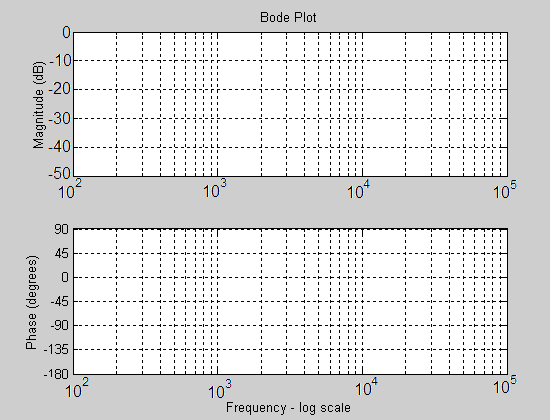
Use the excel file accompanying Lesson 26 to generate the data in the table below – make sure to use the “second order” tab. When complete, plot the data in the empty graph at right to build the phase and magnitude plots.

|  |  |  |
| --- | --- | --- |
| Frequency  Cell B2 | Gain  Cell P10 | Phase  Cell P9 |
| 100 |  |  |
| 300 |  |  |
| 500 |  |  |
| 1,000 |  |  |
| 3,000 |  |  |
| 5,000 |  |  |
| 7,000 |  |  |
| 10,000 |  |  |
| 13,000 |  |  |
| 15,000 |  |  |



Calculate the slope of the Gain (to the right of the cut-off

frequency). Show your work here:

Use the equations presented in class to generate the coefficients

for a 2ond order low pass filter with a cut-off frequency of 300Hz.

|  |  |  |
| --- | --- | --- |
| Coefficient | Decimal | Fixed Point |
| X[n-2] |  |  |
| X[n-1] |  |  |
| X[n-0] |  |  |
| Y[n-1] |  |  |
| Y[n-2] |  |  |

Build a project around the VHDL code for this lesson. Plug the coefficients for your filter above into the right channel filter. Synthesize your design, download and listen to the difference between the left and right audio channels.

-------------------------------------------------------------------------------------

ac97: ac97\_wrapper

port map (

reset => reset, clk => clk,

ac97\_sdata\_out => SDATA\_OUT, ac97\_sdata\_in => SDATA\_IN,

ac97\_sync => SYNC, ac97\_bitclk => BIT\_CLK,

ac97\_n\_reset => AC97\_n\_RESET, ac97\_ready\_sig => ready,

L\_out => LdacValue , R\_out => RdacValue,

L\_in => LadcValue, R\_in => RadcValue);

left\_filter\_lpf1000: entity work.IIR\_Biquad(arch)

-- low pass 2nd order butterworth fl = 1000Hz, Fs = 48000Hz

-- http://www.earlevel.com/main/2013/10/13/biquad-calculator-v2/

generic map(Coef\_b0 => B"00\_00\_0000\_0100\_0000\_0010\_1001\_0110\_1101", -- +0.003916127

Coef\_b1 => B"00\_00\_0000\_1000\_0000\_0101\_0010\_1101\_1010", -- +0.007832253

Coef\_b2 => B"00\_00\_0000\_0100\_0000\_0010\_1001\_0110\_1101", -- +0.003916127

Coef\_a1 => B"10\_00\_1011\_1101\_0001\_0111\_0011\_1010\_0010", -- -1.815341083

Coef\_a2 => B"00\_11\_0101\_0010\_1111\_0011\_0010\_0001\_0001") -- +0.831005589

port map ( clk => clk,

n\_reset => reset,

sample\_trig => ready,

X\_in => LadcValue,

filter\_done => OPEN,

Y\_out => L\_filter\_lpf1000);

process (clk)

begin

if (rising\_edge(clk)) then

if reset = '0' then

LdacValue <= (others => '0'); RdacValue <= (others => '0');

elsif(ready = '1') then

if (filter\_switch = "00") then

LdacValue <= LadcValue; RdacValue <= RadcValue;

else

LdacValue <= L\_filter\_lpf1000; RdacValue <= R\_filter\_lpf1000;

end if;

end if;

end if;

end process;