ECE 417 --- ROBOTICS
Lab 6, Spring 2021

For this lab you will program the Lab-Volt robot to pick up an object at one point and put it down at another, following straight line trajectories along the way. You have the following tasks.

(1) moveXYZ (double X, double Y, double Z);

- Write a (short) subroutine which will move the robot from the current point to a new point. You pass the XYZ coordinates of the desired destination as arguments to the routine. The routine will assume that the desired orientation is one with the gripper pointing straight down; i.e., the orientation is represented by the quaternion q = (0,0,1,0).

(2) moveStraight (double X, double Y, double Z); target XYZ

- Write a subroutine which will move the robot in approximately a straight line (in cartesian space) from the current point to a new point (both points have the gripper pointing straight down). Pass the routine the coordinates of the new point. The routine operates by dividing the line segment between the current point and the new point into 1 cm intervals. The cartesian coordinates of each intermediate point will be determined and the robot will be instructed to move there using the above subroutine.

X

- Write an open gripper and a close gripper routine.

- Your main program will have the robot pick up an object (a small wooden cylinder) placed 45 degrees to its left (i.e., base angle is 45 degrees) and place it 45 degrees to its right. Your program will have to move above the object, open the gripper, move down, close the gripper, move up, move above the setdown position, move down, open the gripper, move up, and then return to home. Your moves to and from the home position do not need to be in a straight line, as the gripper will be changing orientation while moving. The other moves should be in a straight line.

Straight

(1) $(X, Y, Z) \rightarrow \begin{bmatrix} R & \begin{matrix} X \\ Y \\ Z \end{matrix} \\ 0\ 0\ 0 & 1 \end{bmatrix} \Rightarrow$ inverseKinematics $\Rightarrow$ moveTheta

$^0T_5$

in main()

define initialX — moveXYZ( )
define initialY —
define initialZ —

(2) currentXYZ, targetXYZ

static double currentXYZ = { }

step = |targetXYZ − currentXYZ| ← normalize (divide by length)

```
for (i=0; i<length; i++){
    currentXYZ += step
    moveXYZ(currentXYZ)
}
moveXYZ(targetXYZ)
```

currentXYZ = TargetXYZ