Design of Forward Converter:

$V_o = 12V$ $V_s = 100V$ $f_s = 100\ kHz$

$P_{o\,max} = 100$ $V_r = 2\%$

$P_{o\,min} = 50$ $PM = 45^\circ$

$M_{VDC} = \dfrac{12}{100}$ 0.12

$I_{o\,min} = \dfrac{P_{o\,min}}{12} = 4.166\ A$

$I_{o\,max} = \dfrac{P_{o\,max}}{V_O} = 8.33$

$R_{L\,max} = \dfrac{12}{4.16} = 2.88\ \Omega$

$R_{L\,min} = \dfrac{V_o}{I_{o\,max}} = \dfrac{12}{8} = 1.5$

$D = 0.5$

$L_{min} = \dfrac{R_L(1-D)}{2f_s} = \dfrac{2.88 \times (0.5)}{2 \times 10^5} = 7.2\ \mu H$

let $L = 50\ \mu H$

$\Delta i_{L\,max} = \dfrac{V_o(1-D)}{f_s L} = 1.2\ A$

$V_r = 0.02 \times 12 = 0.24V$

$r_{c\,max} = \dfrac{0.24}{0.12 \times 10} = 0.2\ \Omega$

Take $r_c = 50\,m\Omega$

$$C_{min} = \frac{0.6}{2 f_s r_c} = \frac{0.5}{2 \times 10^5 \times 0.05} = 50\,uF$$

let $C = 100\,uF$

Design Procedure:

1. The calculation of parameters was done according to these images.
2. The value of Fs = 100 KHz, Vin = 100V, Vo =12V , L = 50 uH, C = 100 uF, R = 2.8 ohms, r = 50 m ohms, D = 0.5, Phase Margin = 45 degree.
3. The construction of Forward converter is given in the **power_ForwardConverter.slx** file.
4. For the construction of the converter we are using a linear transformer with V1 = 100V and V2 = 25V and V3 = 100V.
5. The pulse generator is connected to the gate of the MOSFET with inputs set as Fs = 100 KHz and D = 0.5.
6. The Circuit diagram is as shown in Fig 1.1.
7. The transfer functions of the Forward converter was designed using these values and is calculated in the MATLAB File – **Bodeplotfwd.m** . The MATlAB code is given in the Appendix. The crossover frequency is chosen as 3 KHz. The resonant frequency was around 2.3 KHz.
8. The transfer function (gc) is shown in Fig 1.2. The formulas used were as given in Mohan's example.
9. Based on the Transfer function of the controller (gc) obtained in the MATLAB file, the controller for the forward converter was designed, shown in **power_ForwardConverter_edit.slx** .
10. The circuit diagram is a shown in Fig 1.3.
11. As we can see, the output Vo of the converter is connected to a subtractor along with the Reference Block. This output is the then connected to the transfer function.
12. The output of the Transfer function is connected to a summer along with the constant value of D=0.5.
13. After saturating the output, this signal is sent to the "greater than equal to" block along with the input of the repeating sequence block.
14. This output is then given to the Gate of MOSFET.
15. For the step load change the step time is given as 0.001 for load and 0.006 for load1. Both the loads would conduct till 0.06 and then there would a step change.

16. The Step reference switches “20% step up from nominal value” at 0.008. The value is set as 9.6 and was also tried for initial value of 6.

Schematics:

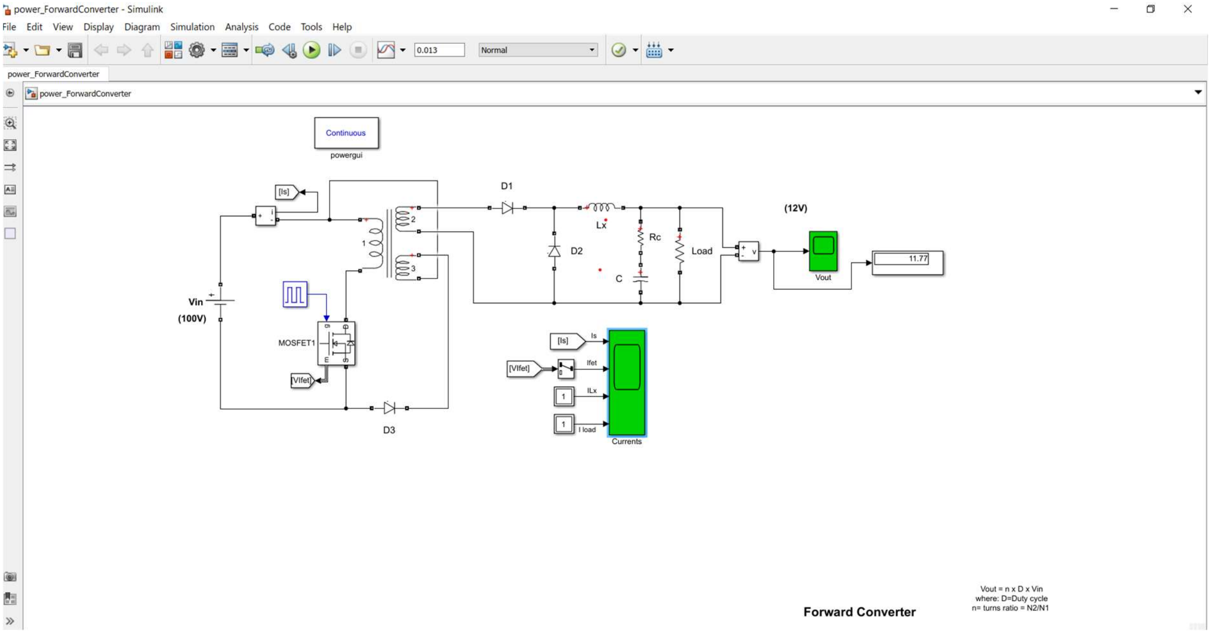

Fig 1.1 - General Design of Forward converter

```
gc =

    2.234e12 s^2 + 3.07e16 s + 1.055e20
  ---------------------------------------
  4.72e07 s^3 + 4.882e12 s^2 + 1.262e17 s
```

Fig 1.2- Transfer function of converter with Type-3 Controller

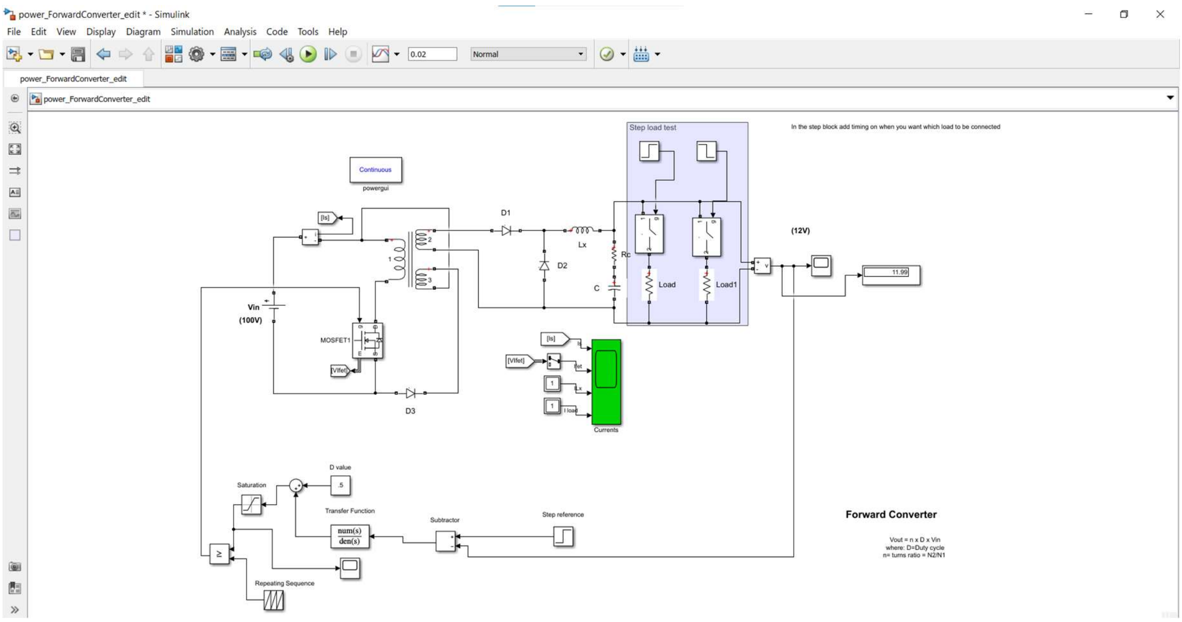

Fig 1.3 – Schematic of Fwd converter with controller

Results and Waveforms:

1. The general Vout and current response of the forward converter was obtained. It is as shown in Fig 2.1 and 2.2.
2. As we can see, We get the desired output voltage of 12 V.
3. The bode plot of the transfer function was obtained and is shown in Fig 2.3. The phiboost is 99.8942 and gain is 0.3336.
4. The Step load response and Step reference voltage response of the forward converter with the controller was obtained and is shown in Fig 2.4 and 2.5.

Fig 2.1- Vout response of converter

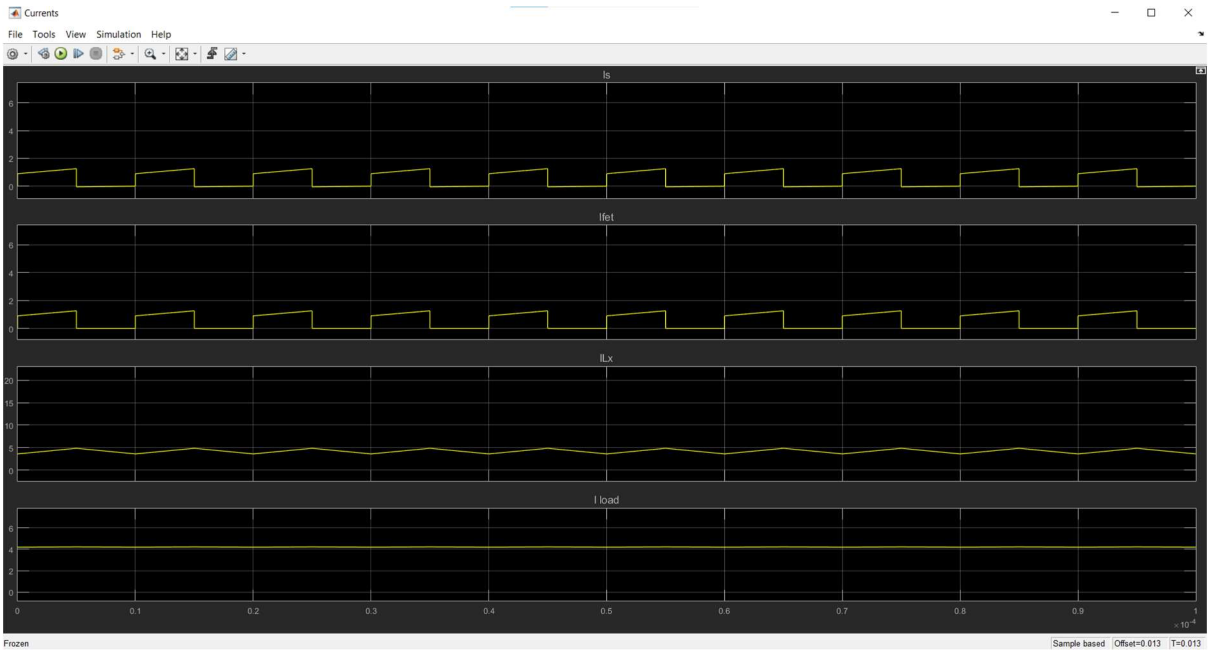

Fig 2.2 – Current waveform

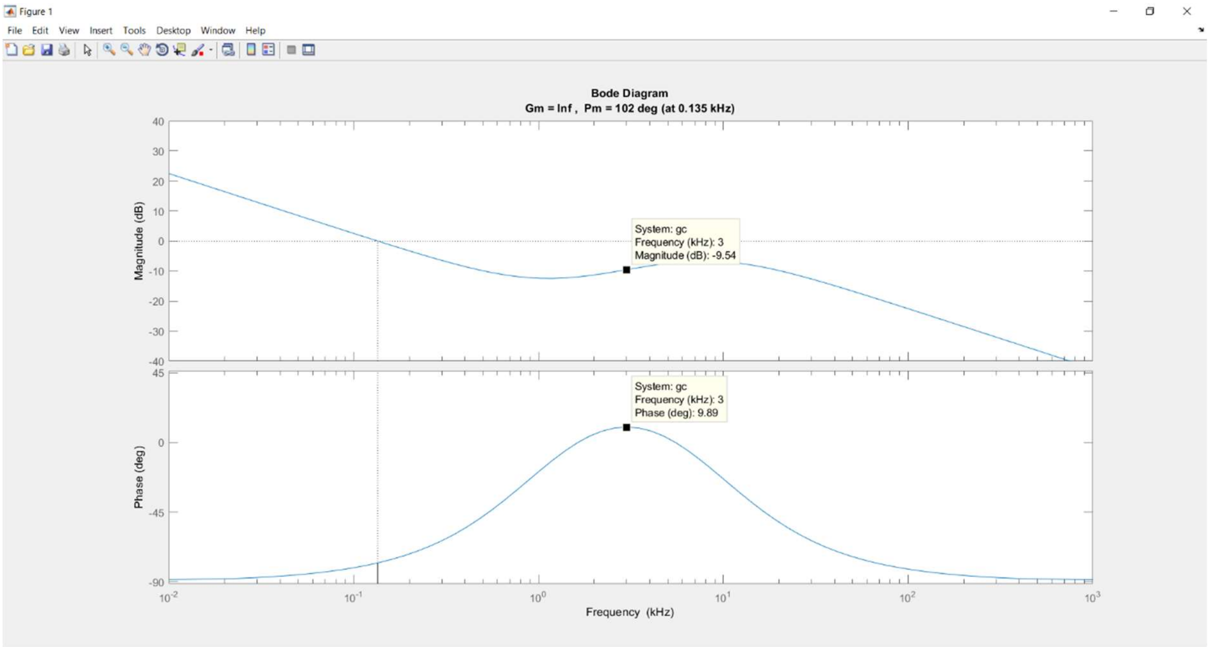

Fig 2.3- Bode plot

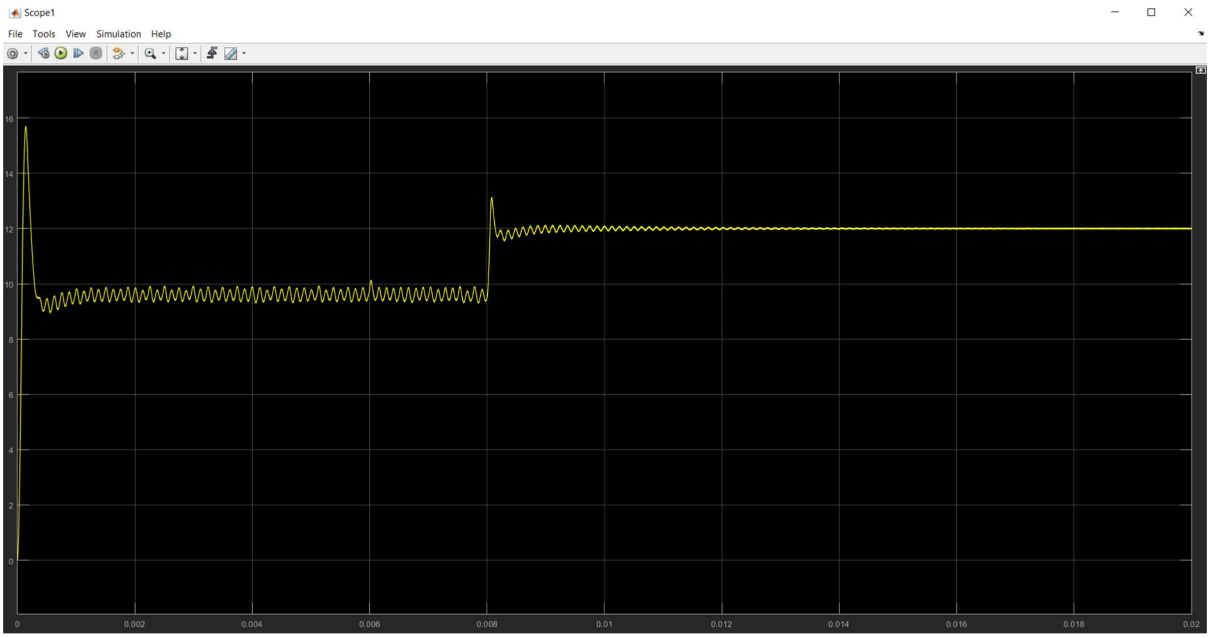

Fig 2.4 – Step Reference 9.6 V to 12 V along with Step Load change

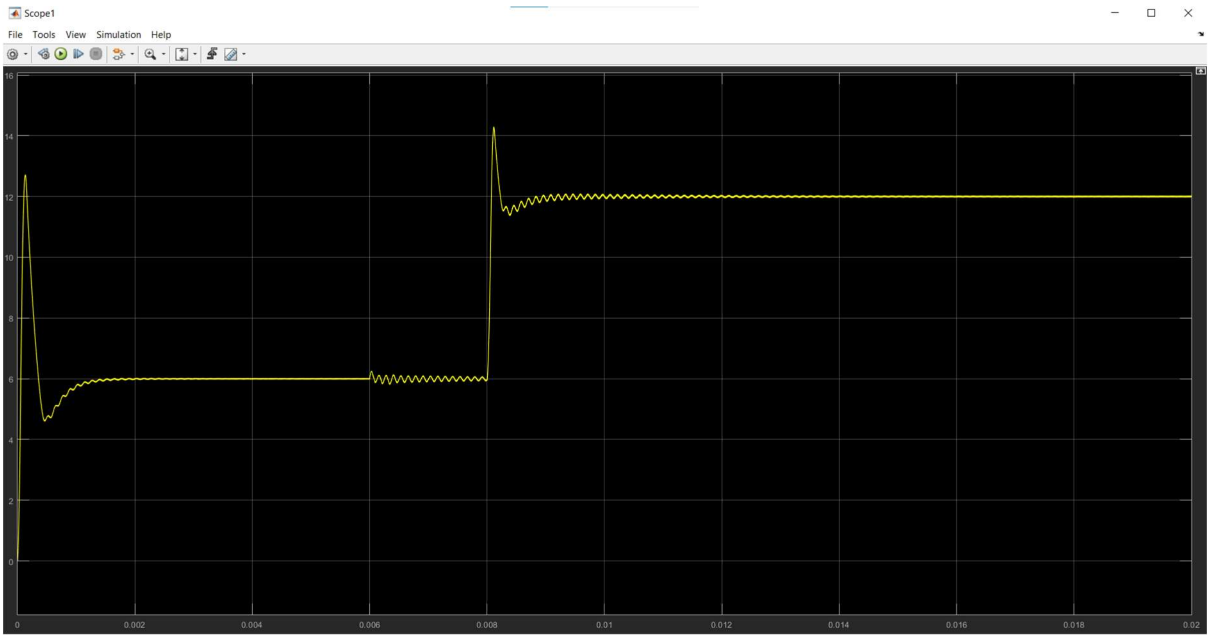

Fig 2.5– Step Reference 6 V to 12 V along with Step Load change

```
Gps =

   2.4e04 s + 4.8e09
  -------------------
  s^2 + 4704 s + 2e08
```

Fig 2.6 – Forward converter transfer function

Appendix:

```
clear
s = tf('s')
opts = bodeoptions('cstprefs');
opts.FreqUnits = 'Hz';
Vin = 100
L = 50e-6
C = 100e-6
r = 50e-3
R = 2.7
n = 0.24     % Ns/Np = .24 D=0.5

Gps = n*Vin/(L*C) * (1+s*C*r) / (s^2 + s*(r/L + 1/(R*C))
+ 1/(L*C) )
% Gps = Vin/(L*C)  / (s^2 + s*( 1/(R*C)) + 1/(L*C) )


fc = 3e3
```

```
pm = 45
kfb = 0.2
Gpwm = 0.556

[gain phase] = bode(Gps, 2*pi*fc)
phiboost = -90+ pm - phase
kboost = tand(45 + phiboost/4)
gaincontroller = 1 / (kfb * Gpwm * gain)
fz = fc/kboost
fp = fc*kboost
kc = gaincontroller * 2*pi*fz/kboost
wz = 2*pi*fz
wp = 2*pi*fp
% controlSystemDesigner('bode',Gps);
% margin(Gps)
gc = (kc/s)*(((1+(s/wz))^2)/((1+(s/wp))^2))
% controlSystemDesigner('bode',gc)
margin(gc)

R1 = 2*(10^3)
C2 = wz/(kc*wp*R1)
C1 = C2*((wp/wz)-1)
R2 = 1/(wz*C1)
R3 = R1/((wp/wz)-1)
C3 = 1/(wp*R3)
```