	*Expression*
$c_{11,l}$(q,qd)	$c_{11,l} = 0$
$c_{12,l}$(q,qd)	$c_{12,l} = 0$
$c_{21,l}$(q,qd)	$c_{21,l} = 0$
$c_{22,l}$(q,qd)	$c_{22,l} = 0$

0 1 2

Table 6: Linearized Euler-Lagrange damping parameters.

3. Fill the table below with the linearized gravitational forces found in Table 4, above.

Inertial Parameter	*Expression*
$g_{1,l}$(q,qd)	$g_{1,l} = 0$
$g_{2,l}$(q,qd)	$g_{2,l} = m_p\, g\, l_p\, q_2$

0 1 2

Table 7: Linearized Euler-Lagrange gravitational torque parameters.

4. The linear Euler-Lagrange equation in [36] can now be solved for the acceleration terms, as described in [37]. First, find the inverse linearized inertial matrix $D_l(q)^{-1}$ and leave it in terms of the linear inertial parameters $d_{ij,l}$.

Solution:
The linearized inertial matrix is

$$D_l(q(t)) = \begin{bmatrix} d_{11,l} & d_{12,l} \\ d_{21,l} & d_{22,l} \end{bmatrix} \quad [s33]$$

.

When inverted, the matrix becomes

$$D_l(q(t))^{[-1]} = \begin{bmatrix} \frac{d_{22,l}}{d_{11,l}d_{22,l} - d_{12,l}d_{21,l}} & -\frac{d_{12,l}}{d_{11,l}d_{22,l} - d_{12,l}d_{21,l}} \\ -\frac{d_{21,l}}{d_{11,l}d_{22,l} - d_{12,l}d_{21,l}} & \frac{d_{11,l}}{d_{11,l}d_{22,l} - d_{12,l}d_{21,l}} \end{bmatrix} \quad [s34]$$

For the SIP system, it is easier to first linearize the equation and then solve for the acceleration terms. This makes the matrix inverse computation less tedious.

0 1 2

5. Show the linear Euler-Lagrange equations when still in matrix form and with respect to the

generalized coordinates, q and qd.

Solution:

Given that the damping terms are all zero, the solved Euler-Lagrange matrix equation becomes

$$\frac{d^2}{dt^2}\mathrm{q}(t) = \mathrm{D}_l(\mathrm{q}(t))^{[-1]}\,(\tau - g_l(\mathrm{q}(t))) \qquad \text{[s35]}$$

Substituting the inverse linear inertial matrix, the linear gravitational vector, and the generalized force vector results in the expression

$$\frac{d^2}{dt^2}\mathrm{q}(t) = \left[\frac{d_{22,l}(\tau_m - B_{arm}\,qd_1 - g_{1,l})}{d_{11,l}d_{22,l} - d_{12,l}d_{21,l}} - \frac{d_{12,l}(B_p\,qd_2 - g_{2,l})}{d_{11,l}d_{22,l} - d_{12,l}d_{21,l}},\; -\frac{d_{21,l}(\tau_m - B_{arm}\,qd_1 - g_{1,l})}{d_{11,l}d_{22,l} - d_{12,l}d_{21,l}} + \frac{d_{11,l}(B_p\,qd_2 - g_{2,l})}{d_{11,l}d_{22,l} - d_{12,l}d_{21,l}}\right] \qquad \text{[s36]}$$

The inertial parameters can now be added to obtain the linear Euler-Lagrange equations (in terms of the generalized coordinates)

$$\frac{d^2}{dt^2}\mathrm{q}(t) = \left[\frac{(J_p + m_p l_p^2)(\tau_m - B_{arm}\,qd_1)}{(J_{arm} + m_p r^2)(J_p + m_p l_p^2) - m_p^2 l_p^2 r^2} - \frac{m_p l_p r (B_p\,qd_2 - m_p g l_p q_2)}{(J_{arm} + m_p r^2)(J_p + m_p l_p^2) - m_p^2 l_p^2 r^2},\; -\frac{m_p l_p r(\tau_m - B_{arm}\,qd_1)}{(J_{arm} + m_p r^2)(J_p + m_p l_p^2) - m_p^2 l_p^2 r^2} + \frac{(J_{arm} + m_p r^2)(B_p\,qd_2 - m_p g l_p q_2)}{(J_{arm} + m_p r^2)(J_p + m_p l_p^2) - m_p^2 l_p^2 r^2}\right] \qquad \text{[s37]}$$

0	1	2

6. These equations are relative to the load output torque, τ_m. However, the input voltage of the servo motor is the control variable. Add the actuator dynamics to make these equations in terms of the input current Vm.

Solution:

The relationship between the load torque and the motor input current is given in Equation [9]. Placing this in terms of the generalized coordinates, given in [2], and substituting this into the above equations gives

$$\begin{aligned}\frac{d}{dt}qd_1(t) = {} & \frac{m_p^2\, l_p^2\, r\, g\, q_2}{(J_{arm} + m_p r^2)(J_p + m_p l_p^2) - m_p^2 l_p^2 r^2} \\ & + \frac{(J_p + m_p l_p^2)\left(-\dfrac{\eta_g K_g^2 \eta_m K_t K_m}{R_m} - B_{arm}\right) qd_1}{(J_{arm} + m_p r^2)(J_p + m_p l_p^2) - m_p^2 l_p^2 r^2} + \frac{(J_p + m_p l_p^2)\,\eta_g K_g \eta_m K_t V_m}{((J_{arm} + m_p r^2)(J_p + m_p l_p^2) - m_p^2 l_p^2 r^2) R_m} \\ & - \frac{m_p l_p r B_p qd_2}{(J_{arm} + m_p r^2)(J_p + m_p l_p^2) - m_p^2 l_p^2 r^2}\end{aligned} \qquad [s38]$$

and

$$\begin{aligned}\frac{d}{dt}qd_2(t) = {} & \\ & -\frac{m_p l_p r\left(\dfrac{\eta_g K_g \eta_m K_t (V_m - K_g K_m qd_1)}{R_m} - B_{arm} qd_1\right)}{(J_{arm} + m_p r^2)(J_p + m_p l_p^2) - m_p^2 l_p^2 r^2} + \frac{(J_{arm} + m_p r^2)(B_p qd_2 - m_p g l_p q_2)}{(J_{arm} + m_p r^2)(J_p + m_p l_p^2) - m_p^2 l_p^2 r^2}\end{aligned} \qquad [s39]$$

0	1	2

4.1.5.2. Introducing the State

Given the state vector

$$x^T = [x_1, x_2, x_3, x_4] \qquad [41]$$

we define the position states as

$$\{\theta(t) = x_1, \alpha(t) = x_2\} \qquad [42]$$

and the velocities as

$$\{\frac{d}{dt}\theta(t) = x_3, \frac{d}{dt}\alpha(t) = x_4\} \qquad [43]$$

.

The SIP plant which is now modeled and represented with state space matrices A, B, C and D, consists of two sensors namely the SRV02 motor shaft encoder and the pendulum encoder. These two sensors provide values for the first two states of the system respectively. The last two states are angular velocities of the first two and in this lab are estimated using high pass filters which differentiate the first two states respectively to obtain a complete 4 element state vector. The controller to be designed will be a proportional-derivative gain K, found using LQR. The reference input for the balance controller is a value of zero degrees (vertical upward position) giving rise to the reference vector [0 0 0 0]. The state vector which consists of the two encoder readings plus their derivatives is subtracted from the reference input vector and fed into the controller.

Follow these exercises to get the state equations:

1. Express the equations of motion obtained in Section 4.1.5.1 in terms of the state.

Solution:
Using the definitions for the angular positions and rates in [42] and [43], we can introduce the state using the following relationships for the generalized coordinates

$$\{q_2 = x_2, q_1 = x_1\} \tag{s40}$$

and

$$\{qd_2 = x_4, qd_1 = x_3\} \tag{s41}$$

Using these, the EOMs become

$$\begin{aligned}\frac{d}{dt}x_3(t) = &-\frac{(J_p \eta_g K_g^2 \eta_m K_t K_m + J_p B_{arm} R_m + m_p l_p^2 \eta_g K_g^2 \eta_m K_t K_m + m_p l_p^2 B_{arm} R_m)\, x_3}{(J_{arm} J_p + J_{arm} m_p l_p^2 + m_p r^2 J_p)\, R_m}\\ &- \frac{m_p l_p r B_p x_4}{J_{arm} J_p + J_{arm} m_p l_p^2 + m_p r^2 J_p} - \frac{(-J_p \eta_g K_g \eta_m K_t - m_p l_p^2 \eta_g K_g \eta_m K_t)\, V_m}{(J_{arm} J_p + J_{arm} m_p l_p^2 + m_p r^2 J_p)\, R_m}\\ &+ \frac{m_p^2 l_p^2 r g x_2}{J_{arm} J_p + J_{arm} m_p l_p^2 + m_p r^2 J_p}\end{aligned} \tag{s42}$$

and

$$\begin{aligned}\frac{d}{dt}x_4(t) = &\frac{(m_p l_p r \eta_g K_g^2 \eta_m K_t K_m + m_p l_p r B_{arm} R_m)\, x_3}{(J_{arm} J_p + J_{arm} m_p l_p^2 + m_p r^2 J_p)\, R_m} + \frac{(R_m J_{arm} B_p + R_m m_p r^2 B_p)\, x_4}{(J_{arm} J_p + J_{arm} m_p l_p^2 + m_p r^2 J_p)\, R_m}\\ &- \frac{m_p l_p r \eta_g K_g \eta_m K_t V_m}{(J_{arm} J_p + J_{arm} m_p l_p^2 + m_p r^2 J_p)\, R_m} + \frac{(-R_m J_{arm} m_p g l_p - R_m m_p^2 r^2 g l_p)\, x_2}{(J_{arm} J_p + J_{arm} m_p l_p^2 + m_p r^2 J_p)\, R_m}\end{aligned} \tag{s43}$$

0	1	2

4.1.5.3. Obtaining the Final State-Space Model

The linear state-space equations are

$$\frac{\partial}{\partial t} x = A\,x + B\,u \tag{44}$$

and

$$y = C\,x + \mathrm{D}\,u \tag{45}$$

The (A,B) matrices can be obtained from the EOMs above. As for the output equation, only the position measurements of the arm and pendulum angles are available and measurement noise will be neglected. Thus the output equation can be written

$$y = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} . \qquad [46]$$

Go through these exercises to obtain the linear state-space representation of the SIP:

1. Enter the first two rows of the state-space matrix A elements in the table below.

0	1	2

Matrix A Element	*Expression*
A[1,1]	0
A[1,2]	0
A[1,3]	1
A[1,4]	0
A[2,1]	0
A[2,2]	0
A[2,3]	0
A[2,4]	1

Table 8: State-space matrix A: first and second row elements.

2. Complete Table 9 with the third row state-space matrix A elements.

Matrix A Element	***Expression***
A[3,1]	0
A[3,2]	$\dfrac{m_p^2\, l_p^2\, r\, g}{J_{arm}\, m_p\, l_p^2 + J_{arm}\, J_p + m_p\, r^2\, J_p}$
A[3,3]	$-\dfrac{m_p\, l_p^2\, \eta_g\, K_g^2\, \eta_m\, K_t\, K_m + m_p\, l_p^2\, B_{arm}\, R_m + J_p\, B_{arm}\, R_m + J_p\, \eta_g\, K_g^2\, \eta_m\, K_t\, K_m}{(J_{arm}\, m_p\, l_p^2 + J_{arm}\, J_p + m_p\, r^2\, J_p)\, R_m}$
A[3,4]	$-\dfrac{m_p\, l_p\, r\, B_p}{J_{arm}\, m_p\, l_p^2 + J_{arm}\, J_p + m_p\, r^2\, J_p}$

Table 9: State-space matrix A: third row elements.

0	1	2

3. Enter the fourth row elements of matrix A in Table 10.

Matrix A Element	***Expression***
A[4,1]	0
A[4,2]	$-\dfrac{m_p\, g\, l_p\, (J_{arm} + m_p\, r^2)}{J_{arm}\, m_p\, l_p^2 + J_{arm}\, J_p + m_p\, r^2\, J_p}$
A[4,3]	$\dfrac{m_p\, l_p\, r\, (B_{arm}\, R_m + \eta_g\, K_g^2\, \eta_m\, K_t\, K_m)}{(J_{arm}\, m_p\, l_p^2 + J_{arm}\, J_p + m_p\, r^2\, J_p)\, R_m}$
A[4,4]	$\dfrac{B_p\, (J_{arm} + m_p\, r^2)}{J_{arm}\, m_p\, l_p^2 + J_{arm}\, J_p + m_p\, r^2\, J_p}$

Table 10: State-space matrix A: fourth row elements.

0	1	2

4. Enter matrix B in Table 11.

Matrix B Element	***Expression***
B[1]	0
B[2]	0
B[3]	$\frac{\eta_g K_g \eta_m K_t (m_p l_p^2 + J_p)}{(J_{arm} m_p l_p^2 + J_{arm} J_p + m_p r^2 J_p) R_m}$
B[4]	$-\frac{m_p l_p r \eta_g K_g \eta_m K_t}{(J_{arm} m_p l_p^2 + J_{arm} J_p + m_p r^2 J_p) R_m}$

0	1	2

Table 11: State-space matrix B.

5. Fill the table below with the C matrix values.

Matrix B Element	***Expression***
C[1,1]	1
C[1,2]	0
C[1,3]	0
C[1,4]	0
C[2,1]	0
C[2,2]	1
C[2,3]	0
C[2,4]	0

0	**1**	**2**

Table 12: State-space matrix C.

6. Fill Table 13 with the matrix D elements.

Matrix B Element	***Expression***
D[1]	0
D[2]	0
D[3]	0
D[4]	0

0	**1**	**2**

Table 13: State-space matrix D.

4.2. Inverted Pendulum Balance Control Design

As mentioned at the beginning of this document, this laboratory consists of two separate controller design steps. One controller will be developed for swinging the pendulum up from the vertical downward position, and another will be designed to balance the pendulum in the vertical upward position. The first controller is an energy based swing up controller while the second is a state-feedback PD controller obtained using LQR. This section is dedicated to designing the balance controller.

The state-feedback controller enters the servo motor and is expressed as

$$I_m(t) = K\,(x_d(t) - \mathrm{x}(t)) \qquad [47]$$

where

$$K = [k_{p,\theta}, k_{p,\alpha}, k_{d,\theta}, k_{d,\alpha}] \qquad [48]$$

is the control gain and

$$x_d(t)^T = [\theta_d(t), 0, 0, 0] \qquad [49]$$

is the desired state as stated above. Remark that this is a proportional-derivative controller with servo and pendulum proportional gains $k_{p,\theta}$ and $k_{p,\alpha}$ and servo and pendulum derivative gains $k_{d,\theta}$ and $k_{d,\alpha}$. Instead the compensator can be expressed in terms of the actual angles, by substituting the states defined in [42] and [43], to get

$$I_m(t) = k_{p,\theta}\,(\theta_d(t) - \theta_l(t)) - k_{p,\alpha}\,\alpha(t) - k_{d,\theta}\left(\frac{d}{dt}\theta_l(t)\right) - k_{d,\alpha}\left(\frac{d}{dt}\alpha(t)\right) \qquad [50]$$

In terms of the control design, it is assumed that all the states are measured. That is, the position and velocity of the servo and the pendulum are measured using sensors. However as mentioned in section 4.1.5.2 in the actual plant there are only sensors measuring the positions of the servo and pendulum. The velocities are computed digitally using high-gain observers and their result is taken as being "measured" for the purposes of the control design. The velocity is computed by taking the derivative of the position and filtering the result using a second-order low-pass filter (LPF). In effect, the state velocities are obtained using a high-pass filter of the form

$$\mathrm{H}(s) = \frac{\omega_f s}{s^2 + 2\,\zeta_f \omega_f + \omega_f^2} \qquad [51]$$

where ω_f is the natural frequency of the filter and ζ_f is the damping ratio of the filter.

4.2.1. LQR

A system is deemed as being *controllable* if its poles can be placed at any desired location via state-feedback. One method of determining if a system is controllable is called the *rank test*:

$$\mathrm{rank}(C_o) = n \qquad [52]$$

The matrix C_o is called the *controllability matrix.* For a four-state system such as the inverted pendulum it is computed by

$$C_o = [B, A\,B, A^2\,B, A^3\,B] \qquad [53]$$

If the rank of the controllability matrix equals the amount of states, i.e. if $n = 4$, then the system is controllable and a state-feedback control can be designed.

Assuming (A,B) is controllable, the control gain K in Equation [48] can be computed using the Linear-Quadratic Regular (LQR) optimization method. For the user-defined weighting matrices Q and R, LQR finds a signal u(t) that minimizes the cost function

$$J = \int_0^\infty \mathrm{x}(t)^T Q\,\mathrm{x}(t) + \mathrm{u}(t)^T R\,\mathrm{u}(t)\,dt \qquad [54]$$

With the state-feedback control

$$u = -K\,x \qquad [55]$$

LQR will compute a gain K that minimizes the *J* expression. For the SIP system, the weighting matrices will be chosen as

$$Q = \begin{bmatrix} q_1 & 0 & 0 & 0 \\ 0 & q_2 & 0 & 0 \\ 0 & 0 & q_3 & 0 \\ 0 & 0 & 0 & q_4 \end{bmatrix} \qquad [56]$$

and

$$R = 15 \qquad [57]$$

Generally speaking if R is kept constant and the diagonal elements in the Q matrix are increased then LQR will work harder to minimize *J* and the gains generated will be larger. Instead, matrix R can be varied. To generate a larger control gain, decrease the value of R while keeping Q constant. This way the algorithm must work harder against the smaller R value to minimize *J* and will yield a larger control gain.

To gain some intuition on LQR go through the following exercises:

1. One of the problems with the LQR method is that there are little guidelines in choosing the

weighting matrices. Typically the Q and R matrices are initially set to their corresponding identity matrices, the optimization algorithm is ran, and the system is simulated using the generated gain to observe the closed-loop response and check whether the specifications are satisfied. However, it can be useful to gain some insight on how the weighting parameters affect the gain being generated. Expand the inside of the cost function

$$x(t)^T Q\, x(t) + u(t)^T R\, u(t) \qquad [58]$$

such that it is in terms of the weighting parameters q_i, the proportional and derivative gains, and the states. Use the value of R given in equation [57].

Solution:

Given that R = 15, the control input is a scalar and the cost function becomes

$$J = (x(t)^T * Q * x(t)) + 15\,u(t)^2 \qquad [s44]$$

Using the state-feedback control in [55], the Q weighting matrices in [56], and gain R given in [57], into the above expression gives

$$J = x_1^2 q_1 + x_2^2 q_2 + x_3^2 q_3 + x_4^2 q_4 + 15(x_1 k_{p\theta} + x_2 k_{p\alpha} + x_3 k_{d\theta} + x_4 k_{d\alpha})^2 \qquad [s45]$$

0	1	2

2. Based on the result, can you determine which gains are primarily affected by which weighting parameter? Give the correlations between the q_i and the gains in K.

Solution:

The result obtained can be expanded and then collected with respect to the states to get

$$\begin{aligned} J_{int} = {} & (k_{p,\theta}^2 + q_1) x_1^2 + (2 k_{p,\theta} x_2 k_{p,\alpha} + 2 k_{p,\theta} x_3 k_{d,\theta} + 2 k_{p,\theta} x_4 k_{d,\alpha}) x_1 \\ & + (q_3 + k_{d,\theta}^2) x_3^2 + (2 x_2 k_{p,\alpha} k_{d,\theta} + 2 k_{d,\theta} x_4 k_{d,\alpha}) x_3 + (q_4 + k_{d,\alpha}^2) x_4^2 \\ & + 2 x_2 k_{p,\alpha} x_4 k_{d,\alpha} + (k_{p,\alpha}^2 + q_2) x_2^2 \end{aligned} \qquad [s58]$$

It is evident from the x_1^2 coefficient that the arm proportional gain is directly affected by the choice of the q_1 parameter. Similarly from the x_2^2 term, the pendulum proportional gain $k_{p,\alpha}$ is primarily affected by q_2. By the same token, the arm and pendulum derivative gains, $k_{d,\theta}$ and $k_{d,\alpha}$, are mainly determined by how the q_3 and q_4 weighting parameters are chosen.

0	1	2

4.3. Inverted Pendulum Swing Up Controller Design

In this section a control scheme is developed for swinging up the pendulum from its vertical downward position. The controller is an energy based controller.

From the earlier sections, the potential energy of the SIP system can be written as:

$$E_p = M_p\, g\, l_p\, (1 - \cos(\alpha(t))) \tag{59}$$

where M_p denotes the pendulum mass, g denotes the gravitational constant and l_p denotes the pendulum length. In addition the kinetic energy of the SIP system is:

$$E_k = \frac{1}{2} J_p \left(\frac{d}{dt}\alpha(t)\right)^2 \tag{60}$$

Where J_p denotes the pendulum moment of inertia.

Summing the above energies we obtain the following for the total energy of the SIP system:

$$E = \frac{1}{2} J_p \left(\frac{d}{dt}\alpha(t)\right)^2 + M_p\, g\, l_p\, (1 - \cos(\alpha(t))) \tag{61}$$

Differentiating with respect to time we obtain:

$$\frac{\partial}{\partial t} E = \left(\frac{d}{dt}\alpha(t)\right)\left(\left(\frac{d^2}{dt^2}\alpha(t)\right) J_p + M_p\, g\, l_p \sin(\alpha(t))\right) \tag{62}$$

1. In order to introduce the control variable *u*, into the above equation solve for *sin(α(t))* in the non-linear equation of motion you found in section 4.1.4 and substitute the resulting expression into [62].

Solution:
Solving for *sin(α(t))* in the non-linear equation of motion found in section 4.1.4 we obtain:

$$\sin(\alpha(t)) = \frac{-J_p\left(\frac{d^2}{dt^2}\alpha(t)\right) + M_p\, u\, l_p \cos(\alpha(t))}{M_p\, g\, l_p} \quad [S59]$$

By substituting [S59] into [62] we obtain:

$$\frac{\partial}{\partial t}E = M_p\, u\, l_p \cos(\alpha(t))\left(\frac{d}{dt}\alpha(t)\right) \quad [S60]$$

0	1	2

A non-linear controller that swings the pendulum up to achieve a given reference energy E_r has the following form:

$$u = (E - E_r)\cos(\alpha(t))\left(\frac{d}{dt}\alpha(t)\right) \quad [63]$$

However for energy to change quickly the magnitude of the control signal must be fairly large. As a result a tunable gain μ is multiplied by the above and the controller is saturated at the maximum acceleration deliverable by the motor u_{max} :

$$u = sat_{u_{max}}\left(\mu\,(E - E_r)\,\mathrm{Sign}\left(\cos(\alpha(t))\left(\frac{d}{dt}\alpha(t)\right)\right)\right) \quad [64]$$

As found in [7] the torque at load gear of the SRV02 is given by:

$$\tau_m = \eta_g K_g \eta_m K_t I_m \quad [65]$$

Where η_g is the gearbox efficiency, K_g is the gear ratio, η_m is the motor efficiency, and K_t is the current-torque constant.

2. Using the following torque and linear acceleration relationship, write down the complete equation describing the relationship between controller acceleration and motor input current.

$$\tau_m = m_{arm} l_{arm} u \quad [66]$$

Solution:

By equating [66] and [65] we obtain the following relationship between the controller acceleration and motor input current:

$$I_m = \frac{m_{arm} l_{arm} u}{\eta_g K_g \eta_m K_t} \quad [S61]$$

0	1	2

3. It was shown in equation [63] that with a given reference energy, a controller can be implemented that swings the pendulum up to achieve that given energy. Calculate the reference pendulum energy that results in the pendulum to balance in its vertical upward position. You can refer to the supplied file *config_sp.m* to obtain system parameter values such as pendulum mass and length if required.

Solution:

From equation [61] we know that the total energy of the pendulum is:

$$E = \frac{1}{2} J_p \left(\frac{d}{dt} \alpha(t) \right)^2 + M_p \, g \, l_p (1 - \cos(\alpha(t))) \qquad \text{[S62]}$$

At the vertical upward position the pendulum angle is constant at 180 degrees. Hence the first term in [S63] is zero since the derivative of the now constant angle is zero. This can also be concluded from the fact that when the pendulum is at its vertical upward position it is not moving and hence its kinetic energy is zero. To evaluate the second term, pendulum mass and length are found from the file *config_sp.m:*

$M_p = 0.127\ kg$

$l_p = 0.156\ m$

Substituting the above values into [S63] we obtain:

$$E_r = 0 + 0.127 * 9.81 * 0.156 (1 - (-1)) = 0.3877$$

0	1	2

5. In-Lab Procedures

The *q_sesip* Simulink diagram shown in Figure 3 is used to control the SRV02 self erecting single inverted pendulum system using the QUARC software. The *SRV02-E+SIP-QuaRC* subsystem contains QUARC blocks that interface with the DC motor and sensors (encoders) of the SIP system.

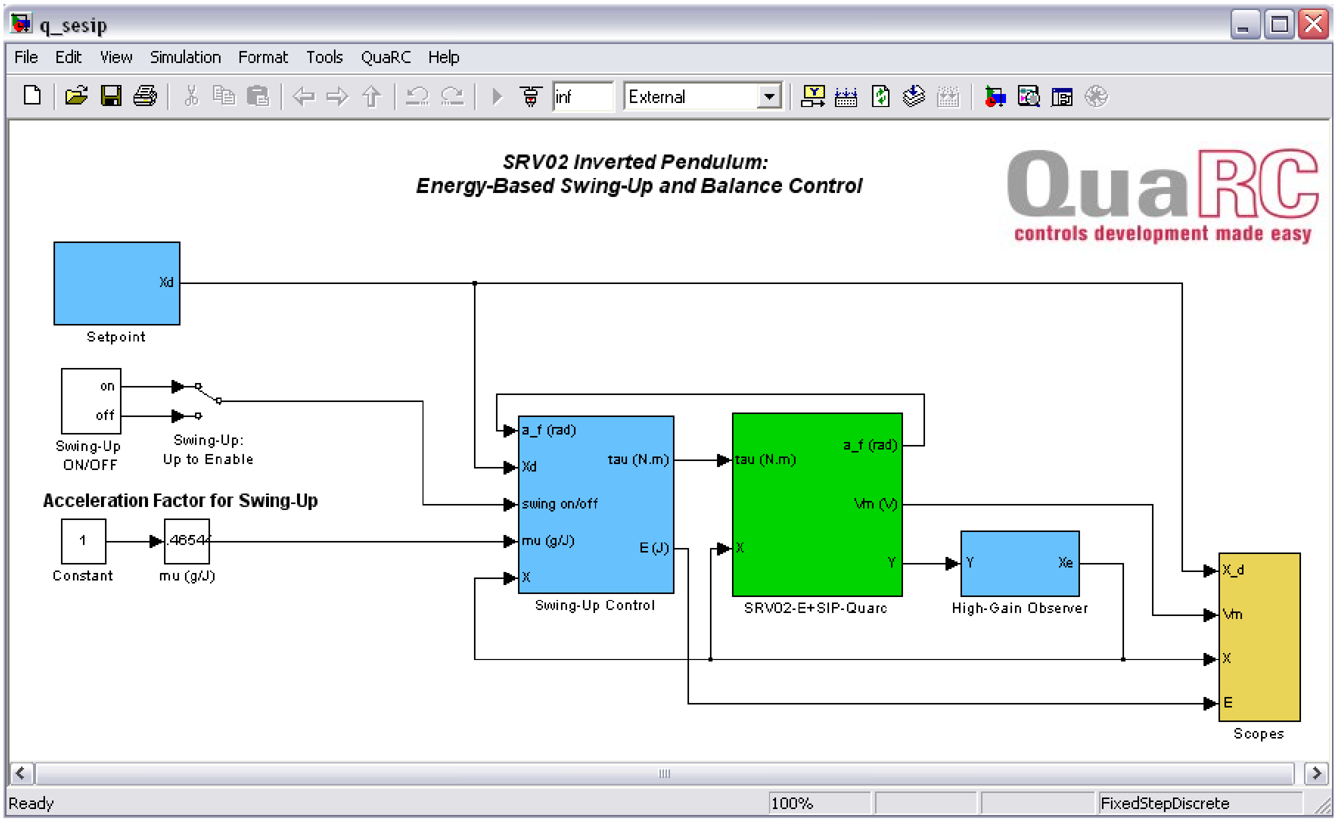


Figure 3: Simulink model used to control the SRV02 self erecting SIP system

In the *Setpoint* block the user can adjust the reference input to the system when the balance controller is running. The reference has been set to a sinewave with a frequency of 0.2 Hz. The amplitude of the reference signal can be entered in degrees in the *Amplitude (deg)* gain block.

The "*Swing-Up: Up to Enable"* block can be used to switch between manual swing up and automatic swing up of the pendulum. In the "Off" position the user must manually swing the pendulum up to its vertical upward position at which point the balance controller kicks in and balanced the pendulum at that position. In the "On" position however the swing up is done by the energy based controller discussed in the Pre-Laboratory section.

The *Swing-Up Control* block consists of three subsystems itself and implements the two controllers used in this laboratory. The *Balance Control* subsystem implements a state-feedback controller obtained using LQR to balance the pendulum in its vertical upward position. The *SRV02 Swing-Up Control* subsystem implements the energy based controller to swing the pendulum up and finally the *Mode-Switching Strategy* block implements the switching strategy between the two controllers. There are two conditions that should be met in order for the switch to take place. First, the pendulum angle has to be within 2.5 degrees of the vertical upward position and second the angular velocity of the pendulum has to be smaller than or equal to 17.6 rad/s. These conditions are checked using the two pre-defined variables *CATCH_ALPHA_UP_LIM* and *CATCH_ALPHA_DOT_LIM* which are set to values mentioned above in the setup script supplied.

The *High-Gain Observer* block contains two second-order high-pass filters that compute the angular rates of the servo and pendulum and outputs the corresponding estimated state, X_e. Remark that although the rates can be computed by the Simulink *State-Space* block, the filtering is used to mimic the actual plant as closely as possible (the high-pass filters are used in the implemented controller).

Follow these steps to implement a complete control system on the actual SIP plant:

1. Load the MATLAB software.
2. Browse through the *Current Directory* window in MATLAB and find the folder than contains the controller files.
3. Double-click on the *q_sesip.mdl* file to open the Simulink diagram shown in Figure 3.
4. Double-click on the *setup_srv02_exp08_sip.m* to open the setup script for this laboratory.
5. **Configure the setup script:** When used with the SIP, the SRV02 must be in the high-gear configuration and no load is to be specified. Make sure the script is setup to match this configuration, i.e. the EXT_GEAR_CONFIG should be set to 'HIGH' and the LOAD_TYPE should be set to 'NONE'. Also, ensure the ENCODER_TYPE, TACH_OPTION, K_CABLE, AMP_TYPE, and VMAX_DAC parameters are set according to the SRV02 system that is to be used in the laboratory. Next, make sure CONTROL_TYPE is set to 'MANUAL'.

Note to Instructor:

Set CONTROL_TYPE = 'AUTO' to automatically calculate the LQR control gains and reference energy for swing up control. If CONTROL_TYPE is set to 'MANUAL' the students are asked to enter their calculated reference energy.

The students should not have access to the script *calculate_qr.m* as this file contains values for parameters that students are supposed to calculate themselves. However, exactly what should be given to the students is at the discretion of the instructor.

5.1. Reference Energy Calculation

Run the setup script *setup_srv02_exp08_sip.m* by selecting the Debug | Run item from the menu bar or clicking on the *Run* button in the tool bar. A message will appear in the MATLAB Command Prompt asking you to enter your calculated reference energy value. Type in the value you found in question 3 of section 4.3 and hit enter. The MATLAB Command Prompt window will show the SRV02 model parameters and a state-feedback gain vector which is currently set to zero. You will design this state-feedback gain using the LQR method as outlined in the next section.

5.2. LQR Control Design

Upon running the setup script *setup_srv02_exp08_sip.m* the linear state space model of the SIP system is loaded into the MATLAB workspace under the matrices A, B, C and D.

1. Using MATLAB and the content in section 4.2.1 show that the SIP system is controllable.

Solution:
As discussed in Section 4.2.1, the system can be deemed controllable by performing the rank test. The controllability matrix for a state space model can be formed by using the MATLAB '*ctrb'* function. You pass the A and B matrices as arguments to this function and it return the controllability matrix associated with that system. The rank of any matrix can be computed using the MATLAB *'rank'* command. The matrix is passed as an argument to this function and it returns the rank. As shown below the above two functions were ran in MATLAB and the result shows that the controllability matrix is indeed full rank and hence the system is controllable.

```
>> ctrb(A,B)

ans =

  1.0e+004 *

         0    0.0274   -0.0001    1.4649
         0    0.0274    0.0153    2.7140
    0.0274   -0.0001    1.4649    1.6339
    0.0274    0.0153    2.7140    3.8513

>> rank(ctrb(A,B))

ans =

     4
```

0	1	2

2. Using the A and B matrices loaded into the MATLAB workspace, the weighting matrices Q and R given below and the MATLAB *'lqr'* command, generate a state-feedback control gain and save it under a variable called *k*. You should first store the Q and R matrices given below in the workspace under the names Q_lqr and R_lqr respectively and then run the *'lqr'* function. Record the MATLAB commands used and the corresponding generated control gain.

$$Q = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \text{ and } R = 10 \quad [68]$$

Solution:

See text below for the commands used to generate a control gain using the MATLAB *'lqr'* command.

```
>> Q_lqr = [1 0 0 0; 0 1 0 0; 0 0 0 0; 0 0 0 0];

>> R_lqr = 10;

>> k = lqr(A,B, Q_lqr, R_lqr)

k =

   -0.31623    1.9146   -0.14956    0.26268

>>
```

0	1	2

5.3. Setup for Control Implementation

Before running the above found controller on the actual SIP system, the *q_sesip* Simulink diagram and the *setup_srv02_exp08_sip.m* script must be confgiured.

Follow these steps to get the system ready for this lab:

1. Setup the SRV02 with the SIP module as detailed in Reference [9].
2. Load the MATLAB software.
3. Browse through the *current directory* window in MATLAB and find the folder that contains the QUARC SIP control file called *q_sesip.mdl*.
4. Double-click on the *q_sesip.mdl* file to open the self erecting single inverted pendulum control Simulink diagram shown in Figure 3.
5. **Configure DAQ**: Ensure the HIL Initialize block in the *SRV02-E + SIP – QuaRC* subsystem is configured for the DAQ device that is installed in your system. See Reference [6] for more information on configuring the HIL Initialize block.

6. **Configure setup script**: Set the parameters in the *setup_srv02_exp08_sip.m* script according to your system setup. See step 5 on page 37 for more details.

5.3.1. Balance and Swing-Up Controller Implementation

In this section the energy based swing up controller explained in section 4.3 and the LQR controller that was designed in question 2 of section 5.2 are ran on the actual SIP system.

Follow these steps to implement the controller:

1. The designed LQR gain found earlier should be already saved in the MATLAB workspace under the variable *k*. In addition the reference energy for swing up control that you found in question 1 of section 5.1 should be already saved in the MATLAB workspace. As mentioned earlier you are asked to enter this value when you run the setup script *setup_srv02_exp08_sip.m*.
2. To run the energy based swing up controller and the LQR controller consecutively, ensure that the *Swing-Up: Up to Enable* switch is set to the **upward position**. If this switch is set to the downward position the swing up of the pendulum must be done manually and once the pendulum is within 2 degrees of its vertical upward position the balance LQR controller kicks in and balanced the pendulum.
3. Select QUARC | Build to build the real-time code for this Simulink model.
   Once the code is successfully built and downloaded to target, select QUARC | Start to begin running the controller.

The energy based swing up controller starts acting by swinging the pendulum left and right until the vertical upward position is reached at which point the LQR balance controller kicks in and balances the pendulum. There are watchdogs implemented in the model that shut down the controller in case the SRV02 motor shaft angle or the pendulum angle exceed a threshold to avoid damage to the device. You might have to run the controller for multiple times in order to achieve a successful switch between the controllers. You can also decrease the swing-up acceleration factor in the *mu(g/J)* slider block if you are not achieving a successful transition between the controllers. Once the balance controller kicks in and successfully balances the pendulum double-click on the *Scopes* sub-system where you can monitor various available signals present in the system such as the SRV02 motor shaft angle, the pendulum angle, input current to the SRV02 motor and the pendulum energy.

1. Provide plots of the LQR controller response, showing the SRV02 angle and the pendulum angle.

Solution:

The closed-loop response of the SIP system using the tuned LQR gain in question 2 of section 5.2 is given below.

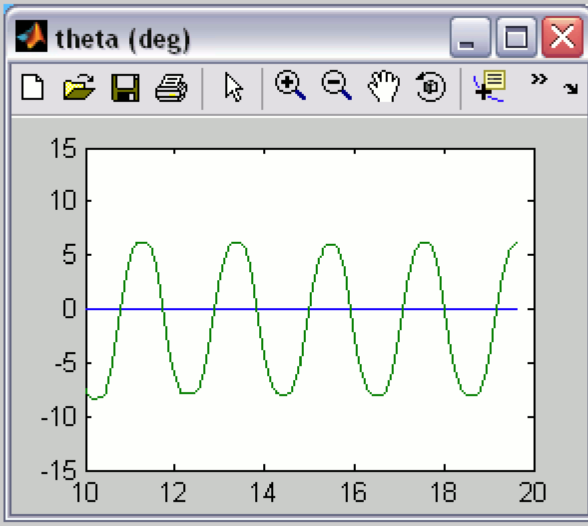


Figure 4: SRV02 Motor Shaft Angle Response

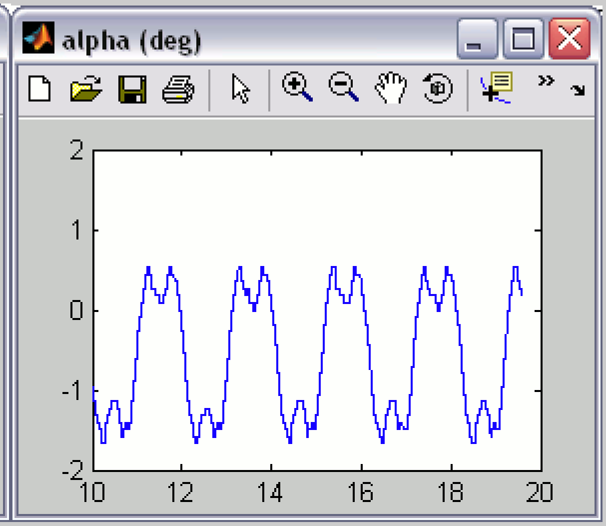


Figure 5: Pendulum Angle Response

Figure 4 shows the SRV02 motor shaft angle response to the LQR controller implemented while Figure 5 shows the pendulum angle under the same controller. As seen from the right figure and pendulum has been balanced about zero degrees (vertical upward position) with small disturbances resulting from movement of the motor shaft which is trying to keep the pendulum balanced.

0	**1**	**2**

2. The two time figure blocks found in the “Scopes” sub-system plot the same data as their corresponding Simulink scopes. However time figures make data analysis easier as they provide the data cursor tool. Using these time figure blocks and the data cursor tool measure the peak to peak amplitude of both responses obtained above.

Solution:

Using the data cursor tool the peak to peak amplitude of the SRV02 motor shaft angle was measured to be approximately 13.36 degrees and the peak to peak amplitude of the pendulum angle response was measured to be approximately 1.93 degrees.

0	**1**	**2**

3. Verify that the reference energy you calculated in question 1 of section 5.1 is correct.

Solution:

In the *q_sesip.mdl* Simulink model and under the *Swing-Up Control/SRV02 Swing-Up Control/Energy-Based Swing-Up Control* the total energy of the pendulum is calculated for energy control purposes. The controller implemented in this sub-system is the same as the one discussed in section 4.3. The second output of this sub-system is E (J) which holds the total energy of the pendulum (potential + kinetic). By looking at the value of this variable when the pendulum is balanced at the vertical upward position we see that it is holding at 0.3876 which is precisely the value found in question 1 of section 5.1.

0	1	2

As you might have noticed in question 1 above, although the pendulum angle response is fairly close to the reference zero degree mark, the SRV02 motor shaft angle has a significant amount of variation. Using the intuition developed in section 4.2.1 tune the Q matrix given in [68] such that these variations are decreased. Follow the steps below to tune this parameter on the fly without stopping the model.

1. Recall from section 5.2 question 2, that the Q matrix should be save under the MATLAB variable name *Q_lqr*. While the model is still running type in the new Q matrix in MATLAB Command Prompt and save it under the variable *Q_lqr*.
2. Calculate the new LQR gain in the same way as was done in question 2 of section 5.2 and save it under the MATLAB variable *k* as before.
3. Type *qc_update_model* in the MATLAB Command Prompt. This function causes the newly calculated LQR gain to be downloaded to the model in real-time and you will see the resulting changes immediately.
4. Follow this tuning procedure until you notice a decrease in the amount of motor shaft angle variations.

5. Provide your final Q matrix and associated LQR gain.

Solution:

Using the weighting matrices:

$$Q = \begin{pmatrix} 1.1 & 0 & 0 & 0 \\ 0 & 1.1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \text{ and } R = 15 \qquad \text{[S64]}$$

we obtain the control gain:

$$K = [-0.2708, 1.7774, -0.1324, 0.2421] \qquad \text{[S65]}$$

0	1	2

6. Provide the response of the motor shaft angle and the pendulum angle under your new tuned LQR controller.

Solution:

Figure 6 and Figure 7 below show these responses.

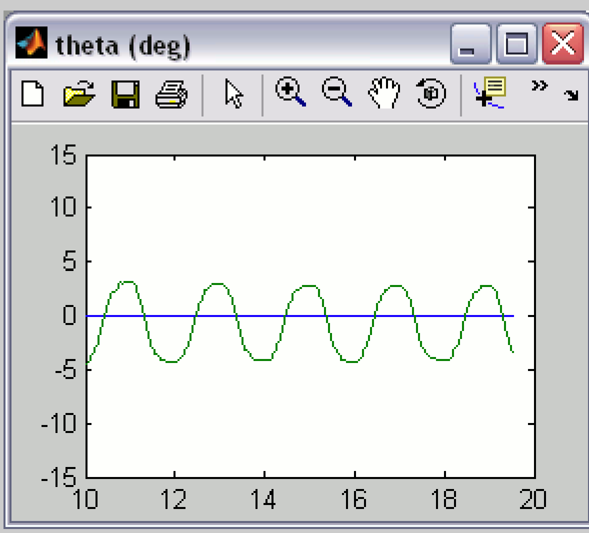


Figure 6: SRV02 Motor Shaft Response

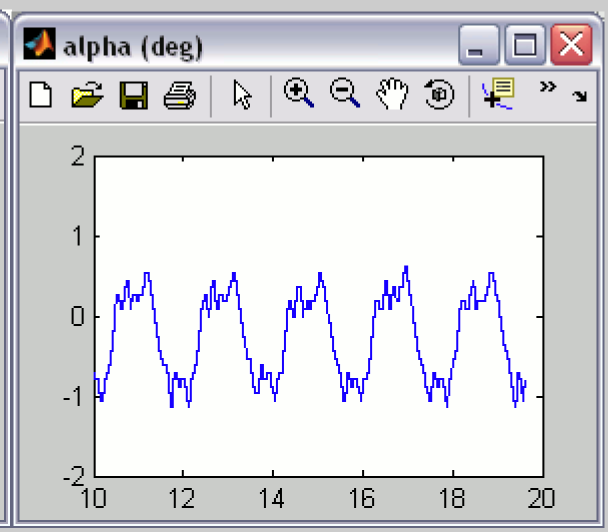


Figure 7: Pendulum Angle Response

0	1	2

7. Using the same Time Figure blocks as discussed in question 2 above, measure the new peak to

peak amplitude of both responses found above and comment on any changes. Provide percentage improvement if any.

Solution:

Using the data cursor tool the new peak to peak amplitude of the SRV02 motor shaft angle response was measured to be approximately 7.38 degrees and the peak to peak amplitude of the pendulum angle response was measured to be approximately 1.58 degrees.

By comparing these numbers with those obtained in question 2 above it is shown that the variations in SRV02 motor shaft angle response have been reduced by 45% and variations in pendulum angle response have been reduced by 18% as a result of choosing higher diagonal entries in the Q weighting matrix.

0	1	2

8. Make sure QUARC is stopped and the amplifier system is shut off if no more experiments will be performed in this session.

6. Results Summary

Fill out Table 14 with the pre-laboratory and in-laboratory results obtained such as the designed gains along with the measured data.

Section	***Exercise #***	***Description***	***Symbol***	***Value***
4.3	3	Reference Swing Up Energy	E_r	0.3877
5.2	2	Initial LQR Gain	k	[-0.1633, 1.4406, -0.0906, 0.1913]
5.3.1	2	Initial peak to peak SRV02 motor shaft angle amplitude	θ_{p-p}	13.36
5.3.1	2	Initial peak to peak pendulum angle amplitude	α_{p-p}	1.93
5.3.1	5	Final tuned LQR Gain	k	[-0.2708, 1.7774, -0.1324, 0.2421]
5.3.1	7	Final peak to peak SRV02 motor shaft angle amplitude	θ_{p-p}	7.38
5.3.1	7	Final peak to peak pendulum angle amplitude	α_{p-p}	1.58

Table 14: SRV02 Exp #8: Self Erecting Single Inverted Pendulum Control Results Summary Table

7. References

[1] Data Acquisition Board User Manual
[2] QUARC User Manual (to access type `doc quarc` in the MALTAB command prompt)
[3] QUARC Installation Manual
[4] Amplifier User Manual
[5] SRV02 User Manual
[6] Rotary Experiment #0: SRV02 QUARC Integration
[7] Rotary Experiment #1: SRV02 Modeling
[8] Rotary Experiment #2: SRV02 Position Control
[9] Rotary Pendulum User Manual