Lab 2

Robot Arm

In this lab we will make the robot arm move through a predefined sequence of movements. Utilizing the work you already did in homework, you will make the arm move.

You should learn or gain experience with:

- Taking to an actuator via a common serial port
- Translate forward/inverse kinematics theory into reality with a real robot arm
- Understand how to translate angles to PWM servo commands
- Gain more experience with Python

Task 1: Calibrate

The robot arm uses toy RC servos[1] to move. These servos are commanded by a pulse width modulated signal[2] (PWM) to set their position. Unfortunately, these toy servos are produced for their low price and not their performance. Therefore, you must determine the correct PWM signal to get your servo to move correctly ... every servo is different.

The Lynx Motino AL5D has 5 servo motors that are able to turn between 0 and 180 degrees. The nominal PWM signals are shown below:

Angle	PWM	Gripper	PWM
0	800	Open	800
180	2300	Closed	2300

```
#!/usr/bin/env python
from __future__ import print_function, division
import pyserial
import time

# open serial port
ser = pyserial.Serial('COM3', 115200)

def angle2pwm(angle):
    # your code here

def move_servo(servo, angle):
    # send a command to a single servo
    # example:
```

[1] https://en.wikipedia.org/wiki/Servo_(radio_control)
[2] https://en.wikipedia.org/wiki/Pulse-width_modulation

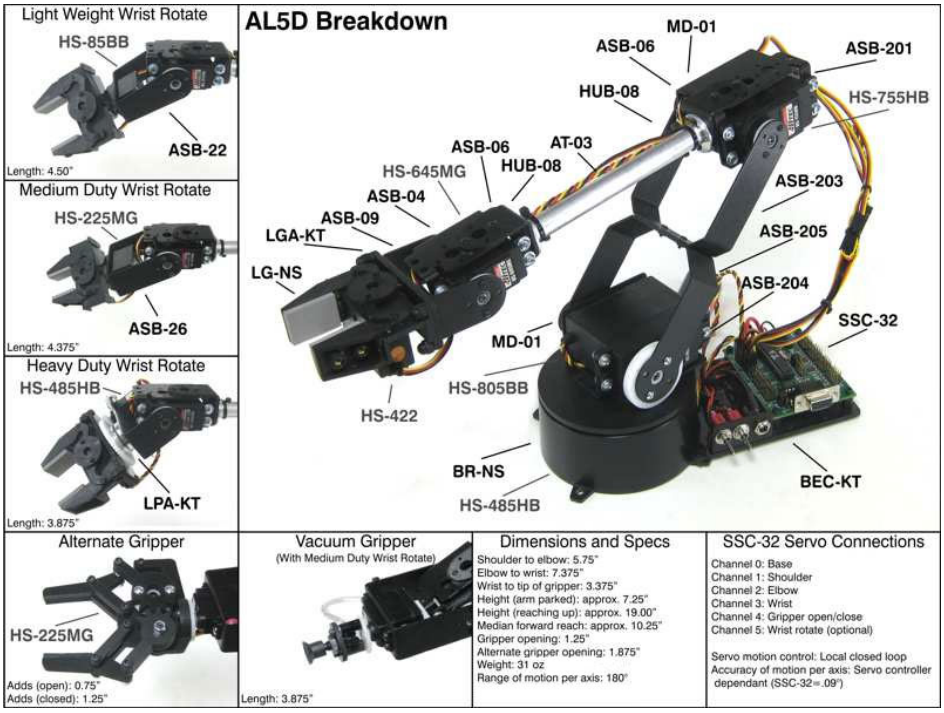

Figure 1: AL5D Robot Arm

```
    #     servo 1  angle 0
    #     send '#1 P800 T2000\r'
    pwm = angle2pwm(angle)
    cmd = '#{} P{}\r'.format(int(servo), int(pwm))
    ser.write(cmd)

if __name__ == "__main__":
    # your code here
```

Task 2: Forward Kinematics

Once you have figured out the best PWM settings for your robot arm, now use your code to move the arm through a sequence of orientations.

Step	Position	Gripper
1	(0, 90, 90, 0)	open
1	(-23.2, 83.8, 102.3, 71.5)	closed
1	(0.0, 111.5, 127.0, 74.5)	closed
1	(0, 90, 90, 0)	open

After each step, pause for 2 seconds. When you have it working, show your instructor.

Task 3: Inverse Kinematics

Now use your code to move the arm through a sequence of positions. Since we are grabbing an object, when the step below says *closed* it really means half way closed or $\frac{PWM_{closed}}{2}$

Step	Position	2 Orientation	Gripper
1	(10.75, 0, 5.75)	0	open
2	(9.5, 0, 4)	0	open
3	(9.5, 0, 0)	0	open

Robots are cool!

```python
def angle2pwm(angle):
    # code

def move_arm(joint_angles):
    # your code here

if __name__ == "__main__":
    sequence = [
        [10.75, 0.0, 5.75, 0, 0], # x, y, z, orientation, gripper
        [...],
        ...
    ]

    for angles in sequence:
        move_arm(angles)
        time.sleep(5)
```