Introduction to EECS II
6.082
Lecture 5

Filtering

- Filter Types
- Filtering using Difference Equations
- Examples

9/18/2006

Frequency Response

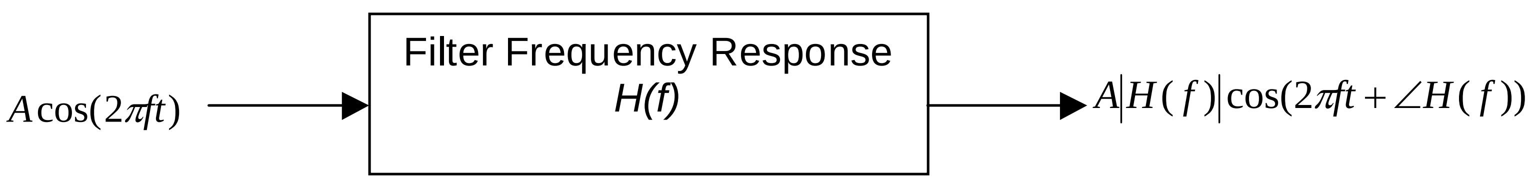
- Input sinusoid of amplitude *A* frequency *f*

$$A\cos(2\pi ft)$$

- Given a linear filter (obeys superposition)

$$\alpha x_1(t) + \beta x_2(t) = \alpha y_1(t) + \beta y_2(t)$$

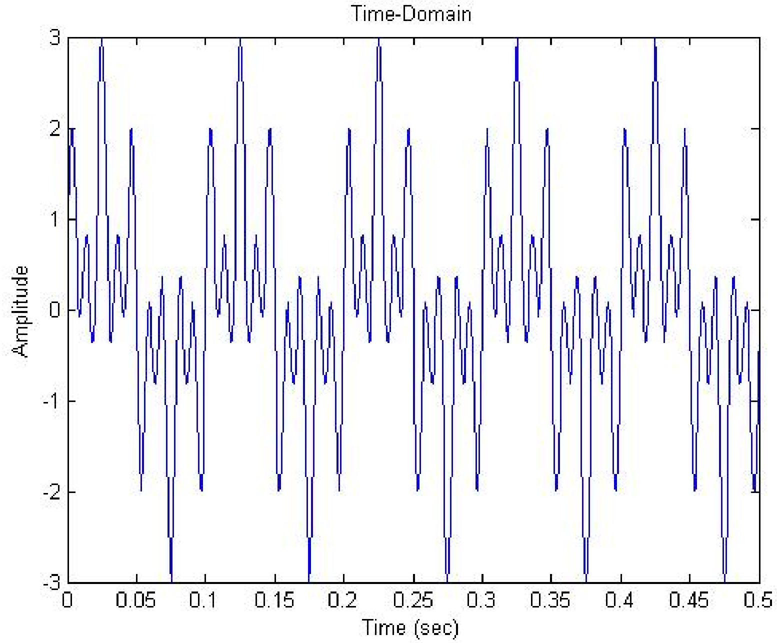
- Output sinusoid has amplitude scaled and phase shifted by the frequency response of the filter

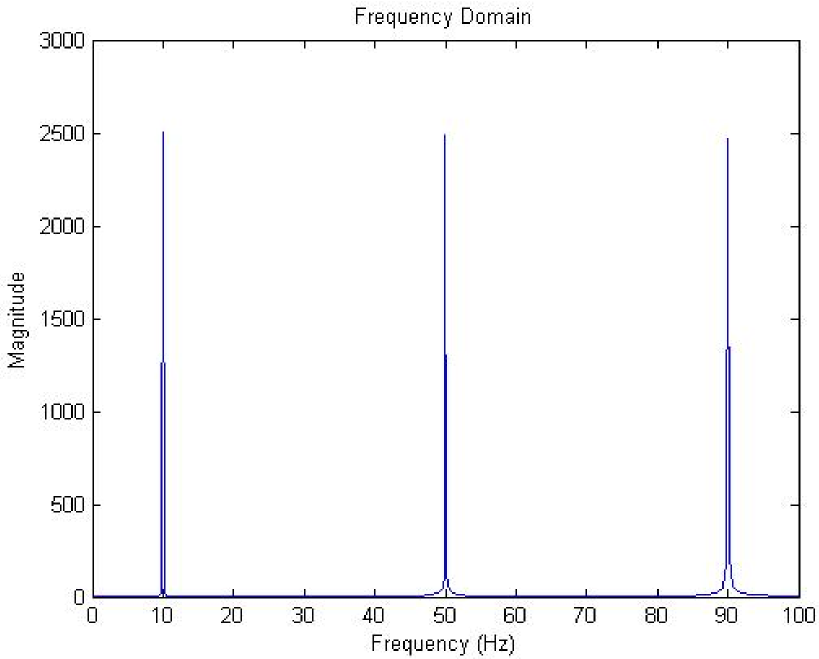


Example Input Signal to Filters

- Input signal - sum of 3 sinusoids
  - 10Hz, 50Hz, 90Hz

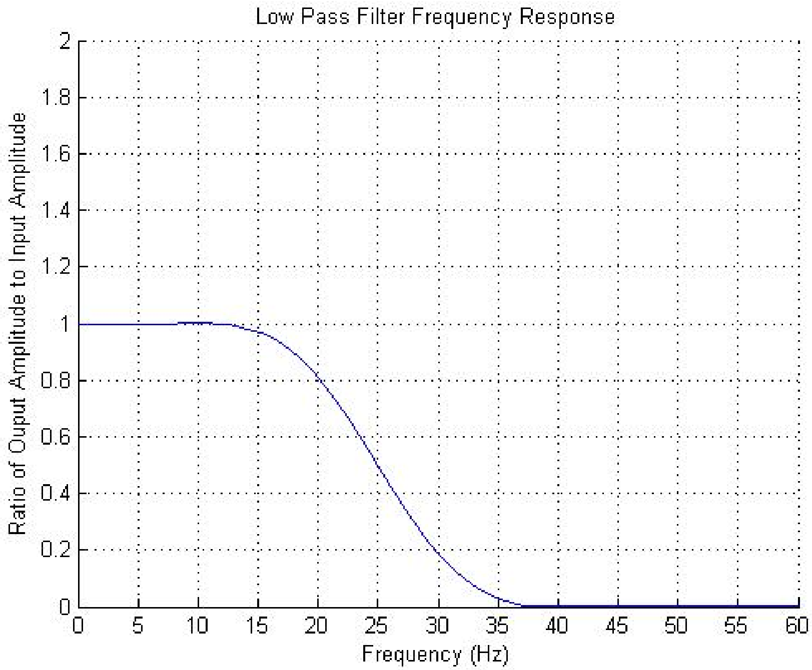
$$x(n) = \sin(2\pi * 10 * n * Ts) + \sin(2\pi * 50 * n * Ts) + \sin(2\pi * 90 * n * Ts)$$



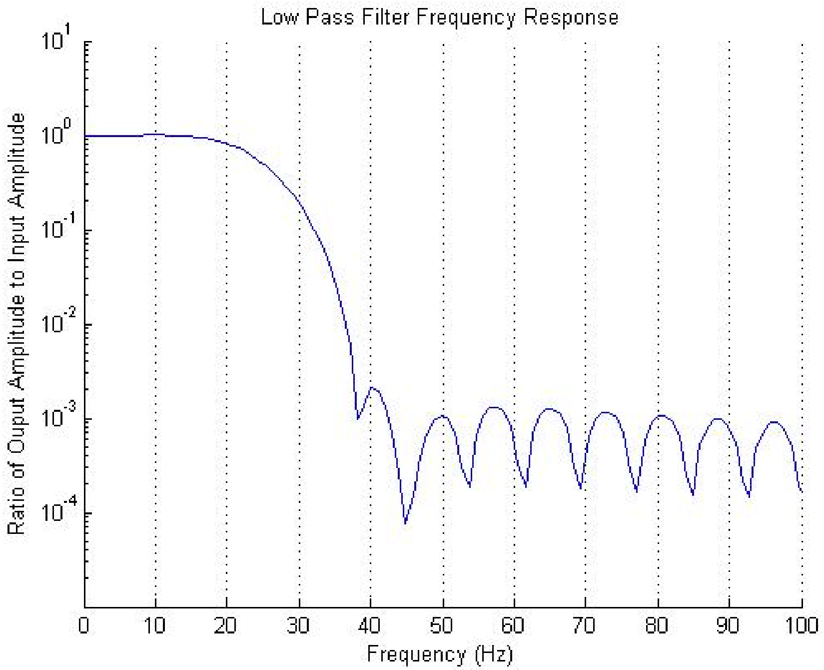


Low Pass Filter Example

- Magnitude of Frequency Response
  - Ratio of output to input amplitude
  - Ignore ripples that are less than 0.1%



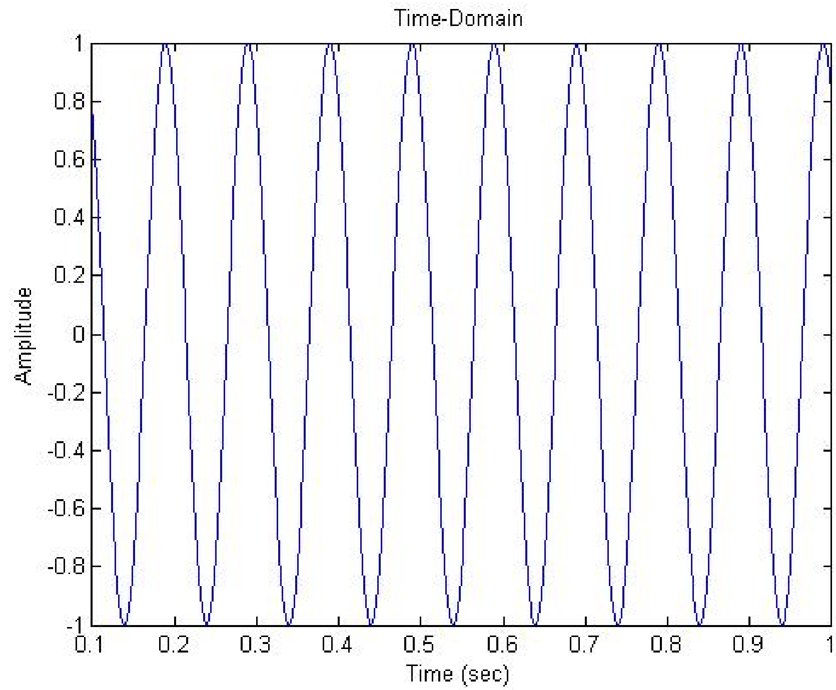
Linear Scale

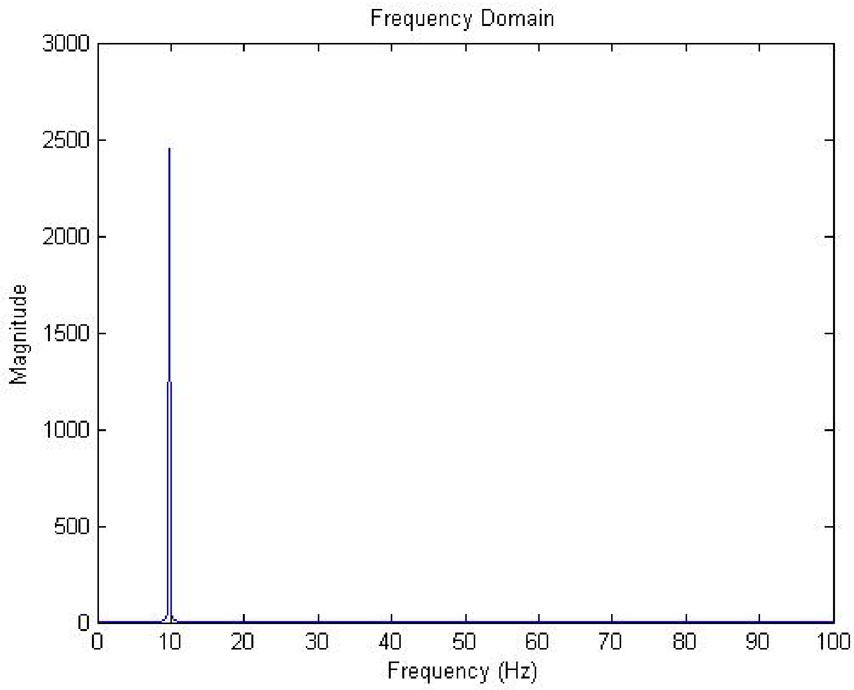


Log Scale

Low Pass Filter Example

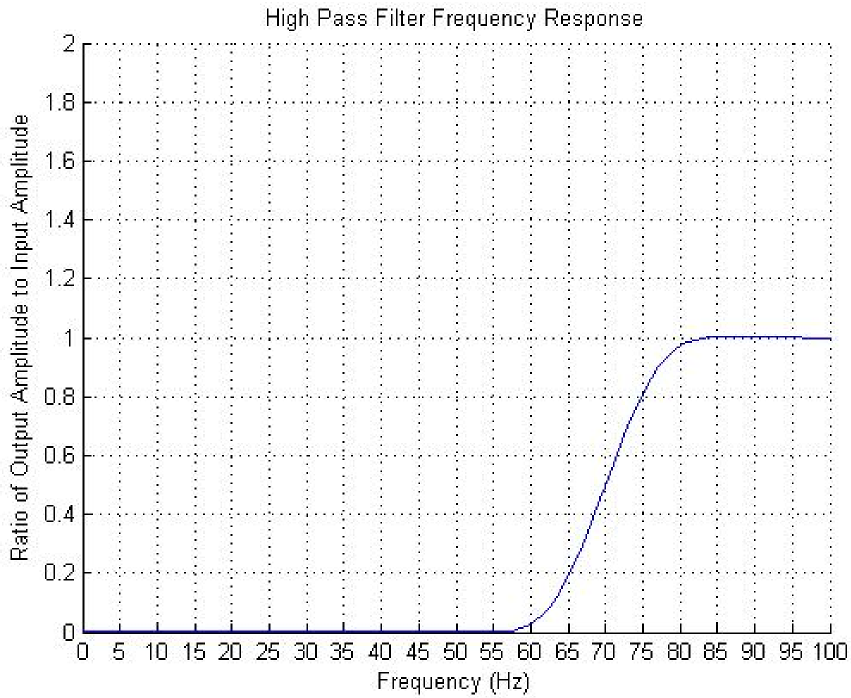
- Output signal in time and frequency domain shows that magnitude of 10Hz signal is nearly unaffected while 50Hz and 90Hz are attenuated.



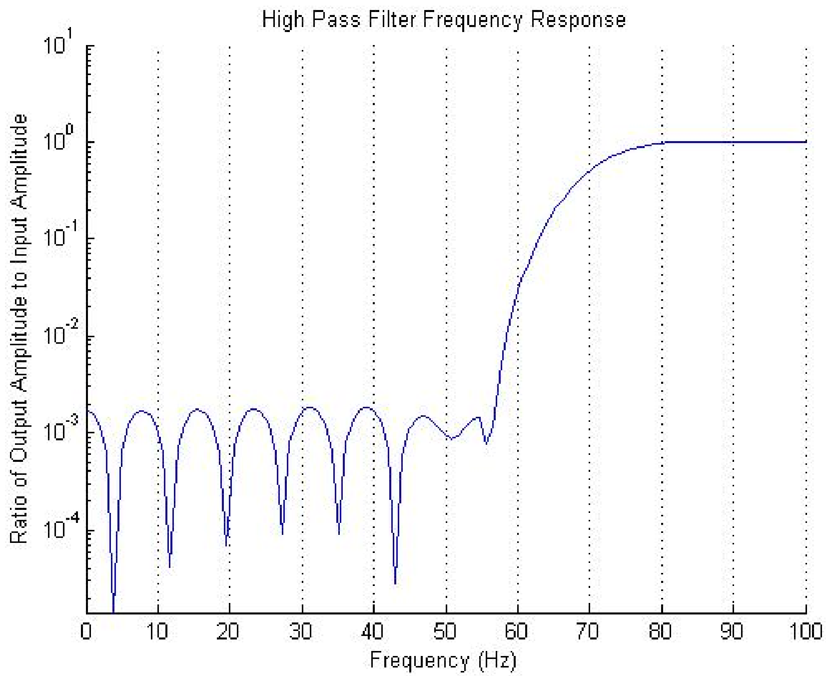


High Pass Filter Example

- Magnitude of Frequency Response
  - Ratio of output to input amplitude
  - Ignore ripples that are less than 0.1%



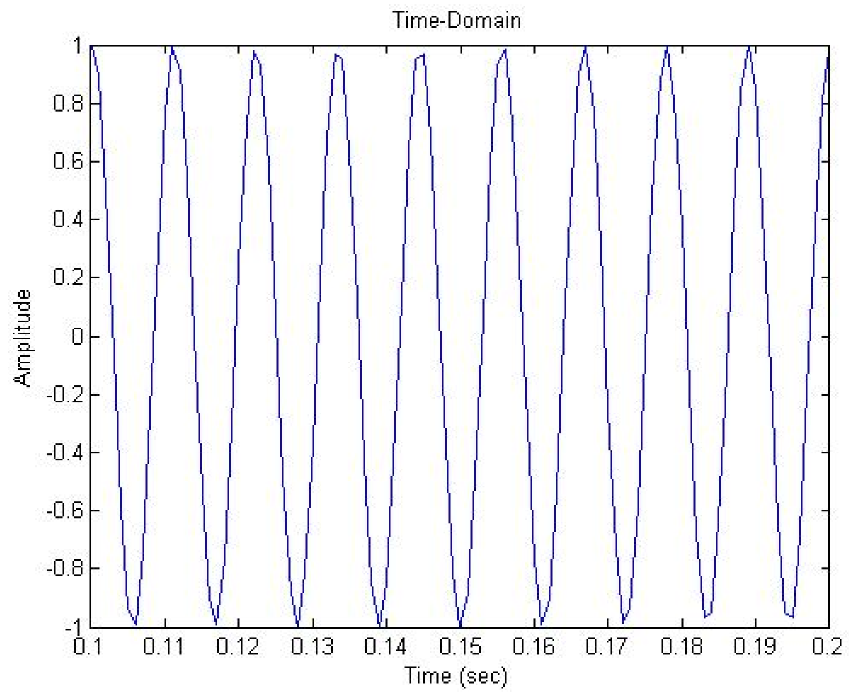
Linear Scale

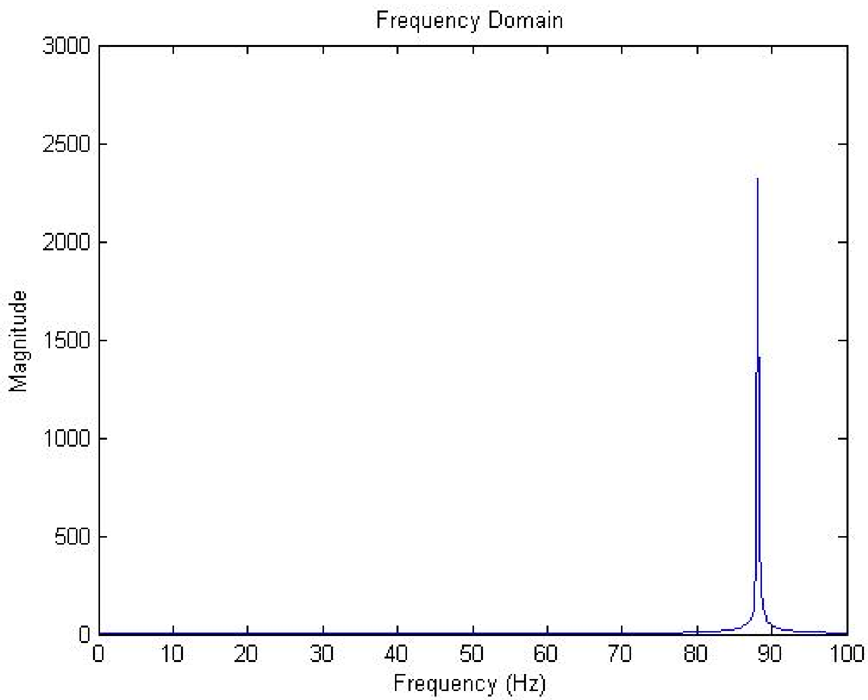


Log Scale

High Pass Filter Example

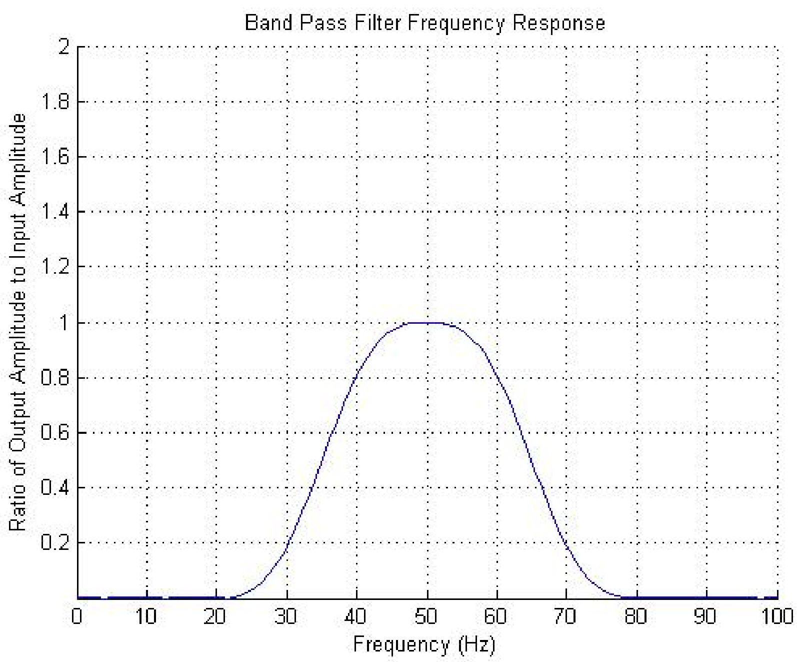
- Output signal in time and frequency domain shows that magnitude of 90Hz signal is nearly unaffected while 10Hz and 50Hz are attenuated.



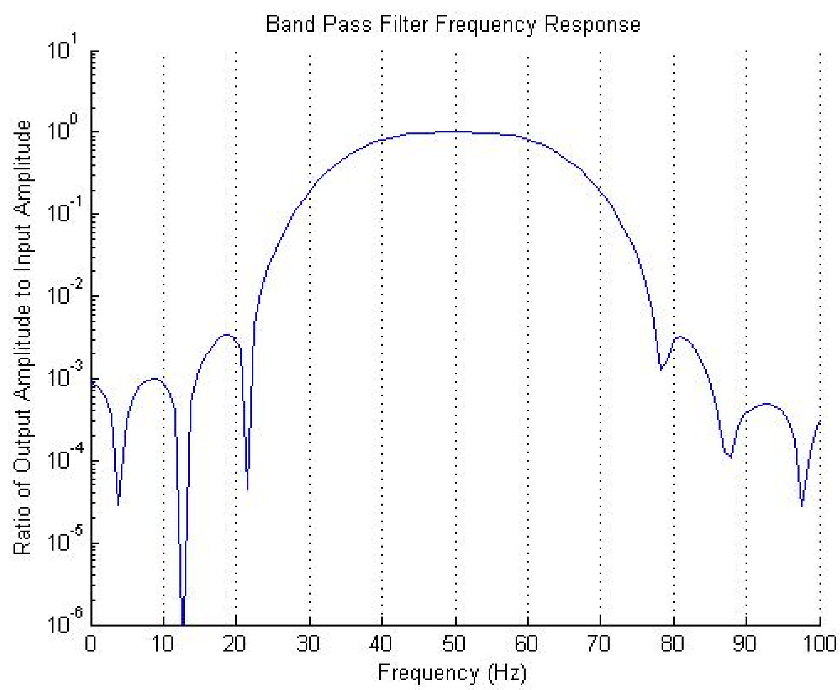


Band Pass Filter Example

- Magnitude of Frequency Response
  - Ratio of output to input amplitude
  - Ignore ripples that are less than 0.1%



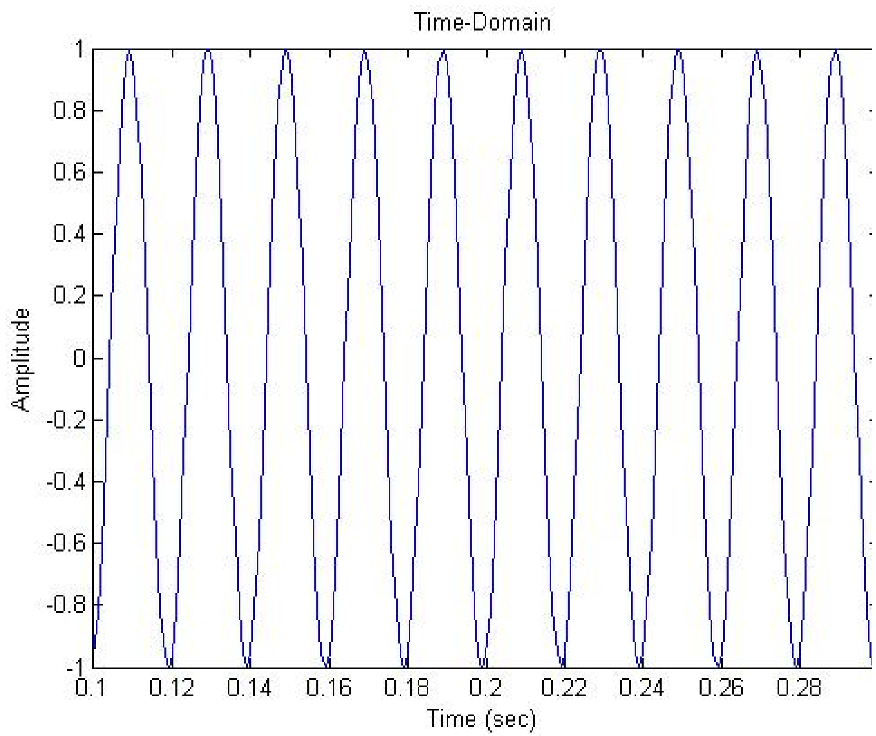
Linear Scale

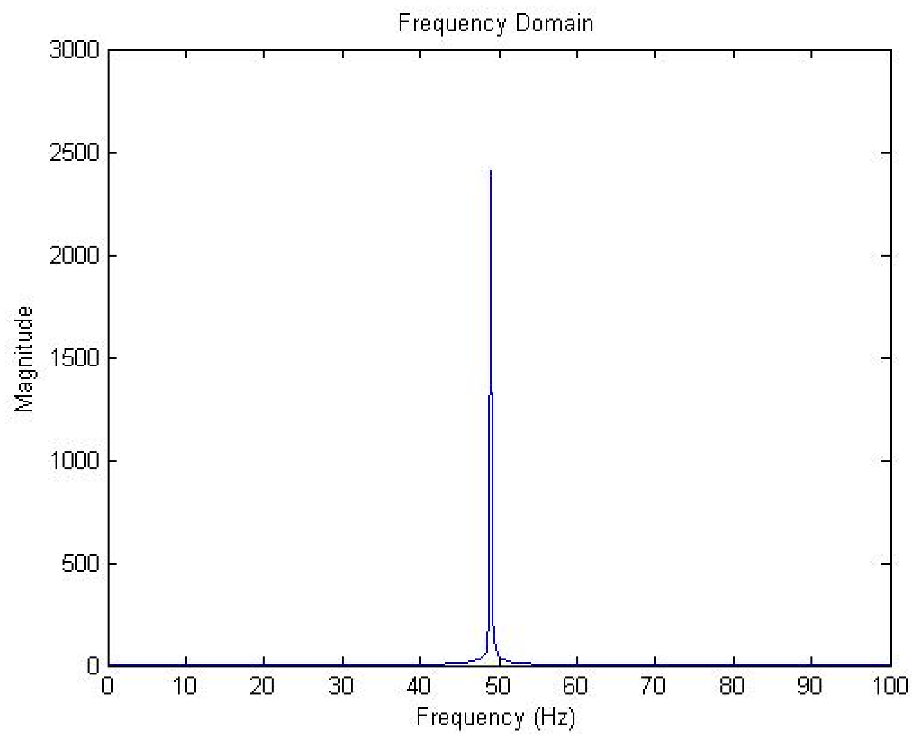


Log Scale

Band Pass Filter Example

- Output signal in time and frequency domain shows that magnitude of 50Hz signal is nearly unaffected while 10Hz and 90Hz are attenuated.





Difference Equation as LPF

- Let y(n) be the arithmetic average of 10 input samples x(n), x(n-1), ….x(n-9)

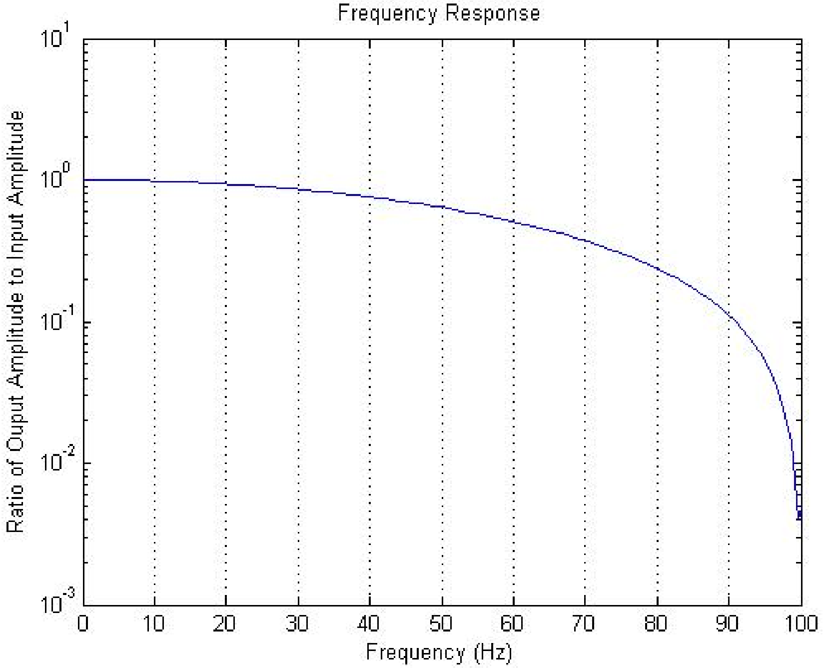
$$y(n) = \frac{1}{10} * x(n) + \frac{1}{10} * x(n-1) + \frac{1}{10} * x(n-2) + \ldots + \frac{1}{10} x(n-9)$$

$$y(n) = \sum_{i=0}^{9} \frac{1}{10} * x(n-i)$$

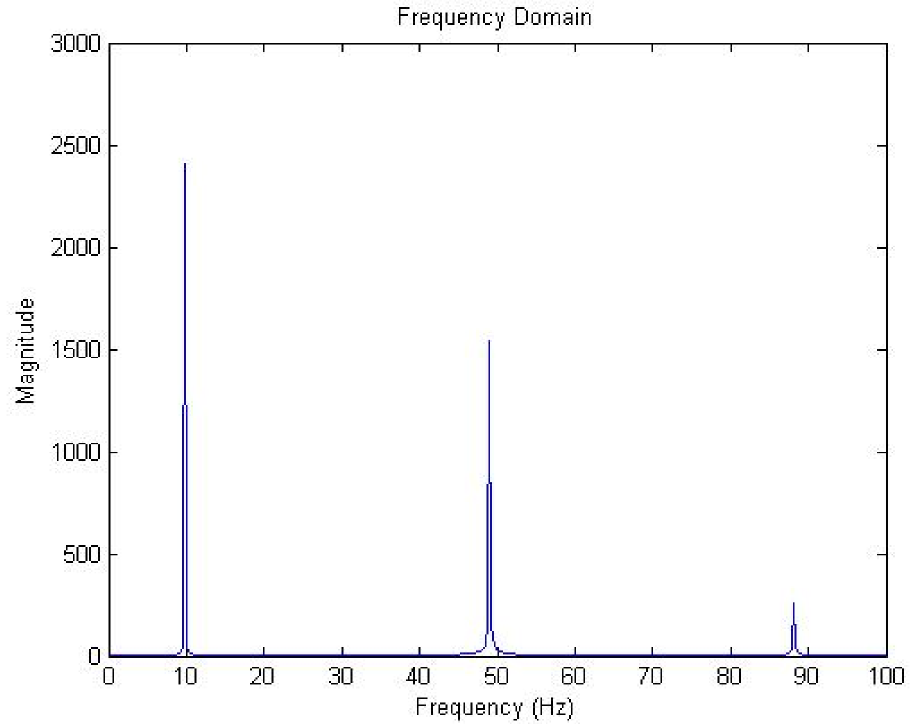
- Averaging operation let's slow changes in input pass to the output. Averaging is a form of low pass filtering

"Arithmetic Average" Response

- The magnitude of the frequency response attenuates 10Hz and does not attenuate 50Hz and 90 Hz very much



Log Magnitude Frequency Response

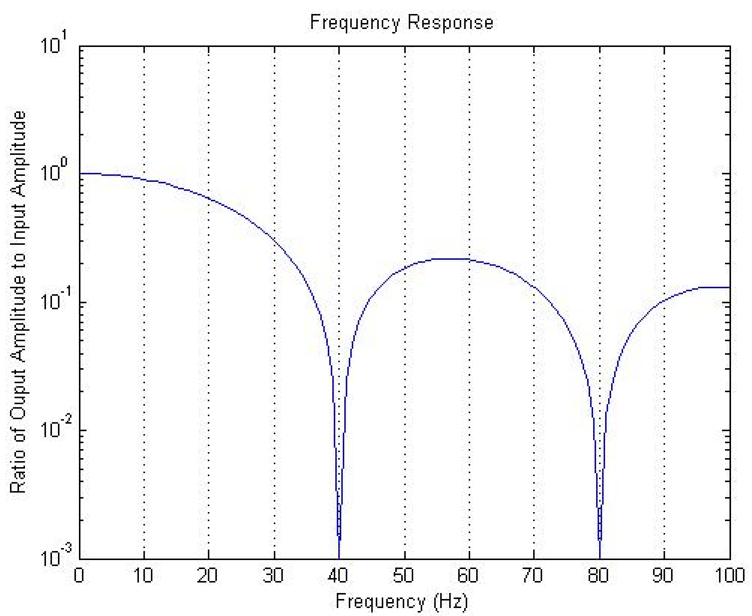


Magnitude of Output in Frequency Domain

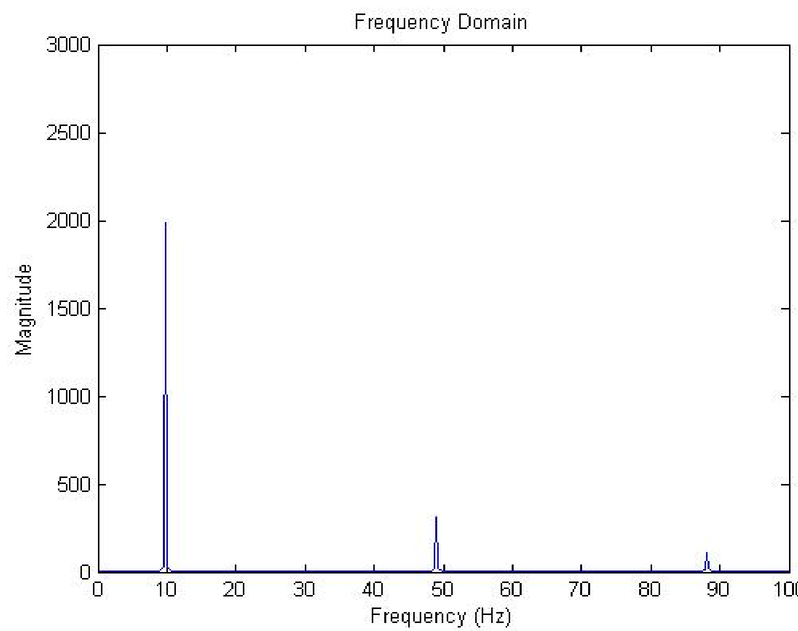
Increase "Order of Filter"

- The order of the filter corresponds to the number of coefficients in the difference equation.

$$y(n)=\frac{1}{35}*x(n)+\frac{1}{35}*x(n-1)+\frac{1}{35}*x(n-2)+\ldots+\frac{1}{35}x(n-34)==\sum_{i=0}^{34}\frac{1}{35}*x(n-i)$$



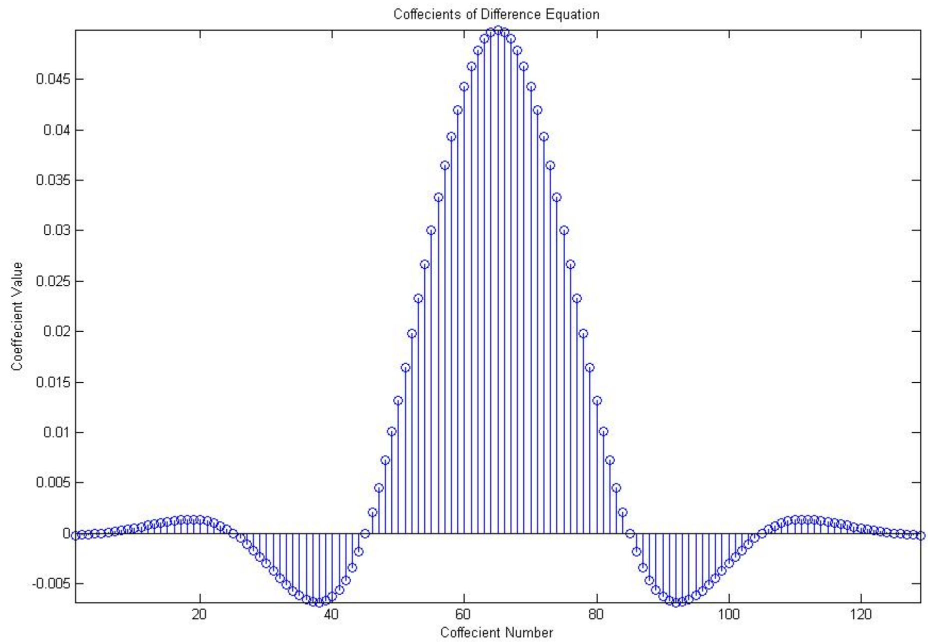
Log Magnitude Frequency Response

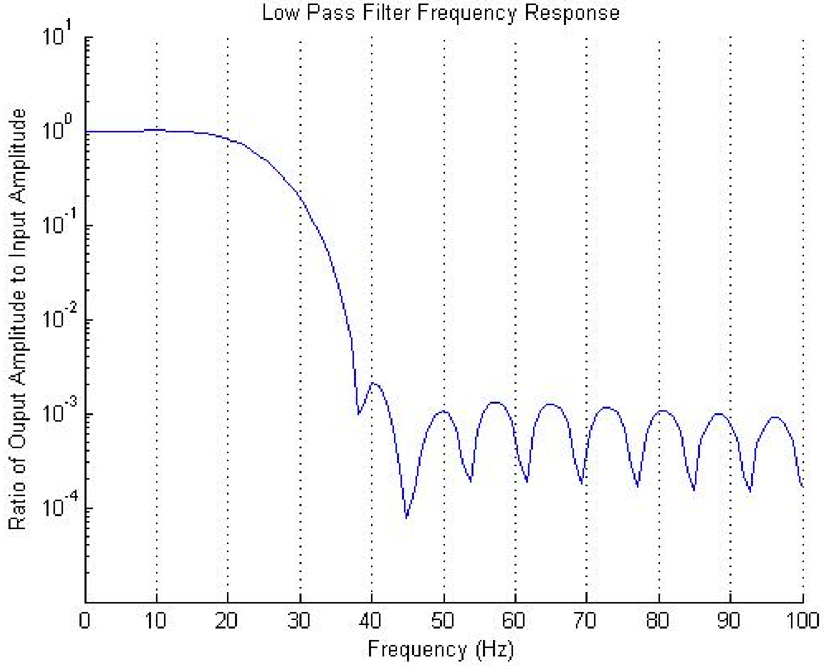


Magnitude of Output in Frequency Domain

Design of LPF

- Use higher order (costs computation)
- Coefficients need to be carefully chosen - Coefficient number 0 multiplies x(n); Coefficient number 1 multiplies x(n-1)
- Use MATLAB to determine coefficients for now and 6.003 to learn methods later

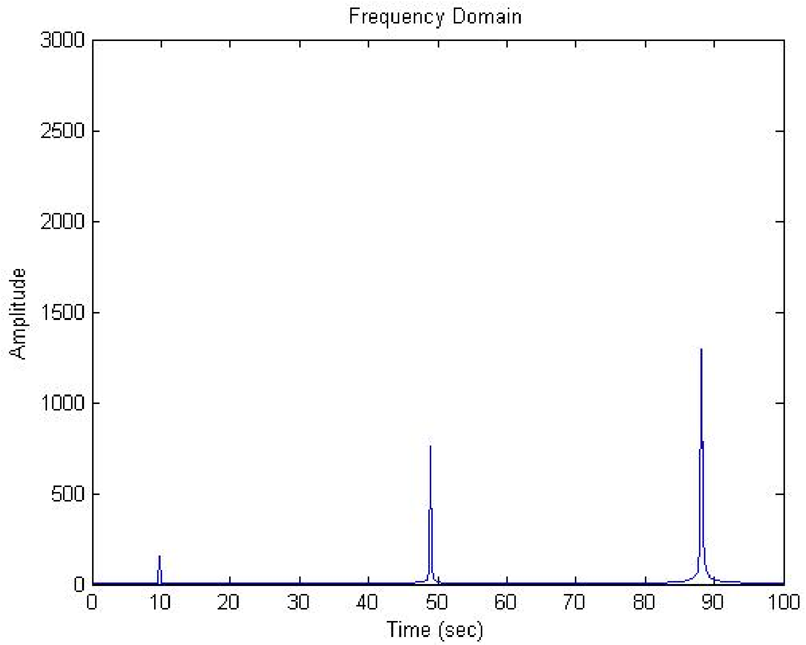




Difference Equation as HPF

$$y(n) = x(n) - x(n-1)$$

- y(n) is the difference between input samples x(n) and x(n-1)
- The difference operation prevents slow changes in the input x(n) (low frequencies) from passing to the output.



Frequency domain view of output

Filter Design with Difference Equations

- Value and number of b_i coefficients determine filter type and "frequency roll-off"
- Higher filter order yields better characteristics at the cost of computation or hardware.

$$y(n) = b_0 x(n) + b_1 x(n-1) + b_2 x(n-2) + \ldots + b_N x(n-N) = \sum_{i=0}^{N} b_i x(n-i)$$