EEEE 585/685 – Principles of Robotics Lab

Experiment #4 Introduction to the AL5B Robotic Arm

I. OBJECTIVE

In this experiment, the student will familiarize themselves with a six degree of freedom (6DOF) arm robot called the AL5B Robotic Arm. This robotic arm will be controlled using the MABL Arm Controller Software and Matlab. Position and orientation information for individual joint frames will be calculated using frame transformations with respect to a universal frame. Using Matlab, relative and absolute joint movements will be demonstrated and contrasted. Also, the final position of the end-effecter will be compared to theoretical calculations for error analysis.

II. THEORY

Frame Transformations:

It is often necessary to determine the position and orientation of the joints in an arm robot. In order to classify a sense of orientation and position, an initial reference frame needs to be established, which is called the Universe Frame (U). The universe frame is expressed in Cartesian coordinates and is composed of three orthogonal unit directional vectors x, y, and z. Once the universe frame is established, additional frames may be placed at the joints of the robot. For this particular robot there are six frames used, excluding the universe frame (U) and tool frame (T). Each of these frames, also known as fixed reference frames, are expressed in Cartesian coordinates and are composed of orthogonal unit directional vectors n, o, and a. The vectors n, o, and a are expressed with respect to the universe frame's x, y, and z components (Figure 1). The origin of the fixed-reference frames is described by a vector, P, with components x, y, and z with respect to the Universe Frame (Figure 2). Notice that in Figure 2, the n, o, a, and P are combined into one matrix to be expressed as homogeneous coordinates. This is done to preserve square matrix proportionality such that special transformations can be applied directly which are discussed below.

$$\bar{n} = \begin{bmatrix} nx \\ ny \\ nz \end{bmatrix} \qquad \bar{o} = \begin{bmatrix} ox \\ oy \\ oz \end{bmatrix} \qquad \bar{a} = \begin{bmatrix} ax \\ ay \\ az \end{bmatrix}$$

Figure 1 - Fixed Reference Frame Unit Directional Vectors

$$F = \begin{bmatrix} nx & ox & ax & Px \\ ny & oy & ay & Py \\ nz & oz & az & Pz \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figure 2 - Fixed-Reference Frame Homogeneous Coordinates

When describing the position and orientation of a particular frame, it is always related to a previous frame(s) or Universe frame. These relationships can be formulated through various frame transformations. These transformations are combinations of pure translations and/or pure rotations with respect to the previous frame (s) or universe frame. By analyzing the geometry of the previous and current frames it is possible to determine the type of translations and rotations needed (Figure 3-6). Each of these transformations can be applied to a fixed-reference frame (Figure 2) by pre-multiplication of the matrices according to the order of the rotations and translations.

$$\begin{bmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & b \\ 0 & 0 & 1 & c \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figure 3 - Pure Translation

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) & 0 \\ 0 & \sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figure 4 - Pure Rotation (X-Axis)

$$\begin{bmatrix} \cos(\phi) & 0 & -\sin(\phi) & 0 \\ 0 & 1 & 0 & 0 \\ \sin(\phi) & 0 & \cos(\phi) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figure 5 - Pure Rotation (Y-Axis)

$$\begin{bmatrix} \cos(\phi) & -\sin(\phi) & 0 & 0 \\ \sin(\phi) & \cos(\phi) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figure 6 - Pure Rotation (Z-Axis)

III. INTRODUCTION

<u>AL5B Robotic Arm Platform:</u>

The AL5B robotic arm is a 5-DOF arm robot. It has 7 servo motors. These motors provide one (1) degree of freedom each to the robotic arm’s base, shoulder, elbow, elbow-twist, wrist, wrist-twist, and its gripper. They also allow for movement within +60^{o} and -60^{o}.

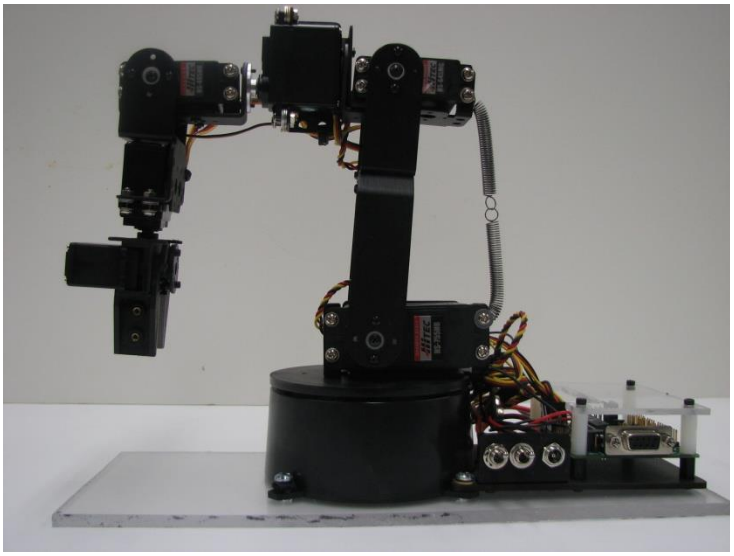


Figure 7 - AL5B Robotic Arm

Servo motor:

Please see Lab 1 and Lab2 for theory on servos and how to control a servo using Pulse Width Modulation (PWM).

The SSC -32 Servo controller:

This arm robot uses the same controller as the Hexapod in Lab1. See Lab1 hand out for more information.

Connecting the Arm Robot:

The servos are be powered by the 6V adapter and using the 5V from the power supply for the SSC-32 logic. Make sure that the jumper on VL = VS1 is disconnected. Connect the serial cable to the SSC-32 as showed below.

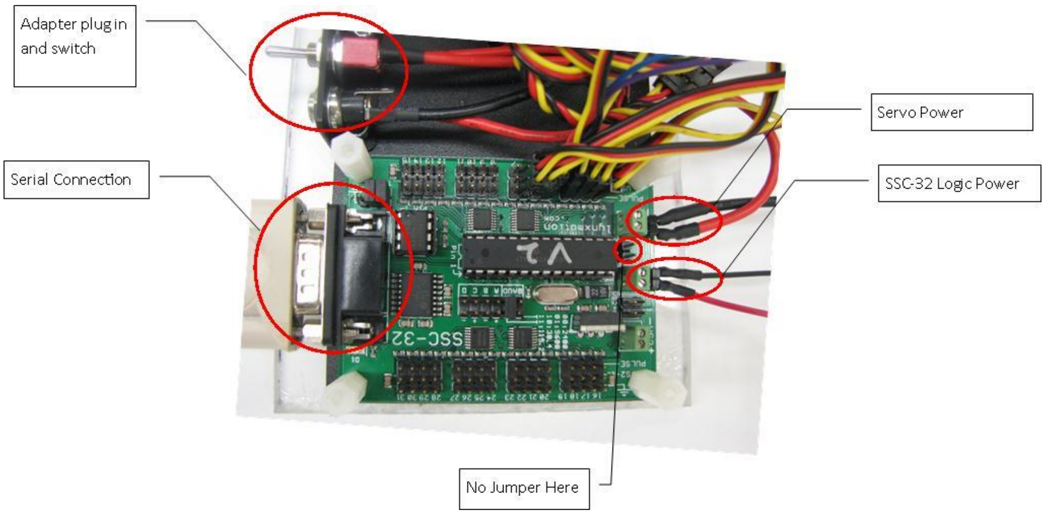


Figure 8 - Arm Robot connection

Getting Documents:

Download the Lab handout and example code from MyCourses.

IV. LAB ASSIGNMENTS

1. Measurements of Robotic Arm:

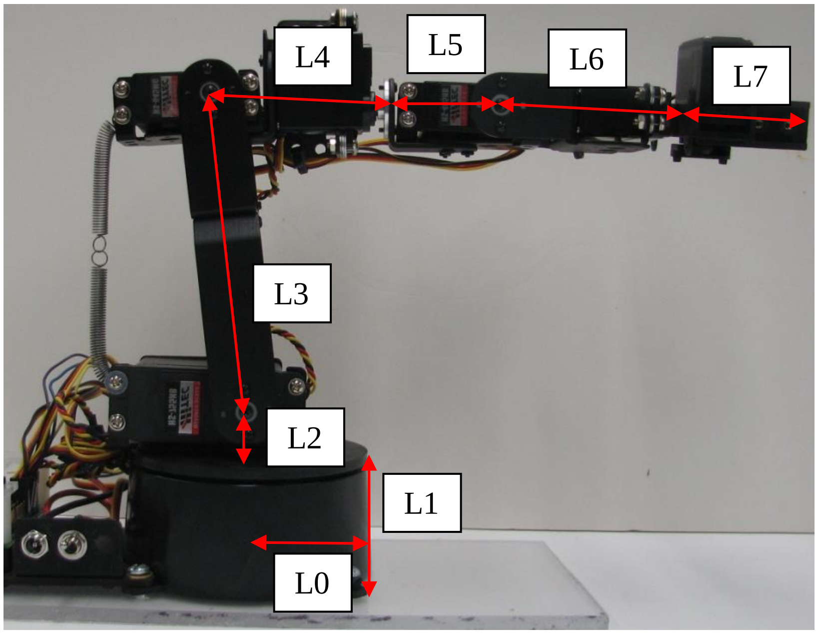


Figure 9 - Home position robot configuration.

a. Using the information given in Figure 10, determine the lengths (cm) of the following links:

Table 1 - Link Lengths

	Length (cm)
L0	
L1	
L2	
L3	
L4	
L5	
L6	
L7	

b. Connect the power to the Arm Robot as shown above in Figure 8, then connecting a serial cable to the robot.

c. Turn on the power supply for the logic controller, followed by the power for the servo controller.

d. By observation and using the information given in Figure 9, determine the angles of freedom for the following robot joints.

   i. Only fill in the "Degree" columns for "Min" and "Max" at this time.

Table 2 - Robot Servo Values and Degrees

Servo/Join	Center Value	Angle of Freedom			
		Min		Max	
		Servo Value	Degree	Servo Value	Degree
Base					
Shoulder					
Elbow					
Elbow Twist					
Wrist					
Wrist Twist					
Gripper					

2. Repeatability of the arm robot:

Write a Movement File to repeat 10 consecutive times, a set of instructions to send the arm of the robot to a certain location and measure the position.

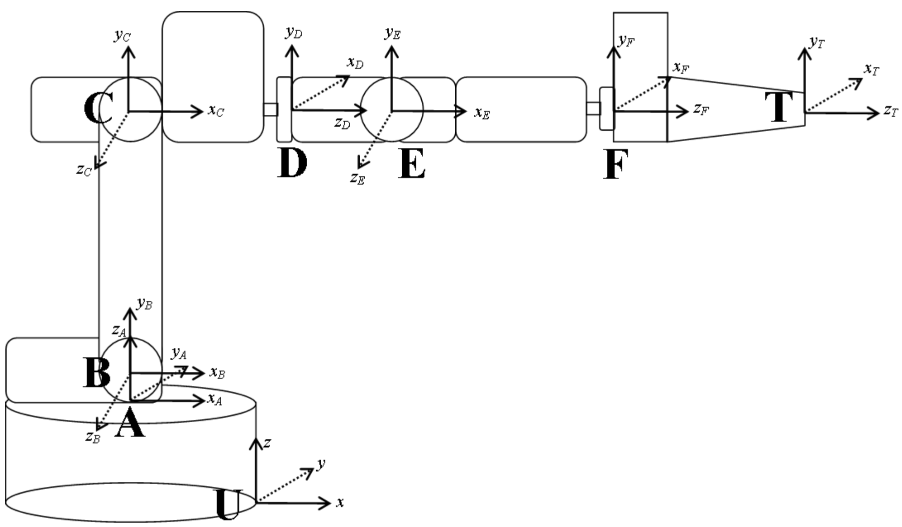


Figure 10 - Robot home position.

Step 1: Move the robot to its home position (Figure 10) with a thin pencil in its grip.

Step 2: Move the tip of the robot (the marker) to a position approximately 12cm away from the square base by moving some joints. Make sure that the marker is touching to paper to create a data point.

Step 4: Return the robot to its home position

Step 5: Repeat this 9 more and then **compute the radius of repeatability from the results.**

Step 6: Give a demo, include the Movement file and results in your report.

3. Initial Matlab Code

a. Start Matlab.
b. Set path to the Example Code directory for Lab4.
c. Open the ArmRobot.m class definition file and Lab4.m script
d. Become familiar with the ArmRobot class definitions (open it and read the comments). Only modify the methods and functions in the ArmRobot class, it is not necessary to change any of sub-classes included in the ArmRobot class (They have been provided to you).
e. Run the Lab4.m script file and be sure there are no errors.
f. ***** If com port redirect popup shows up ignore it, it does not affect the robot communication.**

4. Setting the Bounds

a. Using the Matlab Code, find the decimal servo values that correspond to the bounds of each joint as well as center for the particular robot being used. Record these values in Table 2 for future use.
b. In lab4.m file provide correct arguments to setServoCenters, setServoBounds, and setLinkLengths.
c. Run lab4.m as is to test home position movement.

5. Implement Absolute Joint Movement

a. Write code to implement absolute joint movement, this means given an angle in degrees the robot will move that joint to that corresponding given angle with respect to the home position.
   i. Convert the angle from degrees to servo values and use the *moveJoints()* method.
   ii. Method must be able to take 1 servo and value or an array of servos and corresponding values.

6. Implement Relative Joint Movement

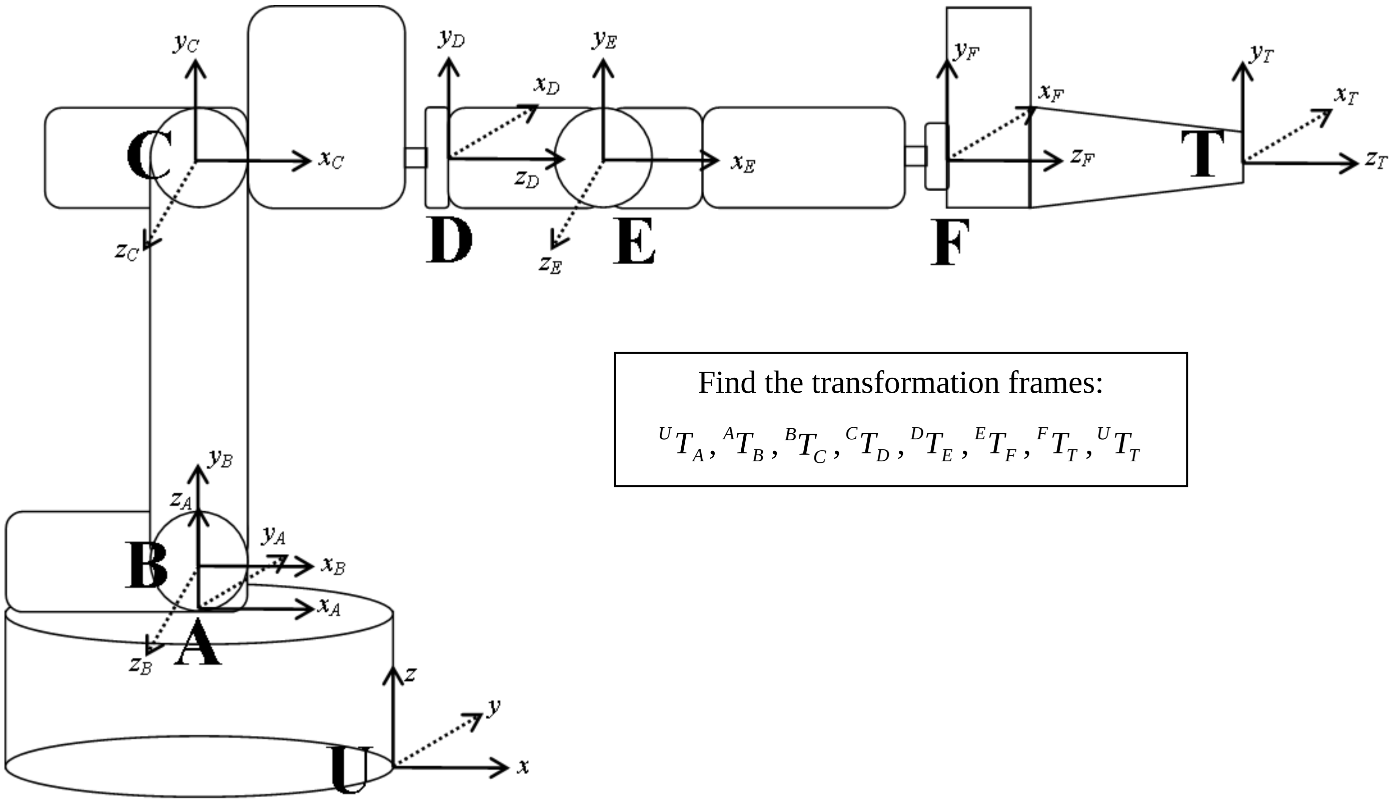
a. Write code to implement relative joint movement, this means given an angle in degrees the robot will move that joint to that corresponding given angle with respect to the current position.
   i. Obtain current servo values from stored properties.
   ii. Convert current servo values to an angle.
   iii. Find new relative angle (addition).
   iv. Convert the new angle from degrees to servo values and use the *moveJoints()* method.
   v. Method must be able to take 1 servo and value or an array of servos and corresponding values.

7. Pseudo Pick and Place Simulation

a. Implement the attached 7 step Pseudo Pick and Place
b. Use the ArmRobot class and newly implemented methods see Lab4.m for starting basis.
c. For ALL transformations use **ONLY** the frame transformation technique.
d. All counter-clockwise motions are positive and clockwise motions are negative
e. **Perform all the calculations using Matlab in a self contained .m file**.
f. **Submit all theoretical transformations in the appendix of your report** (printed nicely!)

g. You may substitute in actual length measurements to reduce the size of your matrices.

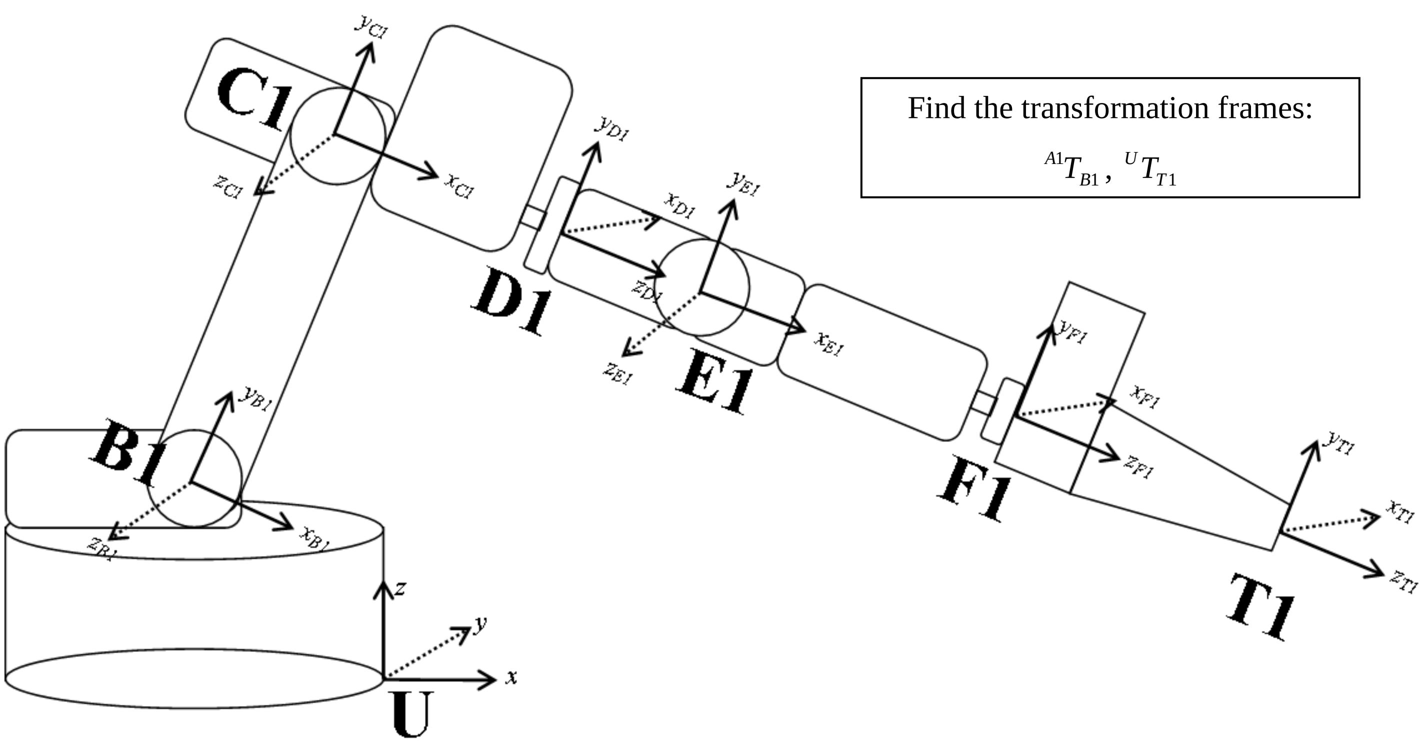
STEP 0: Home position



Find the transformation frames:

$$^{U}T_{A},\ ^{A}T_{B},\ ^{B}T_{C},\ ^{C}T_{D},\ ^{D}T_{E},\ ^{E}T_{F},\ ^{F}T_{T},\ ^{U}T_{T}$$

STEP 1: Rotate shoulder servo about the Z_A axis by -30º.



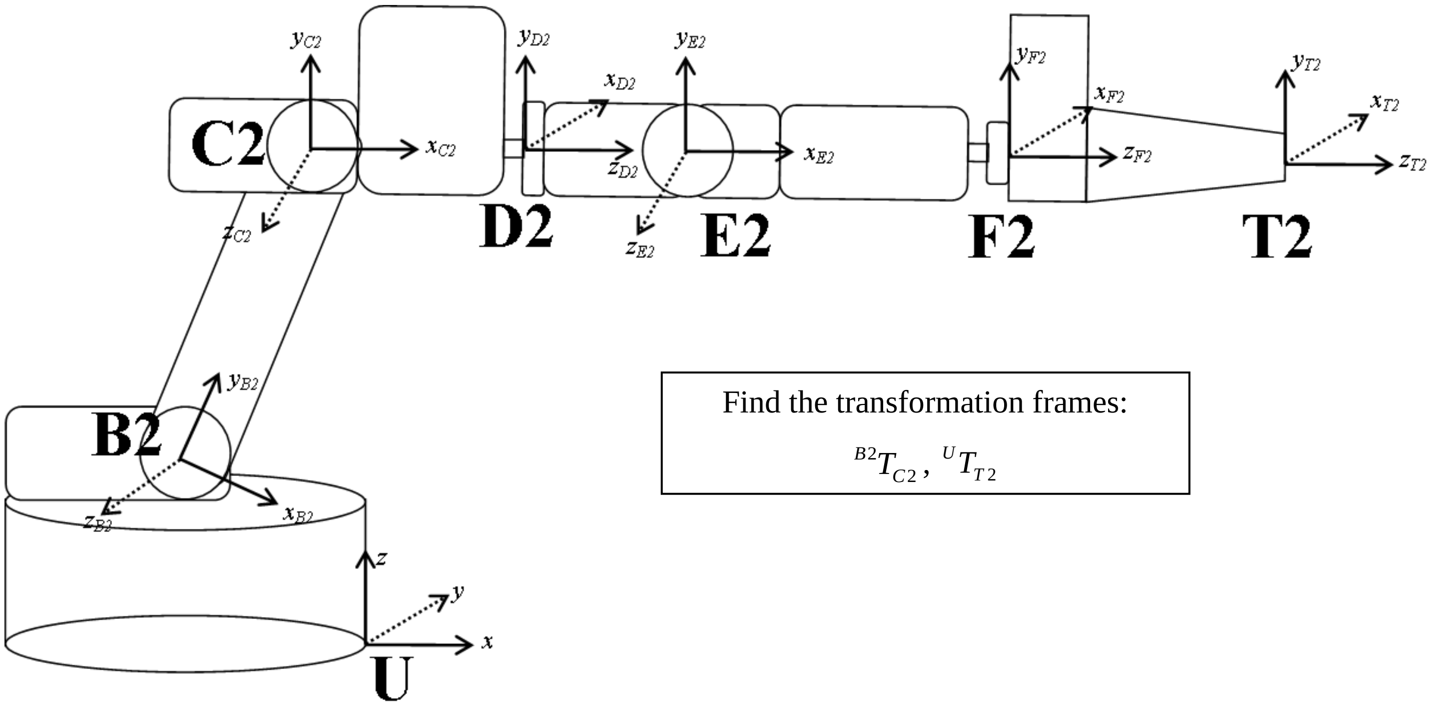
Find the transformation frames:

$$^{A1}T_{B1},\ ^{U}T_{T1}$$

The “A” frame never changes in the pick and place. Therefore it has been removed to reduce graphical clutter.

Other frames should stay the same from the previous step

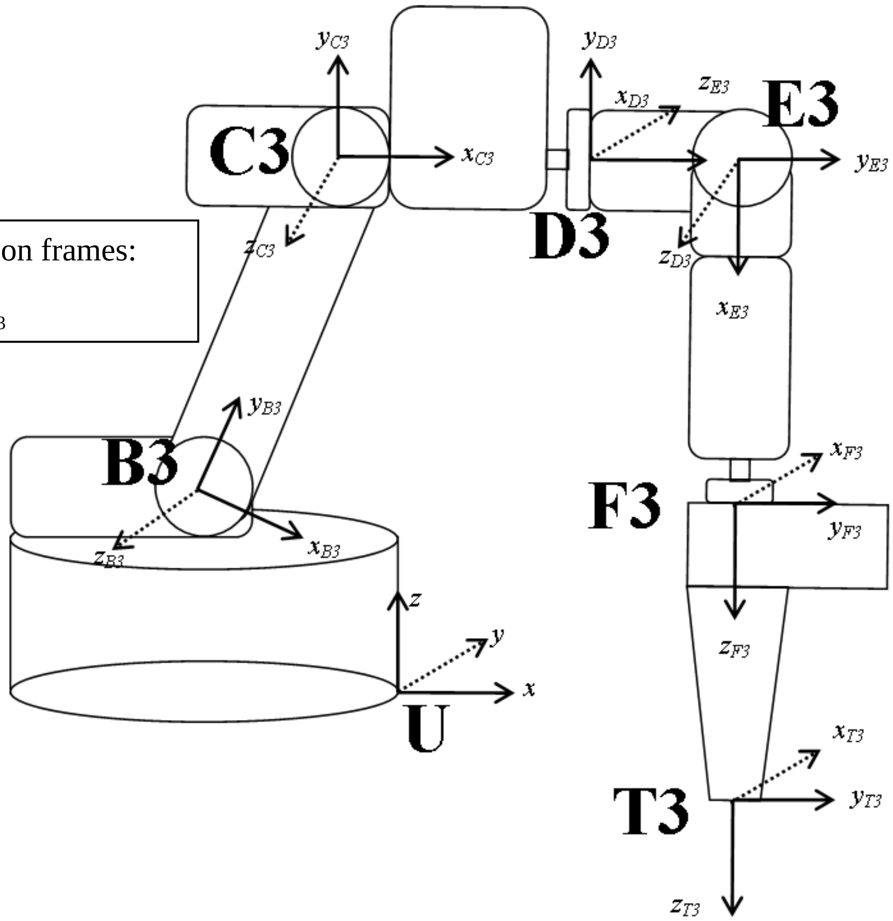
STEP 2: Rotate elbow servo about the Z_{C1} axis by 30º.



Find the transformation frames:

$${}^{B2}T_{C2},\ {}^{U}T_{T2}$$

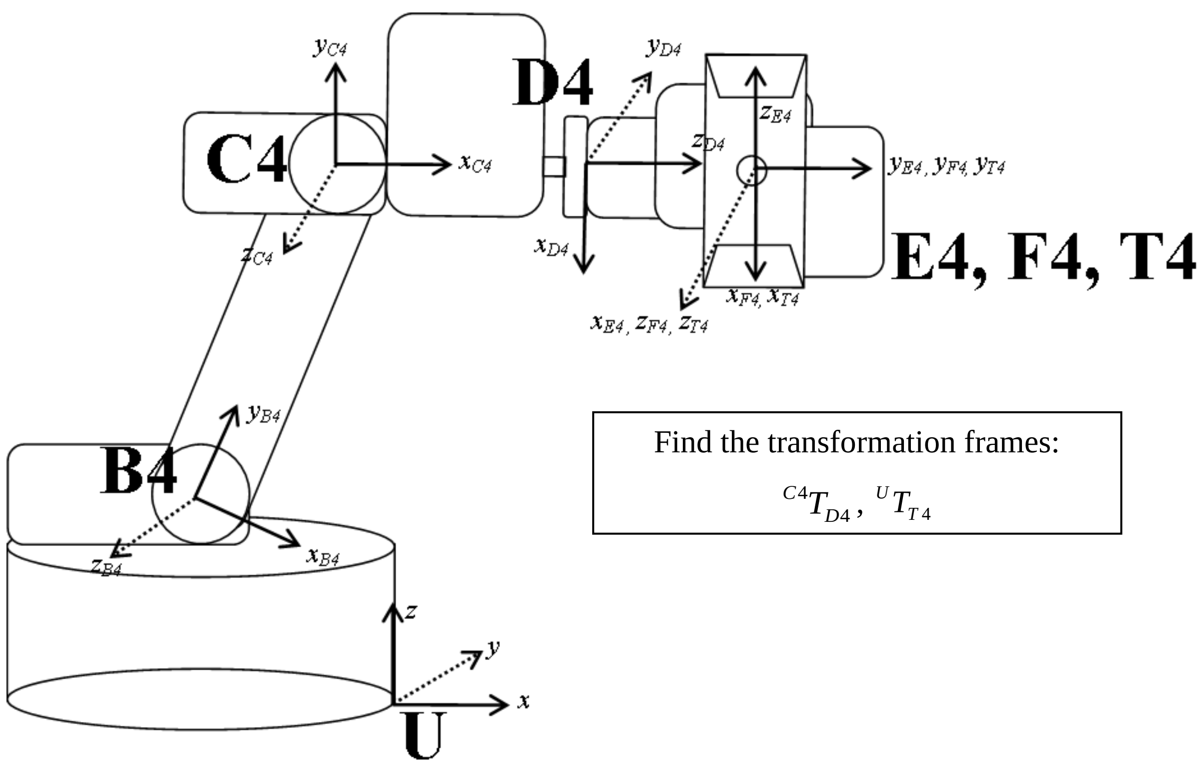
STEP 3: Rotate wrist servo about Z_{D2} axis by -90º.



Find the transformation frames:

$^{D3}T_{E3}$, $^{U}T_{T3}$

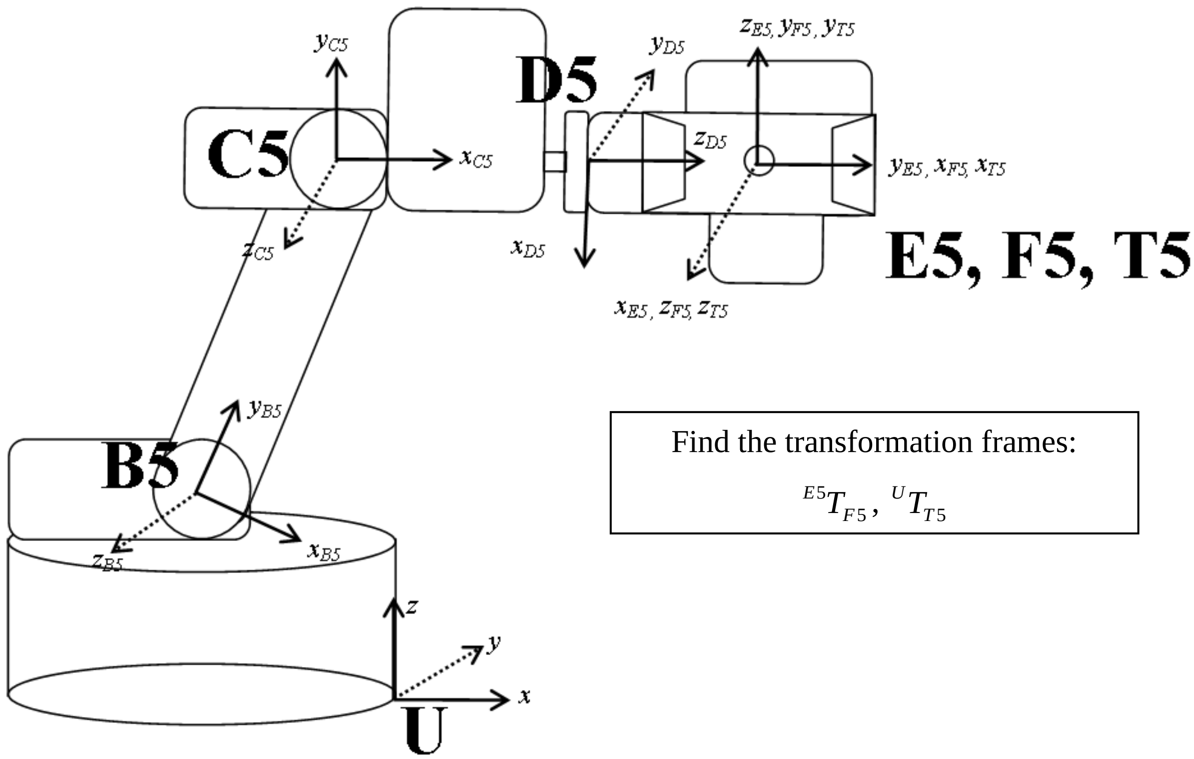
STEP 4: Rotate elbow twist servo about Z_{C3} by -90º.



Find the transformation frames:

$^{C4}T_{D4}$, $^{U}T_{T4}$

STEP 5: Rotate the wrist twist servo about Z_{F4} by 90º.



STEP 6: Close the gripper (No effect on Tool Frame orientation/position)

STEP 7: Open the gripper (No effect on Tool Frame orientation/position)

8. **Gripper Positional Error**
   a. Analyze the error in the calculated positional information of the griper in the theoretical calculations to the actual measured position of the gripper in the simulation.
   b. Do this for each of the 7 steps.
   c. **Create a table and plot to summarize these results.**

V. REPORT

Specific items to this lab that should be included are:

a) Link and Joint measurements.
b) Discussion on relative vs. absolution joint positioning.
c) Repeatability Discussion and results.
d) Output of frame transformations (calculations).
e) Table and Plot of error in gripper position.

VI. QUESTIONS

1. What is the arm robot lacking which would make it an easier robot to work with?
2. What can be done to improve the repeatability of the robot?
3. Draw the workspace for this robot.
4. What are the advantages and disadvantages of the arm robot over prismatic jointed robots?