Haptic Interface Project Report Excerpt

Entails:

1. Project Introduction
2. Complete System Assembly (Solidworks Design)
3. Problem Description and RCGs
4. Complete System Block Diagram

3. INTRODUCTION

For phases one and two of the ELEC 391 Haptic Interface project, the motors and controls groups worked independently on preparing for the final integration phase. The motors group focused on producing a functioning set of two similar low-friction linear actuators, with a position detection method implemented. The controls group practiced control theory and execution on readily available rotary motors and optical encoders, while building digital logic and power circuits to be used in the third (integration) phase.

For phase three, the two groups came together to put together a complete system that involves two linear actuators. The final product assembly is illustrated in (figure1).

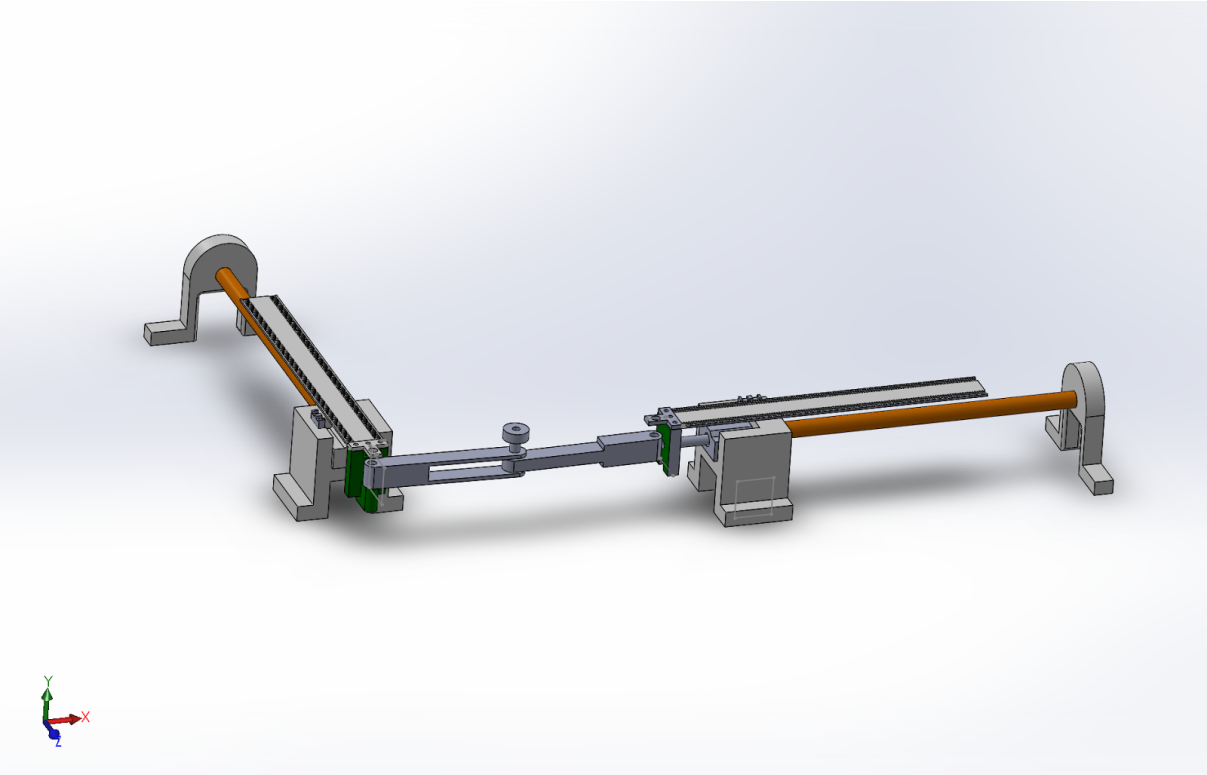

Figure 1: Complete System Assembly

4. PROBLEM DESCRIPTION

For phases one and two of the project, the team was required to prepare for the design and operation the team's haptic interface. The following list describes the RCGs that each group attempted.

4.1 CONTROLS GROUP RCGS:

1. Drive the Motor

Requirement	Control the motor in both directions at variable desired magnitudes of current/voltage
Constraint	In its RC mode, the Sabretooth 2x5 only offers 500 degrees of control over the average voltage to the motor in each direction
Goal	Use PWM to generate an average voltage within the range [-100,100]% of the DC source voltage with as many degrees of control as possible

2. Read Position/Angle of Rotation

Requirement	Have the accumulated angle of the motor available and updating accordingly in the main processing unit
Constraint	The provided encoder (QME-01) has 100 slots per revolution, offering a degree of certainty of 3.6 degrees
Goal	Design a hardware circuit capable of interpreting the encoder's signals and outputting a registered angle value for the Arduino to read whenever it is needed

3. Compute Angular Velocity

Requirement	Compute the rotary motor's angular velocity
Constraint	It is impossible to perform a perfect differentiation on the Arduino
Goal	Implement a code to recalculate velocity every time a movement, or lack thereof, is sensed

4. Calculate Transfer Function of DC Motor

Requirement	Calculate an accurate transfer function that models the behavior of the DC rotary motor
Constraint	No information on the performance constants of the motor was provided
Goal	Generate a step response of the motor's angular velocity to calculate constants, and eventually the motor's transfer function

5. Generate PD Control Constants

Requirement	Simulate the motor's transfer function and tune desired PD values
Constraint	The inaccuracy of the transfer function described in point 4 will affect the effectiveness of the PD constants
Goal	Find a ratio of proportional to derivative constants that would allow the motor to rotate smoothly to its target angle

6. Implement PD Control

Requirement	Program the microcontroller to drive the motor to a desired position/velocity
Constraint	Timing of microcontroller operations, and 3.6 degree of rotary angle certainty
Goal	Implement PD controller which utilizes Kp and Kd from point 5 to constantly calculate the required average voltage, and signal to the Sabretooth 2x5 to achieve the desired functionality

4.2 MOTORS GROUP RCGS:

1. Control a Low Friction Actuator

Requirement	Design and Construct a Linear actuator with full control of the position of the rotor using a slot detector sensor, encoder strip, and PWM
Constraint	Creating a rotor-stator configuration that has minimal friction
Goal	Produce a sponge-like opposing force with linear motion

2. Setup a PWM Circuit

Requirement	Write code to produce 8 PWM signals using the Arduino's Interrupt Service Routine (ISR) and follow datasheet to wire-up L298 H-bridge
Constraint	Directly connecting Arduino pins (PWM signal) to H-Bridge can damage Arduino
Goal	Control the average Power received by linear actuator

3. Configure ESP8266 Wi-Fi Module as a Server

Requirement	Write configuration code to setup Wi-Fi module, as well as, build IPhone User Interface
Constraint	Synchronization of IPhone with ESP8266 Wi-Fi module
Goal	Establish a 2-way connection between Arduino and IPhone

4. Setup and solder Slot-Detector on prototyping board

Requirement	Wire-up a EE_SX1106 slot detector
Constraint	Find an encoder strip that is opaque enough to block whilst being thin enough to fit in the detector
Goal	Produce a high/low signal depending on the position of the encoder strip

5. Design a sturdy Rotor-Joint mechanism

Requirement	Build a smooth joint that connects the rotor with a 3D-printed popsicle-stick
Constraint	Geometry of the rotor-stick joint
Goal	Have low friction between joints

System Level Block Diagram