Creating Bode Plots

Introduction
This short paper demonstrates how to create Bode plots using Octave/MATLAB, LTSpice, and a TI calculator. The transfer function of a certain fourth-order, low pass, inverse Chebyshev filter with 3 dB frequency at 9600 radians/second will be used in all examples. The transfer function is

$$T(s) = \frac{0.1\left((s/9600)^2 + 1.1716\right)\left((s/9600)^2 + 6.8283\right)}{\left(\left((s/9600)^2 + 1.8502(s/9600) + 1.2209\right)\left((s/9600)^2 + 0.41128(s/9600) + 0.65519\right)\right)}$$

This transfer function has zeros (nulls) at 10391 rad/s (1654 Hz) and 25086 rad/s (3993 Hz).

Octave/MATLAB
Octave and MATLAB provide a **bode()** function that can be used to draw a Bode plot quite easily. The **bode()** function requires that a system data structure corresponding to the transfer function be created. The **tf()** or **tf2sys()** functions can be used to create this data structure from the numerator and denominator polynomials of the transfer function. Polynomials are represented in Octave/MATLAB using a vector of the polynomial coefficients, for example the polynomial:

$$p(s) = s^4 + 3s^2 + 5s + 10$$

is represented in Octave/MATLAB as

$$p = [\,1 \quad 0 \quad 3 \quad 5 \quad 10\,];$$

Our example transfer function is in factored form. The **conv()** function can be used to find the result of multiplying two polynomials. It can be used to find the expanded polynomials of the numerator and denominator of our transfer function. The Octave/MATLAB commands to produce a bode plot are:

```
numer = 0.1*conv([(1./9600)^2   0   1.1716], [(1./9600)^2   0   6.8283]);
denom=conv([1/9600^2   1.8502/9600   1.2209],[1/9600^2   0.41128/9600   0.65519]);
sys = tf(numer,denom);
w = logspace(3, 6, 200);    % 200 pt w vector from 1krad/s (10^3) to 1Mrad/s (10^6)
bode(sys, w);
```

The result is shown in Figure 1.

To produce a Bode plot versus frequency in Hertz requires only a bit more effort. The plots have to be created "manually". The **bode()** function can still be used to evaluate the transfer function at the desired frequencies:

```
numer = 0.1*conv([(1./9600)^2 0 1.1716], [(1./9600)^2 0 6.8283]);
denom=conv([1/9600^2 1.8502/9600 1.2209],[1/9600^2 .41128/9600 .65519]);
sys = tf(numer,denom);
f = logspace(2, 5, 200);   % 200 pt f vector from 100 Hz (10^2) to 10kHz (10^5)
[mag, phase] = bode(sys,2*pi*f);  % Use bode to determine mag and phase
subplot(2,1,1);
semilogx(f, 20*log10(abs(mag)));
subplot(2,1,2);
semilogx(f, phase);
```

LTSpice
It is extremely useful to be able to produce a Bode plot of a desired transfer function in LTSpice. The desired function can be compared to the response of a filter designed to have the same transfer function. LTSpice can also be used to create a Bode plot of a theoretical transfer function, but we have to create a dummy circuit to force LTSpice to do a AC Analysis. A dummy circuit can consist of only a voltage source with ground and labeled node as shown in Figure 3. The transfer function is included as a "comment" on the schematic. This allows the transfer function to be saved as part of the schematic file. Note that the complex frequency variable *s* has been replaced with *i*w*.

An AC Analysis of this circuit produces no plot. To plot the transfer function, right-click on the plot window and select "Add Trace" from the menu. In the pop-window that appears, copy and paste the transfer function from the schematic into the text entry box at the bottom of the window. The result is shown in Figure 4. The nulls at 1654 Hz and 3993 Hz are clearly visible. (The frequency axis is in Hertz. Compare this plot to the Bode plot shown in Figure 2 which was created in Octave/MATLAB).
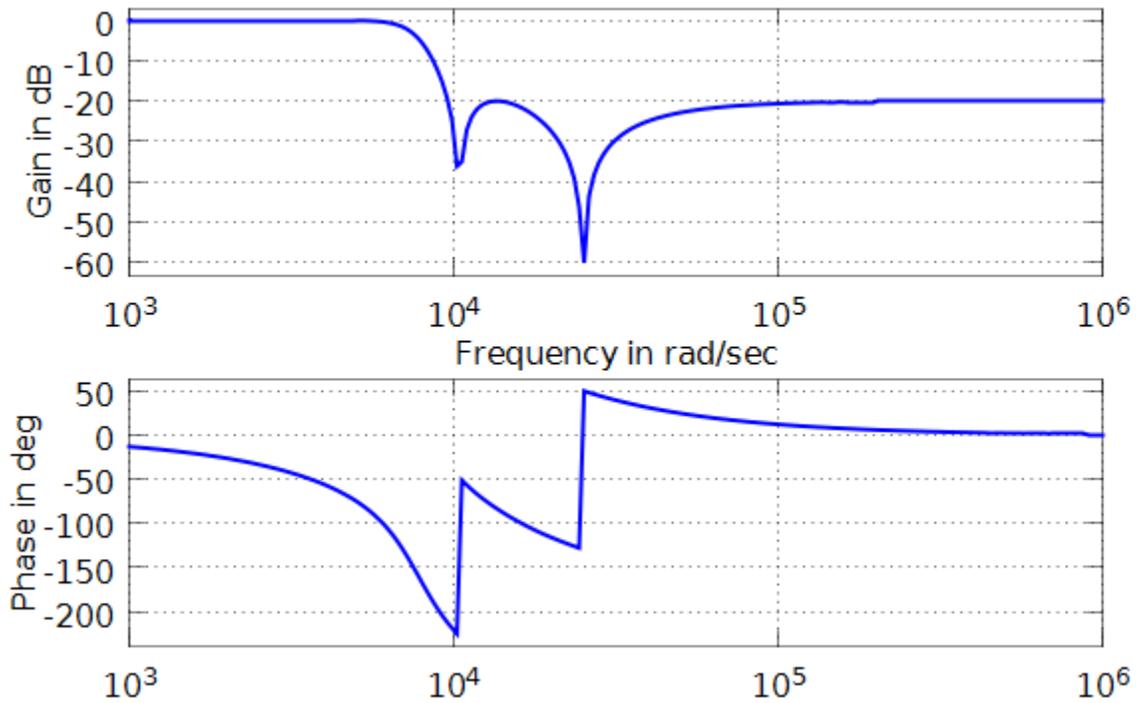
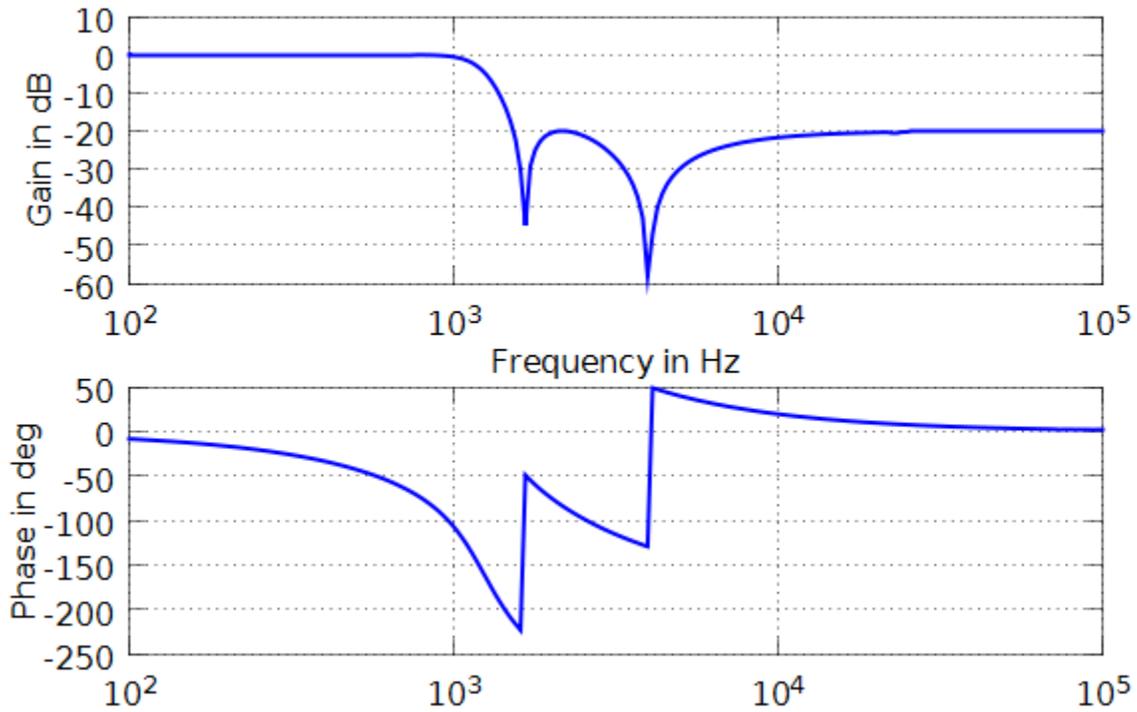Figure 1: Bode Plot from Octave/MATLAB (freq. in rad/s)

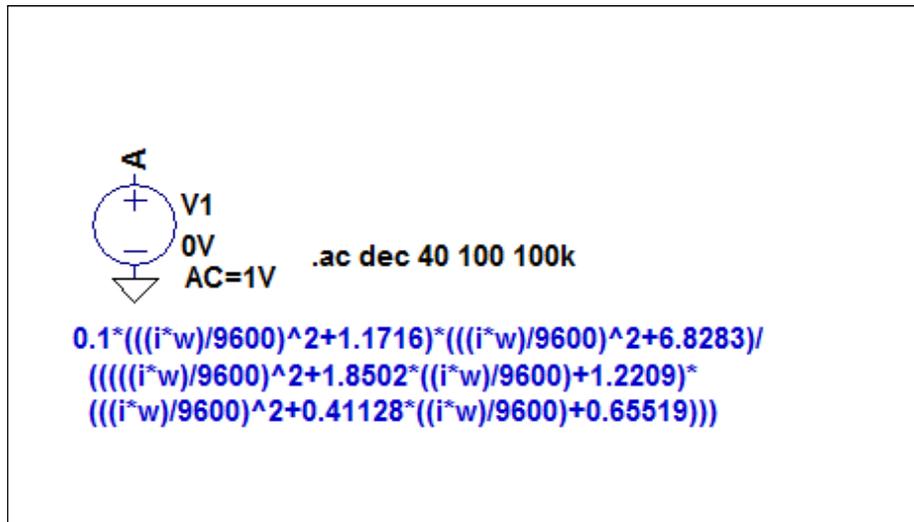Figure 2: Bode Plot from Octave/MATLAB (freq in Hertz)

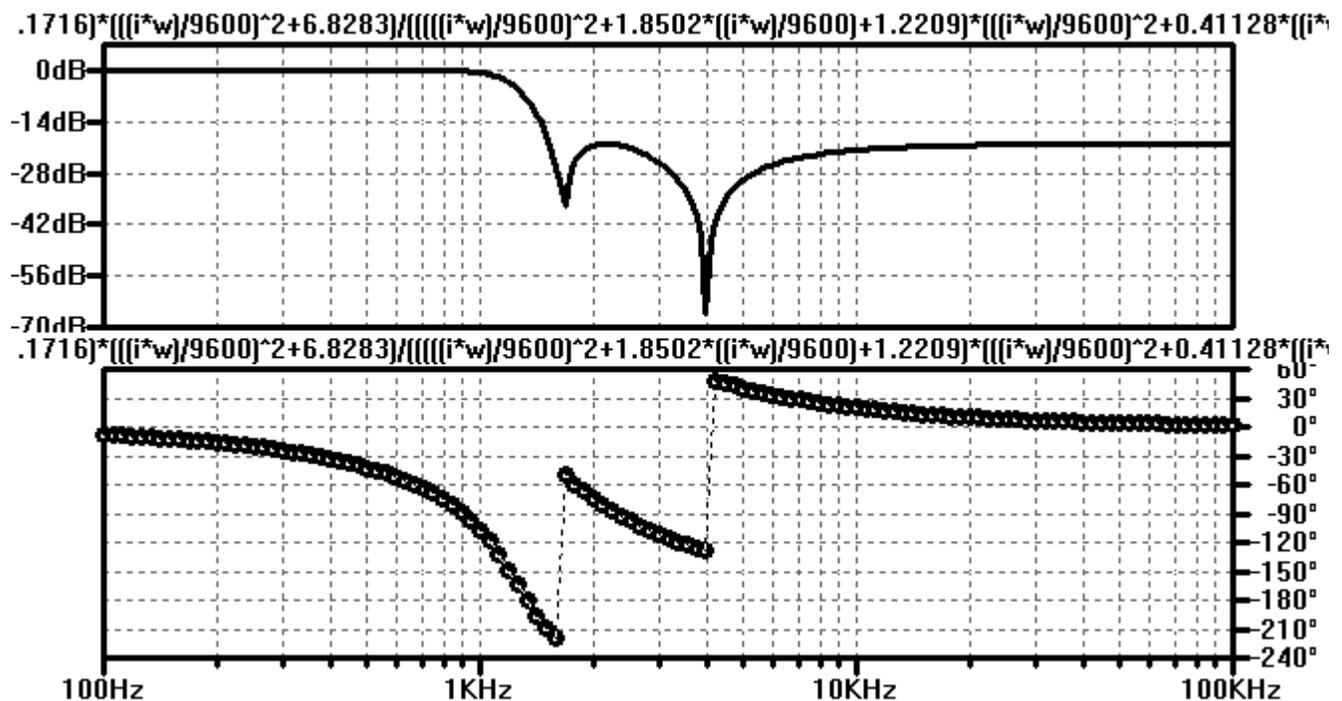Figure 3: LTSpice "Circuit" to Create Bode Plot (freq in Hertz)

Figure 4: Bode Plot from LTSpice (freq in Hertz)

To create a Bode plot versus frequency in rad/s is slightly more difficult. You must define a new parameter w that is stepped over the desired frequency range in rad/s. The AC analysis is then done at frequencies of $w/(2\pi)$. Figure 5 shows the statements that produce a Bode plot with frequency in rad/s. An analysis of this circuit again produces no plot. You must use the "Add Trace" feature to plot the transfer function. The transfer function has exactly the same form as that used in Figure 4. The results are shown in Figure 6.
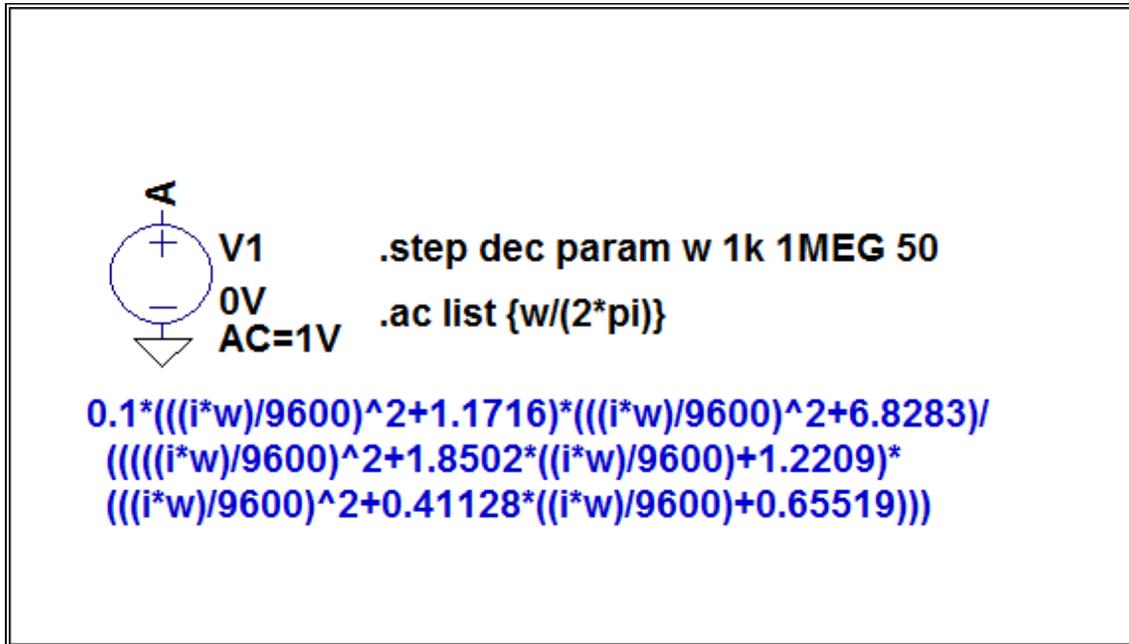
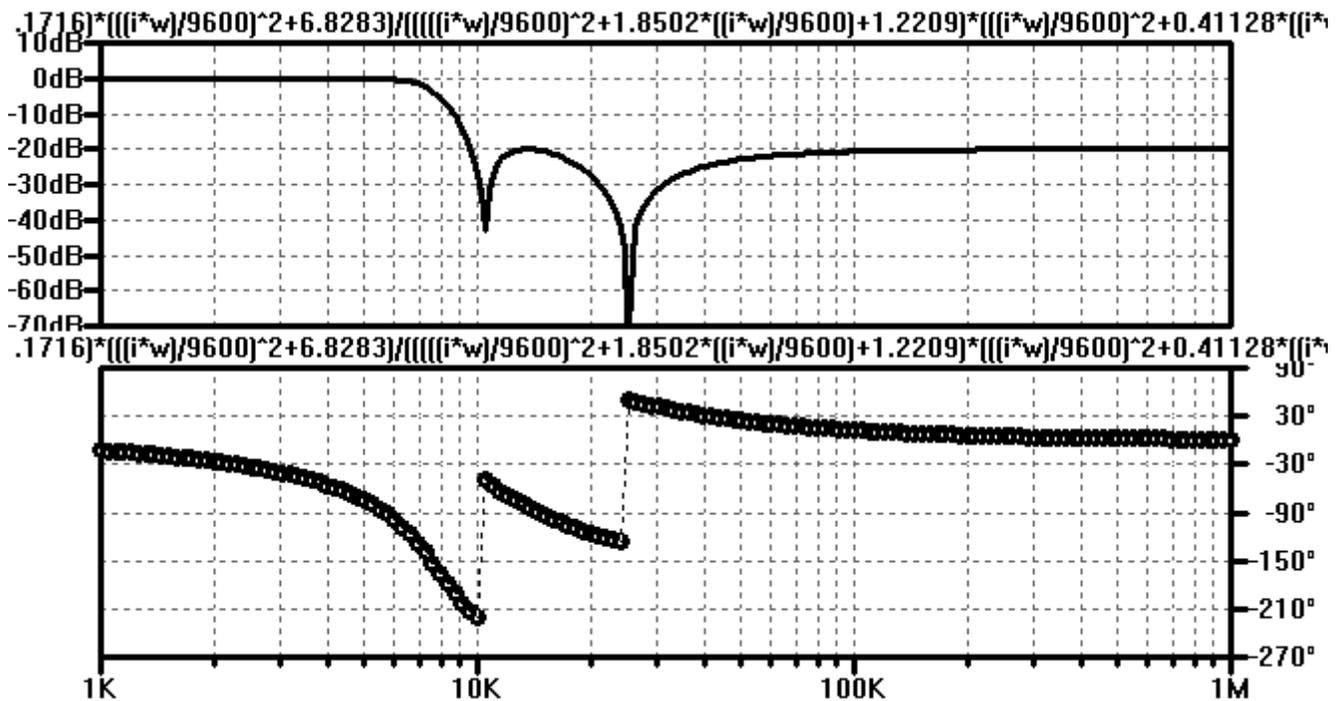Figure 5: LTSpice "Circuit" to Create Bode Plot (freq in rad/s)

Figure 6: Bode Plot from LTSpice (freq. in rad/s)

TI-85/TI-89/TI-92

The TI-85 does not directly produce semilog plots, but it can be "tricked" into doing so. To plot the example transfer function on a TI-85, define the following functions:

```
y1 = 10^x
y2 = (0, y1)
y3 = 0.1*((y2/9600)^2+1.1716)*((y2/9600)^2+6.8283)
y4 = ((y2/9600)^2+1.8502*(y2/9600)+1.2209)*((y2/9600)^2+0.41128*(y2/9600)+0.65519)
y5 = 20*log(abs(y3/y4))
y6 = angle(y3/y4)
```

y1 represents the frequency variable ω. The values along the x-axis are then log(ω) values. y2 is the complex variable j ω and y2 is used in place of s in all expressions. y3 is the numerator and y4 is the denominator of the transfer function. y5 is the magnitude function and y6 is the phase function. Set xMin = 3, xMax = 6, and xScl = 1 this corresponds to $10^3 < ω < 10^6$. When plotting y5 set yMin = -70, yMax = 10, and yScl = 10. When plotting y6 set yMin = -180, yMax = 180, and yScl = 30. When using the trace function on the calculator to find points of interest the actual frequency corresponds to 10^x.

The TI-89 and TI-92 calculators use slightly different notation for representing functions and complex numbers.. The required definitions are:

```
y1(x) = 10^x
y2(x) = i y1(x)
y3(x) = 0.1*((y2(x)/9600)^2+1.1716)*((y2(x)/9600)^2+6.8283)
y4(x) = ((y2(x)/9600)^2+1.8502*(y2(x)/9600)+1.2209)*
                            ((y2(x)/9600)^2+0.41128*(y2(x)/9600)+0.65519)
y5(x) = 20*log(abs(y3(x)/y4(x)))
y6(x) = angle(y3(x)/y4(x))
```

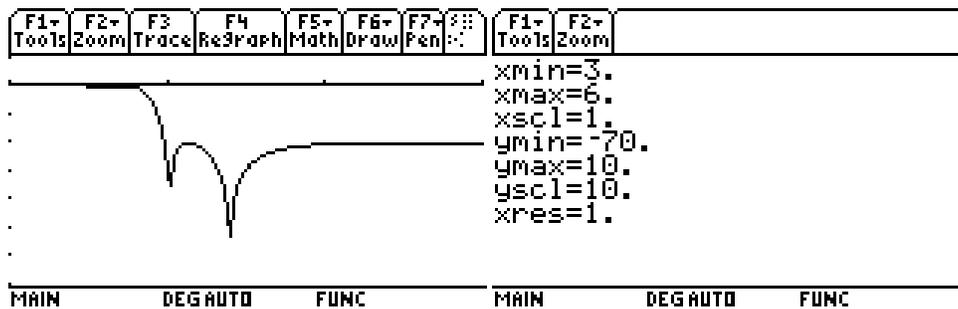The resulting plots are shown in Figures 7 and 8.
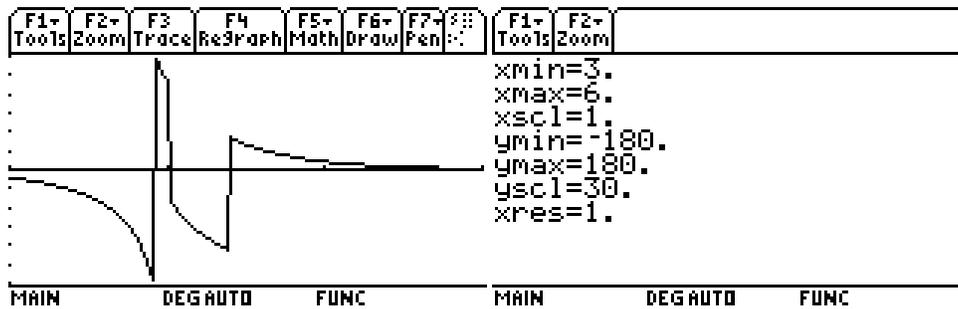

Figure 7: Magnitude Plot from a TI-89

Figure 8: Phase Plot from a TI-89

Note that Octave/MATLAB and LTSpice both "unwrap" the phase. This gives a smoother phase plot, but results in phase angles outside the normal -180 to +180 degree range. The TI-89 does not do this and so all phase angles are in the -180 to +180 degree range, but there may be jumps of 360 degrees in the plot.

For plotting in Hertz set y2 = (0, 2 π y1) on the TI-85 and y2(x) = i 2 π y1(x) on the TI-89.