ECOLE CENTRALE DE NANTES

MODELLING AND CONTROL OF MANIPULATORS

LAB REPORT – 5

Submitted by

REGULAN GOPI KRISHNAN

RAMACHANDRAN RAGESH

November 2017

1. Objective

The main objective of the present lab is to study the behavior of a Kuka LWR Robot model in the Matlab environment by applied some classical linear and non-linear robotic control (PID, Computed Torque) in the joint space and/or the task space and then to compare it with results obtained with Adams.

Co-simulated Model

Co-simulated method takes advantages of two software, with enhancing dynamic performance of robot arm, improving efficiency, reducing the cost and saving time. For the complex control system, it is a good solution selected.

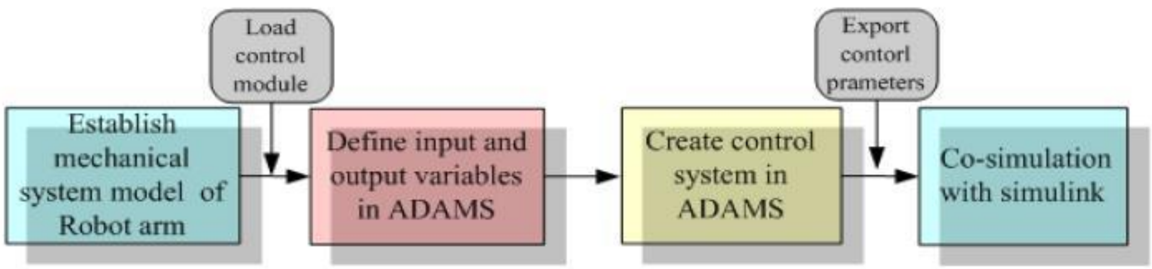


Figure 1: Co-simulation flow chart of ADAMS and MATLAB.

2. Establishment of Kuka LWR robot dynamics model

The IDM (Inverse Dynamic Model) of a robot calculates the torques _ as a function the motor positions (q), velocities (qdot) and accelerations (qdotdot). It can be obtained from the Newton Euler or the Lagrangian equations. The kuka LWR robot has a serial structure with n = 7 rotational joints. Each motor has encode which measures the motor positon.

j	σ_j	α_j	d_j	θ_j	r_j
1	0	0	0	θ_1	0
2	0	$\pi/2$	0	θ_2	0
3	0	$-\pi/2$	0	θ_3	$r_{l_3} = 0.4$
4	0	$-\pi/2$	0	θ_4	0
5	0	$\pi/2$	0	θ_5	$r_{l_5} = 0.39$
6	0	$\pi/2$	0	θ_6	0
7	0	$-\pi/2$	0	θ_7	0

Table 1: MDH Parameters of Kuka LWR robot.

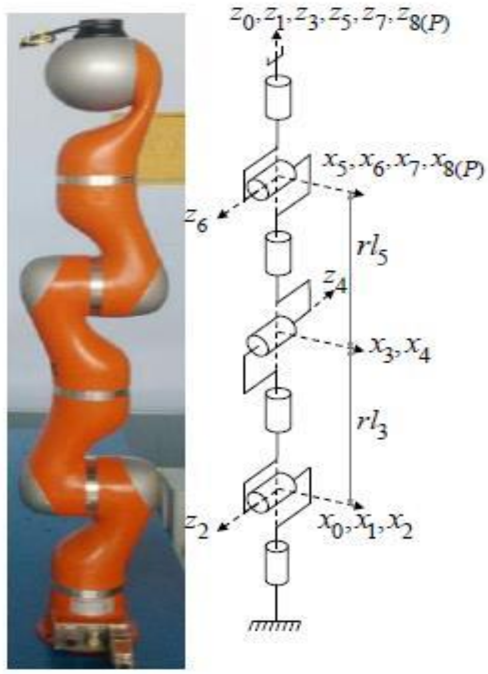


Figure 2: Link frames of the Kuka LWR.

The kinematics od serial robots is defined using the Modified Denavit and Hartenberg (MDH) notation.

Modeling validation

After opening "simu_lwr_rig_essentiels.mdl" the Simulink diagram is shown as in fig. 4. It has LWR MDD block, PID controller block and the unit delay is given before the direct dynamic model. The output from the MDD block are qs, qds, and qdds. The qs from the MDD clock is given to the PID block as one input and the other inputs to the PID block are qr, qdg and antiwindup. The antiwindup contains a column matrix of 1*7 zeros. The output from the PID block is given to the unit delay and then to the LWR MDD block.

LWR MDD FULL block

The input tau is given to the direct dynamic model and the integrator is used twice to get the qds and qdds. The qds and qdds is given as the inputs to the direct dynamic model and the loop is executed. The qs, qds and qdds are found out from the output. The block is shown in Fig. 5

PID block

PID block is shown in Fig. 6. PID block does the controller part which controls the deviation from the input and the output.

Unit Delay

The unit delay is given before the LWR MDD block to ensure the deviation from the time. The error is multiplied and given in the same equation and therefore the unit delay is given. For eg. In the equation, X = y + Z*Z + 7 – X, the X is given as the same value so the error repeats. In order to eliminate the equation has to be changed to $X = y + Z*Z + 7 - X_{n-1}$. Now this is given by unit delay.

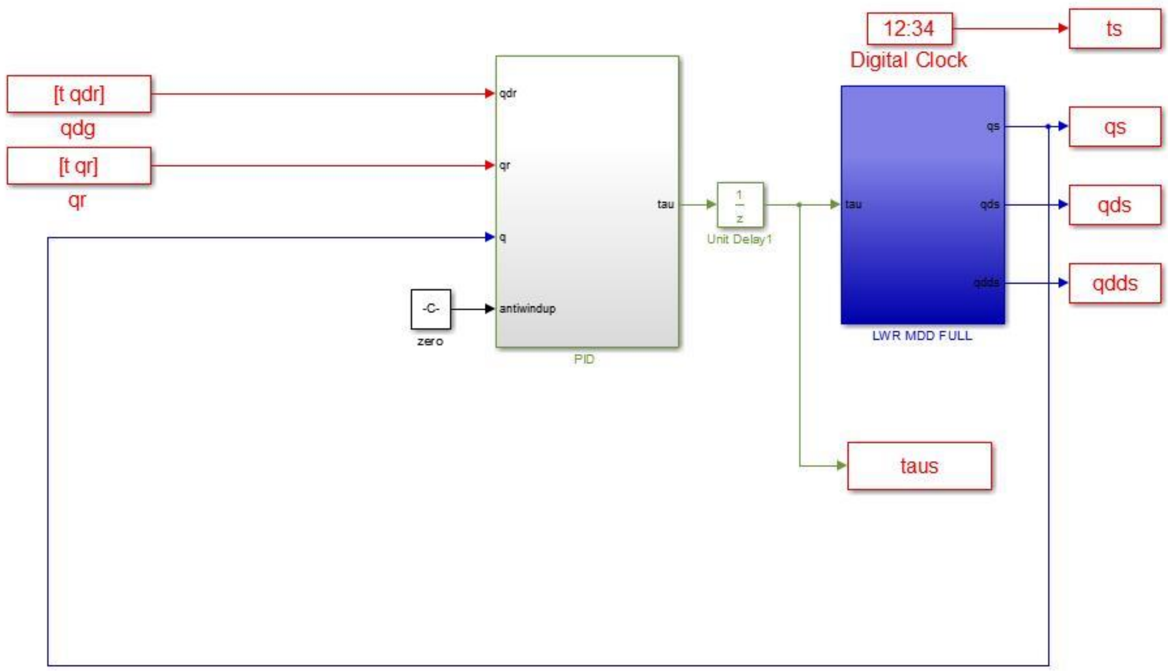


Figure 4: First simulator of the Kuka LWR robot

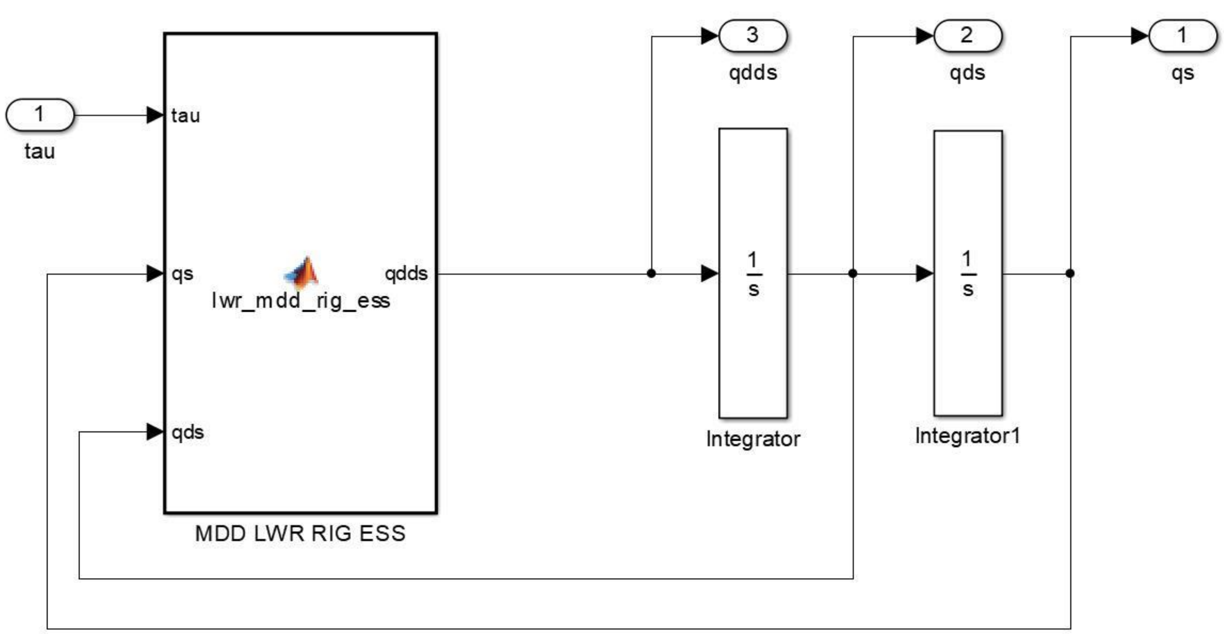


Figure 5: LWR MDD FULL block

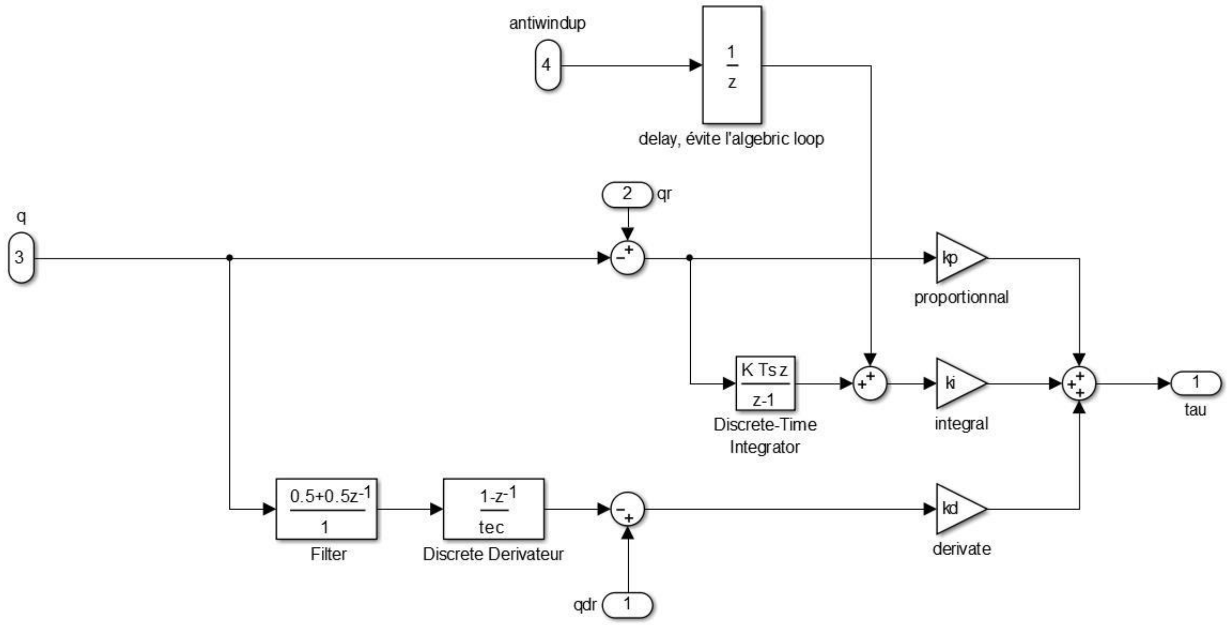


Figure 6: PID controller block

Consignes.mat

Consignes.mat contains the following datas:

qddr	<27744x7 double>	-11.45...	11.4499
qdr	<27744x7 double>	-3.4949	3.2073
qr	<27744x7 double>	-2.5831	2.6005
t	<27744x1 double>	0	27.7430

3. Establishment of robot arm co-simulation model with ADAMS and Matlab/Simulink

Co-simulation with ADAMS and MATLAB /SIMULINK means that build multi-body system in ADAMS, output parameters related to system equation and then import information from ADAMS into MATLAB/SIMULINK and set up control scheme. During calculation process, there exchanges data between virtual prototype and control program, where, ADAMS solves the mechanical system equation and MATLAB solves the control system equation. They both finish the whole control process.

3.1 Building the interface between two software

The model built in ADAMS, as a sub-system, need to be imported into MATLAB/SIMULINK, on which SIMULINK constructs the co-simulation system. First, exchange data between ADAMS and MATLAB/SIMULINK through ADAMS/CONTROL interface. Second, define 21 system variables (Creation of the Input/output of the plant) which is needed in co-simulation such as:

_ input variable: tau1, tau2, tau3, tau4, tau5, tau6, tau7;
_ output variable:

1. q1, q2, q3, q4, q5, q6, q7 (joint positions);
2. q1dot, q2dot, q3dot, q4dot, q5dot, q6dot, q7dot (joint velocities).

After defining the Input/Outputs variables, we export the Matlab model from ADMAS/CONTROL. This will generate three _les (.m, .cmd and .adm _le) which is useful in data-exchange between ADAMS and MATLAB. In our case, these are the files: CPU_jnt_space.m, CPU_jnt_space.cmd and CPU_jnt_space.adm

3.2 Simulation

3.2.2 Developing in Matlab/Simulink a simulator to control the Kuka robot (Adams model) with computed control law and in the joint space.

Refer to the Program "test_lwr_trq_control_joint".

The Simulink diagram for the Kuka robot (Adams model) with computed torque control law and in the joint space is given below (Fig. 7). The interpreted MATLAB function called "slk_trajectory" is given as the input. The desired paths of the seven variables are taken respectively from q_d, qd_d, qdd_d. q_d and qd_d are given to the kp and kv which are called gain. From this error is taken respectively from both the functions. q_d, qd_d, qdd_d are given to the interpreted MATLAB function called "slk_idym". Then this output is given to the KUKA KWR4+ ADAMS. Then the output is given to the measured values of all the seven variables. w + tau is also taken.

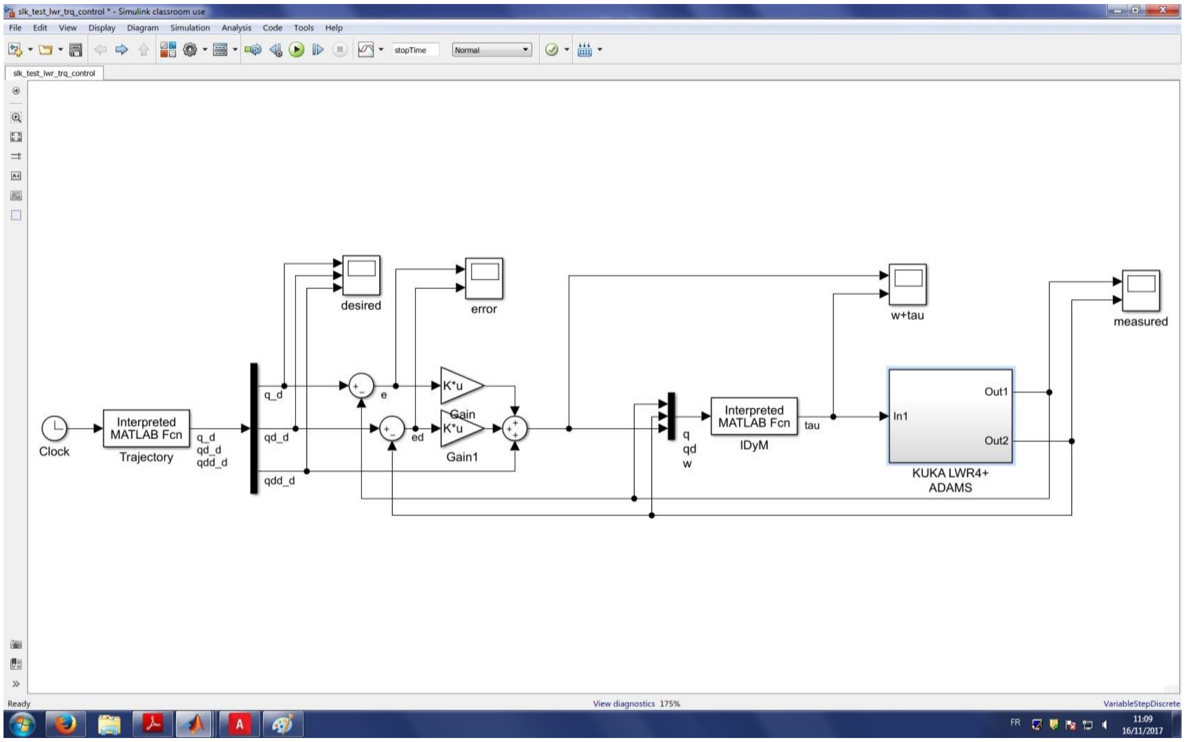


Figure 7: Computed torque control diagram

The following figure explains the adams_sys model where the S-function and stat-space variables are defined. The adams_sub block defines that it has totally 21 variables.

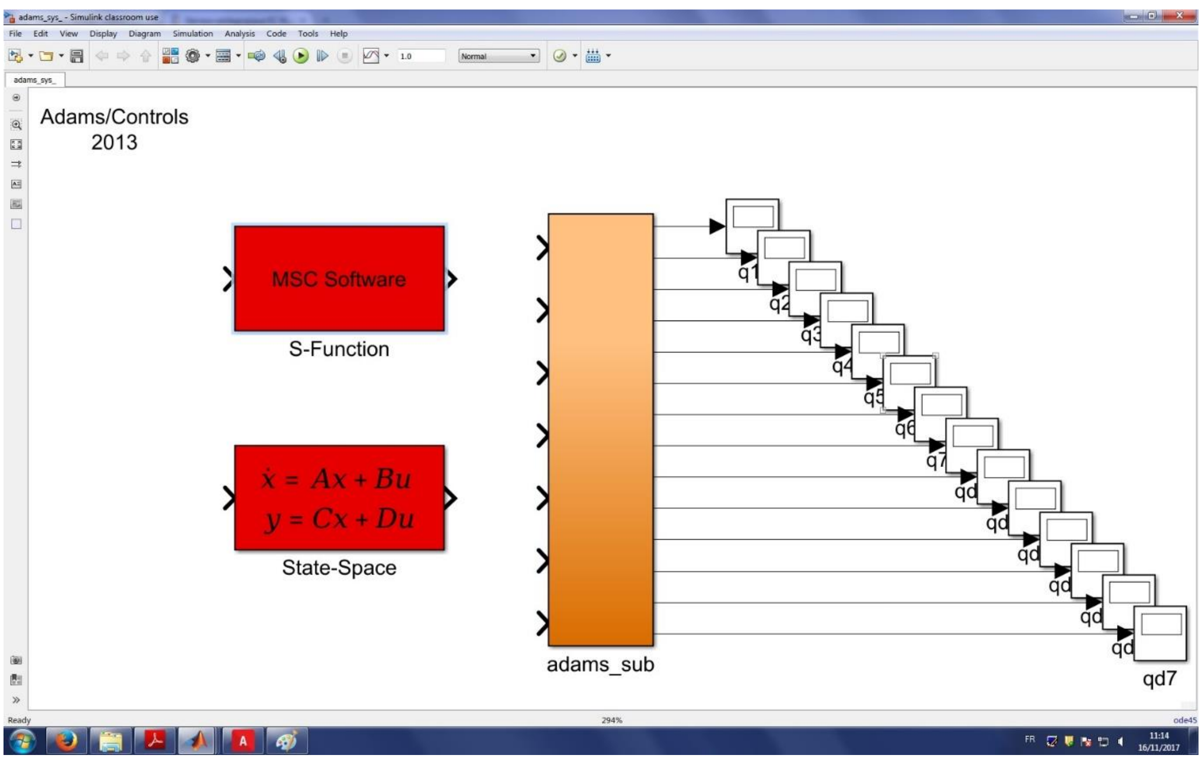


Figure 8: adams_sys

The graph of desired q, qd and qdd are shown in fig. 9. The path tells the q contains seven variables and qd contains seven variables. These variables follows the sinusoidal function which gradually increasing in magnitude from the first to the seventh variable respectively. qdd follows the trapezoidal profile.

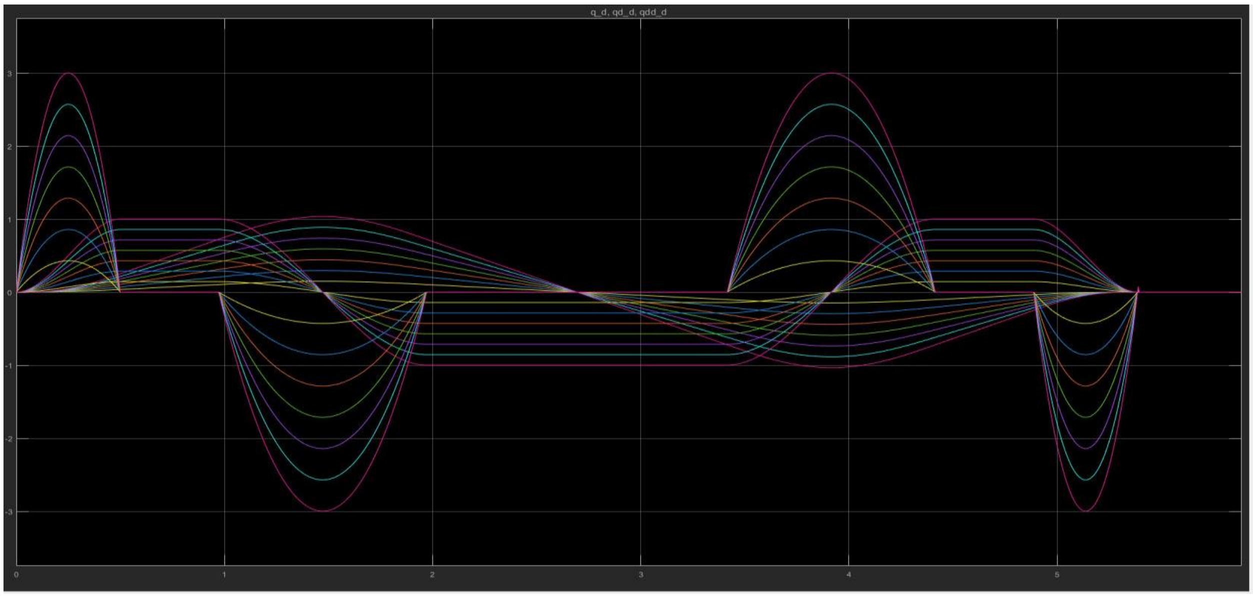


Figure 9: Desired q, qd, qdd

Fig. 10 explains the error of e and ed. The graph shows that there is a very minimum error of +0.015 to -0.015.

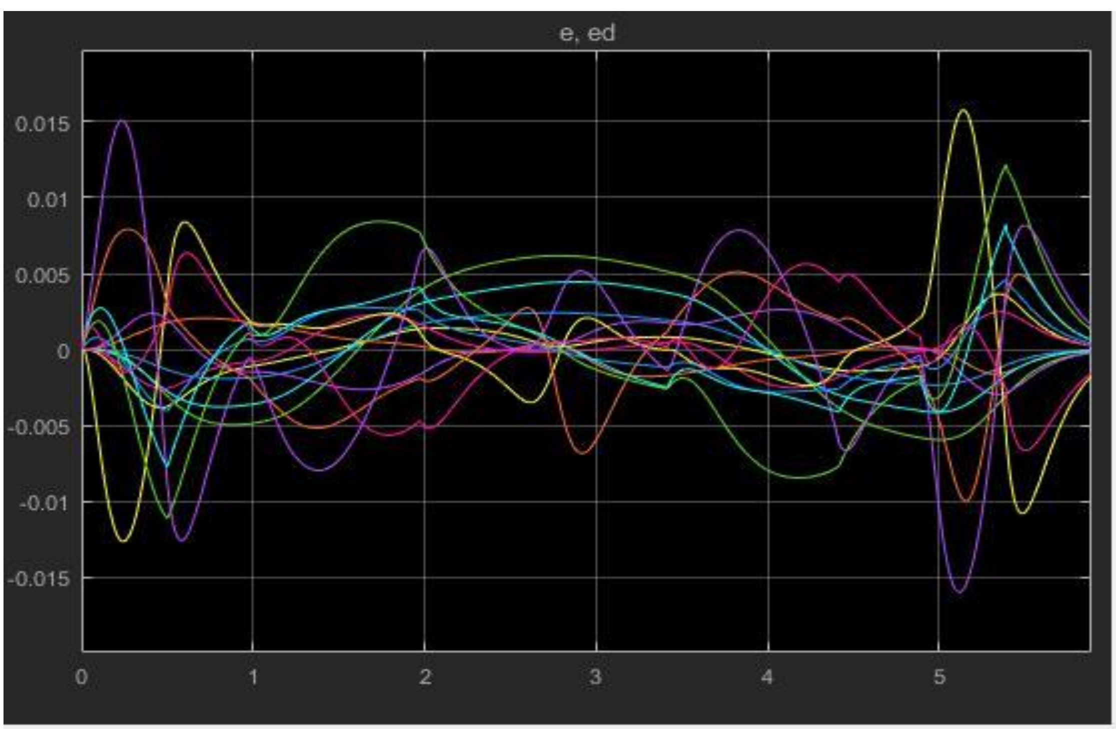


Figure 10: Error CTC e and ed

The KUKA LWR4+ ADAMS block is built as shown in Fig.11. The S-function which is adams model contains 14 variables as the input and is multiplexed into two variables and given memory blocks separately and then to the output. This is remained as the output obtained should be store every time in the memory block.

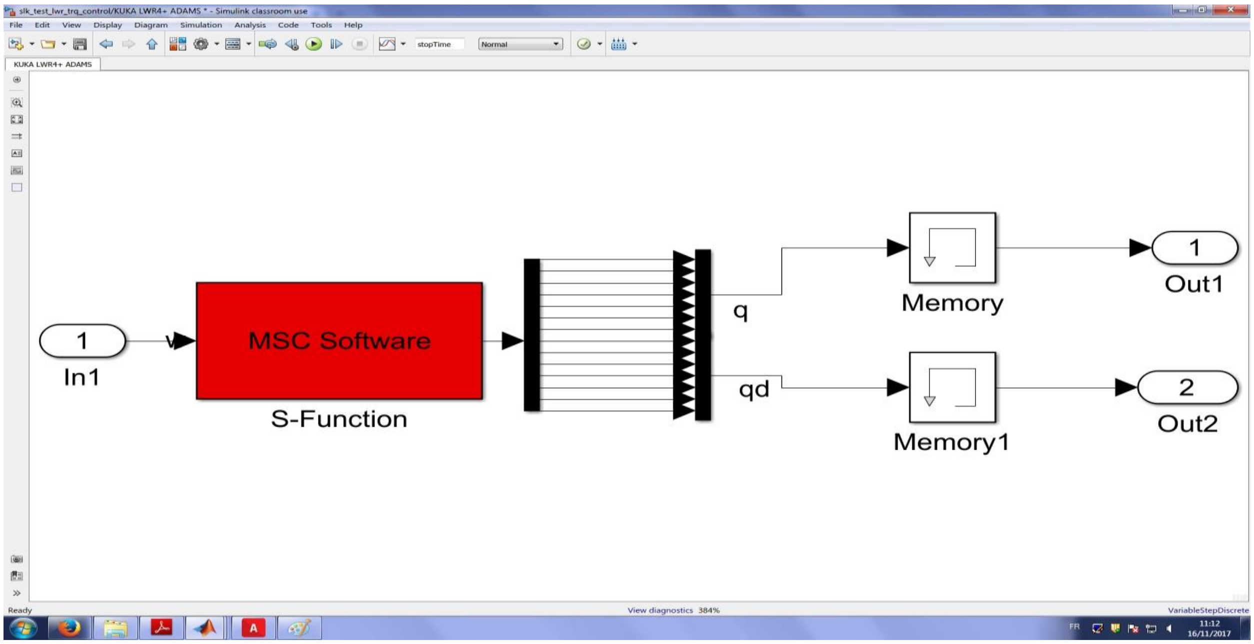


Figure 11: Inside adams

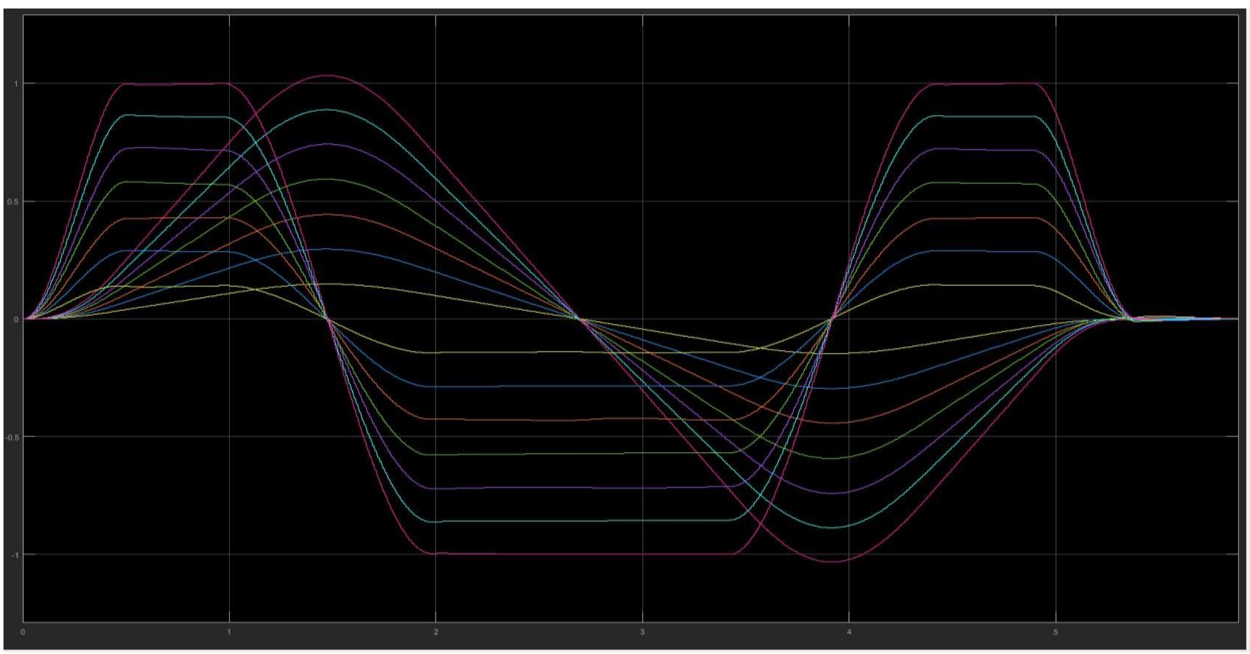


Figure 12: Measured ctc

Fig. 12 explains the measured value of output. These are given in sinusoidal and trapezoidal profile.

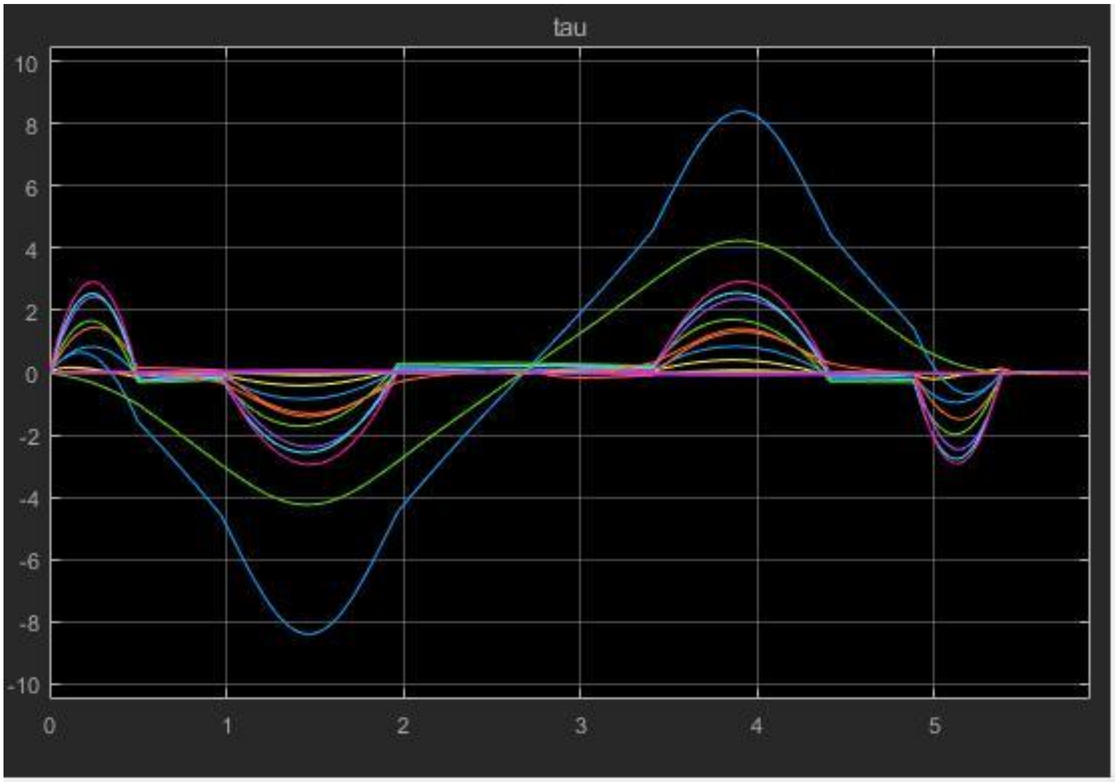


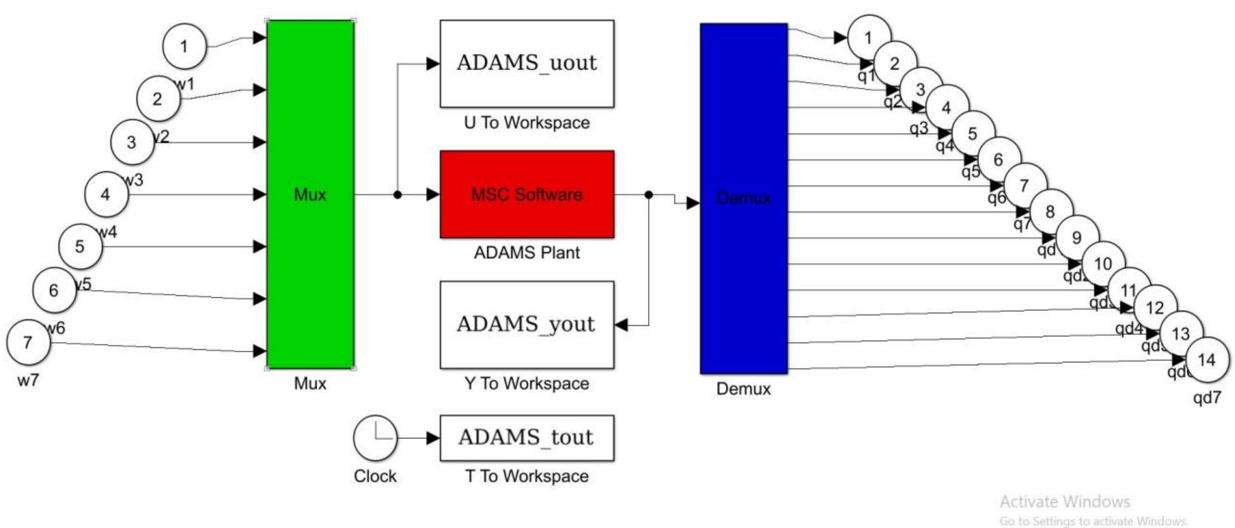
Figure 13: w+tau

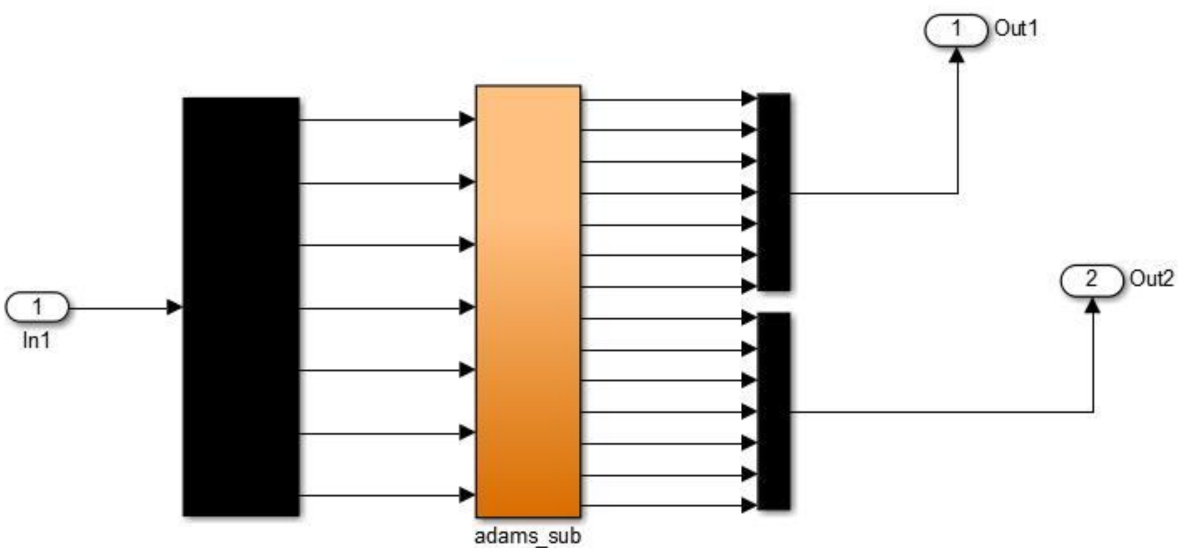
Fig. 13 explains the addition of output from the Interpreted MATLAB function “slk_idym” and the output from the input variables. The graph explains that the combination of sinusoidal and trapezoidal profiles which result in sinusoidal at one moment and then converges to zero for some time and then again sinusoidal continues. Some variables are fully sinusoidal not influenced by the trapezoidal functions and the maximum value of tau will be +8.3 to -8.3.

3.2.3 Develop in Matlab/Simulink a simulator to control the Kuka robot (Adams model) with computed control law and in the task (cartesian) space.

Refer to the Program "test_lwr_trq_control_task".

The computed torque control in the joint space is given below in the diagram.





Refer to the Simulink diagram "slk_test_lwr_trq_control_Task_space.mdl"