Lab Manual # 01

Roll #: **Name:**

Title: Robix – Robot Development Kit Experiment 1

Objective:

1. Learning the use of Usbor software for controlling servo motors connected to Usbor Servo Controller.
2. Assemble a robotic manipulator and drive it through the software.
3. Figure out the work-envelope of the assembled robot.

Apparatus

- Robix – Robot Development Kit
- PC with Usbor software installed.

Introduction

Robix – Robot Development Kit consist of several components necessary for making and testing basic educational robots. A basic robot will consist of multiple joints and rigid links joining these joints. The joints in-turn are made of, normally, motors, gearing system, and feedback sensor. In this kit, a DC Servo motor will act as a complete joint, which will be controlled through an Usbor Servo Controller.

The links are mechanically fastened with the joints in form of a kinematic open or closed chain robot. Once assembly is complete the servo motor joints are driven through the servo controller based on the program written in the Usbor software. The Usbor software employs an easy-to-understand high level language to generate control commands for the servo controller. Moreover, a software-based teach pendant is also there that can be used to move servos 'manually' and learn the movement with a single click of a button. The learned movement is automatically converted into the high-level language code so that it may be used later.

Refer to the Usbor Software Tutorial PDF file for instructions to install the Usbor software are getting it connected with the Usbor Servo Controller.

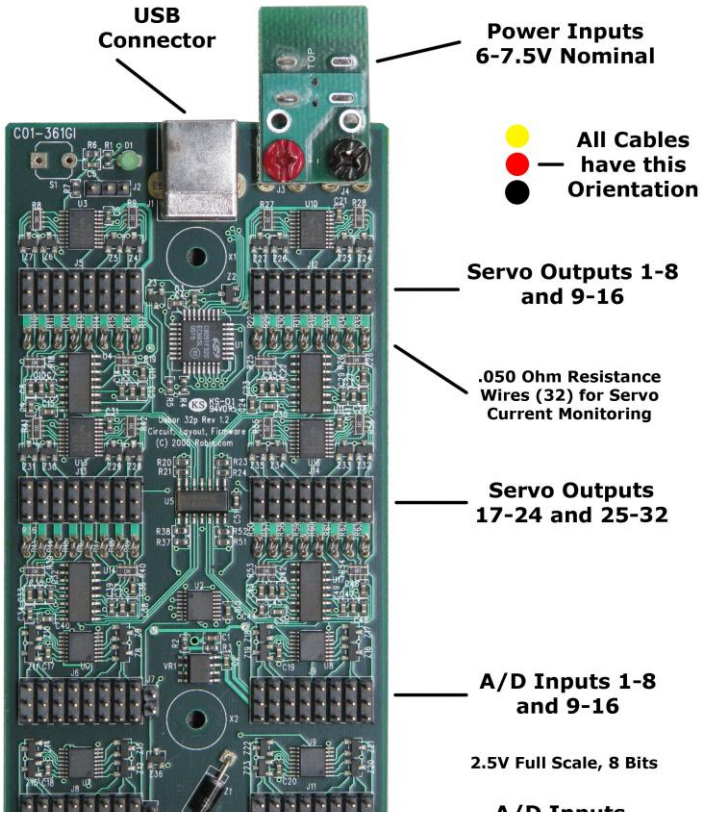
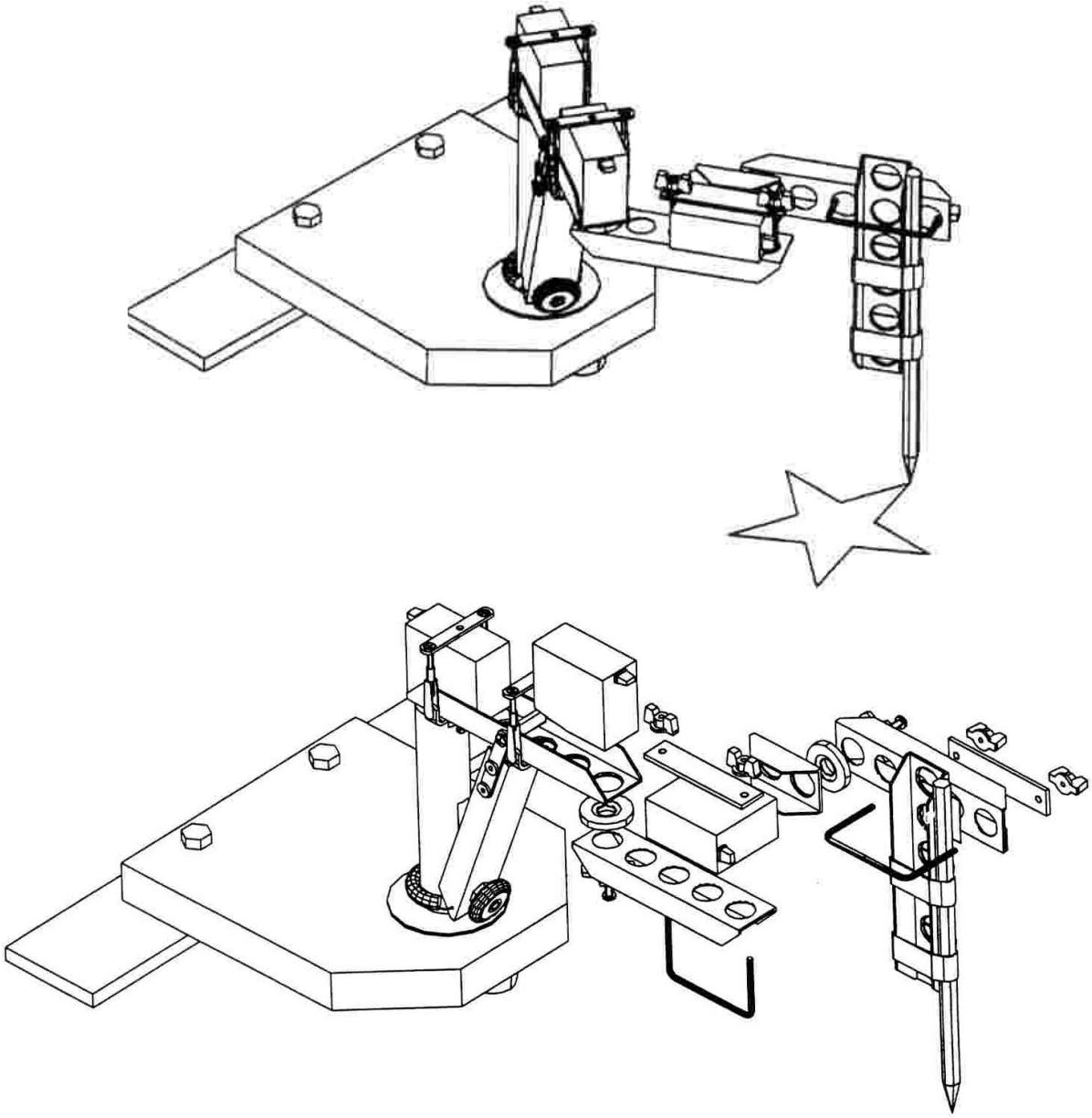


Figure 1: Usbor Servo Controller

Experimental Performance

Task 1:

Assemble the **Draw Robot** as per the following diagram and use the teach pendant to **draw a square.** Make sure to teach pen lifting and placement correctly and make use of the workspace efficiently.



Task 2:

Use the following code to draw a star on a sheet of paper. However, before running the robot through the following code, make sure you have understood it perfectly, that is you should know what each macro (function) and command is doing.

This script is provided as a sample. Use if for guidance rather than

```
# depending on it to work “as-is”. The move commands below were all created using
# Teach mode, which is also the easiest way for you to proceed.
# Sketch a straight-line figure, such as a star, on a pad of paper, then
# define macro pendown for your robot and execute it.
# Next, teach your robot the figure by moving in succession to each of the points.
# Since axes 1, 2 in our robot did not move perfectly parallel to the table, pen pressure
# was increased through part of the drawing where the line was faint and he
# the pen was wiggling. Adding pressure damped the servo wiggle and darkened the line.
invert all off # clear any left-over invert settings
invert 1, 2 on # invert axes to make teach-mode keys more intuitive
maxspd all 12 # low speed,
accdec all 20 # moderate acceleration
macro penup; move 3 to 0; end; # pen up
macro pendown; move 3 to -375; end; # pen down
macro start; move 1 to -979, 2 to 1400 end
macro star;
start;
move 1 to -499, 2 to 1261; move 1 to -154 , 2 to 1240
move 1 to -226, 2 to 937;
move 1 to 173, 2 to 286, 3 by -50 # increase pen pressure slightly
move 1 to -298, 2 to 739; move 1 to -106 , 2 to 202
move 1 to -532, 2 to 796
move 1 to -859, 2 to 883, 3 by 50 # ease off pen pressure
move 1 to -745, 2 to 1111
end
penup; start # don't make a line moving to start
pendown; star 0 # pen down then cycle continuously through the figure
```

Question 1: Draw the top view of the workspace of Draw Robot. Workspace shouldn’t include the lifted pen position. Insert the image below.

Question 2: What is macro in Usbor Software?

Question 3: How can you improve the drawing accuracy of your Draw Robot?

Question 4: The draw-robot uses 2 joints to position the attached pen at any suitable position in the workspace. Comment whether this robot is a redundant robot or not. Whether 'yes' or 'no', explain your answer by highlighting what a redundant robot is in this case.

Question 5: Draw the free-body-diagram of the robot you have assembled. This free-body-diagram will be later used to design the robot in the simulation environment (MATLAB SimScape). Insert the image below.