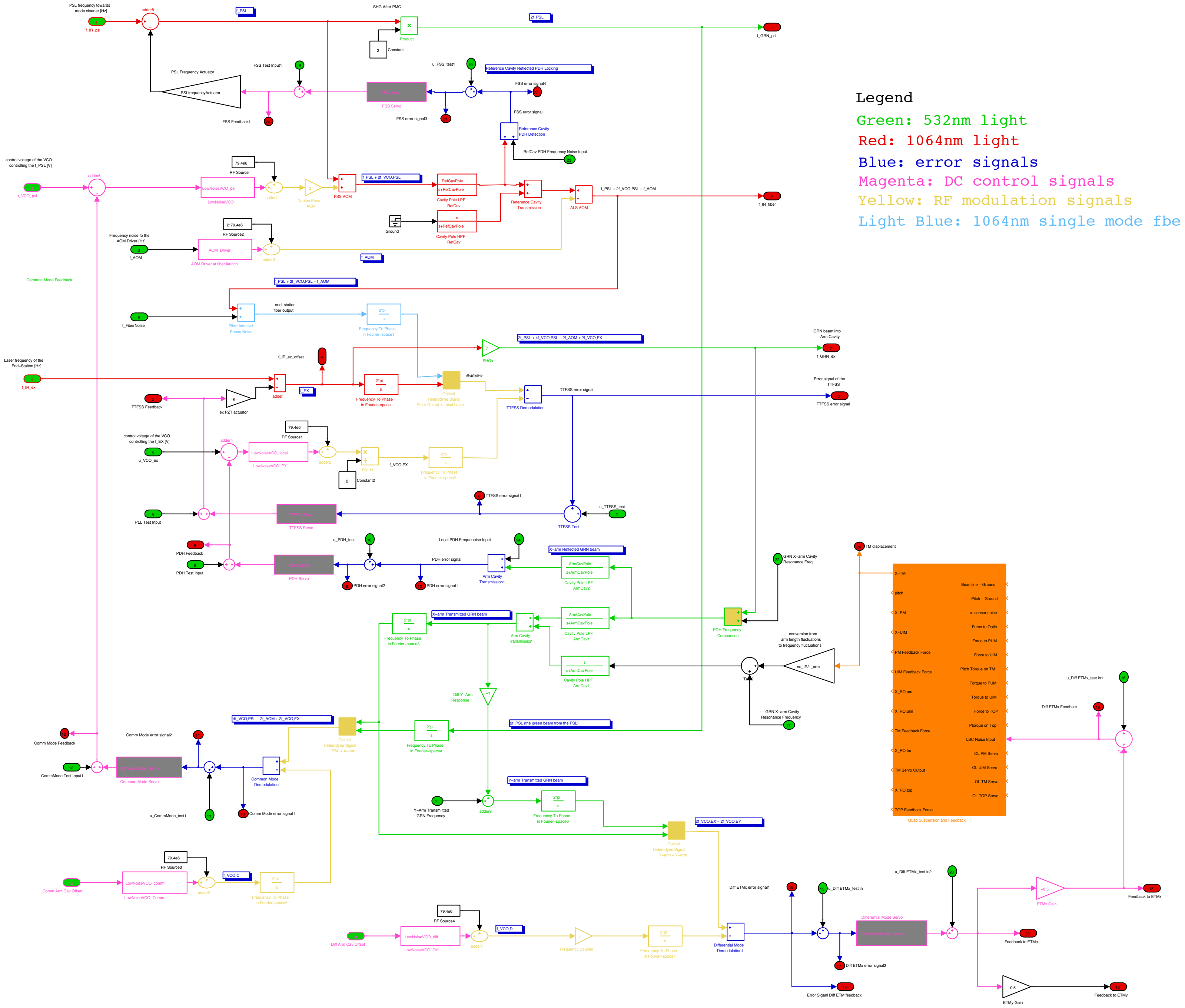
ALS Locking Strategy - 3



Legend

Green: 532nm light

Red: 1064nm light

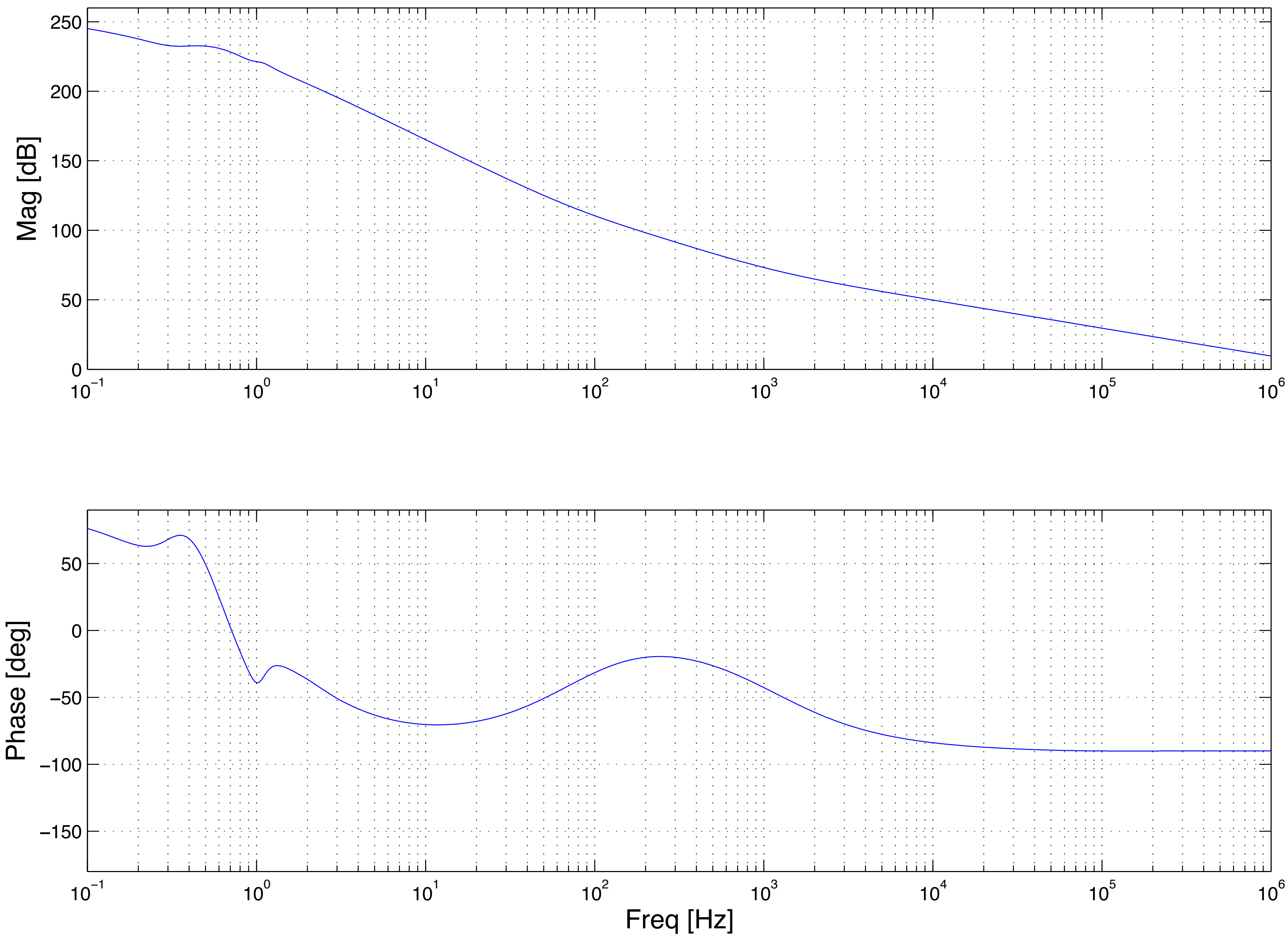
Blue: error signals

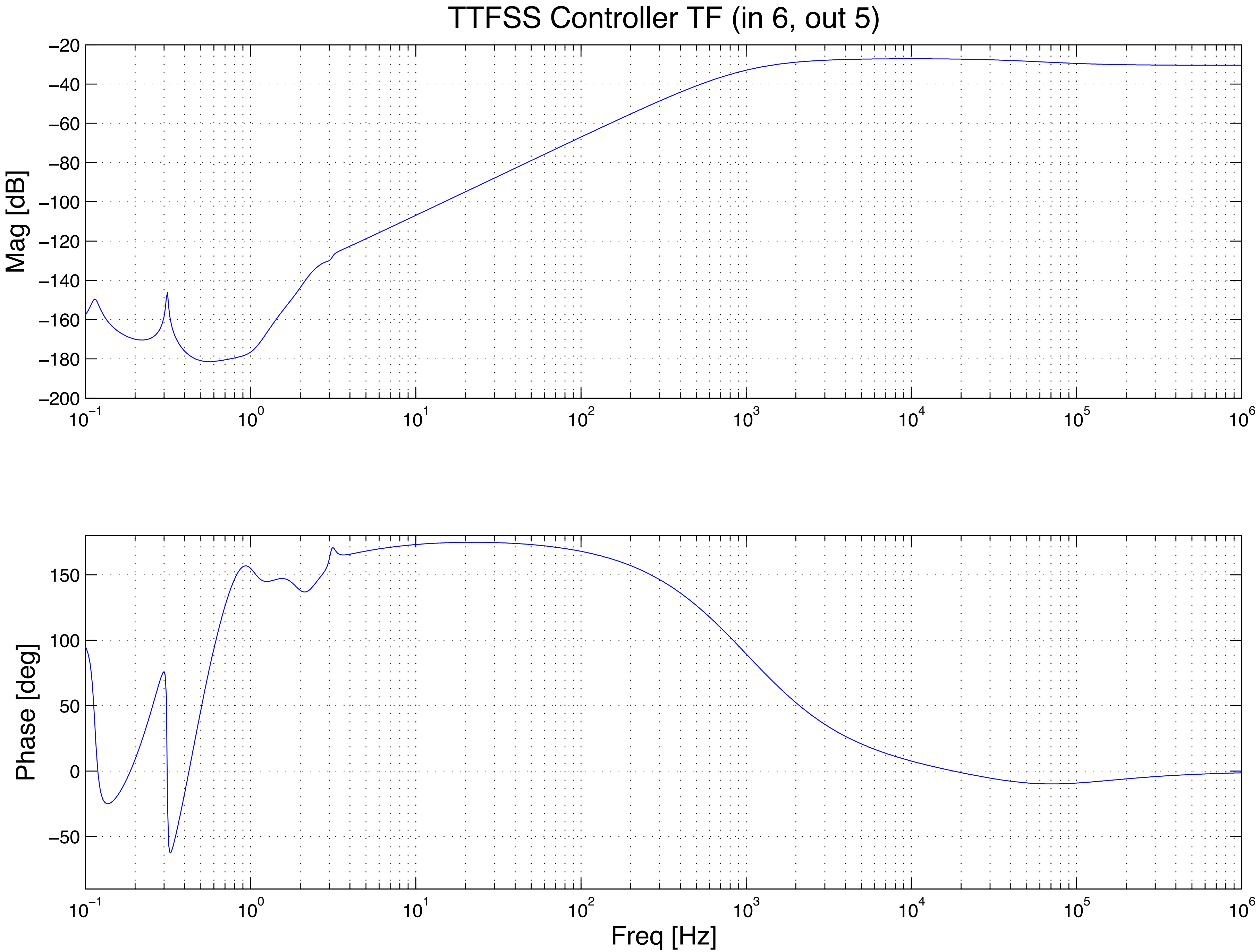
Magenta: DC control signals

Yellow: RF modulation signals

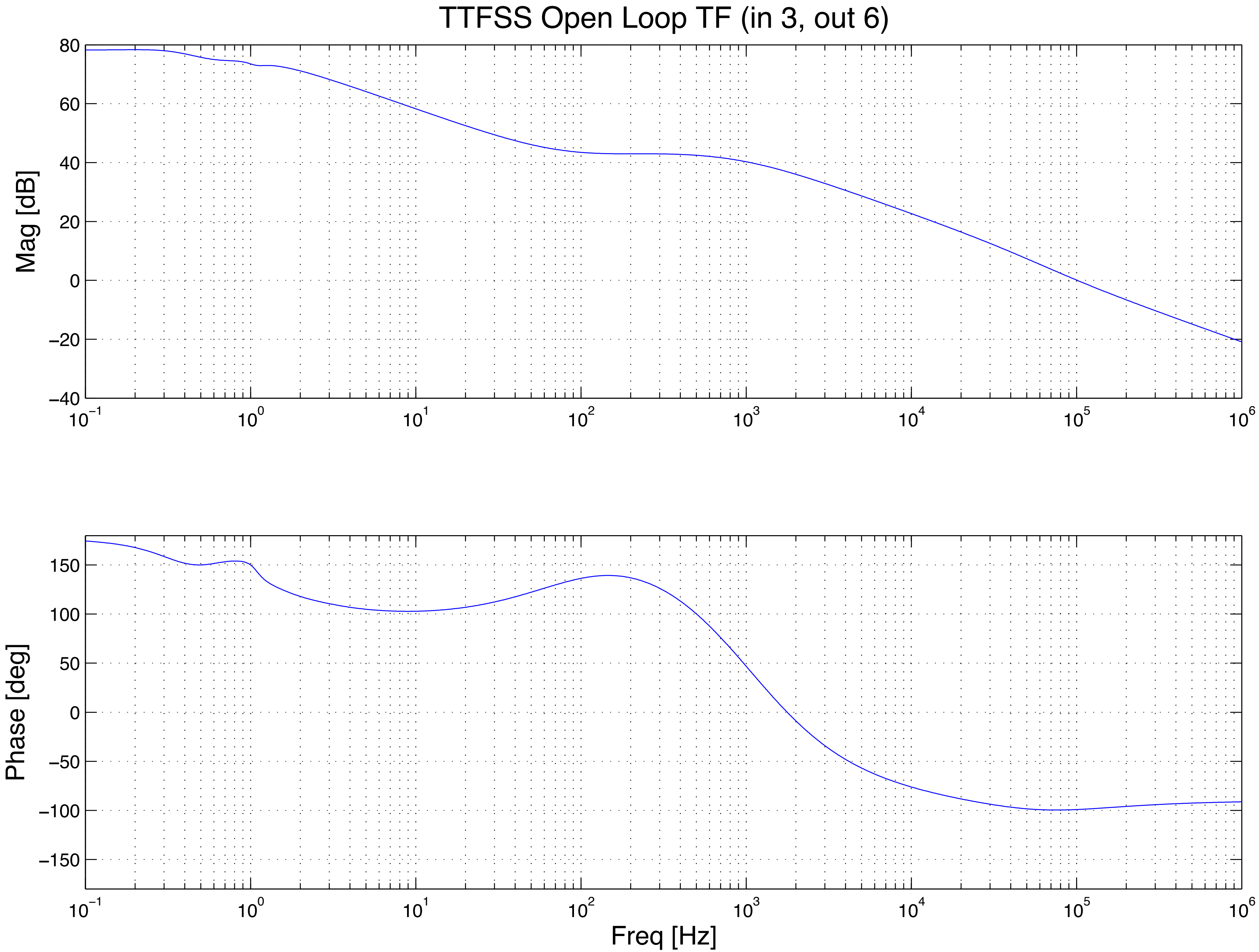
Light Blue: 1064nm single mode fbe

Local laser -to- TTFSS error signal TF (in 5, out 3)

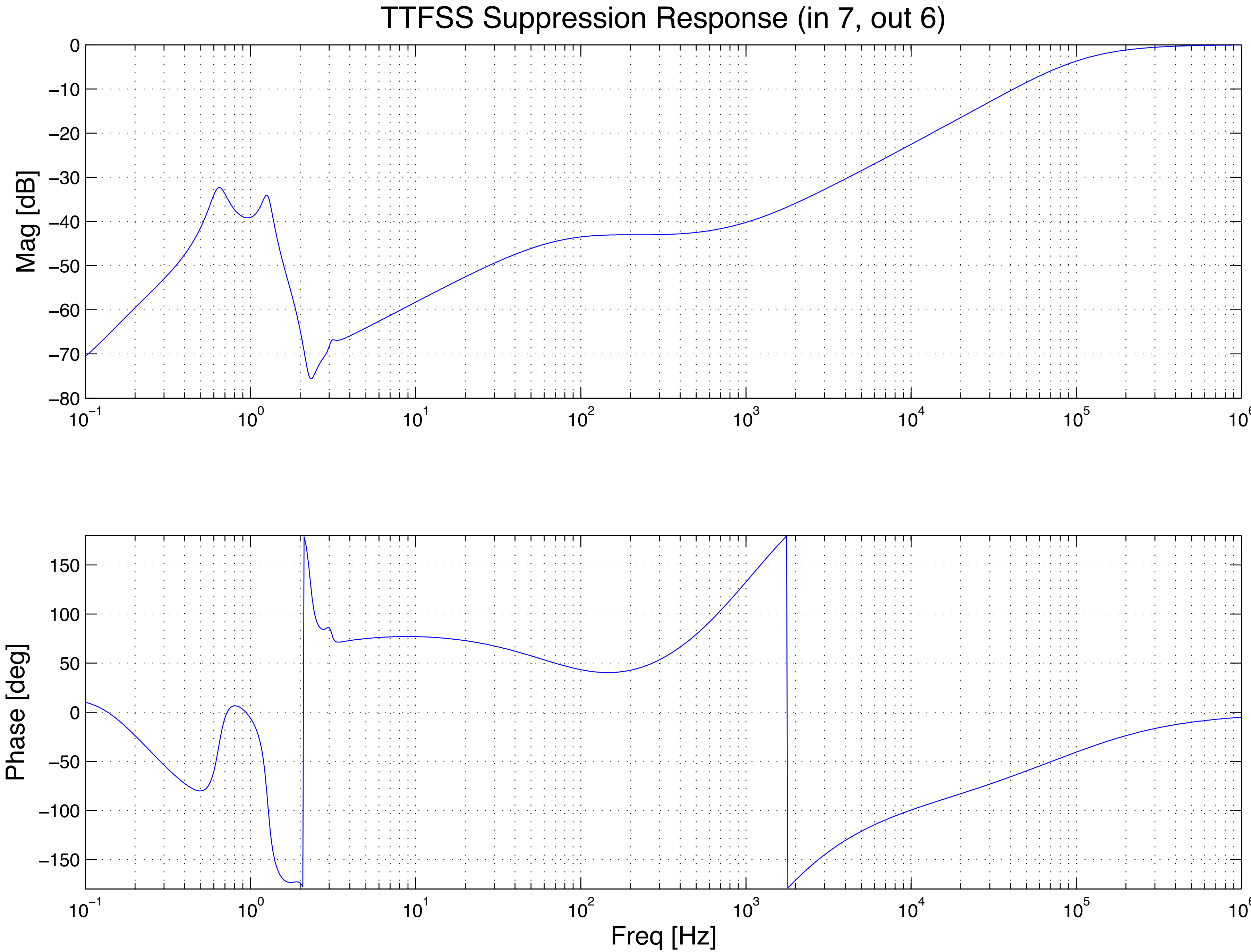




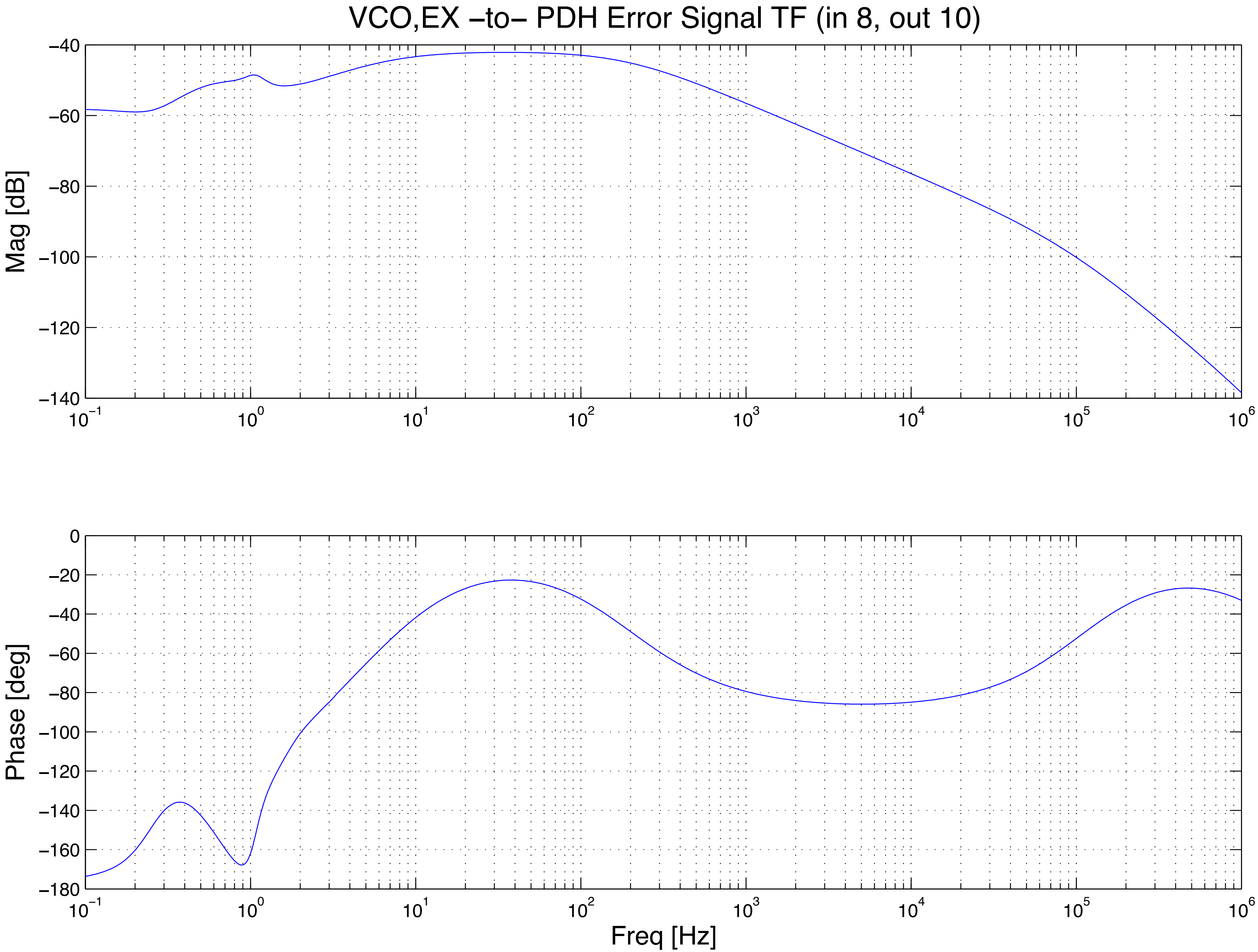
TTFSS Controller TF (in 6, out 5)
Mag [dB]
−20
−40
−60
−80
−100
−120
−140
−160
−180
−200
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
Phase [deg]
150
100
50
0
−50
Freq [Hz]



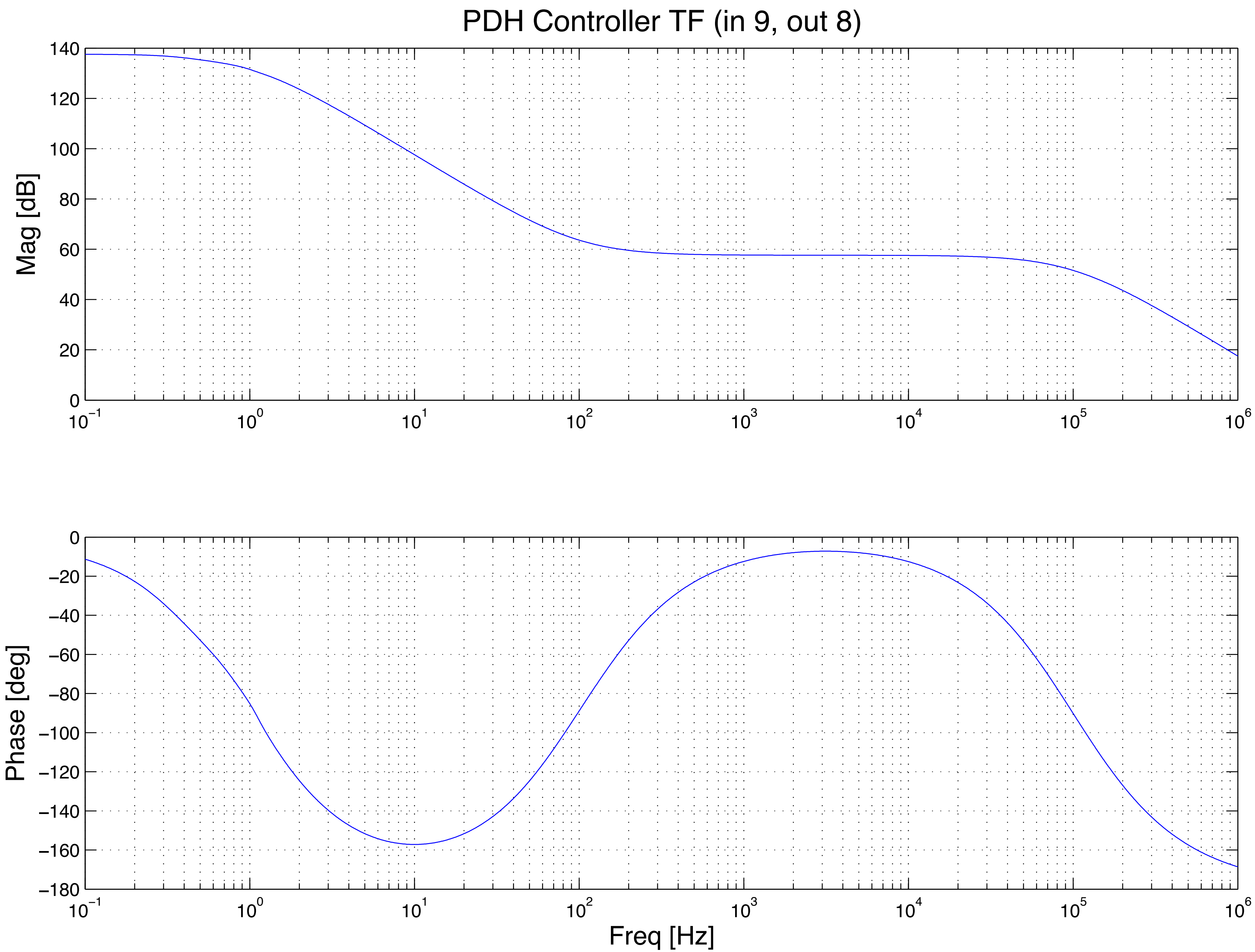
TTFSS Open Loop TF (in 3, out 6)
Mag [dB]
80
60
40
20
0
−20
−40
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
Phase [deg]
150
100
50
0
−50
−100
−150
Freq [Hz]



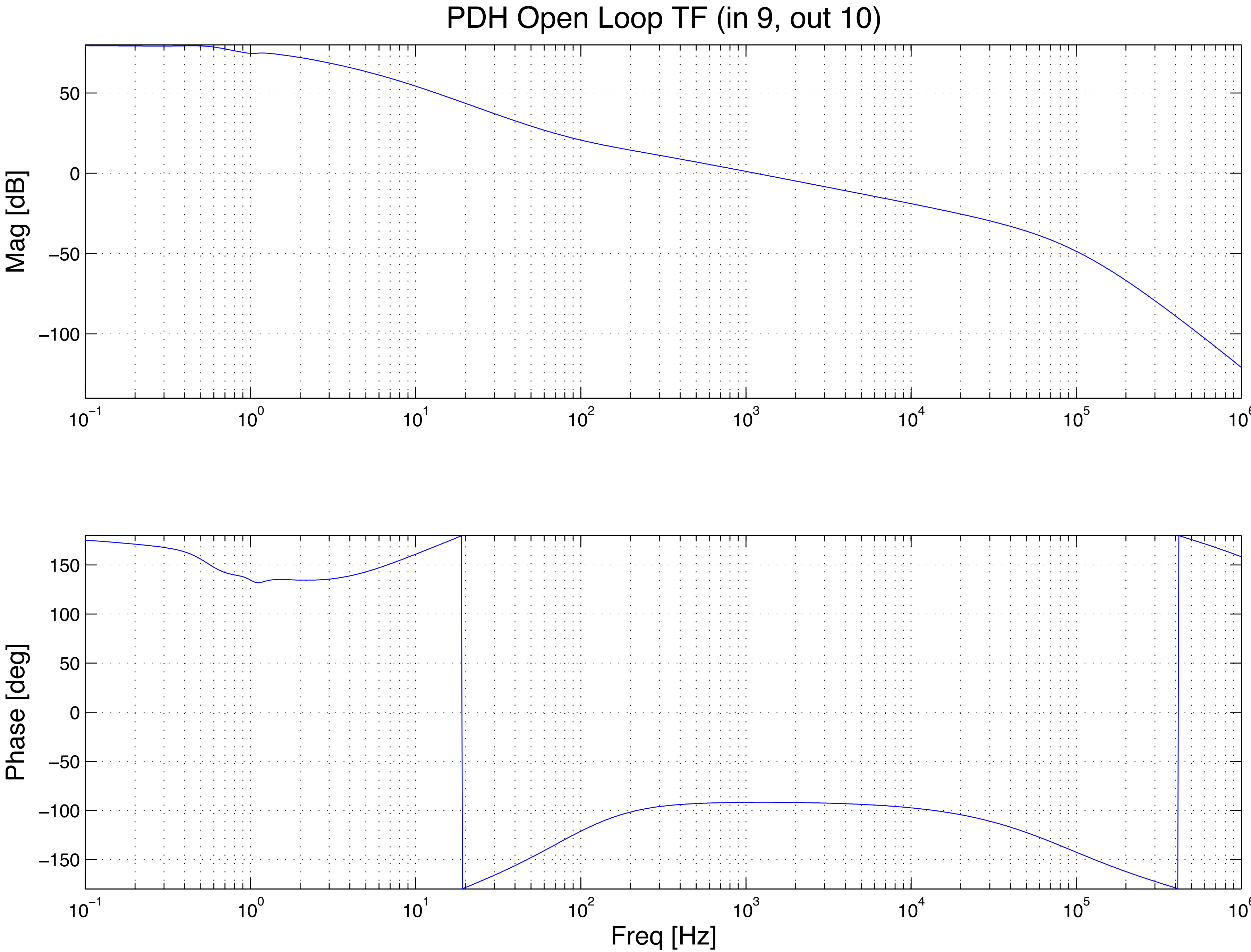
TTFSS Suppression Response (in 7, out 6)
Mag [dB]
0
−10
−20
−30
−40
−50
−60
−70
−80
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
Phase [deg]
150
100
50
0
−50
−100
−150
Freq [Hz]



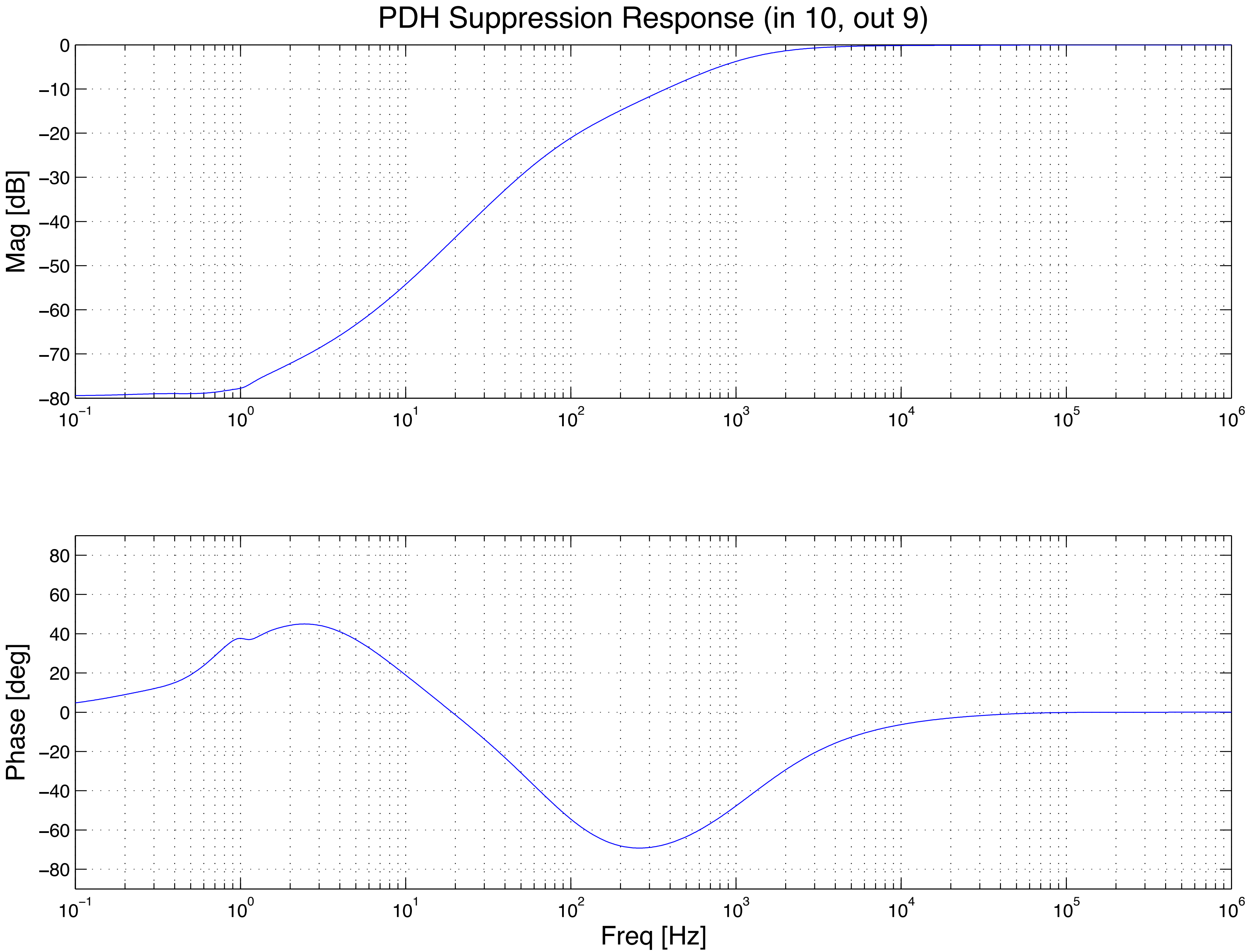
VCO,EX –to– PDH Error Signal TF (in 8, out 10)
Mag [dB]
−40
−60
−80
−100
−120
−140
Phase [deg]
0
−20
−40
−60
−80
−100
−120
−140
−160
−180
10^{-1} 10^{0} 10^{1} 10^{2} 10^{3} 10^{4} 10^{5} 10^{6}
Freq [Hz]

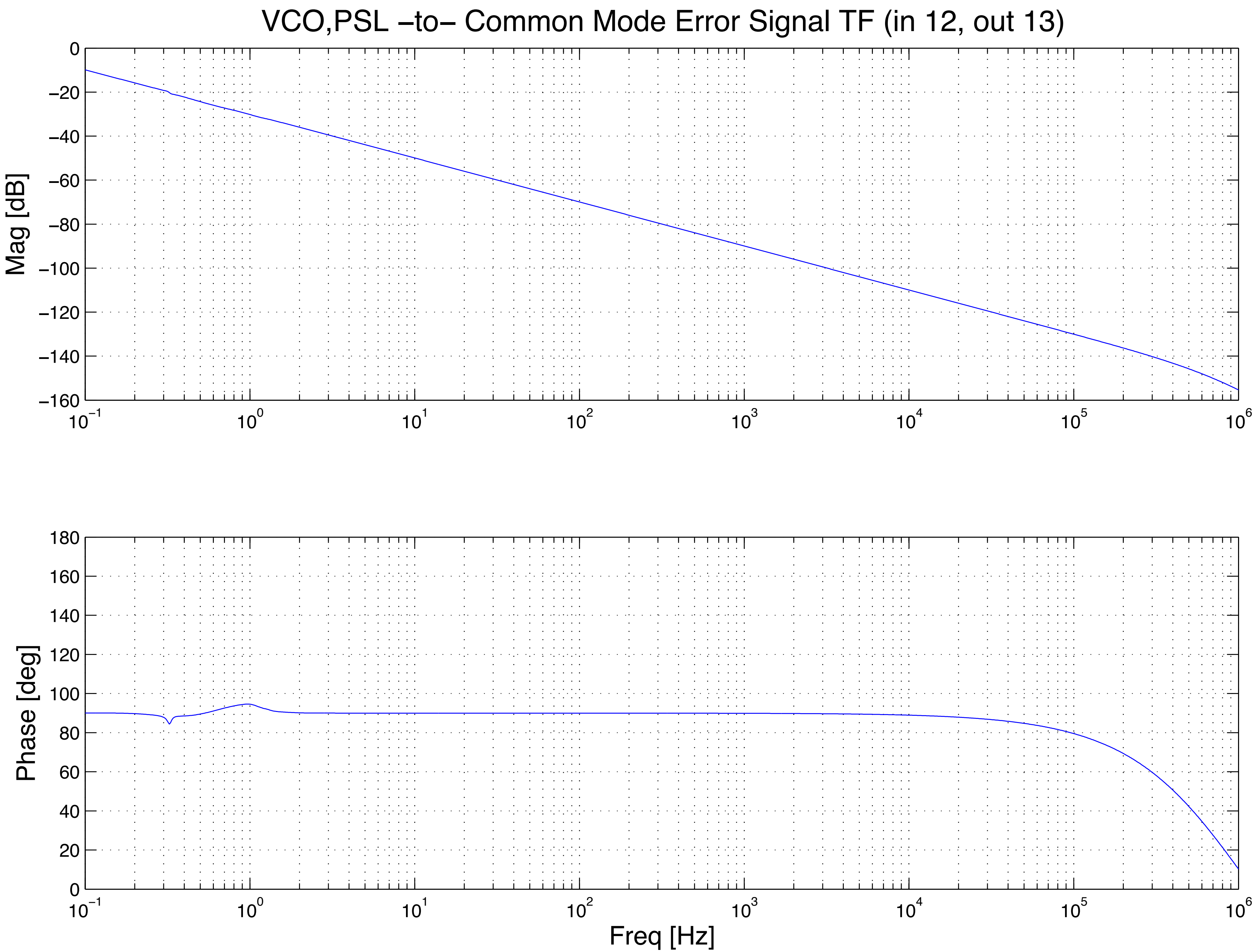


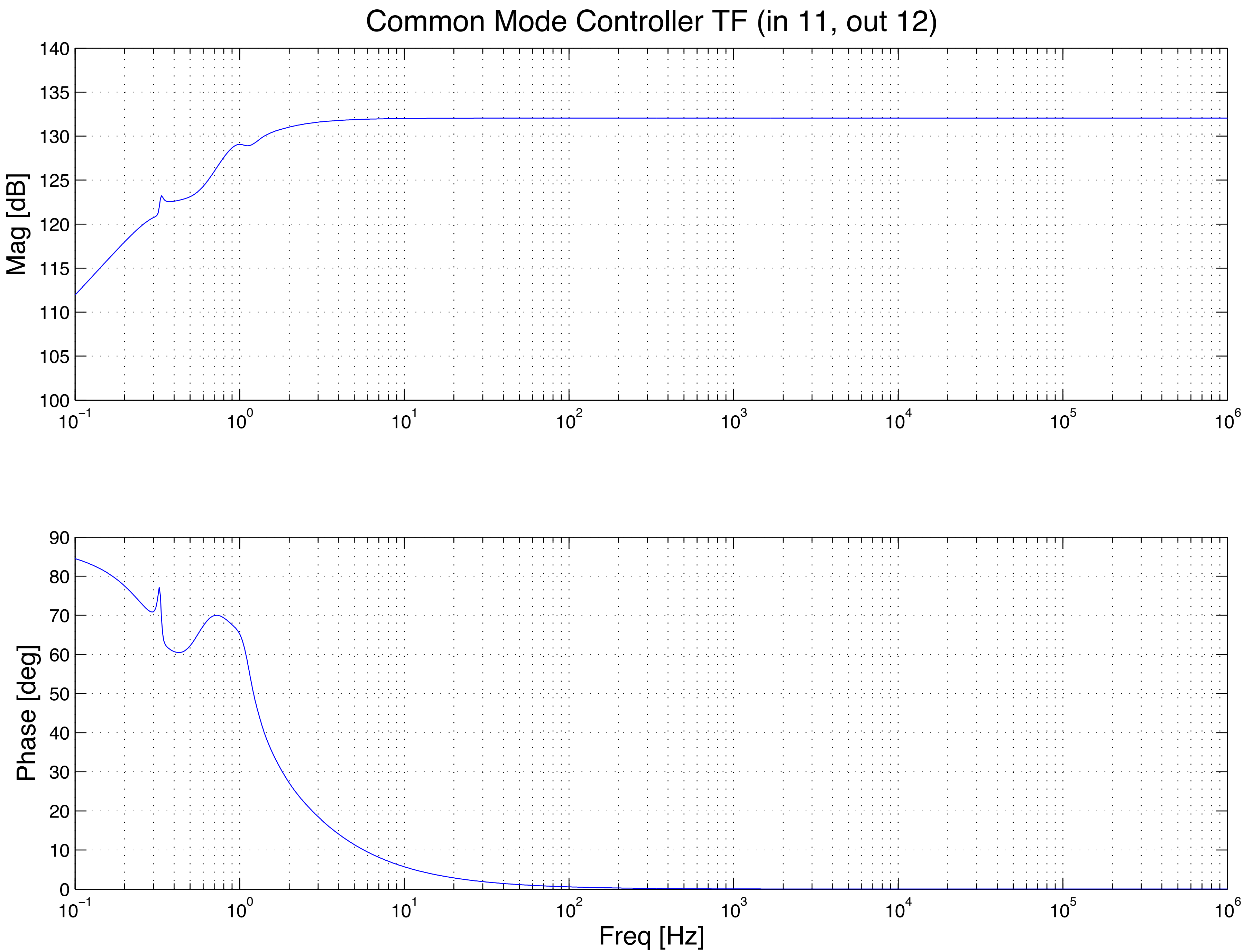
PDH Controller TF (in 9, out 8)
Mag [dB]
140
120
100
80
60
40
20
0
Phase [deg]
0
−20
−40
−60
−80
−100
−120
−140
−160
−180
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
Freq [Hz]



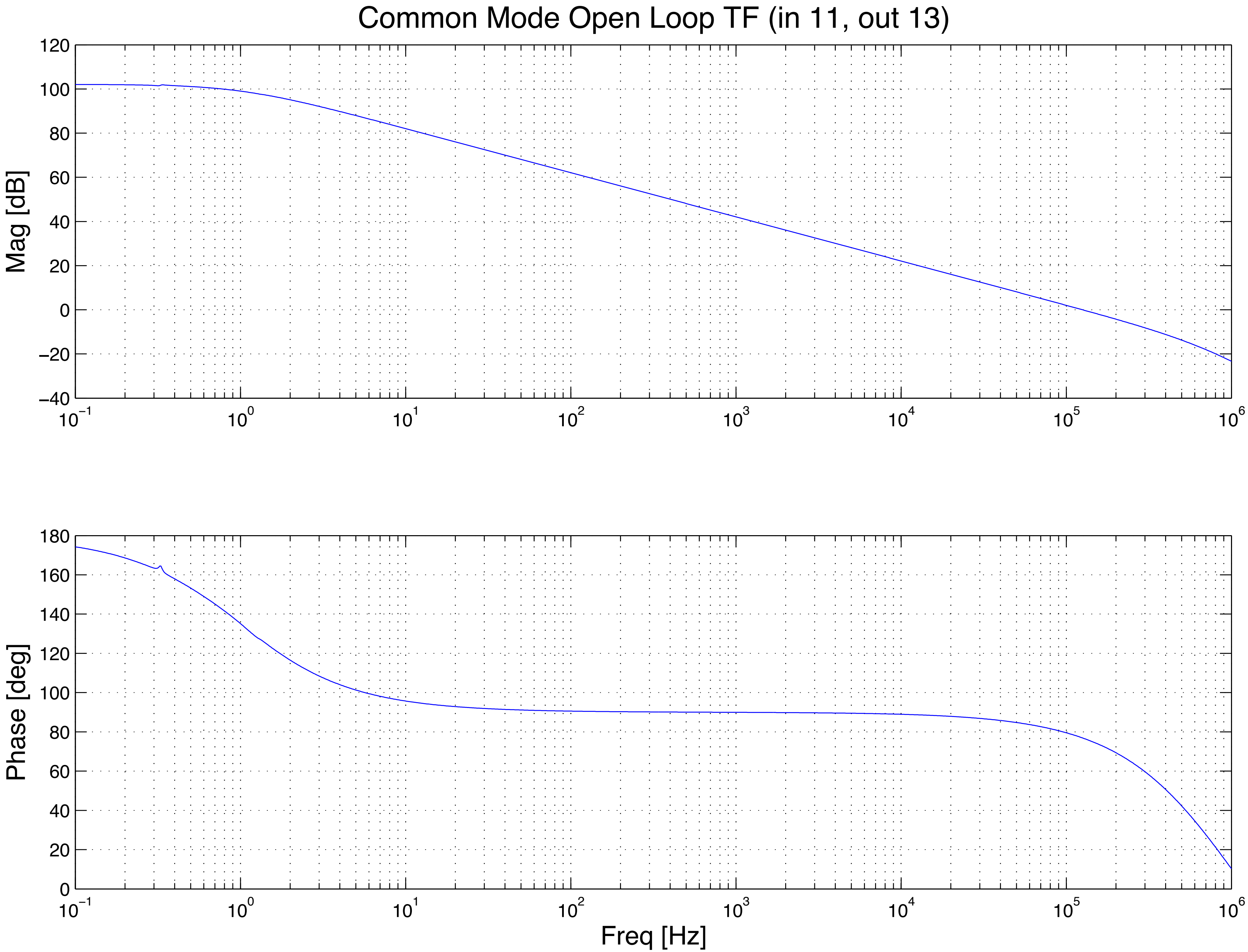
PDH Open Loop TF (in 9, out 10)
Mag [dB]
50
0
−50
−100
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
Phase [deg]
150
100
50
0
−50
−100
−150
Freq [Hz]


PDH Suppression Response (in 10, out 9)
Mag [dB]
0
−10
−20
−30
−40
−50
−60
−70
−80
10^{-1} 10^{0} 10^{1} 10^{2} 10^{3} 10^{4} 10^{5} 10^{6}
Phase [deg]
80
60
40
20
0
−20
−40
−60
−80
10^{-1} 10^{0} 10^{1} 10^{2} 10^{3} 10^{4} 10^{5} 10^{6}
Freq [Hz]

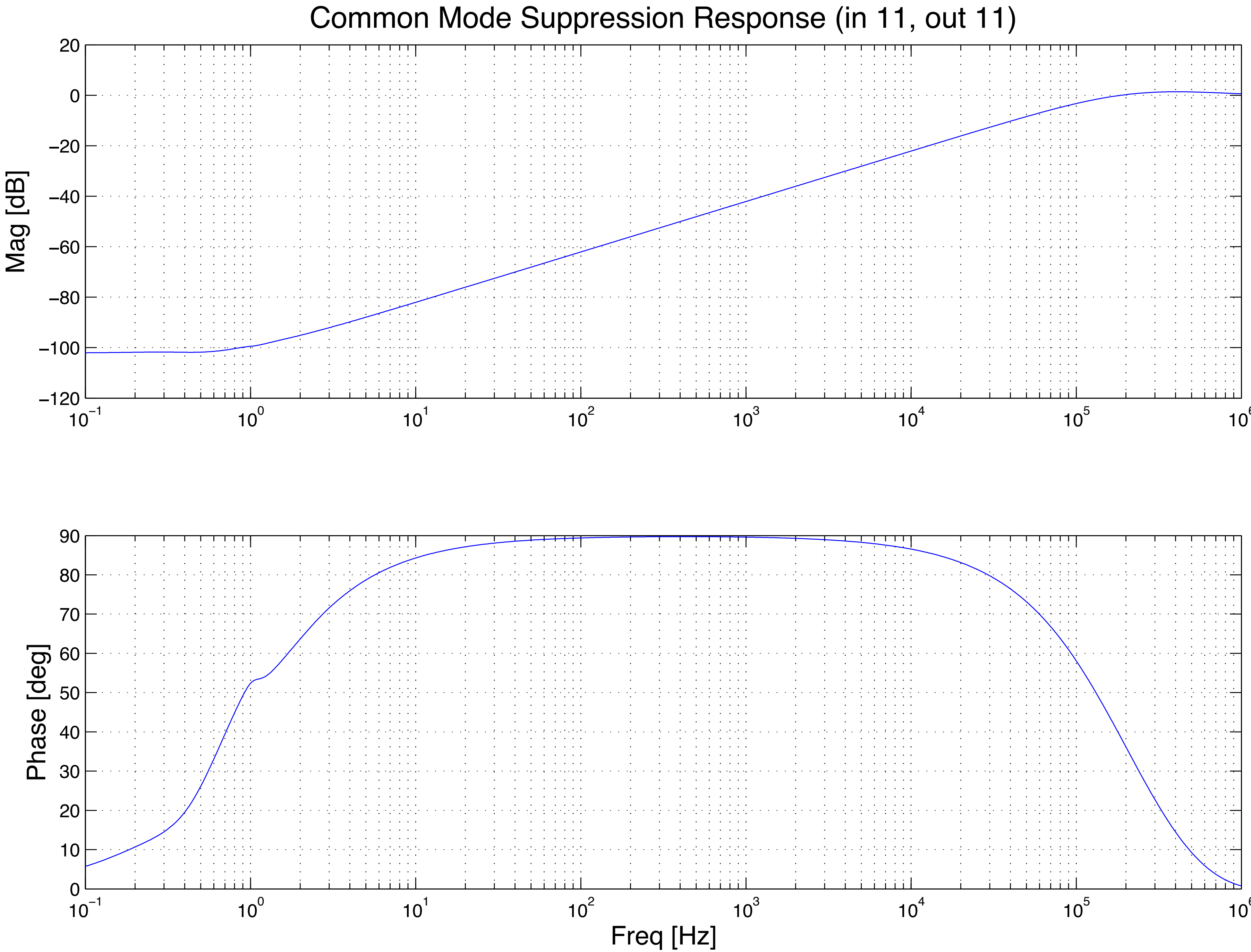

VCO,PSL −to− Common Mode Error Signal TF (in 12, out 13)
Mag [dB]
0
−20
−40
−60
−80
−100
−120
−140
−160
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
Phase [deg]
180
160
140
120
100
80
60
40
20
0
Freq [Hz]



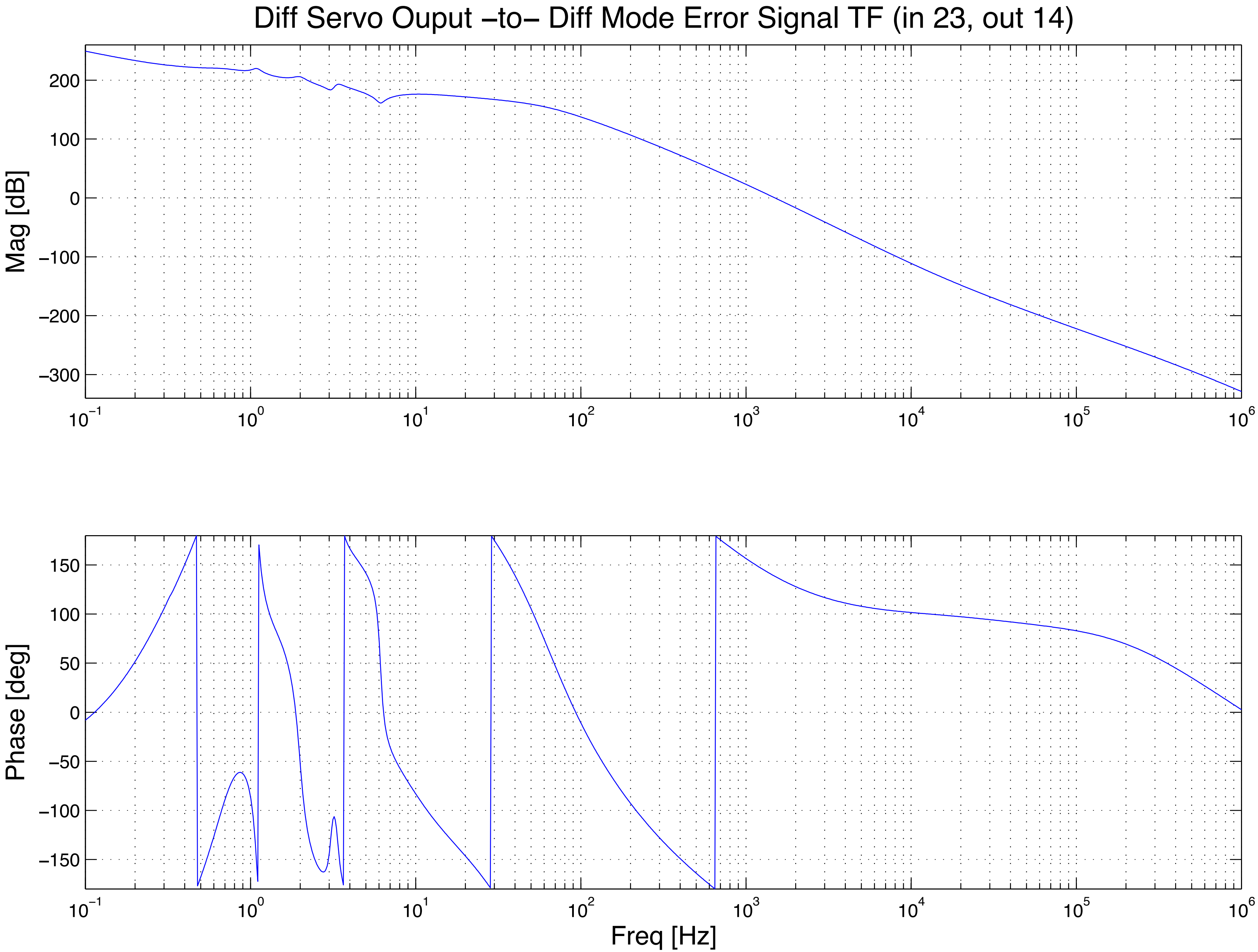
Common Mode Controller TF (in 11, out 12)
Mag [dB]
140
135
130
125
120
115
110
105
100
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
Phase [deg]
90
80
70
60
50
40
30
20
10
0
Freq [Hz]



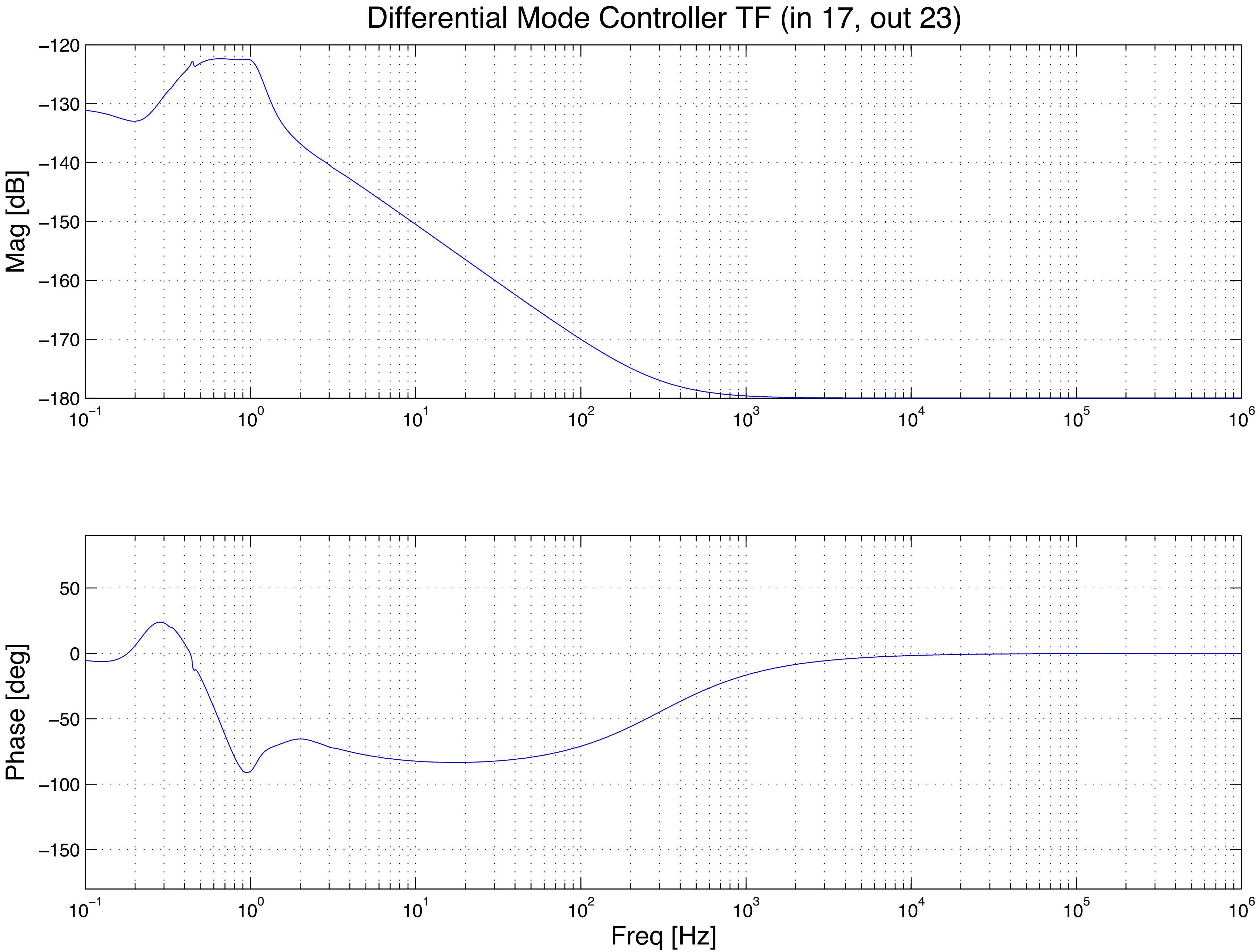
Common Mode Open Loop TF (in 11, out 13)
Mag [dB]
Phase [deg]
Freq [Hz]
120
100
80
60
40
20
0
−20
−40
180
160
140
120
100
80
60
40
20
0
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}



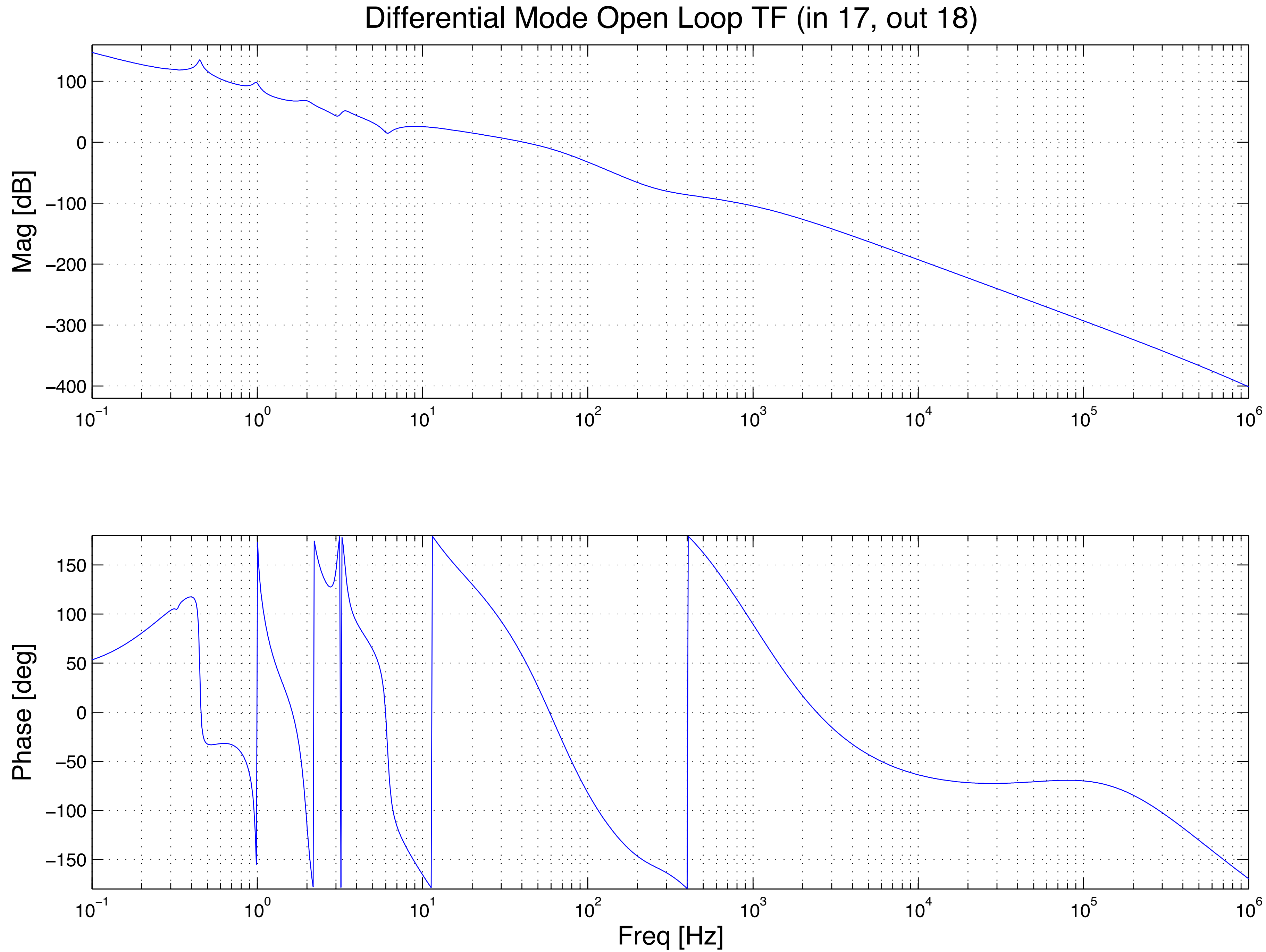
Common Mode Suppression Response (in 11, out 11)
Mag [dB]
20
0
−20
−40
−60
−80
−100
−120
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
Phase [deg]
90
80
70
60
50
40
30
20
10
0
Freq [Hz]

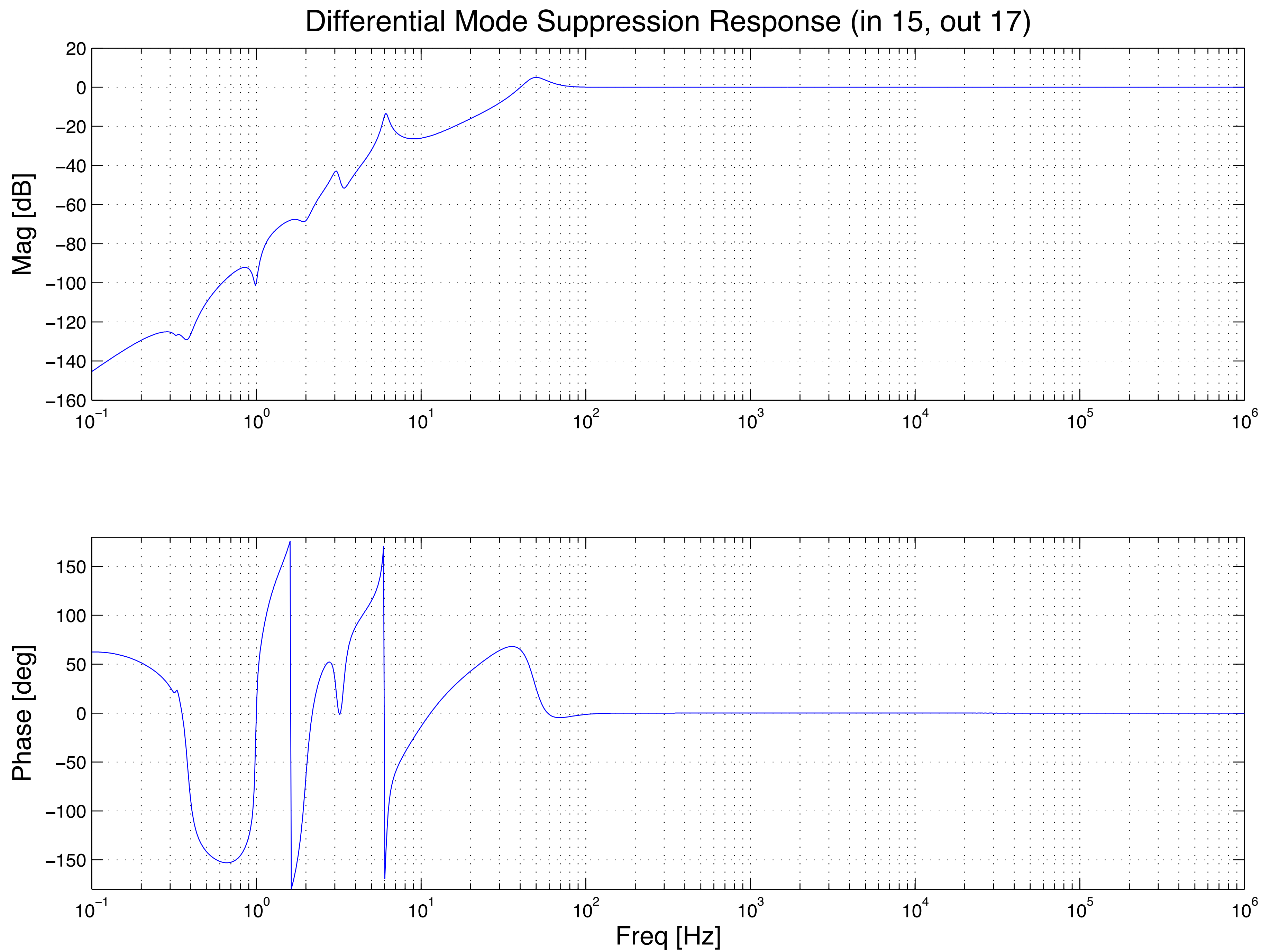


Diff Servo Ouput –to– Diff Mode Error Signal TF (in 23, out 14)
Mag [dB]
200
100
0
−100
−200
−300
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
Phase [deg]
150
100
50
0
−50
−100
−150
Freq [Hz]

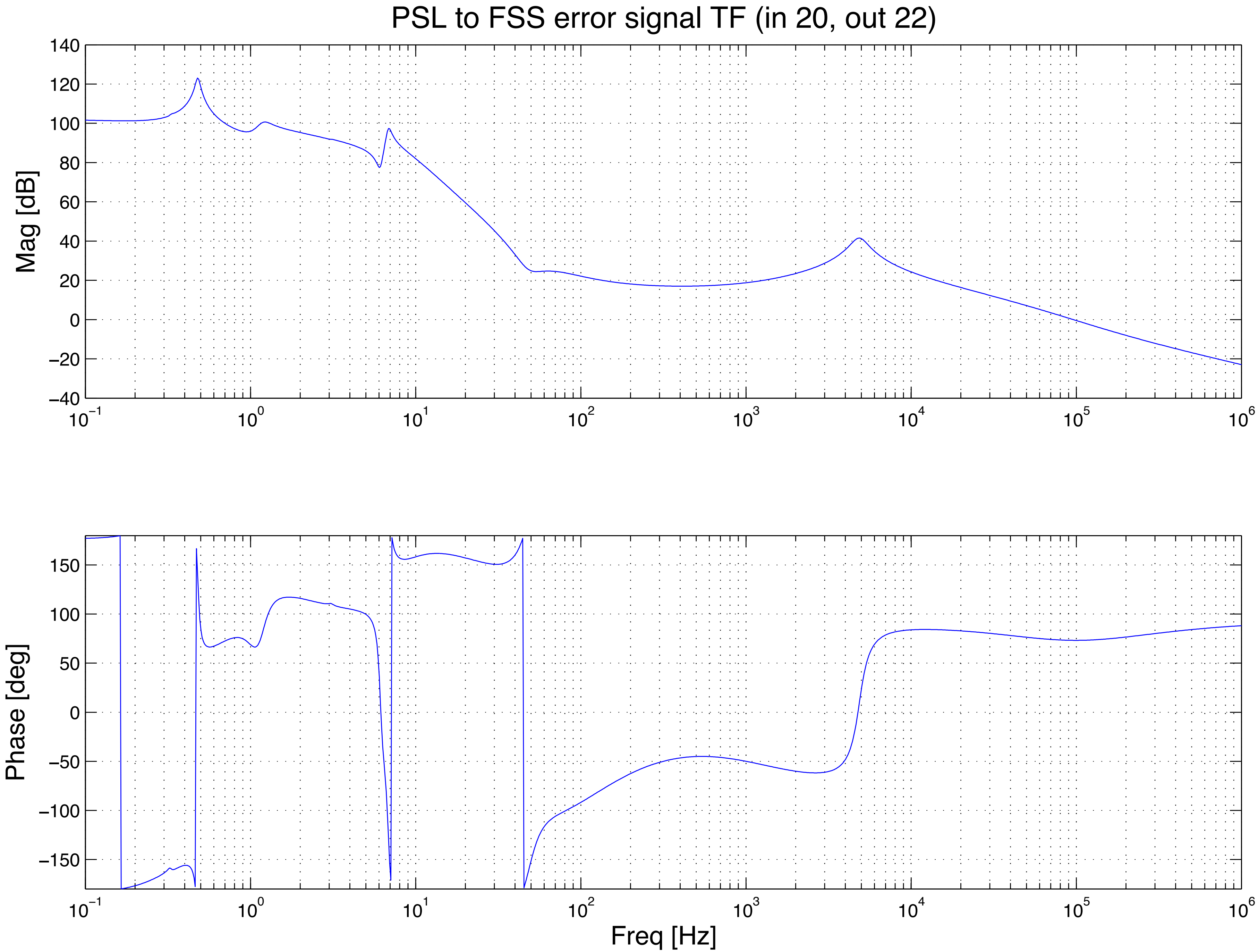


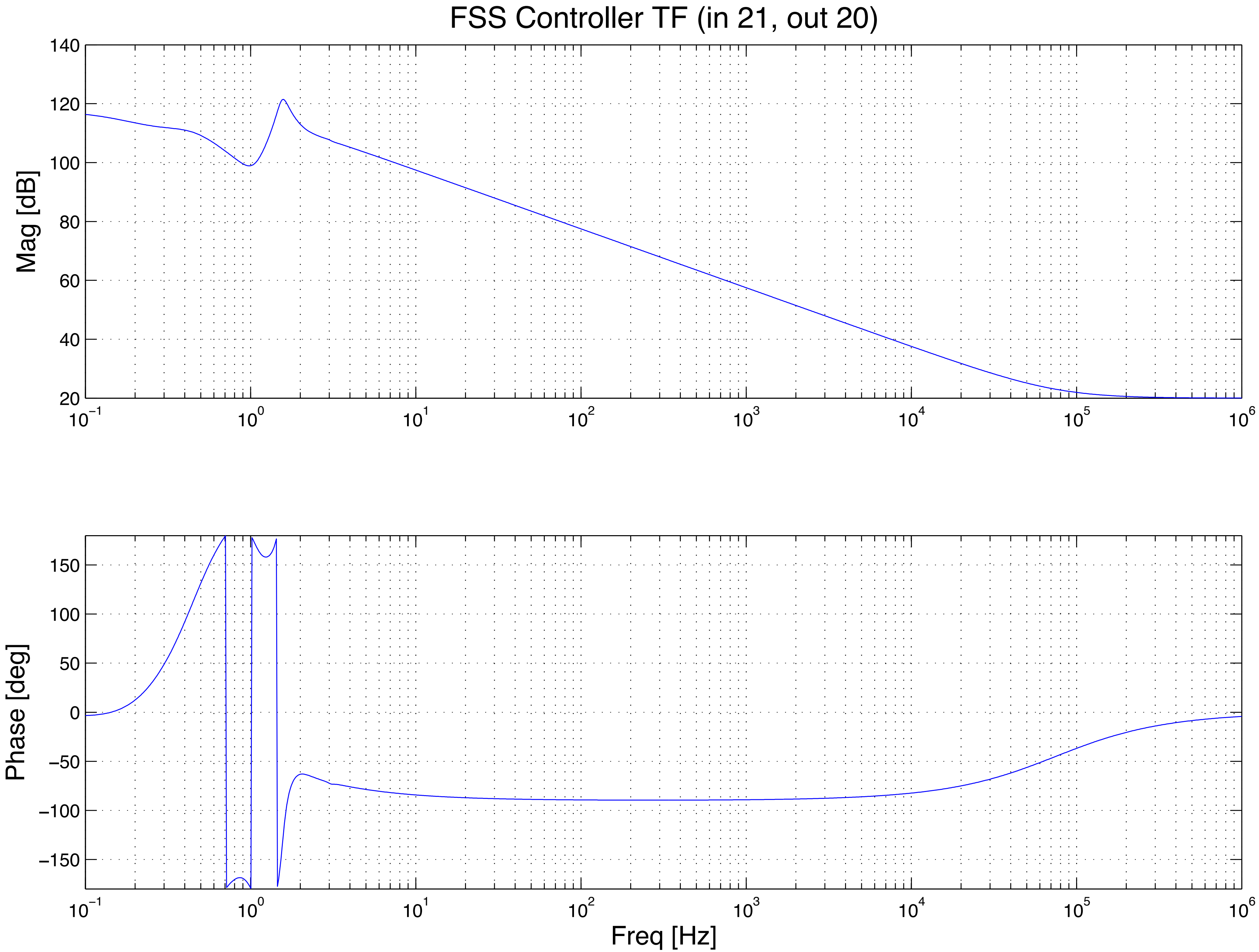
Differential Mode Controller TF (in 17, out 23)
Mag [dB]
−120
−130
−140
−150
−160
−170
−180
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
Phase [deg]
50
0
−50
−100
−150
Freq [Hz]

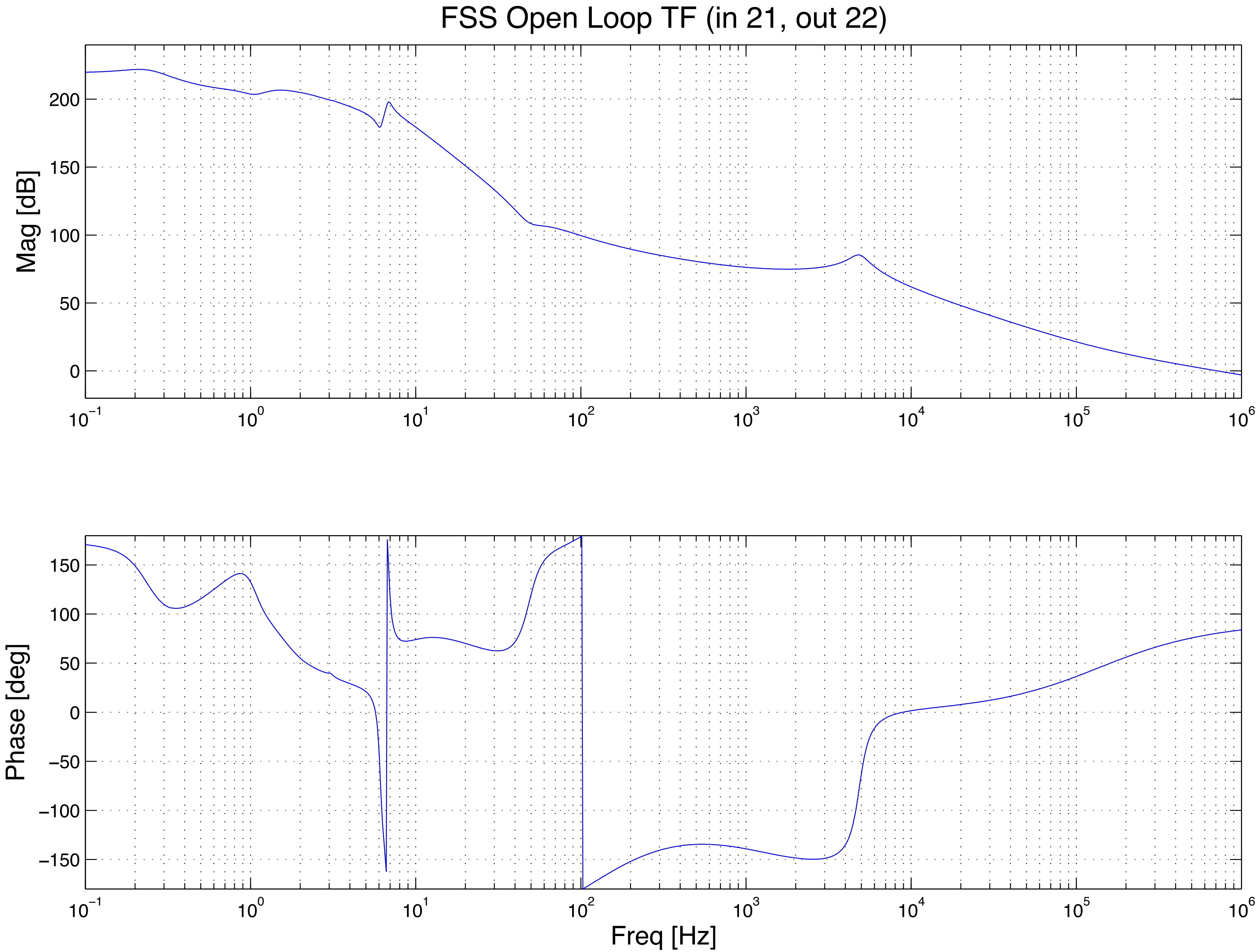

Differential Mode Open Loop TF (in 17, out 18)
Mag [dB]
100
0
−100
−200
−300
−400
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
Phase [deg]
150
100
50
0
−50
−100
−150
Freq [Hz]

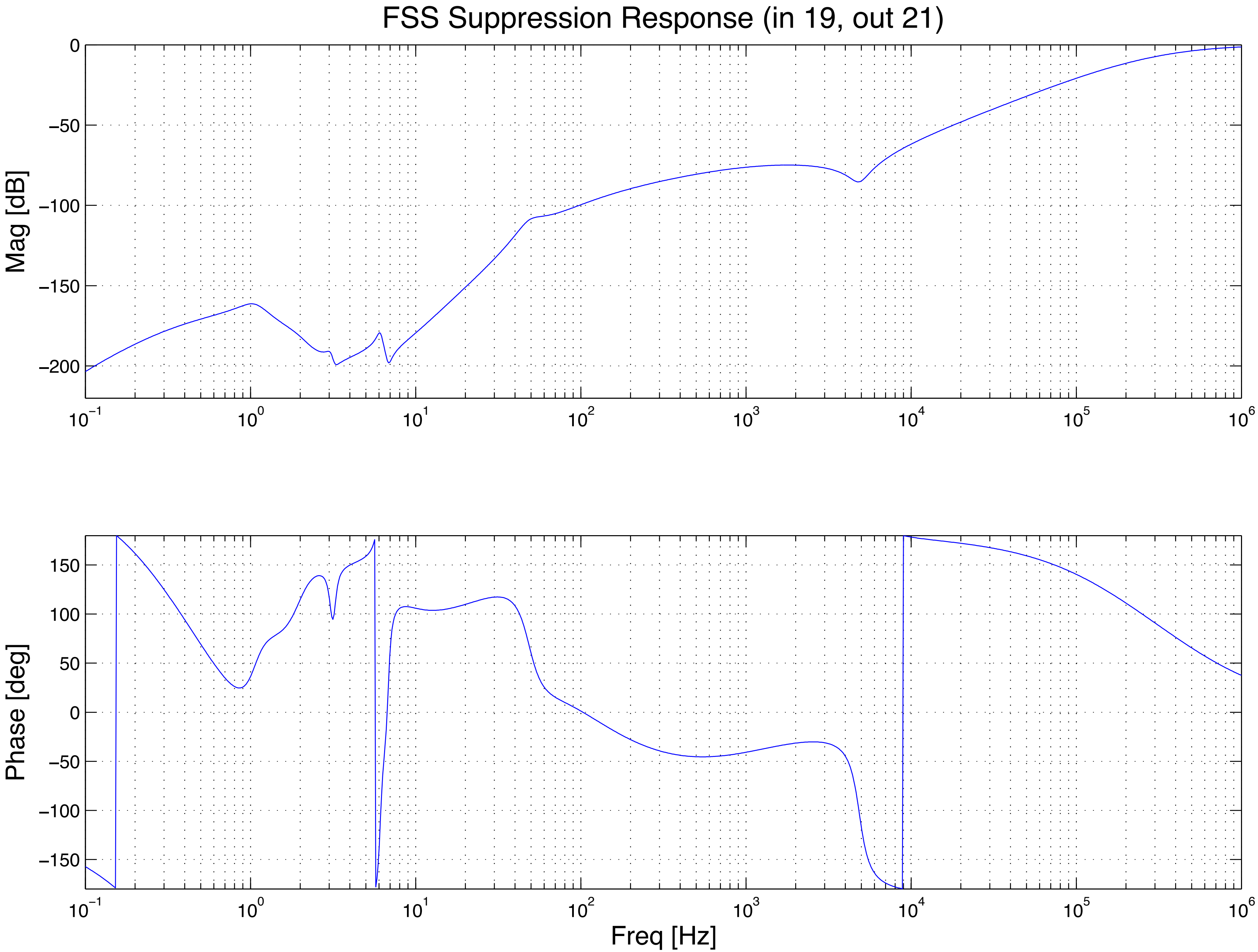


Differential Mode Suppression Response (in 15, out 17)
Mag [dB]
20
0
−20
−40
−60
−80
−100
−120
−140
−160
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
Phase [deg]
150
100
50
0
−50
−100
−150
Freq [Hz]


PSL to FSS error signal TF (in 20, out 22)
Mag [dB]
140
120
100
80
60
40
20
0
−20
−40
Phase [deg]
150
100
50
0
−50
−100
−150
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
Freq [Hz]


FSS Controller TF (in 21, out 20)
Mag [dB]
Phase [deg]
Freq [Hz]


FSS Open Loop TF (in 21, out 22)
Mag [dB]
Phase [deg]
Freq [Hz]
200
150
100
50
0
−50
−100
−150
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}



FSS Suppression Response (in 19, out 21)
Mag [dB]
0
−50
−100
−150
−200
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
Phase [deg]
150
100
50
0
−50
−100
−150
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}
Freq [Hz]

```
% ALS Locking Strategy
%
% Daniel Sigg's idea using 4 VCO's
%
%
% BS - 10 May 2010
%

clear all;

% Constants
c = 299792458; % [m/s]

lambda_IR = 1064e-9;
lambda_GRN = 532e-9;

f = logspace(-1, 6, 1e3);

%% Setting up the PSL section
%

% Ref Cav Transmission TF
RefCavFSR = c / (2 * 0.2); % Ref Cav length 20cm?
RefCavFIN = 5000; % Ref Cav Finesse
RefCavPole = 2*pi * RefCavFSR / (2* RefCavFIN); % Ref Cav Pole frequency, 470kHz

% PSL FSS
PSLfrequencyActuator = 1;     % This is just a gain for the FSS feedback to the PSL
(Temp, PZT and Pockell)
zzz = [RefCavPole/2/pi];
ppp = [1];
kkk = 10;
FSS_Servo = zpk(-2*pi*zzz, -2*pi*ppp, kkk);


% LowNoiseVCO -> FSS low noise VCO driver TF, [Hz/V]
zzz = [];
ppp = 2e6;                  % Range of the VCO
kkk = ppp / 20;             % VCO Full tuning range (2 MHz) / VCO Input voltage range
(+/-20V)
LowNoiseVCO_psl = zpk(-2*pi*zzz, -2*pi*ppp, kkk);


% AOM Driver at fiber launch
AOM_Driver = 1;

%% Setting up the X-End station
%

FiberPhaseNoise = 100;   % flat fiber induced phase noise, 100 Hz/rtHz
freqnoiseNPRO = abs(1e4 ./ (1 + i.*f/1)); % Freerunning NPRO, 100 Hz/rtHz at 100 Hz

% Arm Cav Transmission TF
L_arm = 3995;
ArmCavFSR = c / (2 * L_arm); % Ref Cav length 20cm?
ArmCavFIN = 100; % Ref Cav Finesse
ArmCavPole = 2*pi* ArmCavFSR / (2* ArmCavFIN); % Arm Cav Pole frequency, ~1178 Hz

% LowNoiseVCO,EX -> TTFSS low noise VCO driver TF, used to demodulate the
% heterodyne signal in the end-station
zzz = [];
ppp = 2e6;
kkk = ppp / 20;             % VCO Full tuning range (2 MHz) / VCO Input voltage range
(+/-20V)
LowNoiseVCO_local = zpk(-2*pi*zzz, -2*pi*ppp, kkk);
```

```
% End-Station laser feedback
uPZT_local = 3e6;   % PZT Volts to Frequency conversion, 3 MHz/V

% TTFSS Servo -> TTFSS Locking Servo, lock the laser to the heterodyne beatnote.
zzz = [0 1 RefCavPole/2/pi];
ppp = [1e3 1e3 0.5e5];            % limited by the feedback to the laser PZT?
kkk = 0.03;
TTFSS_Servo = zpk(-2*pi*zzz, -2*pi*ppp, kkk);


% PDH Servo -> PDH locking servo to lock the laser frequency to the arm cavity
% add notch at 1 Hz, Q=10, depth 30 dB (using foton:)

%- When DIFF and COMM are not engaged
% zzz = [1 300 0.003+i*0.999949 0.003-i*0.999949];
% ppp = [1.001+i*0.994872 1.001-i*0.994872 1e5 1e5];
% kkk = 100000000000000;%6000000;

zzz = [100 100];
ppp = [1 1 1e5 1e5];
kkk = 3e14;
PDH_Servo = zpk(-2*pi*zzz, -2*pi*ppp, kkk);


%% Setting up the Vertex ALS Demodulation
%

% LowNoiseVCO,Comm -> Vertex ALS Common Mode low noise VCO driver TF
zzz = [];
ppp = 2e6;                  % Frequency range, Hz
kkk = ppp / 20;             % VCO Full tuning range (2 MHz) / VCO Input voltage range
(+/-20V)
LowNoiseVCO_comm = zpk(-2*pi*zzz, -2*pi*ppp, kkk);

% LowNoiseVCO,Diff -> Vertex ALS Differential Mode low noise VCO driver TF
LowNoiseVCO_diff = zpk(-2*pi*zzz, -2*pi*ppp, kkk);

%% Common Mode Servo -> Common Mode locking servo to lock the PSL frequency
%% to the common mode arm cavity length fluctuations
zzz = [0];
ppp = [1];
kkk = 4e6;
CommonMode_Servo = zpk(-2*pi*zzz, -2*pi*ppp, kkk);


%% Differential Signal Feedback to the ETM Quads
nu_IR = c / lambda_IR;

% Quad response of error signal input to TM displacement
zzz = [];
ppp = [0.005+1i 0.005-1i];
kkk = 10;
LSCquad = zpk(-2*pi*zzz, -2*pi*ppp, kkk);
%mybodesys(LSCquad,f);

% Quad LSC feedback servo response
zzz = [1 0.003+i*0.999949 0.003-i*0.999949];
ppp = [1.001+i*0.994872 1.001-i*0.994872 1e4];
kkk = 1;
Quad_Servo = zpk(-2*pi*zzz, -2*pi*ppp, kkk);

% Differential Mode Servo
zzz = [ 300];
ppp = [1 ];
kkk = 0.000000001;
DifferentialMode_Servo = zpk(-2*pi*zzz, -2*pi*ppp, kkk);
```

```
%% Setting up the Quad Feedback and Control Block
global pend

% Angular radiation pressure torque coefficients
k_major = 0;
k_minor = 0;
k_ospring = 0;

gLP = 0;
ServoTM = 0;
ServoPM = 0;
ServoUIM = 0;
ServoTOP = 0;

damper = 1; % ECD
% damper = 2; % GEO Damping
% damper = 3; % Damping with fancy LPF
% damper = 4; % no damping

%************************************
ssmake4pv2eMB2; % better blade modeling from MATHEMATICA, Mark Barton
%************************************
localdamp;

if ~exist('k_ospring')
  k_ospring = 0;
else
  % MEVANS
  warning('NOT including optical spring.');
  k_ospring = 0;
end

% Run the Quad Servo Script
PDHservo_All_2010_05_21_11_00_19

% Set the Signal Path Switches
gLP = 0;    % Keep the loop open to make it run within the overall simulation
gTM = 1;    % close the TM feedback
gPM = 1;    % close the PM feedback
gUIM = 1;   % close the UIM feedback
gTOP = 1;   % close the TOP feedback

%% Implementing the Simulink Model
%
modelname = 'ALS_freq3v3';
%%
[AAA,BBB,CCC,DDD] = linmod2(modelname); % linearise the Simulink model
SYS = ss(AAA,BBB,CCC,DDD);              % ceates a state-space model of the Simulink
model

%% Print the Simulink model with all its colours
% set_param(modelname, 'ShowPageBoundaries', 'on');
% print(['-s' modelname], '-dpdf', [modelname '.pdf']); % print the simulink model with
its colors...

%% Obtaining the transfer functions
save_figure = 1;    % controls is the figures are save as .pdf or not

if save_figure
    FSS=1;
    TTFSS=1;
    PDH=1;
    COMM=1;
    DIFF=1;
```

```
else
    FSS = 0;    % Plots the FSS loops of th PSL servo
    TTFSS = 0;  % Plots the TTFSS loops in the end-station
    PDH = 0;    % plots the PDH loops in the end-station
    COMM = 0;
    DIFF = 1;
end         % end save_figure

if FSS
%% FSS Feedback of the laser to the Reference Cavity
    %
%%  % PSL laser -to- FSS error signal
    hdl= figure(101)
    a = SYS(20,18);
    b = SYS(22,18);
    G = mybodesys(b/a,f);
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('PSL to FSS error signal TF (in 20, out 22)', 'FontSize', 16);
    ylabel('Mag [dB]', 'FontSize', 14);
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
    ylabel('Phase [deg]', 'FontSize', 14);
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
%    set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end


%%  % FSS Open Loop response
    hdl= figure(102)
    a = SYS(21,19);
    b = SYS(20,19);
    G = mybodesys(b/a,f);
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('FSS Controller TF (in 21, out 20)','FontSize', 16);
    ylabel('Mag [dB]', 'FontSize', 14);
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
    ylabel('Phase [deg]', 'FontSize', 14);
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
%    set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end


%%  % FSS Close Loop Response Response
    hdl= figure(103)
    a = SYS(21,19);
```

```
    b = SYS(22,19);
    G = mybodesys(b/a,f);
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('FSS Open Loop TF (in 21, out 22)', 'FontSize', 16);
    ylabel('Mag [dB]', 'FontSize', 14);
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
    ylabel('Phase [deg]', 'FontSize', 14);
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
%    set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end

%%  % FSS Supression Response
    hdl= figure(104)
    G = mybodesys(SYS(21,19),f);
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('FSS Suppression Response (in 19, out 21)', 'FontSize', 16);
    ylabel('Mag [dB]', 'FontSize', 14);
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
    ylabel('Phase [deg]', 'FontSize', 14);
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
%    set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end

end         % end FSS



if TTFSS
%% TTFSS Feedback of the laser to the Heterodyen Signal
    %
    % Local laser to TTFSS error signal
    hdl= figure(1)
    a = SYS(5,8);
    b = SYS(3,8);
    G = mybodesys(b/a,f);
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('Local laser -to- TTFSS error signal TF (in 5, out 3)', 'FontSize', 16);
    ylabel('Mag [dB]', 'FontSize', 14);
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
```

```
    ylabel('Phase [deg]', 'FontSize', 14);
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
%    set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end


    % TTFSS Open Loop response
    hdl= figure(2)
    a = SYS(6,7);
    b = SYS(5,7);
    G = mybodesys(b/a,f);
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('TTFSS Controller TF (in 6, out 5)','FontSize', 16);
    ylabel('Mag [dB]', 'FontSize', 14);
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
    ylabel('Phase [deg]', 'FontSize', 14);
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
%    set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end


    % TTFSS Close Loop Response Response
    hdl= figure(3)
    a = SYS(6,7);
    b = SYS(3,7);
    G = mybodesys(b/a,f);
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('TTFSS Open Loop TF (in 3, out 6)', 'FontSize', 16);
    ylabel('Mag [dB]', 'FontSize', 14);
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
    ylabel('Phase [deg]', 'FontSize', 14);
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
%    set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end

    % TTFSS Supression Response
    hdl= figure(4)
    G = mybodesys(SYS(6,7),f);
```

```
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('TTFSS Suppression Response (in 7, out 6)', 'FontSize', 16);
    ylabel('Mag [dB]', 'FontSize', 14);
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
    ylabel('Phase [deg]', 'FontSize', 14);
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
%     set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end

end          % end TTFSS

if PDH
%% PDH Feedback of the laser frequency to the arm cavity, via the VCO,EX
    %
    % Local VCO,EX to PDH error signal
    hdl = figure(5)
    a = SYS(8,9);
    b = SYS(10,9);
    G = mybodesys(b/a,f);
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('VCO,EX -to- PDH Error Signal TF (in 8, out 10)', 'FontSize', 16);
    ylabel('Mag [dB]', 'FontSize', 14);
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
    ylabel('Phase [deg]', 'FontSize', 14);
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
%     set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end


    % PDH Open Loop response
    hdl= figure(6)
    a = SYS(9,10);
    b = SYS(8,10);
    G = mybodesys(b/a,f);
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('PDH Controller TF (in 9, out 8)', 'FontSize', 16);
    ylabel('Mag [dB]', 'FontSize', 14);
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
    ylabel('Phase [deg]', 'FontSize', 14);
```

```
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
%    set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end


    % PDH Close Loop Response Response
    hdl = figure(7)
    a = SYS(9,10);
    b = SYS(10,10);
    G = mybodesys(b/a,f);
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('PDH Open Loop TF (in 9, out 10)', 'FontSize', 16);
    ylabel('Mag [dB]', 'FontSize', 14);
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
    ylabel('Phase [deg]', 'FontSize', 14);
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
%    set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end

    % PDH Supression Response
    hdl = figure(8)
    G = mybodesys(SYS(9,10),f);
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('PDH Suppression Response (in 10, out 9)', 'FontSize', 16);
    ylabel('Mag [dB]', 'FontSize', 14);
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
    ylabel('Phase [deg]', 'FontSize', 14);
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
%    set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end

end          % end PDH

if COMM
%% Common Mode Feedback of the PSL frequency to the common mode arm cavity
%% length fluctuations, via the VCO,C
    %
    % Vertex VCO,PSL to Common Mode error signal
```

```
    hdl = figure(9)
    a = SYS(12,12);
    b = SYS(13,12);
    G = mybodesys(b/a,f);
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('VCO,PSL -to- Common Mode Error Signal TF (in 12, out 13)', 'FontSize',
16);
    ylabel('Mag [dB]', 'FontSize', 14);
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
    ylabel('Phase [deg]', 'FontSize', 14);
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
%    set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end

    % Common Mode Servo Open Loop response
    hdl= figure(10)
    a = SYS(11,11);
    b = SYS(12,11);
    G = mybodesys(b/a,f);
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('Common Mode Controller TF (in 11, out 12)', 'FontSize',16);
    ylabel('Mag [dB]', 'FontSize', 14)
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
    ylabel('Phase [deg]', 'FontSize', 14);
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
%    set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end


    % Common Mode Close Loop Response
    hdl = figure(11)
    a = SYS(11,11);
    b = SYS(13,11);
    G = mybodesys(b/a,f);
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('Common Mode Open Loop TF (in 11, out 13)', 'FontSize', 16);
    ylabel('Mag [dB]', 'FontSize', 14);
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
    ylabel('Phase [deg]', 'FontSize', 14);
```

```
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
%    set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end
%%
    % Common Mode Supression Response
    hdl = figure(12)
    G = mybodesys(SYS(11,11),f);
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('Common Mode Suppression Response (in 11, out 11)', 'FontSize', 16);
    ylabel('Mag [dB]', 'FontSize', 14);
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
    ylabel('Phase [deg]', 'FontSize', 14);
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
%    set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end
end         % end COMM

if DIFF
%% Differential Mode Feedback to both the ETMs (out of phase). This
%% requires the Quad response and its servo, for now I have a single
%% pendulum replacing the Quad...
    %
    % Diff Mode Servo input to Differential Mode error signal
    hdl = figure(13)
    a = SYS(23,20);
    b = SYS(14,20);
    G = mybodesys(b/a,f);
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('Diff Servo Ouput -to- Diff Mode Error Signal TF (in 23, out 14)',
'FontSize', 16);
    ylabel('Mag [dB]', 'FontSize', 14);
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
    ylabel('Phase [deg]', 'FontSize', 14);
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
%    set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end
```

```
    % Diff Mode Controller response
    hdl= figure(14)
    a = SYS(17,15);
    b = SYS(23,15);
    G = mybodesys(b/a,f);
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('Differential Mode Controller TF (in 17, out 23)', 'FontSize',16);
    ylabel('Mag [dB]', 'FontSize', 14)
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
    ylabel('Phase [deg]', 'FontSize', 14);
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
%    set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end

    % Differentail Mode Close Loop Response
    hdl = figure(15)
    a = SYS(17,15);
    b = SYS(18,15);
    G = mybodesys(b/a,f);
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('Differential Mode Open Loop TF (in 17, out 18)', 'FontSize', 16);
    ylabel('Mag [dB]', 'FontSize', 14);
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
    ylabel('Phase [deg]', 'FontSize', 14);
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
%    set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end

    % Differential Mode Supression Response
    hdl = figure(16)
    G = mybodesys(SYS(17,15),f);
    %
    subplot(211)
    semilogx(f,20*log10(abs(G)))
    tt= title('Differential Mode Suppression Response (in 15, out 17)', 'FontSize', 16);
    ylabel('Mag [dB]', 'FontSize', 14);
    axis([min(f) max(f) floor(min(log10(abs(G))))*20 ceil(max(log10(abs(G))))*20])
    grid on
    subplot(212)
    semilogx(f,angle(G)*180/pi)
    ylabel('Phase [deg]', 'FontSize', 14);
    xlabel('Freq [Hz]', 'FontSize', 14);
    axis([min(f) max(f) 90*floor(min(angle(G)*180/pi)/90) 90*ceil(max(angle(G)*180/pi)/
90)])
```

```
%    set(gca,'YTick',[-1800:90:1800])
    grid on
    if save_figure == 1
        orient(hdl, 'landscape');
        print('-dpdf', ['sim/',num2str(hdl,'%.3d'), ' ',get(tt,'string'), '.pdf']);
    end

end         % end DIFF
```