Name: Megha Veerendra

Net Id: mv1807

PROJECT-3 REPORT

Q1) Trajectory Generation

The trajectory for the scara manipulator is generated using the trapezoidal velocity profile which is follows:

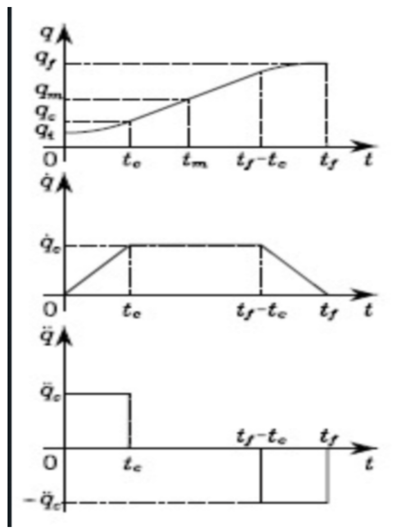


Fig1.1: Trapezoidal velocity profile

The trajectory that is generated as per the given positions is as shown in Fig 1.2. The general idea is to pick a cruise time t_c . The robot is assumed to accelerate from 0 to t_c seconds. After this the velocity becomes constant from t_c to t_c -t_f , where t_f is the final time. Finally from t_c -t_f to t_f the robot is assumed to decelerate. There are four positions that were given in the assignment. For each position the above explained method is used to derive the trajectory. The total time of the trajectory is 4 seconds.

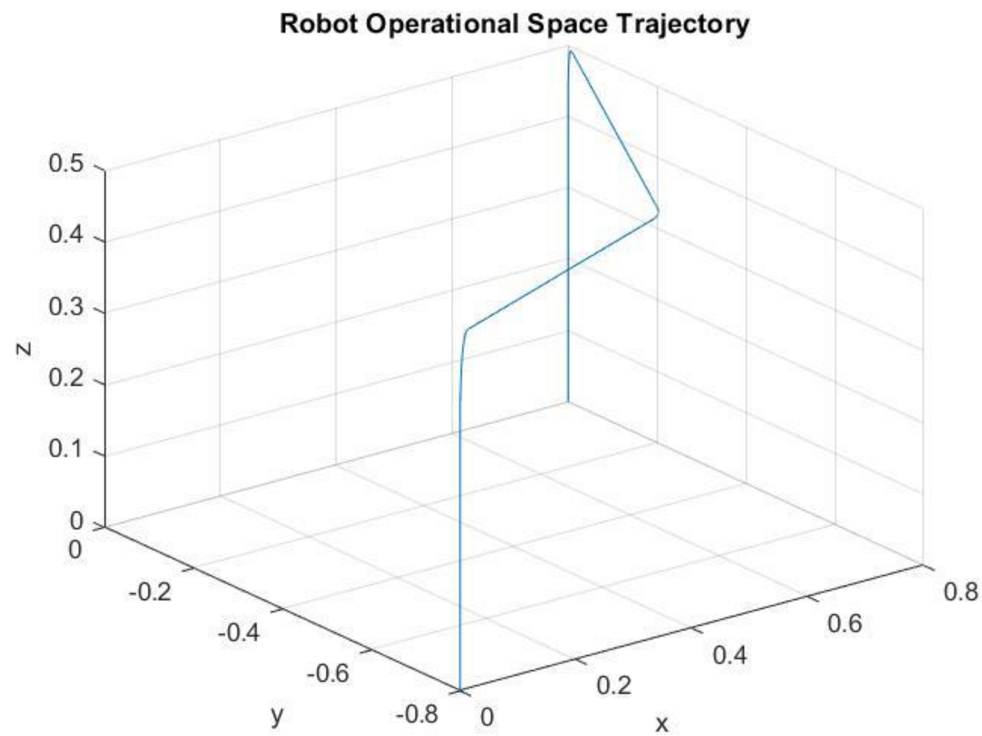
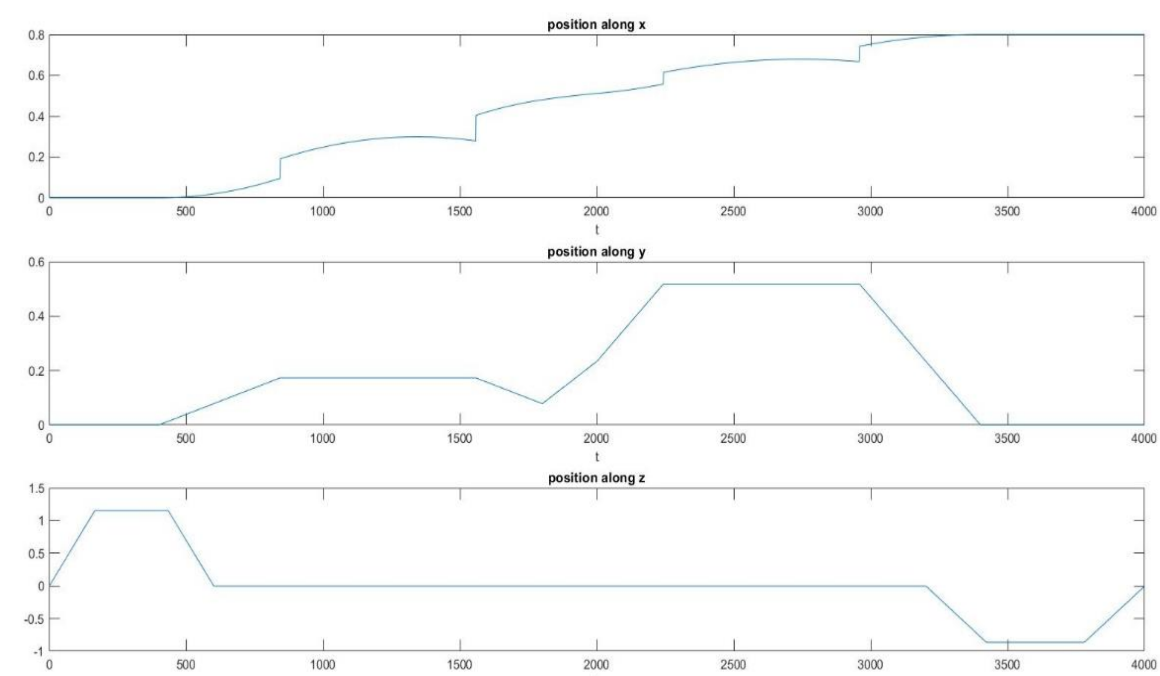
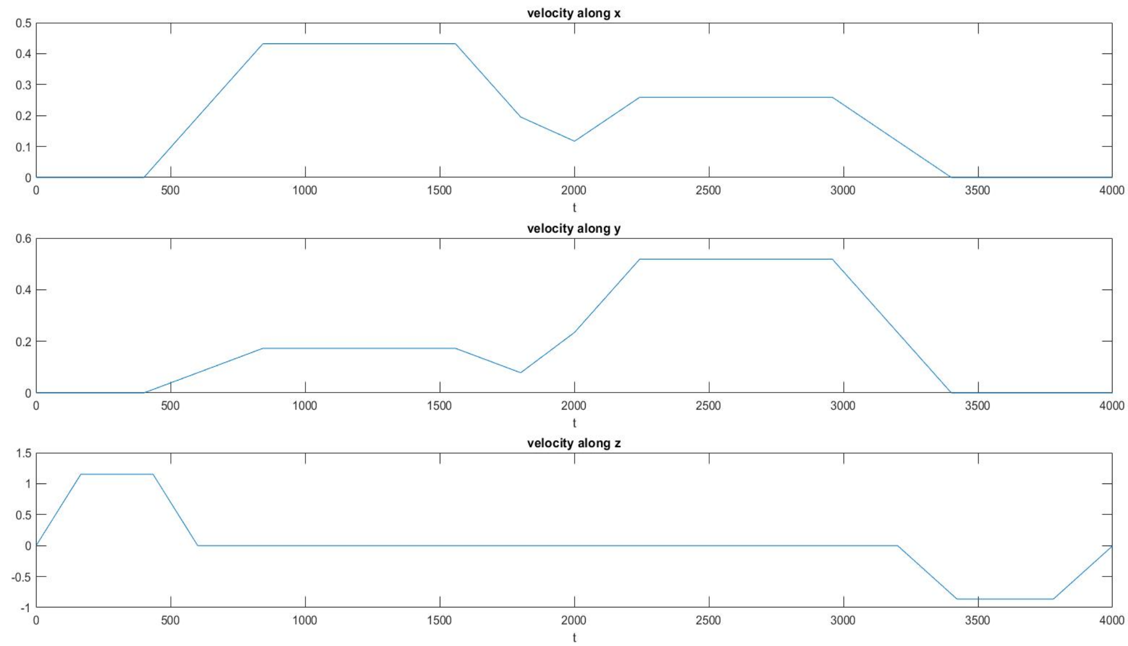
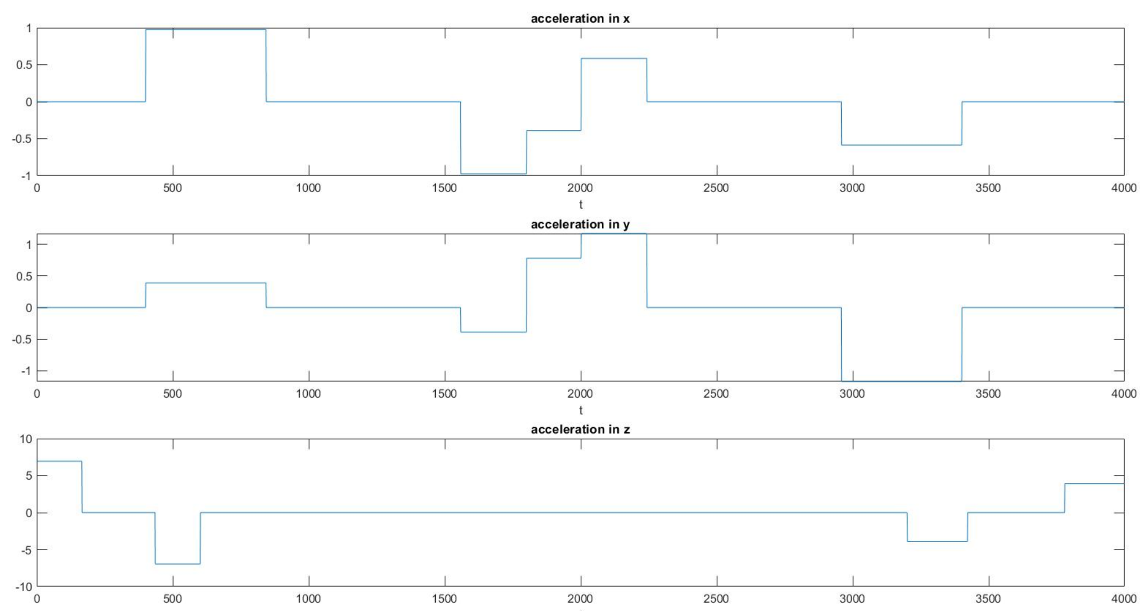


Fig1.2: operational space trajectory

The position, velocity and acceleration in the individual axes with respect to time is shown below:







Q2) **Inverse Dynamic Control**

The Inverse dynamic control approach following block diagram in order to follow the joint space trajectory.

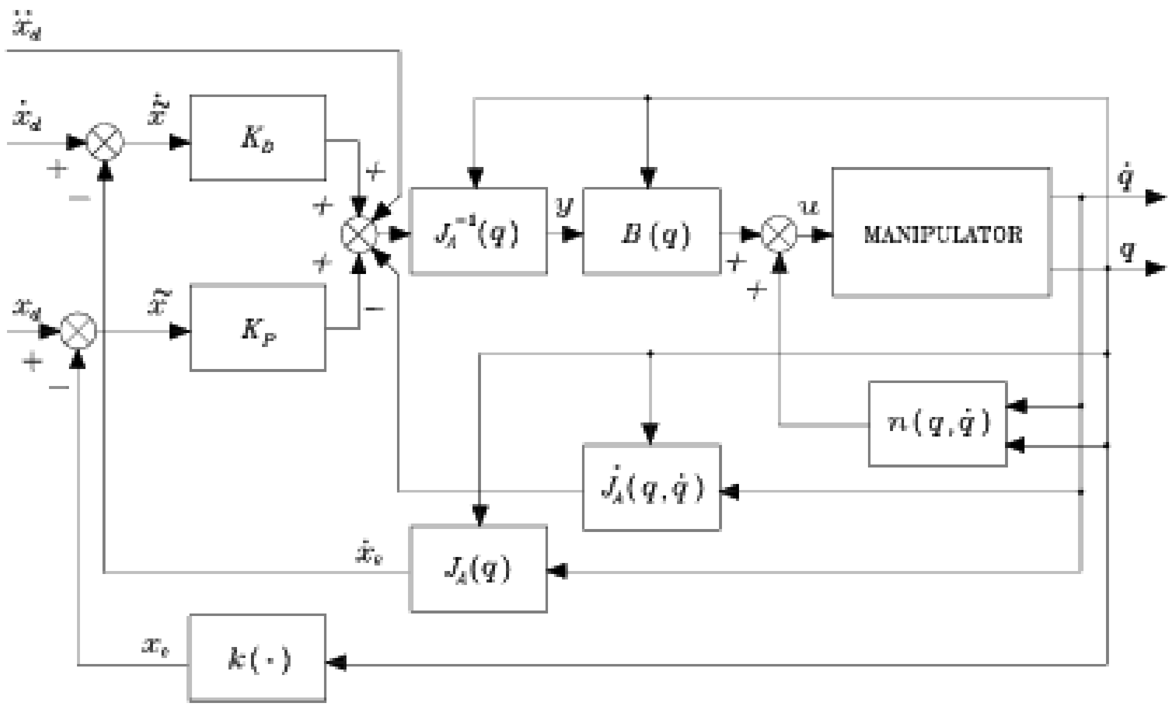


Fig 2.1 : Inverse dynamic control block diagram

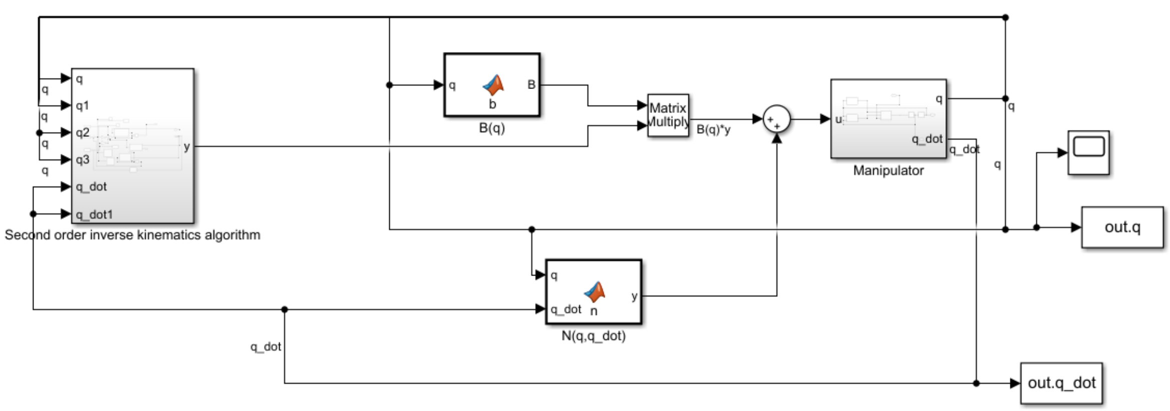


Fig 2.2: Simulink Model of the inverse dynamic control approach

- Fig 2.2 shows the Simulink model used to follow the joint space trajectories.
- The model contains two sub systems: (a)the second order inverse kinematics algorithm and (b)the manipulator.
- In subsystem (a), q1 = q2 = q3 = q and q_dot = q_dot1. Subsystem (a) gives as its output y = q_dot_dot.
- B(q) contains the Matlab code used generate the B(q) matrix

- N(q,q_dot) contains the Matlab code used generate the N(q,q_dot) matrix. N(q,q_dot) is calculated using the equation

$$n(q,\dot{q}) = \dot{B}(q)\dot{q} - \frac{1}{2}\left(\frac{\partial}{\partial q}\left(\dot{q}^T B(q)\dot{q}\right)\right)^T + \left(\frac{\partial \mathcal{U}(q)}{\partial q}\right)^T.$$

$$\left(\frac{\partial \mathcal{U}(q)}{\partial q}\right)^T = g(q)$$

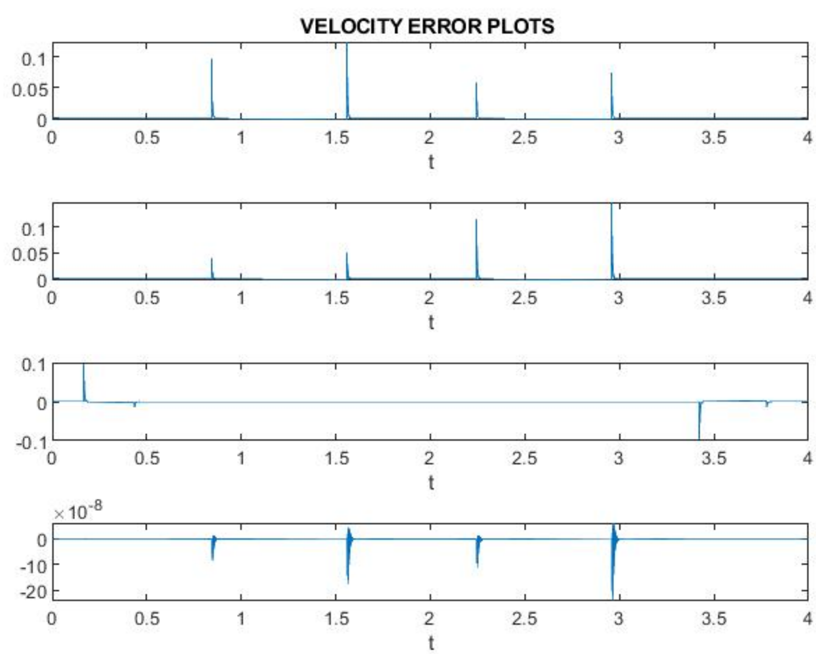
- B(q)+N(q,q_dot) = u, which forms the input to the Manipulator subsystem. The manipulator dynamics are given by the equation below.

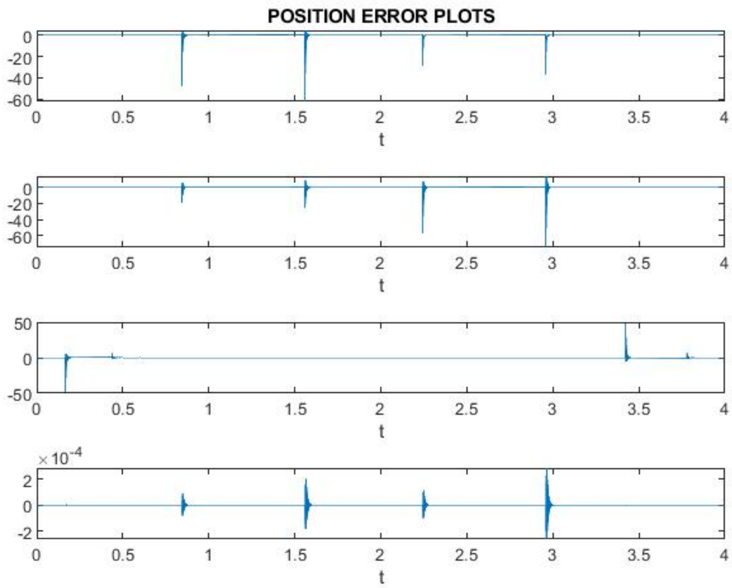
$$B(q)\ddot{q} + C(q,\dot{q})\dot{q} + F_v\dot{q} + g(q) = \tau$$

- The manipulator generates the joint positions and the joint accelerations which on integration gives the joint velocities. Integration of the joint velocities gives the joint positions.

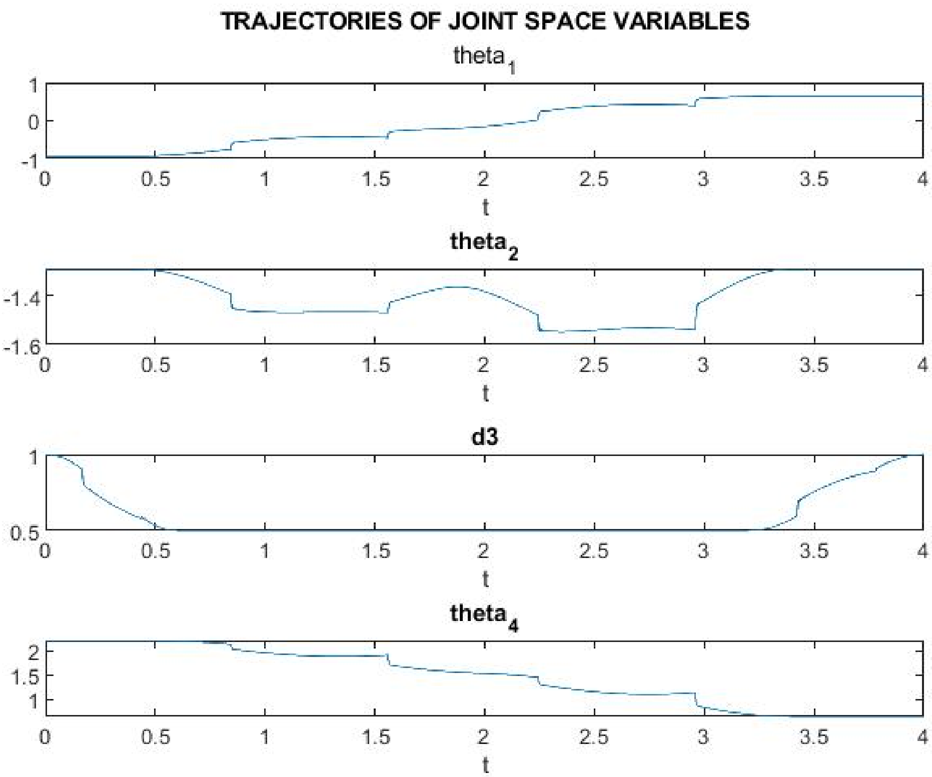
Error Plots

The position error plots and the Velocity error plots are as follows. A the error almost goes to zero, we conclude that the controller works effectively.





The trajectories of the joint space are shown as follows


TRAJECTORIES OF JOINT SPACE VARIABLES
theta_1
theta_2
d3
theta_4
t
0
0.5
1
1.5
2
2.5
3
3.5
4
-1
1
-1.4
-1.6
0.5
1.5