TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

NAME:PALLAB GANGULY :: STUDENT NUMBER:267602
NAME: LE DUC TOAN :: STUDENT NUMBER:267945

IHA-2576 PROJECT WORK 2017-2018

Documentation quality (clear and well documented pictures etc.)	**/ 2**
Model (clear and logical structure, not too many blocks on a single level)	**/ 2**
M-files (logical structure, well commented etc.)	**/ 1**
Model based pressure compensator (working 0.5 doc. 0.5)	**/ 1**
Sum-flow control (working 0.5 documentation 0.5)	**/ 1**
Switching cost function (working 0.5 documentation 0.5)	**/ 1**
Cross-flow cost function (working 0.5 documentation 0.5)	**/ 1**
Simulation results (correctness, well documented)	**/ 5**
Simplified PNM controller (bonus)	**/ max 1**
Feedback	**/ 1**
Total	**/ 15**

1. ABSTRACT

The objective for the project work is to work with Digital Valve system and to design a model-based controller. The system consists of a supply unit along with a digital valve system and an asymmetric cylinder moving load mass vertically. The following subsystems were designed for the above system.

- Upper level controller
- Model based valve controller
- Valve system
- Cylinder and load

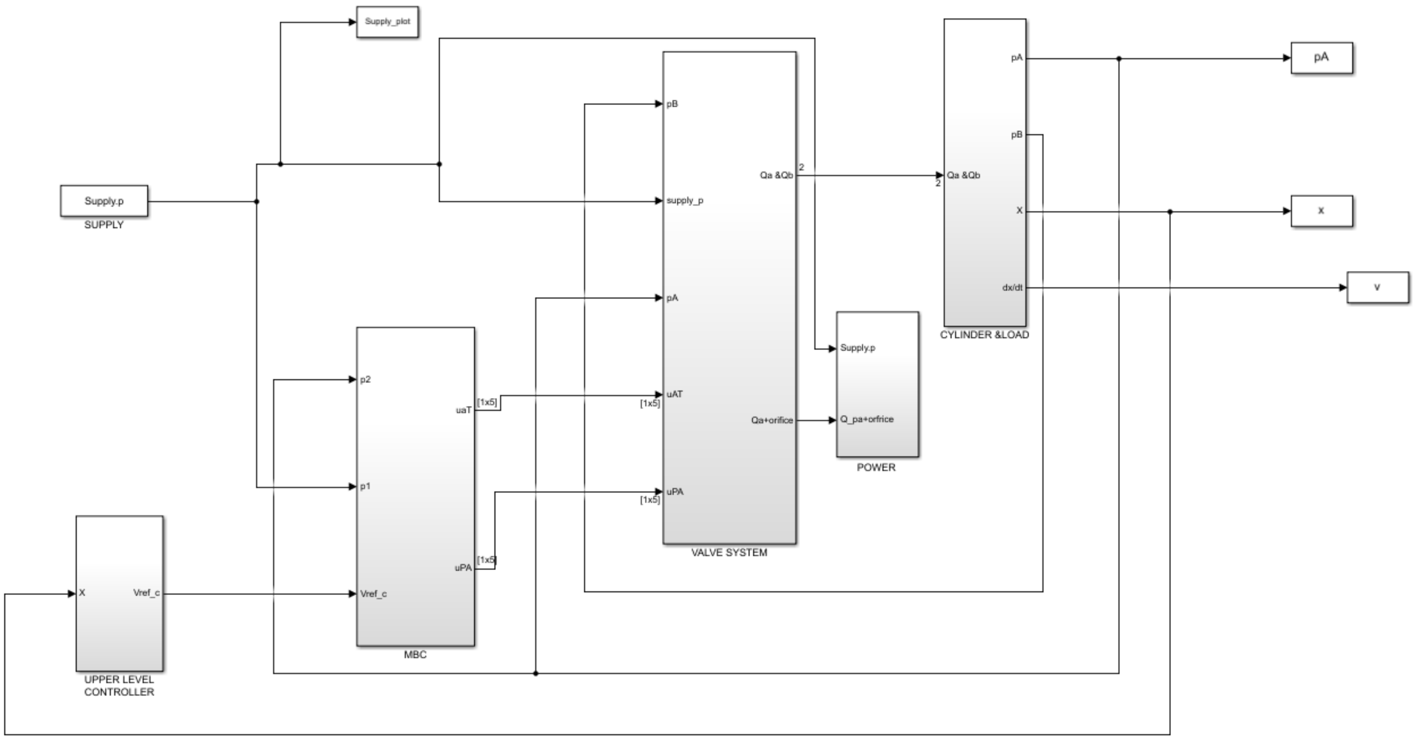


Figure 1:System Overview

2. SYSTEM ANALYSIS

2.1 CONTROLLERS:

In our project work we have designed two levels of controller

- Upper-level controller
- Core level controller

UPPER-LEVEL CONTROLLER:

Inside the upper level controller, we first used a velocity feedforward in the system which enables only following certain trajectory. It is a open loop system, because of that the motion control depends only on the accuracy of the model.

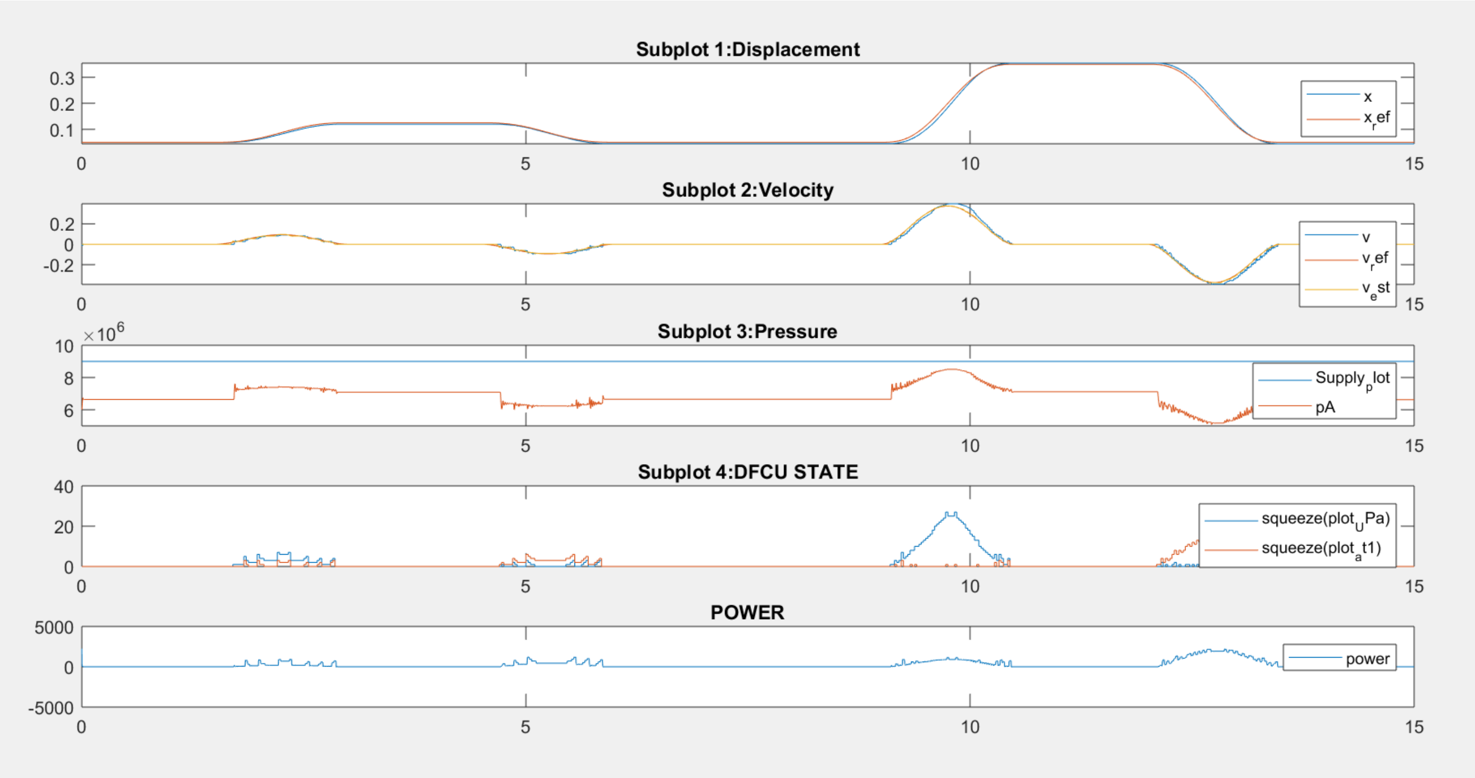


Figure 2:Velocity feedforward

```
Kff=1;%Gain of the controllers Kff and Kp
Kp=0;%Kp=0 when we consider only velocity feedforward
```

The simulation above in *Fig:2* illustrates that when velocity feedforward controller is used, the position and velocity reference does not actually follow that the measured position and velocity from the piston. This results in instability and not considered a good way to approach so we use a position feedback P-controller .

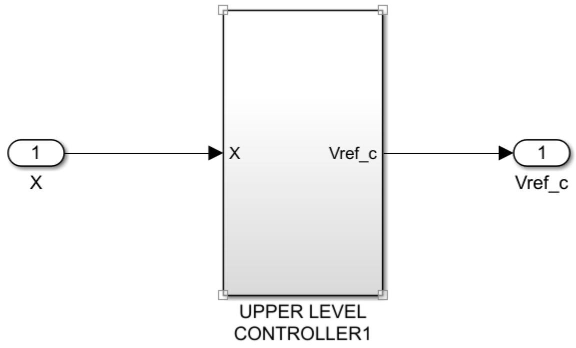


Figure 3:Upper level controller subsystem

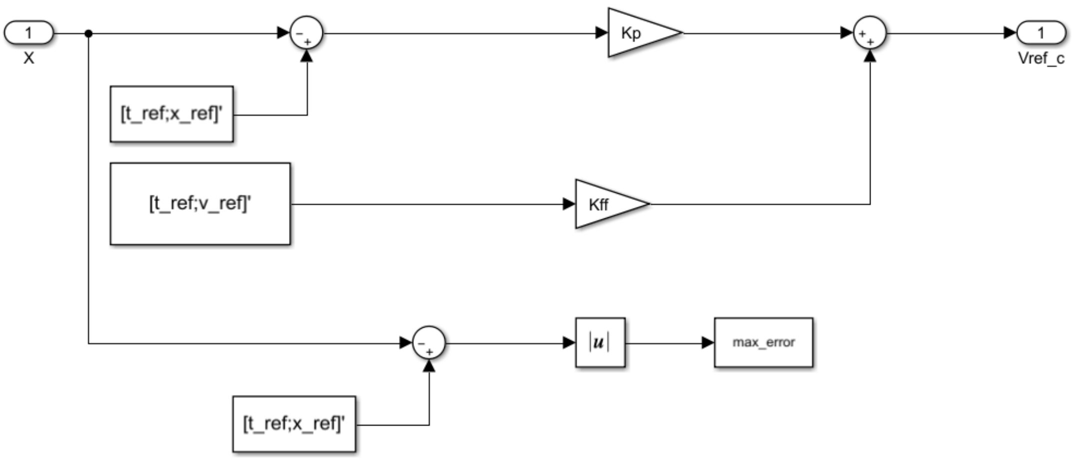


Figure 4:Upper level controller blocks

- ***Kp*** and ***Kff*** are the controller gains and are tuned to better results

CORE LEVEL CONTROLLER:

Theoretical Background:

The core level controller we designed a SUM-FLOW controller which includes two DFCU's working simultaneous. Among them one is considered a primary DFCU and the other is a secondary DFCU. The purpose of using a sum flow controller is to increase the controllability and secondly using two DFCU's at the same time allows the system to work with higher velocities that means the secondary DFCU can close in higher velocities. When the movement of the piston is the upward direction DFCU PA is considered as a primary DFCU because flow due the pressure (supply) is generated from the port PA due to which the piston is moving up and DFCU AT at that point is considered secondary and the role is reversed when the piston is moving down. With such a combination we can control the piston velocity more accurately. The control matrix used is a combination of both the DFCU's and is a function of number of valves.

```
ctrl_mtrx = fliplr(double((dec2bin(0:2^N-1))=='1'));
ctrl_mtrx2=combvec(ctrl_mtrx',ctrl_mtrx')'; %for sum flow controller using binary coding
```

```
ctrl_mtrx =tril(ones(N+1,N),-1); %for single DFCU
ctrl_mtrx2=combvec(ctrl_mtrx',ctrl_mtrx')'; %for sum flow controller using equal coding
```

The above equations depicts the control matrix for ***SUM FLOW CONTROL*** using both binary and equal coded system. Where ***N*** denotes the number of valves used in each of the DFCU's.

After we designed the sum flow controller it was observed that there were unnecessary valve switching's which caused adverse effects in the system such as unwanted noise, valve wear and also it made the system little unstable, moreover having too many valve switching uses lot of energy consumption so it was necessary to introduction some cost function which could reduce the unnecessary valve switching's so ***SWITCHING-COST*** was introduced and it worked quit effectively.

In the sum flow controller, we have merged the two DFCU's and the idea was that they should work simultaneously one as a primary DFCU and the other as secondary. And the control matrix designed was also a combination of both the DFCU's. Now the challenge before us was that a condition must be given to the controller that if the piston is moving in the positive direction DFCU PA is primary and AT secondary and vise-verse for implementing this idea ***CROSS-FLOW*** was implemented which divided the control matrix into two different parts DFCU-PA & DFCU-AT and then a switching condition was applied.

<u>The High level Architecture of Core-level Controller:</u>

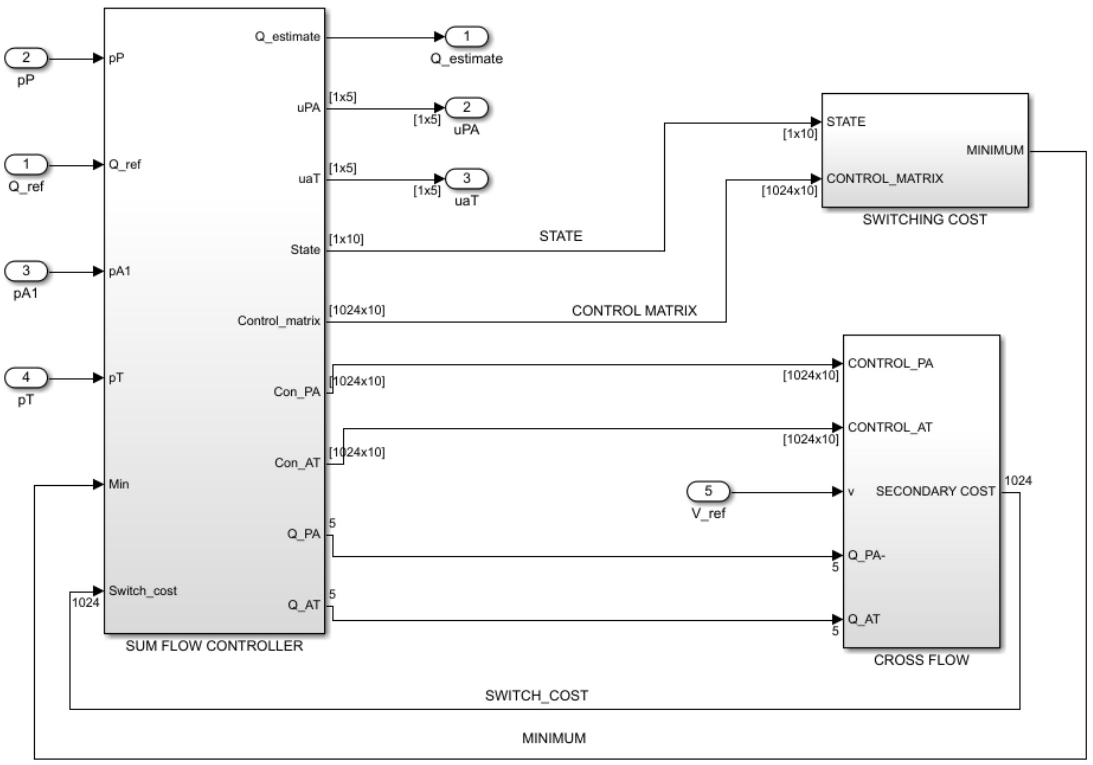


Figure 5:Core level Controller

Model analysis of SWITCHING COST and CROSS-FLOW;

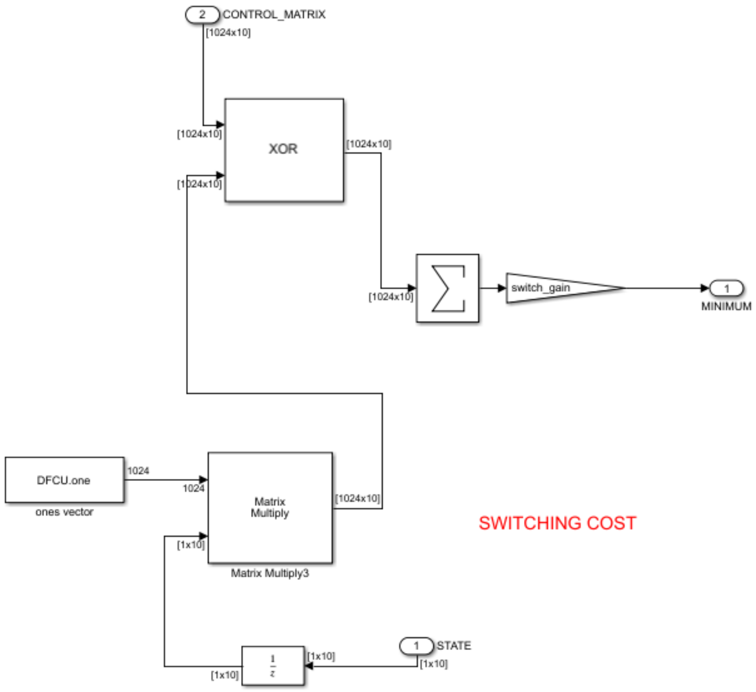


Figure 7:Switching cost blocks

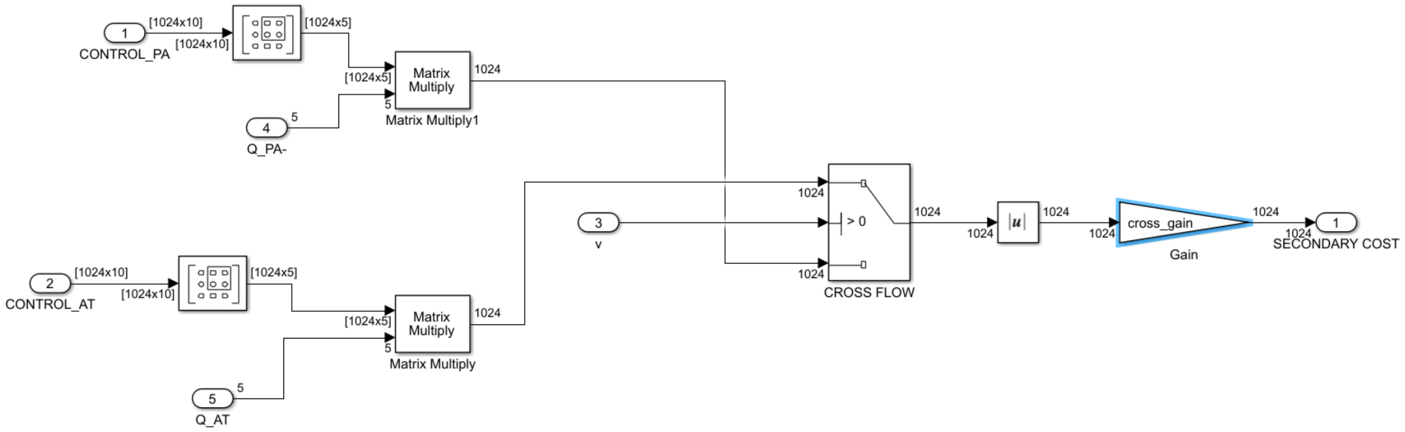


Figure 6:Cross flow blocks

In *Fig:6* the switching cost calculates the number of valve switching that is necessary to achieve any state from the current state. The output is a vector and contains number of switches from every state. A delay block is necessary to prevent unnecessary algebraic looping.

In *Fig:7* we divided control state of both the DFCU's using a sub matrix blocks and gave a condition using a switch block to ensure positive and negative movement of the piston during each stroke of the cylinder.

3. SIMULATION STUDY

3.1 System properties vs. Cost function gains.

Considering cost function gains=1

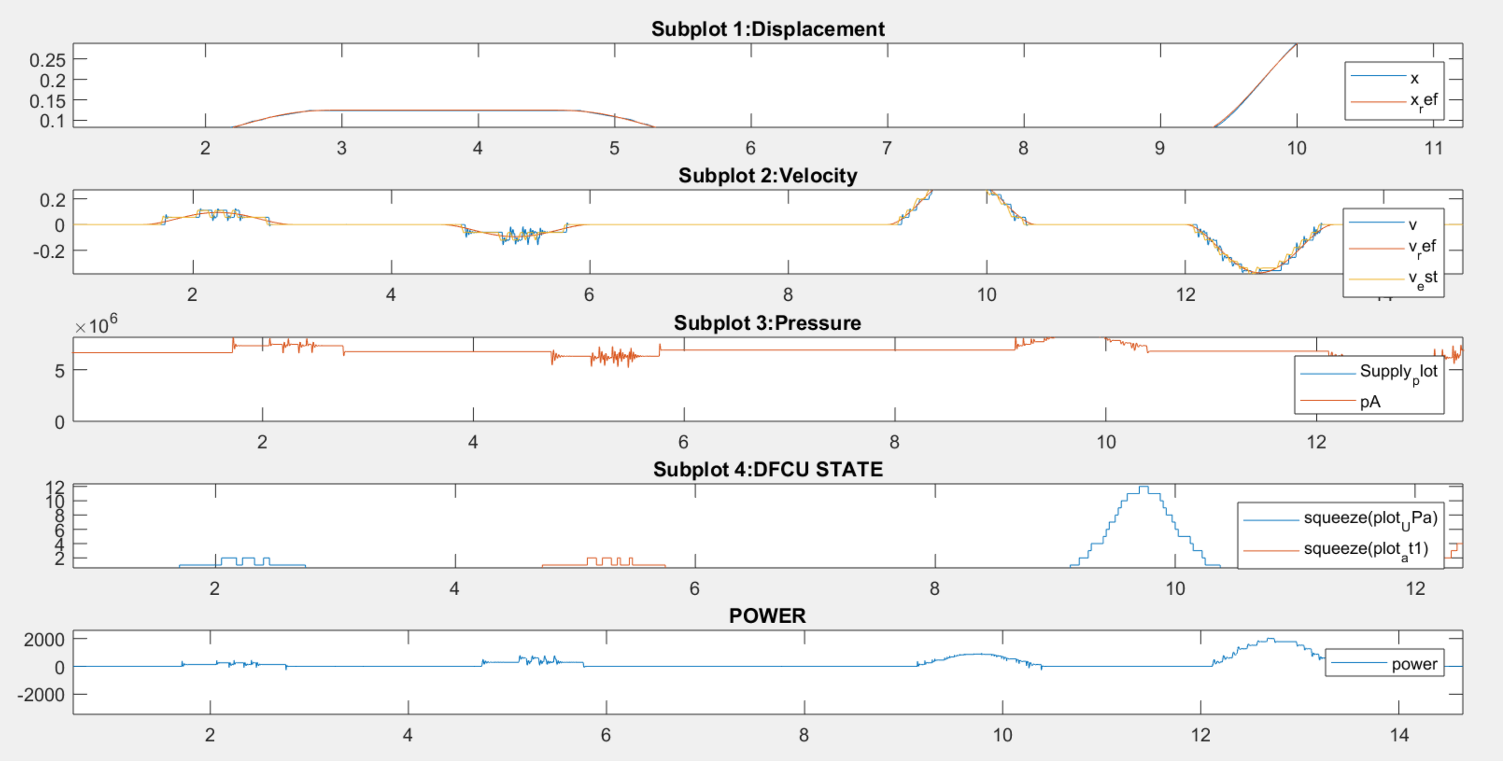
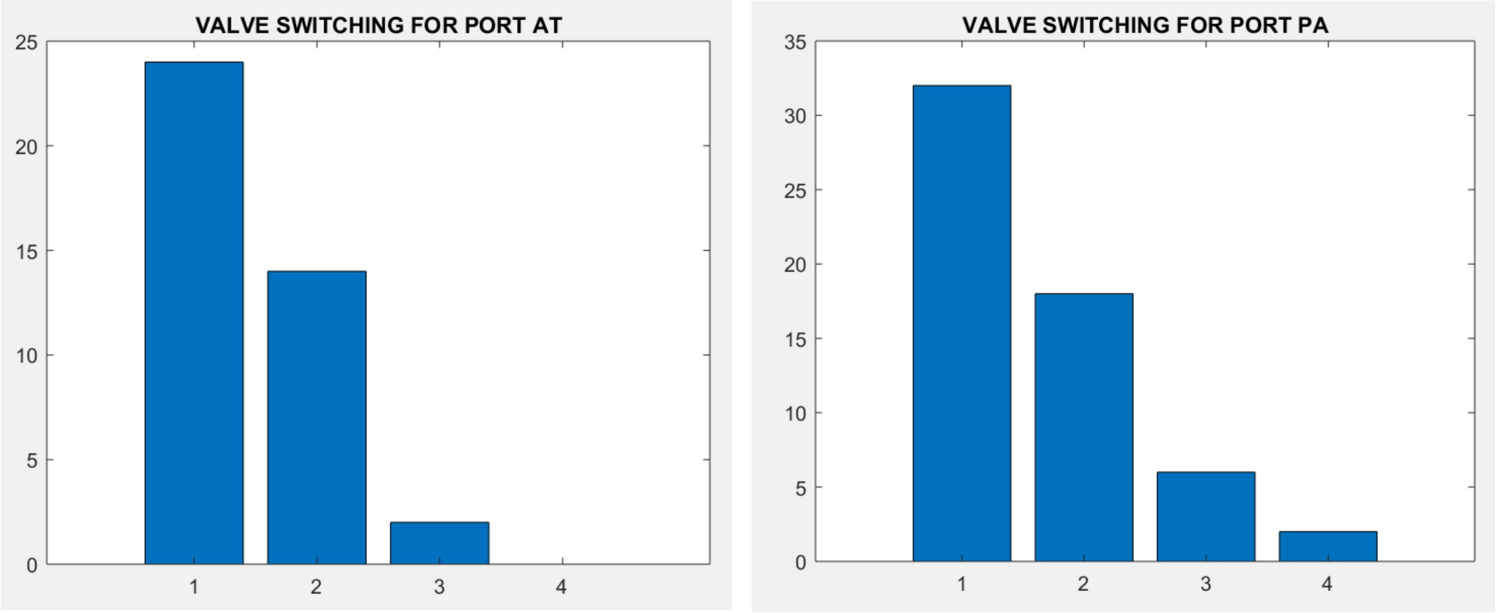


Figure 8:Cost function Gain=1



Valve Switching's at Cost function gain =1.

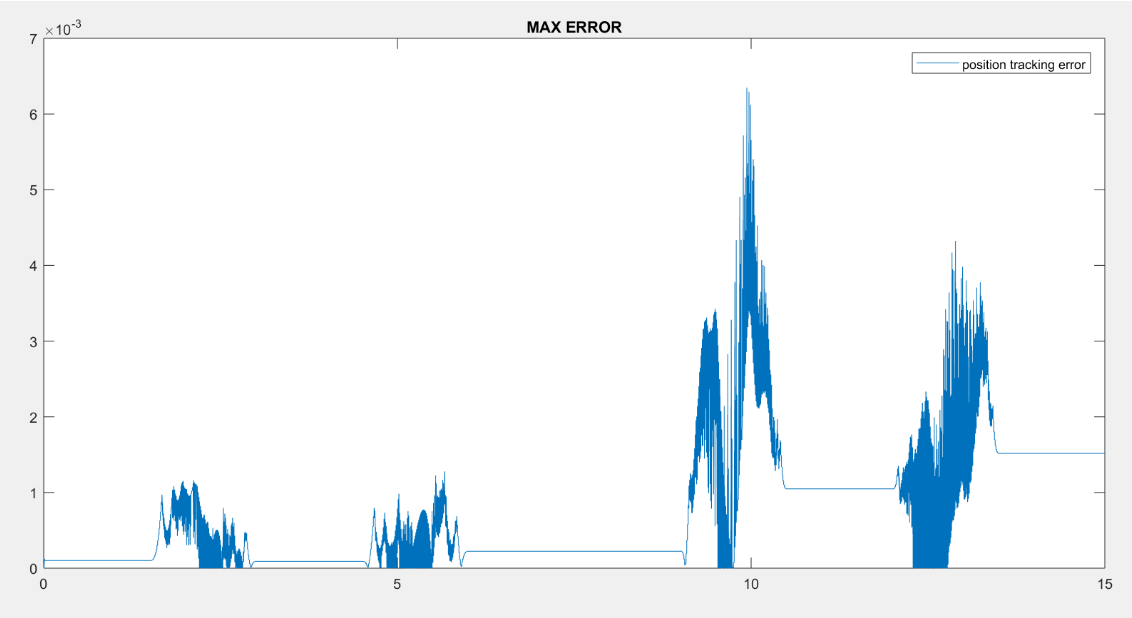


Figure 9:Max_error at gain=1

When Cost function gain is 5e-2:

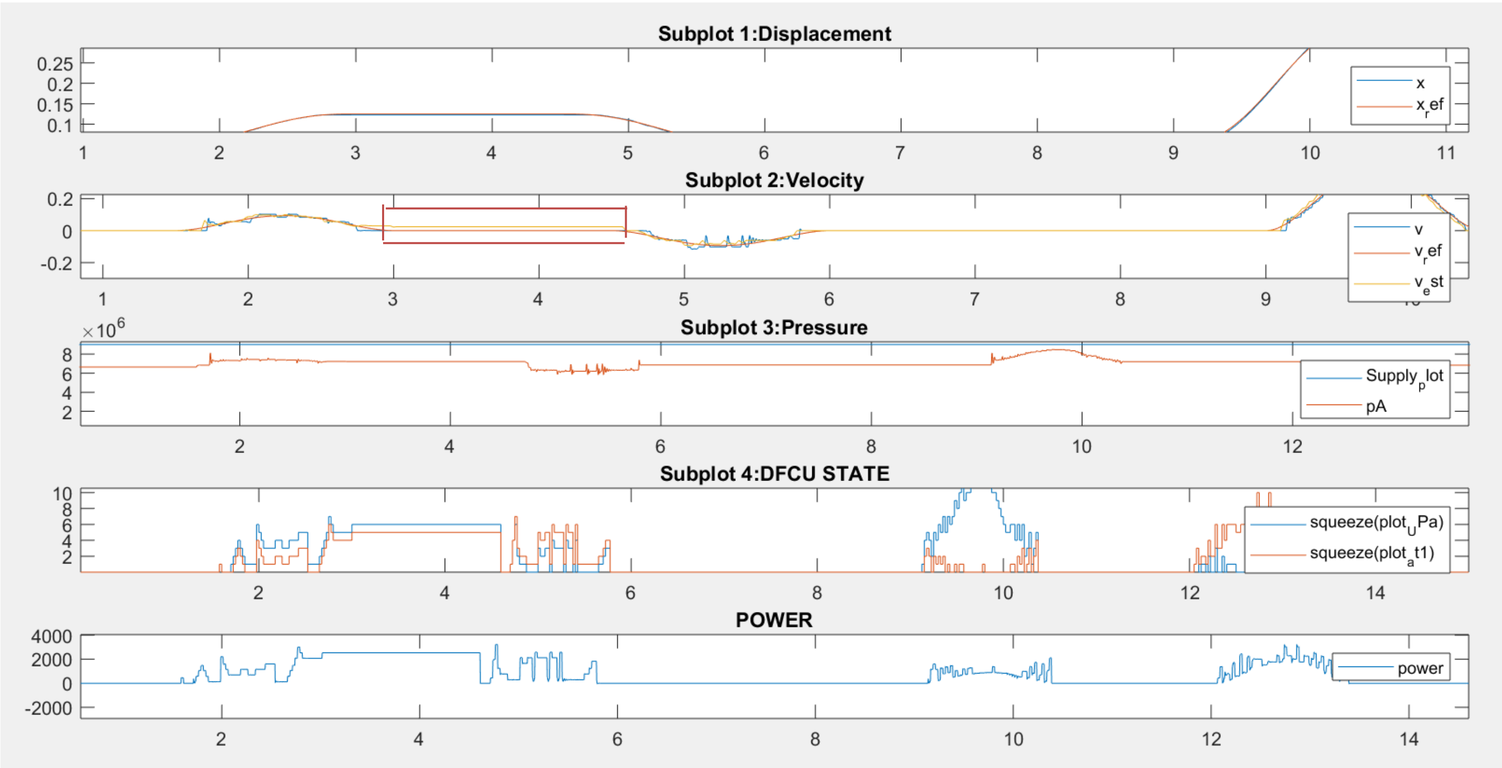
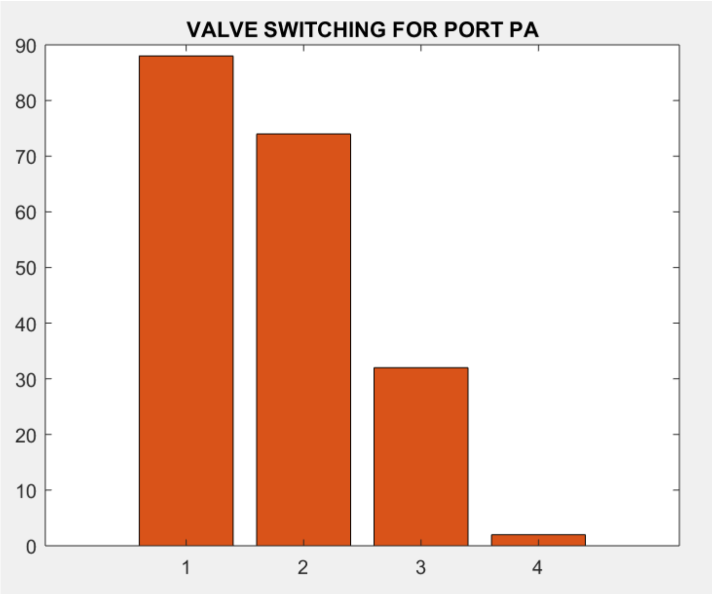
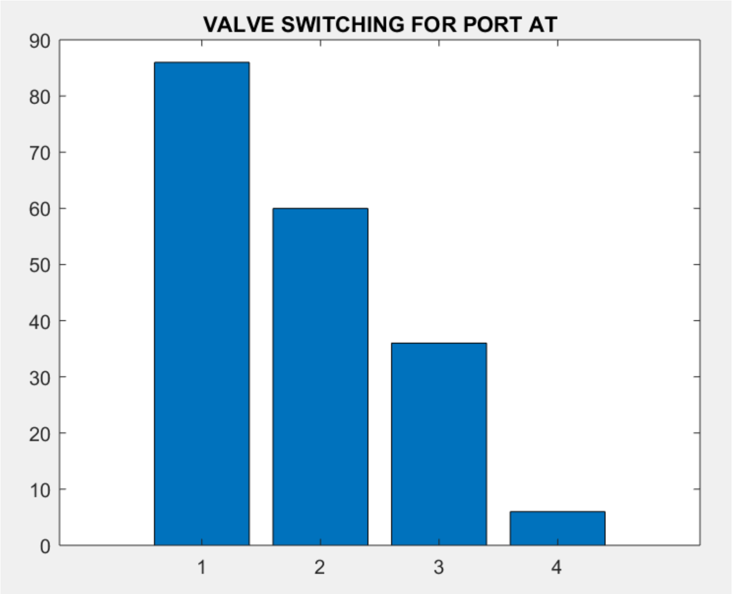


Figure 10:Cost function gain=5e-2





Valve Switching's at Cost function gain =5e-2

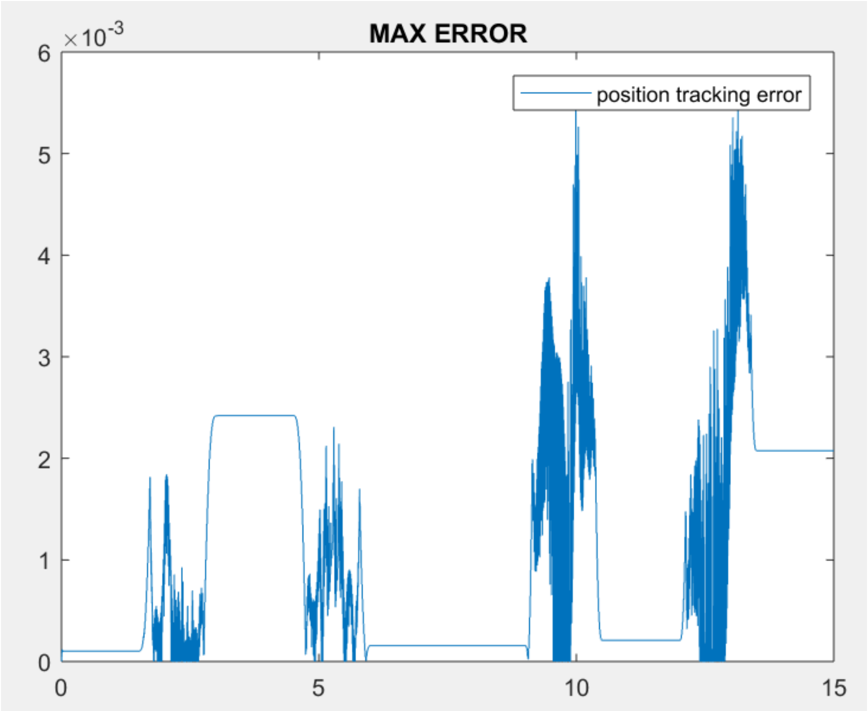


Figure 11:Max_error at gain=5e-2

Cost function gain is 5e-6:

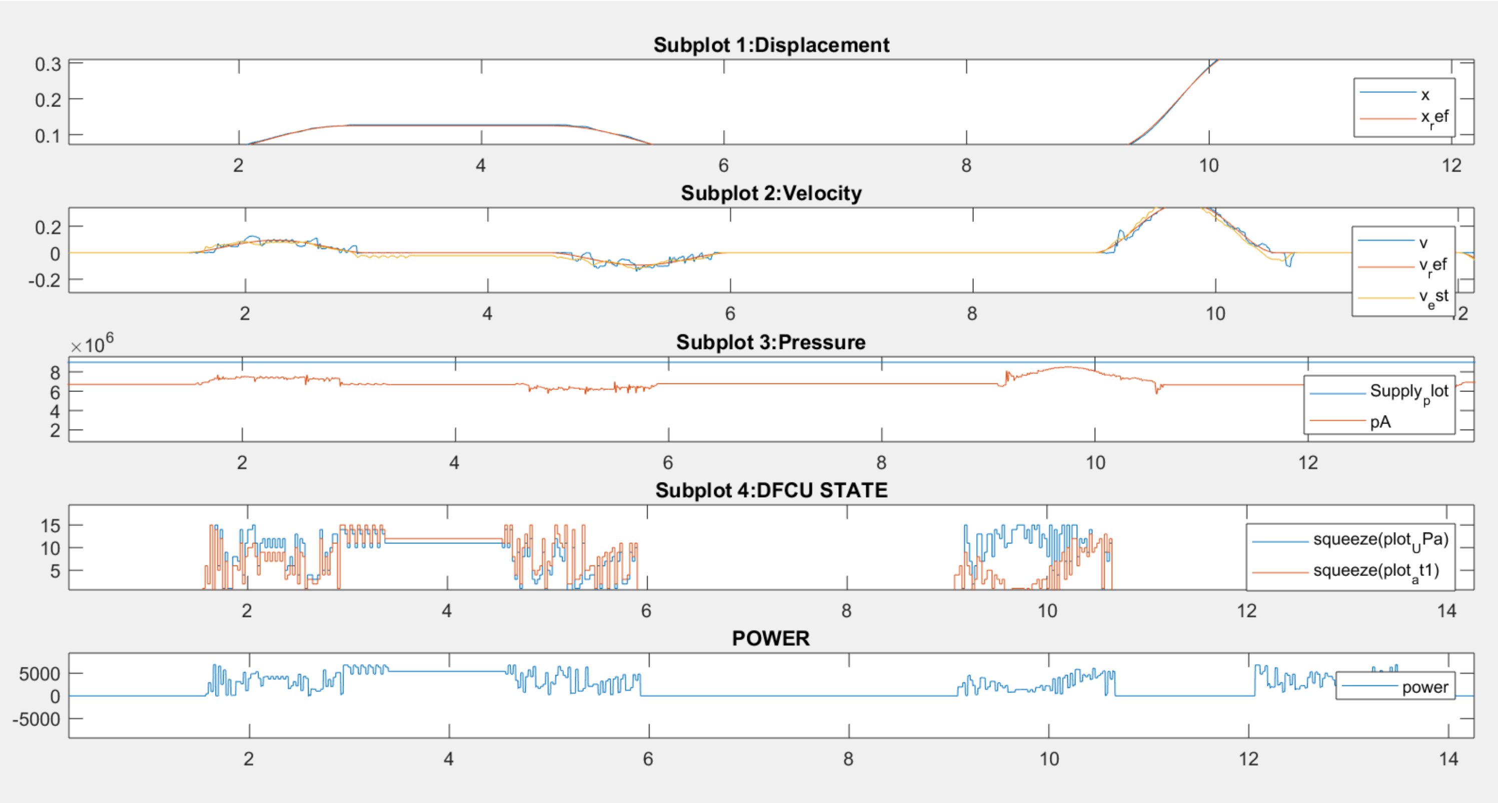
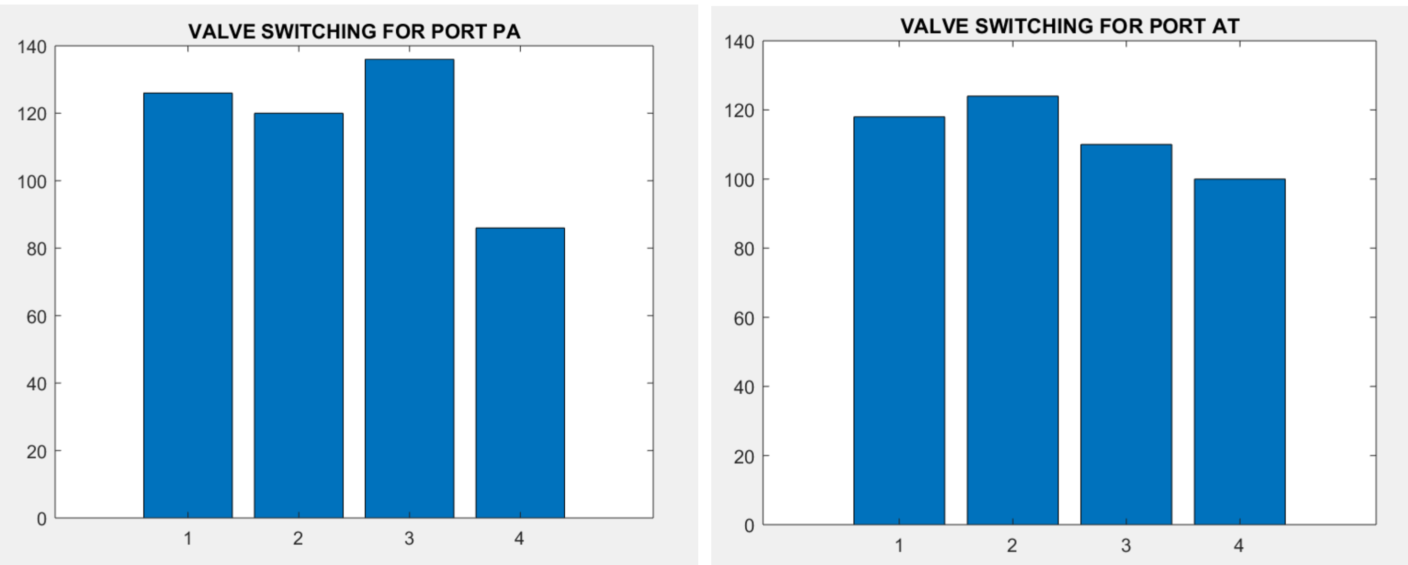


Figure 12:Cost function gain=5e-6



Valve Switching's at Cost function gain=5e-6

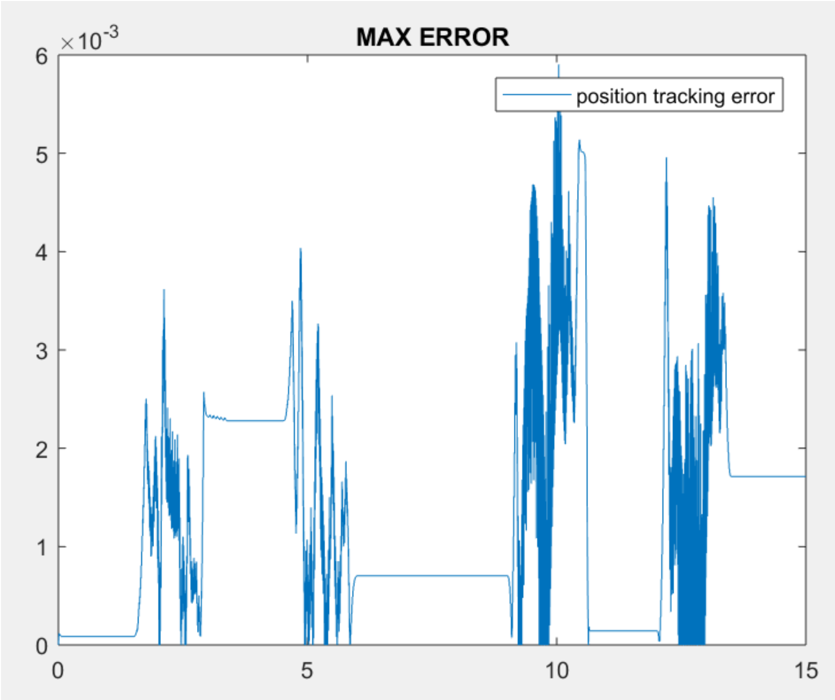


Figure 13:Max_error at gain=5e-6

From Figures 8, 10 and 12 considering Cost function gains 1, 5e-2 and 5e-6 respectively we notice a significant change in “VELOCITY”, “DFCU STATE” and “POWER” plots.

When a high cost function gain is used which was 1, the valve switching was considerably less as it is seen clearly from “DFCU STATE” plot causing considerable use of power.

The valve switching can be more clearly justified when the valve switching of port PA and AT are shown using bar graph. (from figures above)

Whereas when cost function gain 5e-2 is used there was increase in valve switching which resulted in increase in POWER.

But when too less cost function gains are used which almost tends towards zero we noticed that the Velocity reference and estimated velocity does not follow each other any longer as highlighted in the plot which may cause instability to the system. Too many valve switching occur, consuming unnecessary power from the system, Which can be clearly illustrated from the POWER plot and valve switching bar graphs of port AT and PA.

After analyzing the conditions and testing with different cost function gains , it is observed that *switch_gain* and *cross_gain* valves can be taken as 5e-2 as the valve switching are optimum around 60-80 along with the POWER consumption.

3.2 Resolution of binary coded DVS.

When 5-bit DFCU is used:

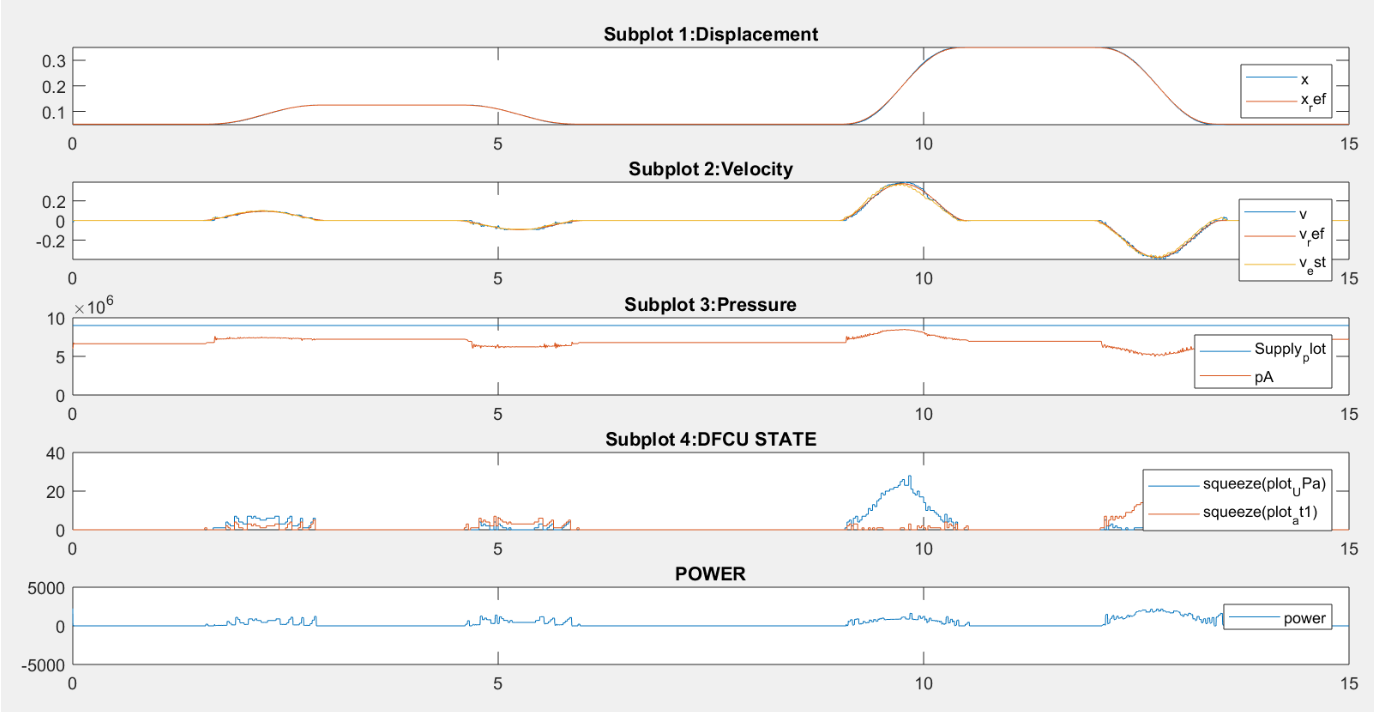


Figure 14:5-Bit DFCU

When 4-Bit DFCU is used:

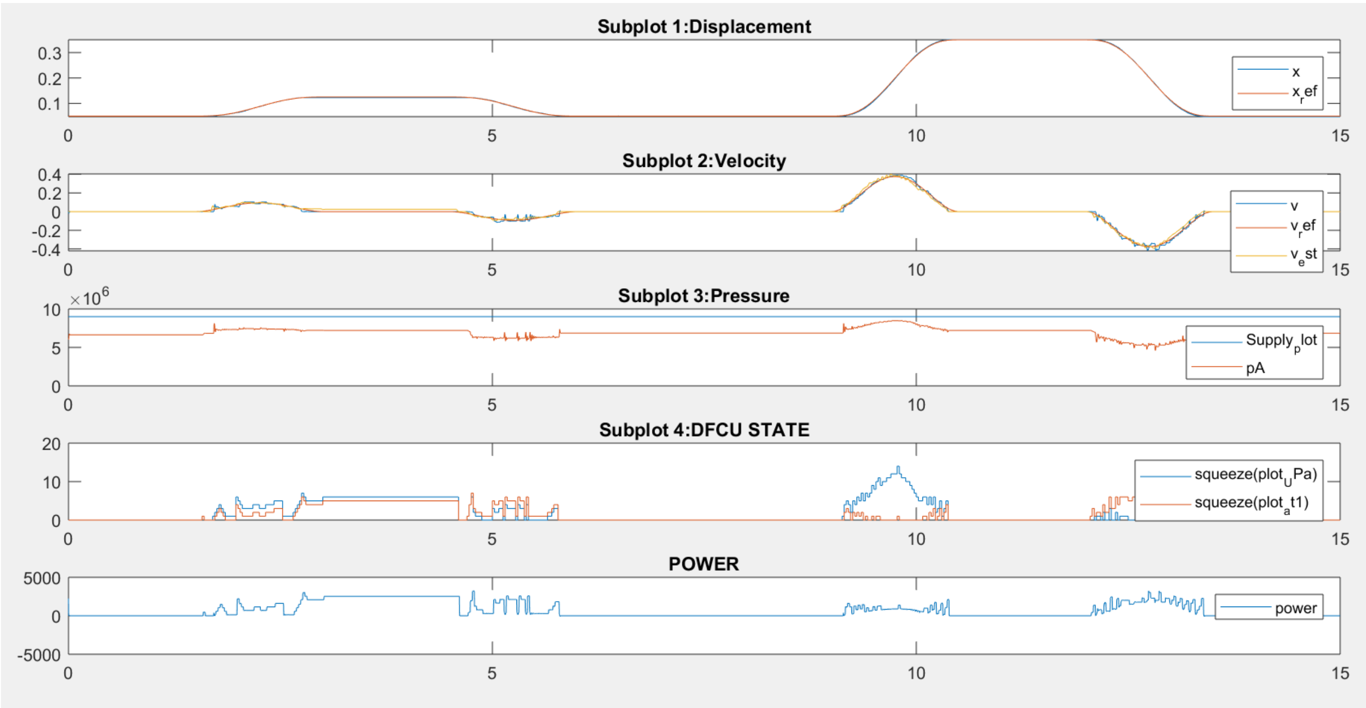


Figure 15:4-Bit DFCU

<u>When 3-bit DFCU is used</u>

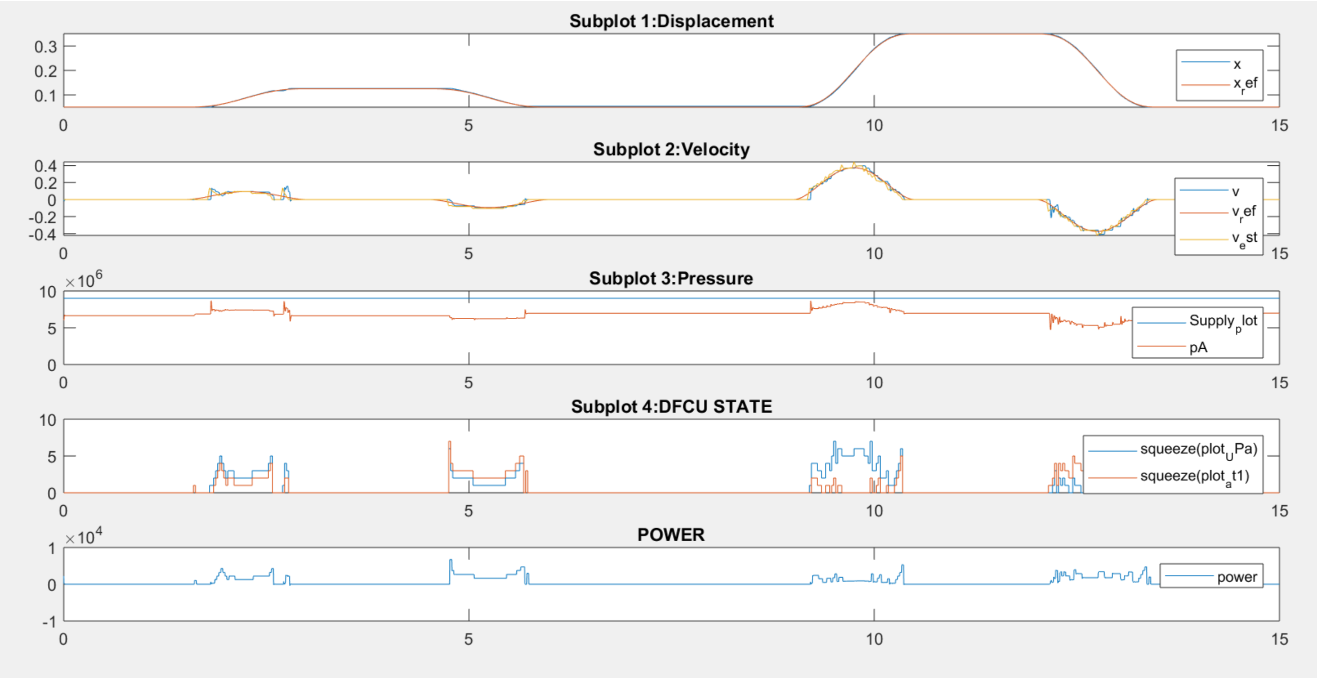


Figure 16:3-Bit DFCU

<u>When 2-bit DFCU is used</u>

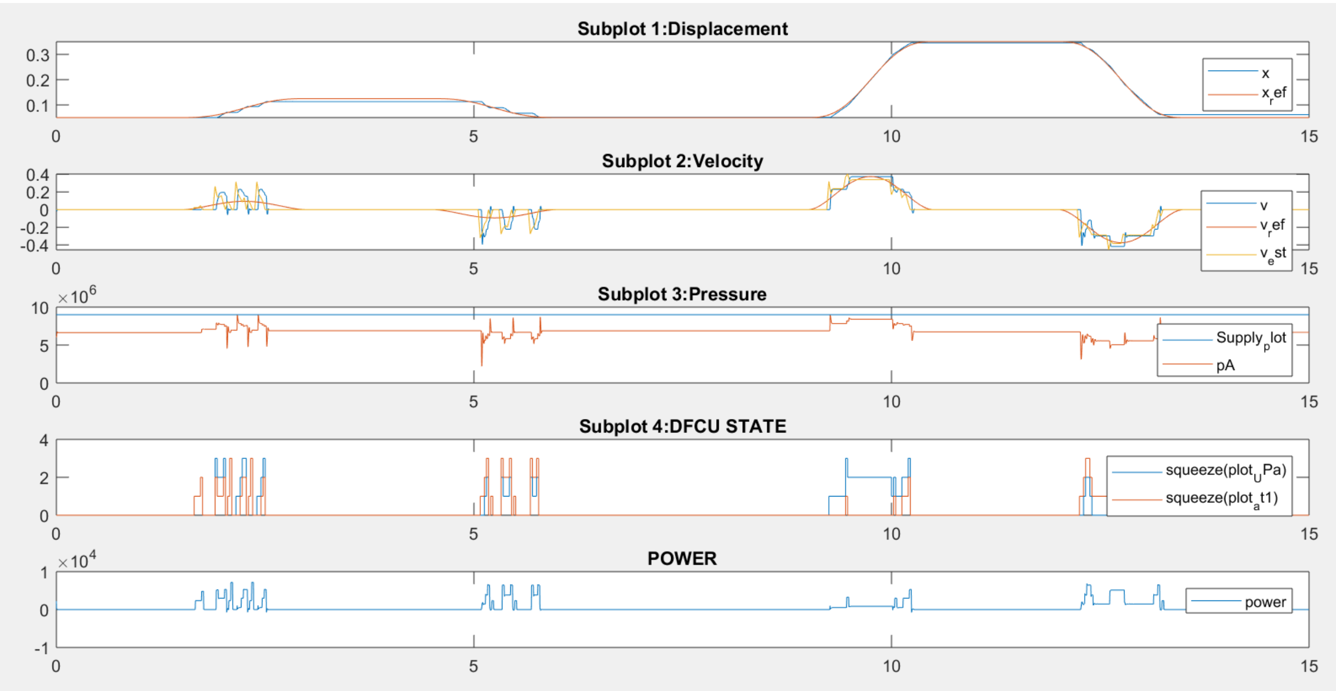


Figure 17:2-Bit DFCU

The main difference we noticed when using 5,4,3,2 bit DFCU's was there was a considerate drop of valve switching between ports AT and PA which we can estimate from the

DFCU STATE plots. In 5-bit DFCU has maximum number of valve switching and 2-bit DFCU has the least. The less valve switching is because of the control matrix, since the size of the control matrix is directly proportional to the bit's used and seen in the equation below:

```
ctrl_mtrx = fliplr(double((dec2bin(0:2^N-1))=='1'));
ctrl_mtrx2=combvec(ctrl_mtrx',ctrl_mtrx')'; %for sum flow controller using binary coding
```

So there will be more number of switching options when higher bits of DFCU's are used.

The less valve switching results to less power consumption in lower bit DFCU's which can be seen in POWER plot.

<u>Comparing 5-bit DFCU without considering Cross-flow with 3 & -bit DFCU considering Cross-flow.</u>

We can have a similar velocity tracking using 5-bit DFCU as compared to 3 &4 bit DFCU's. In 5-bit DFCU we did not consider the cross flow as the *cross_gain* is considered as zero. We tested with different *switch_gain* values to have a similar velocity tracking , and noticed higher the gain values better is the result. So. We took the *switch_gain=1* and the simulation of velocity tracking as follows:

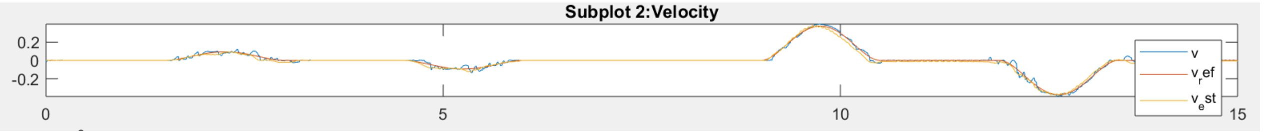


Figure 18:5-bit with cross_gain=0

Cross_gain=0 means that both the DFCU's can now work anonymously causing the velocity of the piston totally uncertain .

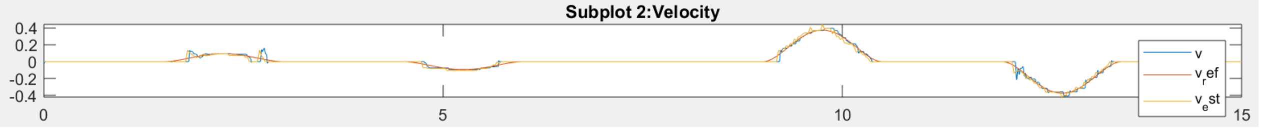


Figure 19:3-Bit DFCU

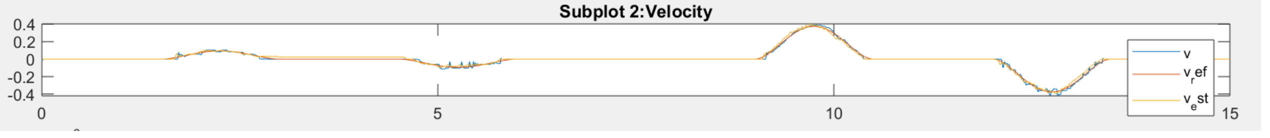


Figure 20:4-Bit DFCU

When it come to power consumption , since in 5-bit DFCU without cross flow there is un restricted valve switching resulting in high power consumption compared to that of 3 & 4 bit DFCU's.

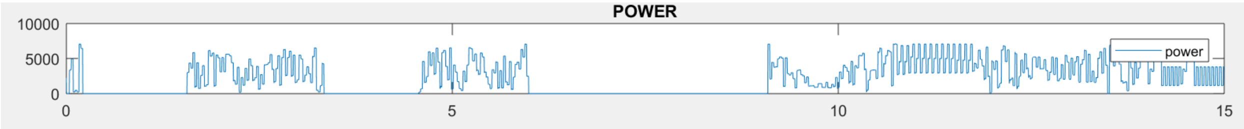


Figure 21:5-bit DFCU POWER CONSUMPTION

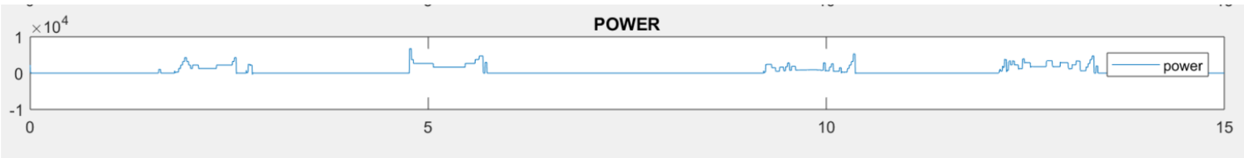


Figure 22:3-bit DFCU POWER CONSUMPTION

3.3 Equal coding vs. binary coding with plain feedforward control

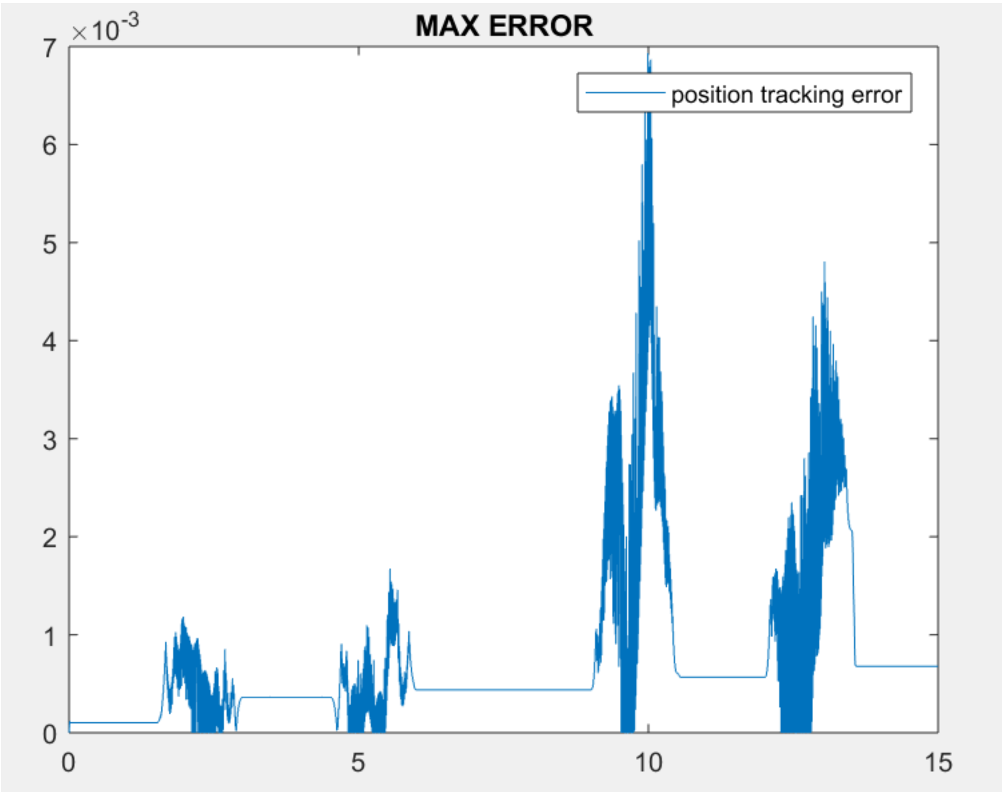


Figure 23-5-bit DFCU(binary)

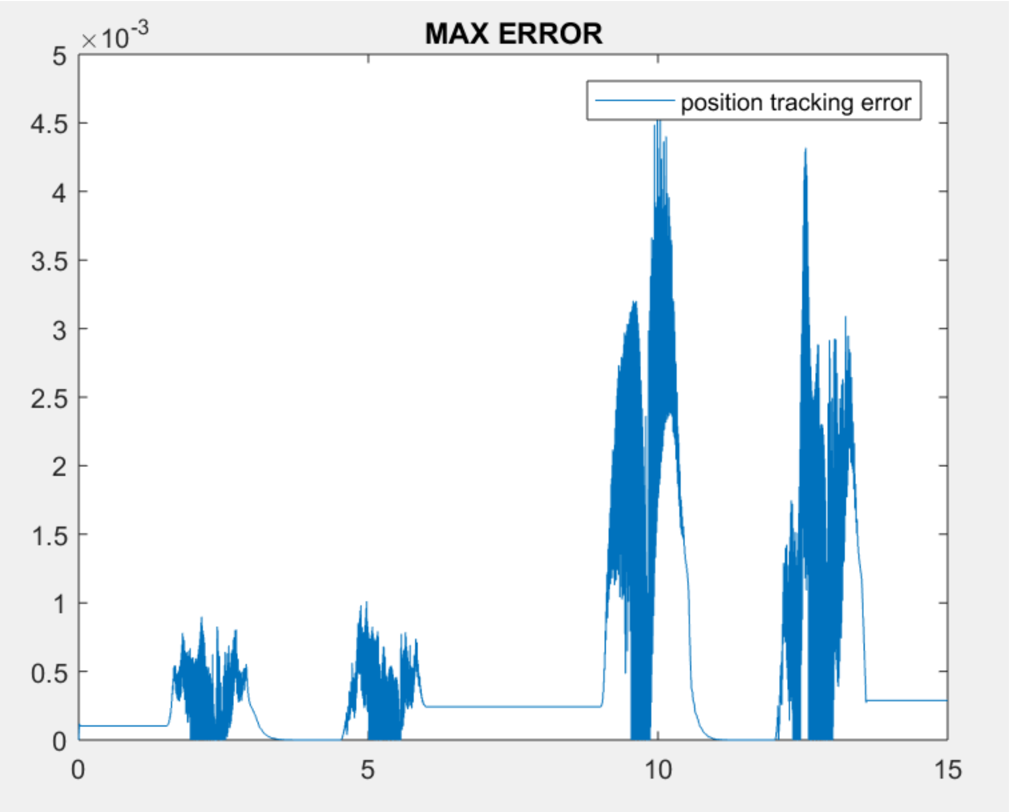


Figure 24:31-bit DFCU(equal)

The maximum position tracking error was almost the same as 31-bit equal coded DFCU for almost 9 second of simulation. But in 5-bit binary coded DFCU the maximum error reaches up to 7*10^3 at around 10 seconds ,whereas in 31-bit equal coded DFCU the error reaches the maximum of around 4.5*10^3 which is much less than the binary coded DFCU. As we know in equal coded systems the valves are nominal that implies position obtained from the cylinder and the reference position will not vary much as compared to binary coded DFCU.

3.4 Pressure peaks

Pressure peaks occurs when there is significant delay in valve switching. To observe the pressure peaks in 4-bit DFCU.

When valve switching delay is:

```
Valve.switch_o_delay = 15e-3*ones(1,N);
```

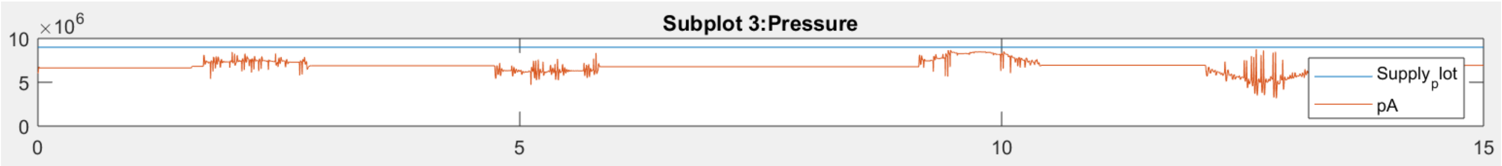


Figure 25:pressure peak for 15ms

When valve switching delay is:

```
Valve.switch_o_delay = 12e-3*ones(1,N);
```

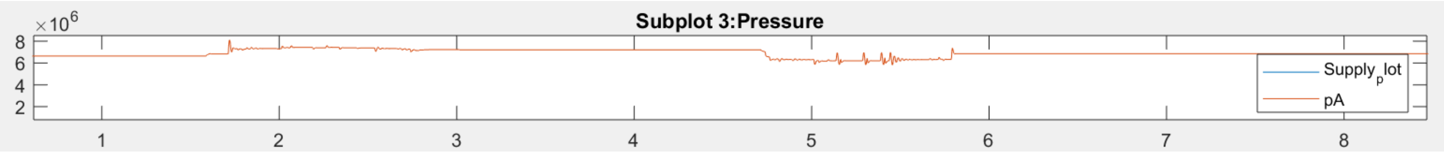


Figure 26:pressure peak for 12ms

From the simulation study of fig 25 and 26 we noticed that with increase in opening delay time of the valves there causes pressure peaks which are not considered good for the system.

4. FEEDBACK

We just wanted to mention that it was one of the most interesting project we did till now in our masters because we could understand each and every part of the systems which we modelled in Simulink. This was all because we got two excellent supervisor's Miika Paloniitty and Mikko Huova they explained the project so well and took out there valuable time for us whenever we needed them. We consider our self truly blessed to have them as our supervisor's.

Digital Hydraulic was totally a new concept that we learned and found it most interesting, the function's like selector , minimum blocks we used for the first time and designing the controller was much a learning process for us. We are definitely looking forward to take the advanced course in digital hydraulics if implemented in future.

Basically we did not miss a single exercise session of this course last year and attended few this year as it was interesting a much learning. So most of the project work was done in the exercise sessions only apart from that we spend around 1 or 2 days in a week for tuning the parameters modifying the system and writing the report.