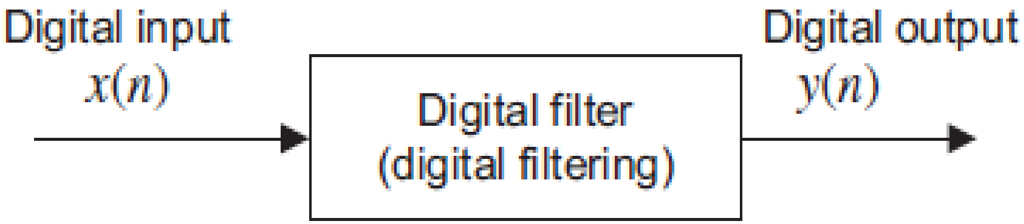
Chapter 6

Digital Filters

Digital Filtering: Realization

Definition: A digital filter is a DSP system, where $x(n)$ and $y(n)$ are the DSP system's input and output, respectively.



The Digital filter difference equation:

$$y(n) = \sum_{i=0}^{M} b_i x(n-i) - \sum_{j=1}^{N} a_j y(n-j)$$

Where a_i and b_i represent the coefficients of the system.

Matlab Implementation:

3-tap (2ndorder) IIR filter

```
>> B = [0 1];A = [1 0 -0.5];
>> x = [1 0.5 0.25 0.125];
>> y = filter(B, A, x)
   y =
   0   1.0000   0.5000   0.7500
```

Example

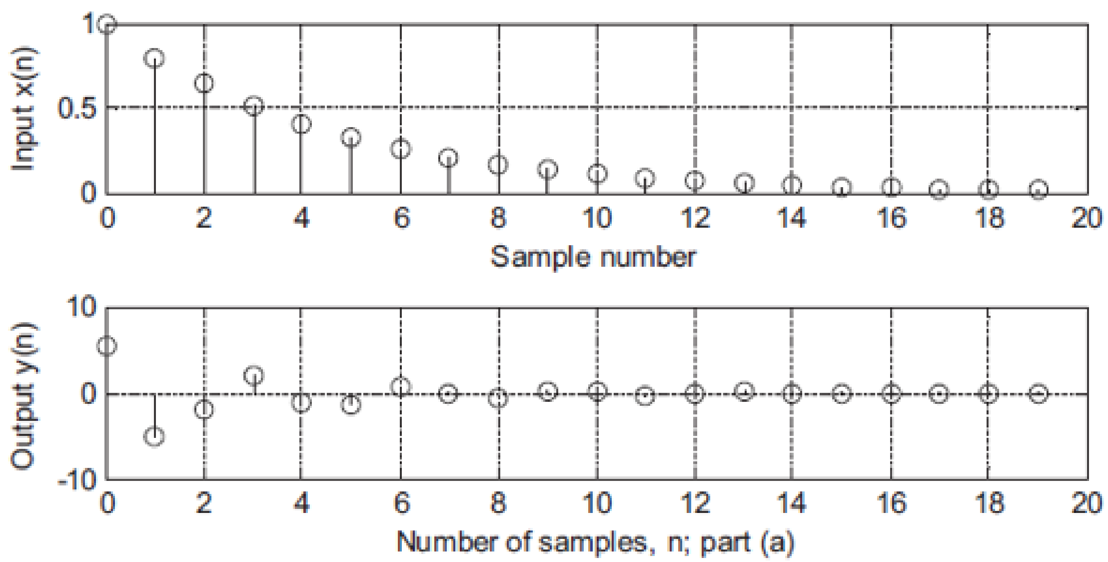
Given the DSP system $y(n) = 2x(n) - 4x(n-1) - 0.5y(n-1) - y(n-2)$

with initial conditions $y(-2) = 1, y(-1) = 0, x(-1) = -1,$ and the input $x(n) = (0.8)^n u(n),$

Compute the system response $y(n)$ for 20 samples using MATLAB.

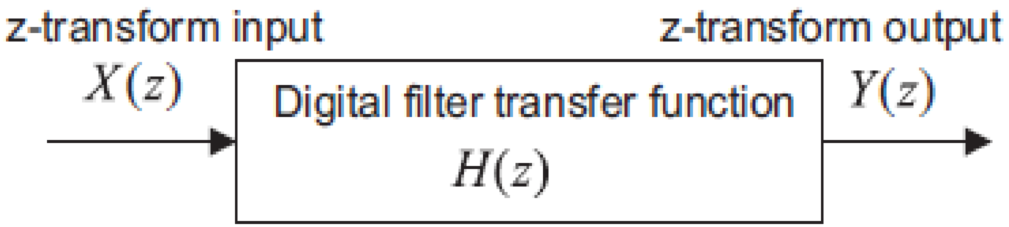
Solution:

```
% Compute y(n)=2x(n)-4x(n-1)-0.5y(n-1)-0.5y(n-2)
% Nonzero initial conditions:
% y(-2)=1, y(-1)=0, x(-1) =-1, and x(n)=(0.8)^n*u(n)
%
y = zeros(1,20);       % Set up a vector to store y(n)
y = [ 1 0 y];          % Add initial condition of y(-2) and y(-1)
n=0:1:19;              % Compute time indexes
x=(0.8).^n;            % Compute 20 input samples of x(n)

x = [ 0 -1 x];                                   % Add initial condition of
                                                   x(-2)=0 and x(-1)=1

for n=1:20
  y(n+2)= 2*x(n+2)-4*x(n+1)-0.5*y(n+1)-0.5*y(n); % Compute 20 outputs of y(n)
end
n=0:1:19;
subplot(3,1,1);stem(n,x(3:22));grid;ylabel('Input x(n)');xlabel('Sample number');
Subplot(3,1,2); stem(n,y(3:22)),grid;
xlabel('Number of samples, n; part (a)'); ylabel('Output y(n)');
```



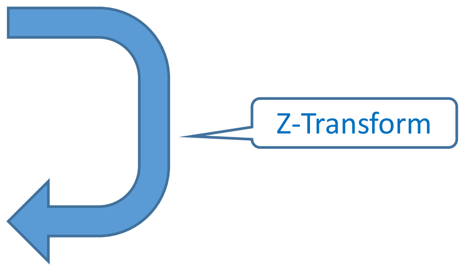
Transfer Function

If $x(n)$ and $y(n)$ are the input and the output of the Digital filter (DSP) respectively, and $X(z)$ and $Y(z)$ are their Z-transforms,



Differential Equation:

$$y(n) = b_0x(n) + b_1x(n-1) + \cdots + b_Mx(n-M) - a_1y(n-1) - \cdots - a_Ny(n-N)$$



Z-Transform:

$$Y(z) = b_0X(z) + b_1X(z)z^{-1} + \cdots + b_MX(z)z^{-M} - a_1Y(z)z^{-1} - \cdots - a_NY(z)z^{-N}$$

Transfer Function:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{b_0 + b_1z^{-1} + \cdots + b_Mz^{-M}}{1 + a_1z^{-1} + \cdots + a_Nz^{-N}} = \frac{B(z)}{A(z)}$$

Example: Transfer Function

Given a DSP: $y(n) = x(n) - x(n-2) - 1.3y(n-1) - 0.36y(n-2)$ Find the its transfer function H(z).

Z-Transform: $Y(z) = X(z) - X(z)z^{-2} - 1.3Y(z)z^{-1} - 0.36Y(z)z^{-2}$

Z-transform on both sides of the difference equation

Rearrange: $Y(z)(1 + 1.3z^{-1} + 0.36z^{-2}) = (1 - z^{-2})X(z)$

factoring Y(z) on the left side and X(z) on the right side

Transfer Function: $H(z) = \dfrac{Y(z)}{X(z)} = \dfrac{1 - z^{-2}}{1 + 1.3z^{-1} + 0.36z^{-2}}$

Given H(z) : $H(z) = \dfrac{z^2 - 1}{z^2 + 1.3z + 0.36}$ Find the difference equation of the system

Rearrange: $H(z) = \dfrac{(z^2 - 1)/z^2}{(z^2 + 1.3z + 0.36)/z^2} = \dfrac{1 - z^{-2}}{1 + 1.3z^{-1} + 0.36z^{-2}} = \dfrac{Y(z)}{X(z)}$

Dividing the numerator and denominator by z^2

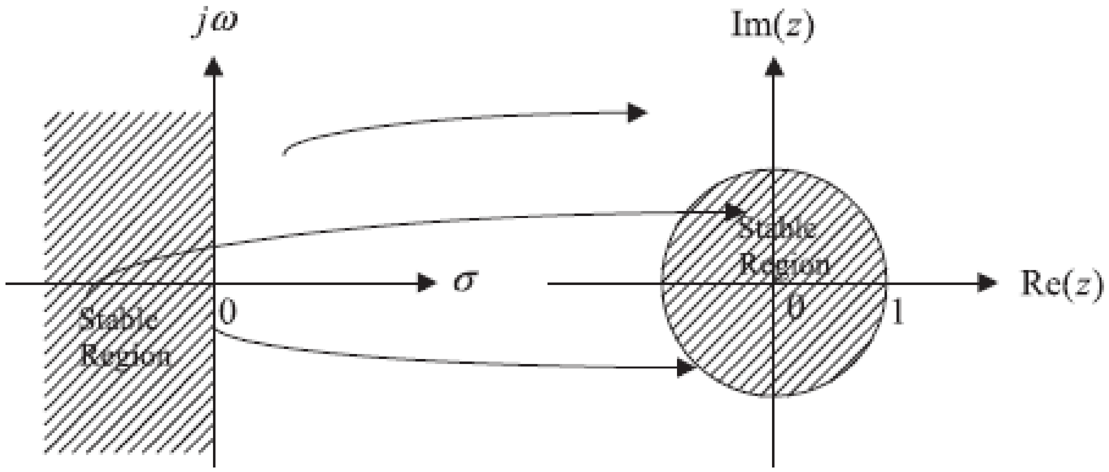
⇒ $Y(z)(1 + 1.3z^{-1} + 0.36z^{-2}) = X(z)(1 - z^{-2})$ ⇒ $Y(z) + 1.3z^{-1}Y(z) + 0.36z^{-2}Y(z) = X(z) - z^{-2}X(z)$

Differential Equation: $y(n) = x(n) - x(n-2) - 1.3y(n-1) - 0.36y(n-2)$

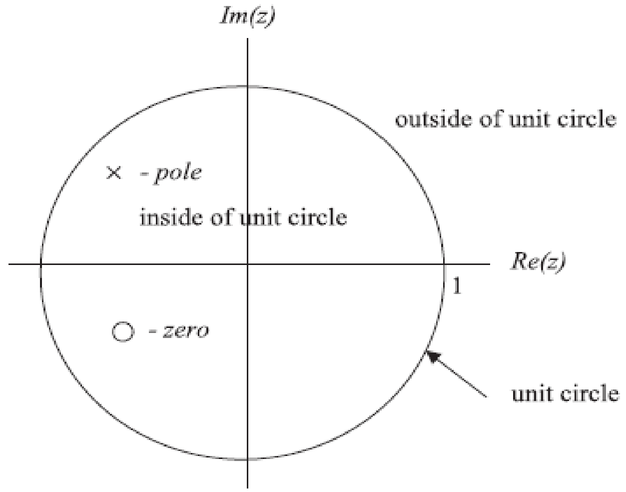
Applying the inverse z-transform and using the shift property

Pole –Zero from Transfer Function

- A digital transfer function can be written in the pole-zero form.
- The z-plane pole-zero plot is used to investigate characteristics and the stability of the digital system.
- Relationship of the sampled system in the Laplace domain and its digital system in the z-transform domain



mapping: $z = e^{sT}$



- The z-plane is divided into two parts by a unit circle.
- Each pole is marked on z-plane using the cross symbol x, while each zero is plotted using the small circle symbol o.

Example: Pole-zero plot

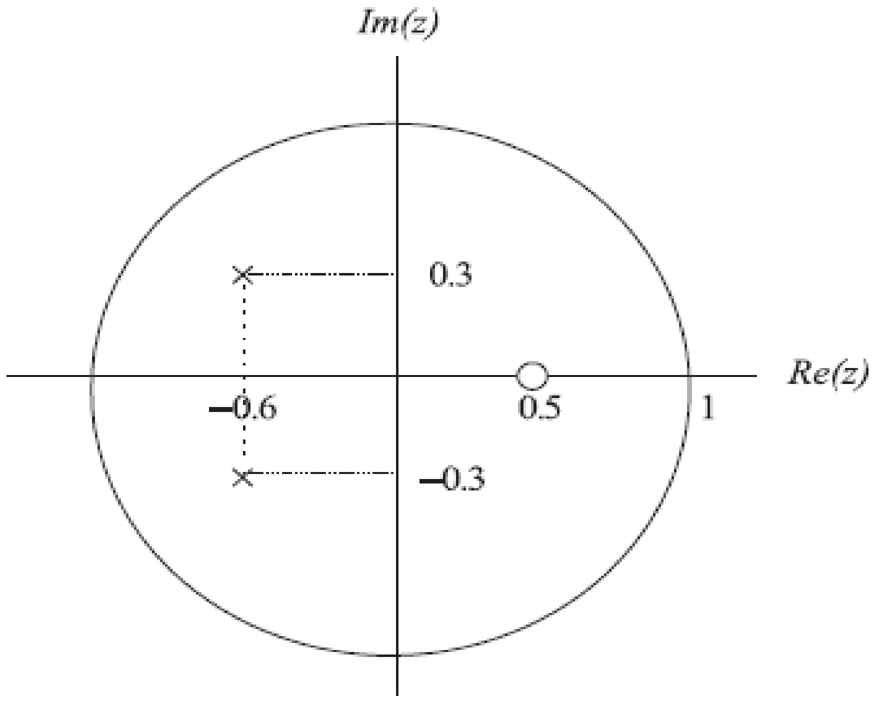
Given the digital transfer function: $H(z) = \dfrac{z^{-1} - 0.5z^{-2}}{1 + 1.2z^{-1} + 0.45z^{-2}}$ Plot poles and zeros

$$H(z) = \frac{(z^{-1} - 0.5z^{-2})z^2}{(1 + 1.2z^{-1} + 0.45z^{-2})z^2} = \frac{z - 0.5}{z^2 + 1.2z + 0.45}$$

multiplying the numerator and denominator by z^2

$$= \frac{(z - 0.5)}{(z + 0.6 - j0.3)(z + 0.6 + j0.3)}$$

$$= \frac{(z - zero1)}{(z - pole1)(z - pole2)} \Rightarrow \begin{cases} zero1 = 0.5 \\ pole1 = -0.6 + j0.3 \\ pole2 = -0.6 - j0.3 \end{cases}$$

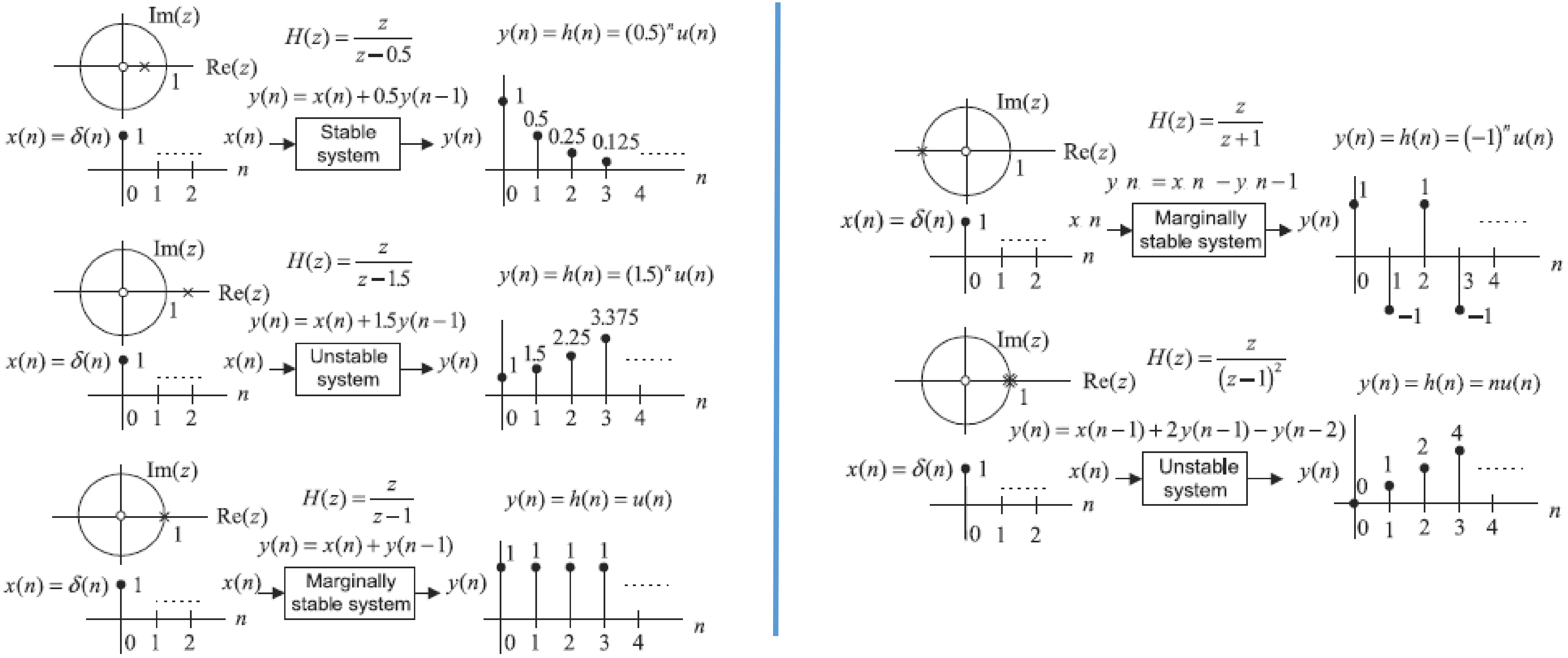


The system is stable.

The zeros do not affect system stability.

System Stability (Depends on poles' location)

- If the outmost poles of the DSP TF H(z) are inside the unit circle on the z-plane pole-zero plot, then the *system is stable*.
- If the outmost poles are first-order poles of the DSP TF H(z) and on the unit circle on the z-plane pole-zero plot, then the system is *marginally stable*.



Example: System Stability

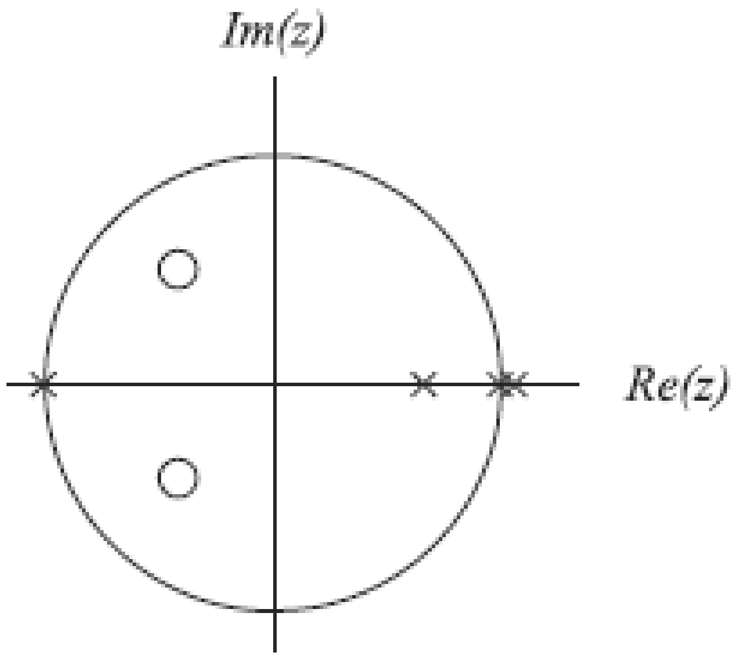
Sketch the z-plane pole-zero plot and determine the stability for the system:

$$H(z) = \frac{z^2 + z + 0.5}{(z-1)^2(z+1)(z-0.6)}$$

Zeros are $z = -0.5 \pm j\,0.5$.

Poles: $z = 1,\ |z| = 1;\ z = 1,\ |z| = 1;\ z = -1,\ |z| = 1;,\ z = 0.6\ \ |z| = 0.6 < 1.$

Since the outermost pole is multiple order (2nd order) at z = 1 and is on the unit circle, the *system is unstable*.



Digital Filter: Frequency Response

- From the Laplace transfer function, we can achieve the analog filter frequency response $H(j\omega)$ by substituting $s = j\omega$ into the transfer function H(s).

$$H(s)\big|_{s=j\omega} = H(j\omega)$$

- Similarly, in a DSP system, we substitute $z = e^{sT}\big|_{s=j\omega} = e^{j\omega T}$ into the Z-transfer function of the system's transfer function H(z) to acquire the digital frequency response

$$H(z)\big|_{z=e^{j\omega T}} = H(e^{j\omega T}) = \left|H(e^{j\omega T})\right| \angle H(e^{j\omega T})$$

Magnitude frequency response

Phase response

Putting $\Omega = \omega T$

normalized digital frequency

$$H(e^{j\Omega}) = H(z)\big|_{z=e^{j\Omega}} = \left|H(e^{j\Omega})\right| \angle H(e^{j\Omega})$$

Frequency Response Example

Problem Given the digital system with a sampling rate of 8,000 Hz, determine the frequency response.

$$y(n) = 0.5x(n) + 0.5x(n-1)$$

Solution

z-transform : (on both sides on the difference equation)

$$Y(z) = 0.5X(z) + 0.5z^{-1}X(z)$$

Frequency response:

transfer function →

$$H(z) = \frac{Y(z)}{X(z)} = 0.5 + 0.5z^{-1}$$

$z = e^{j\Omega}$ →

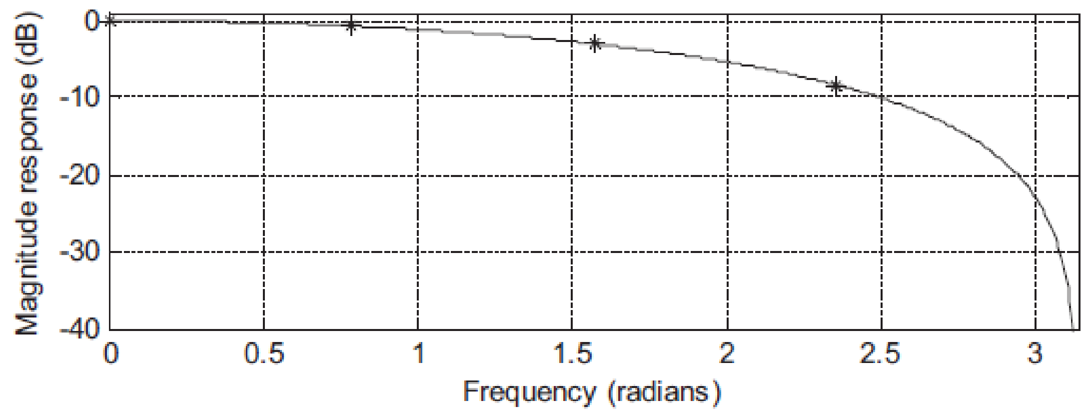
$$H(e^{j\Omega}) = 0.5 + 0.5e^{-j\Omega}$$
$$= 0.5 + 0.5\cos(\Omega) - j0.5\sin(\Omega).$$

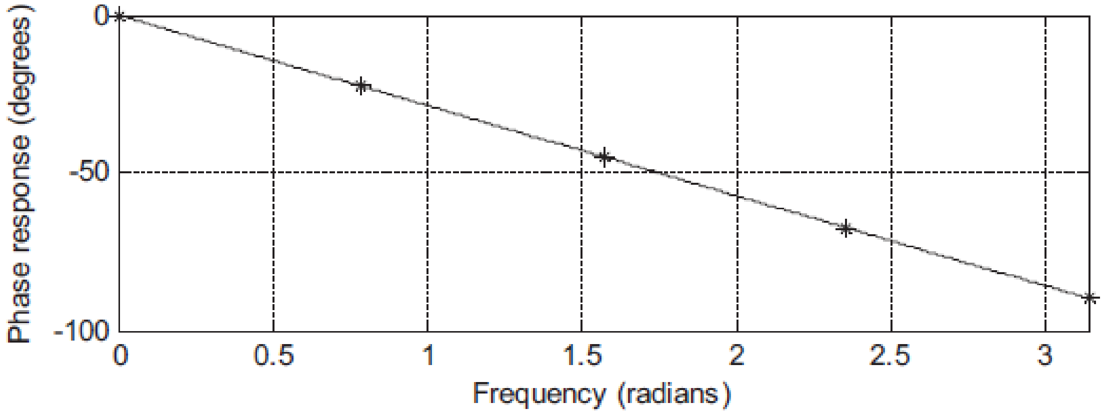
Magnitude frequency response →

$$\left|H(e^{j\Omega})\right| = \sqrt{(0.5 + 0.5\cos(\Omega))^2 + (0.5\sin(\Omega))^2}$$

Phase frequency response →

$$\angle H(e^{j\Omega}) = \tan^{-1}\left(\frac{-0.5\sin(\Omega)}{0.5 + 0.5\cos(\Omega)}\right)$$

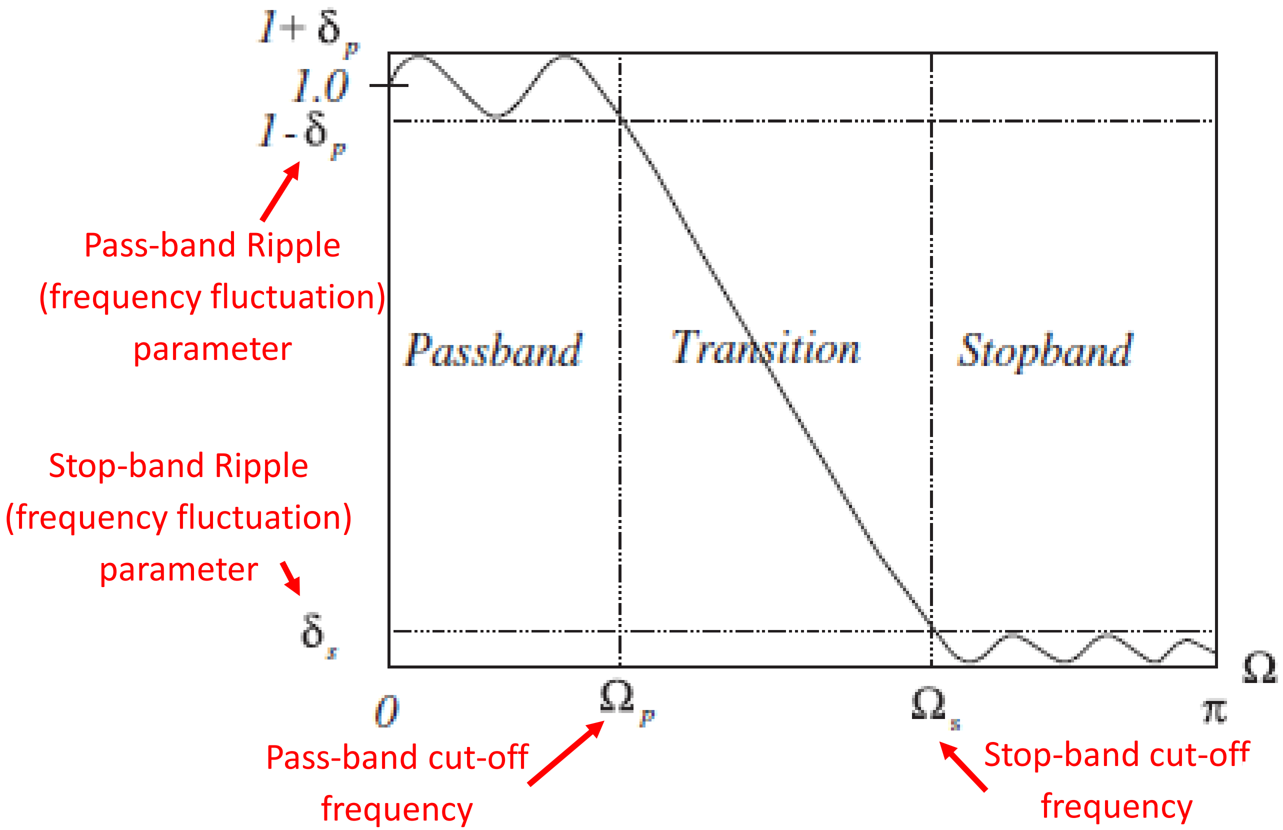




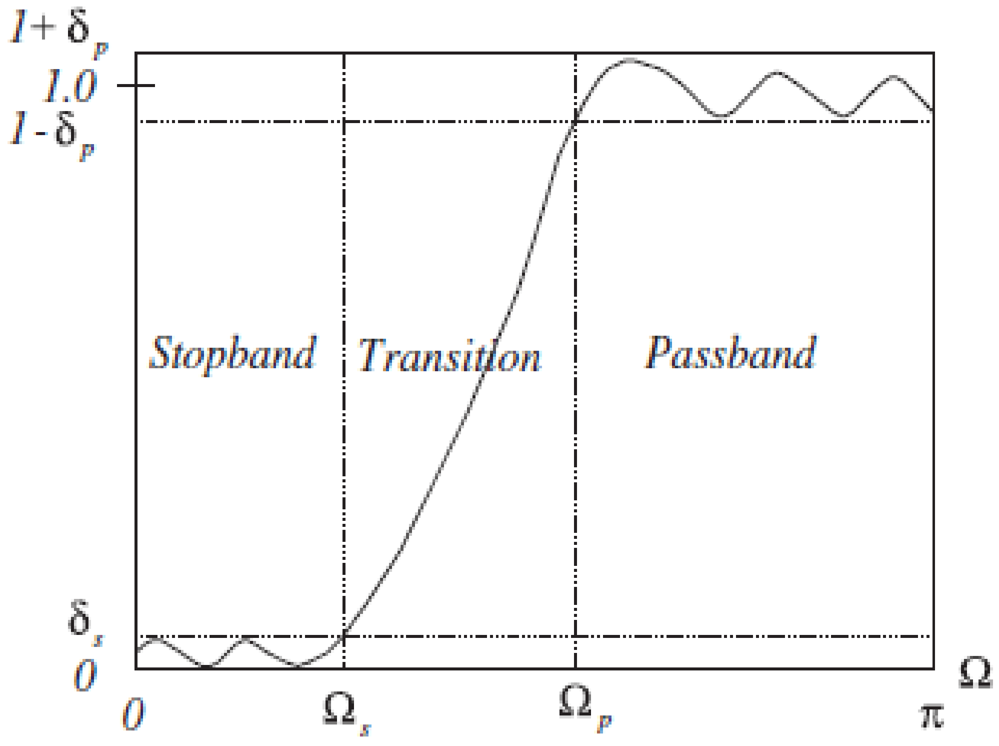
It is observed that when the frequency increases, the magnitude response decreases. The DSP system acts like a digital low-pass filter, and its phase response is linear.

Digital Filter: Frequency Response -contd.

BASIC TYPES OF FILTERING



Low-pass filter (LPF)



High-pass filter (HPF)

Matlab: Frequency Response: **[h, w] = freqz(B, A, N)**

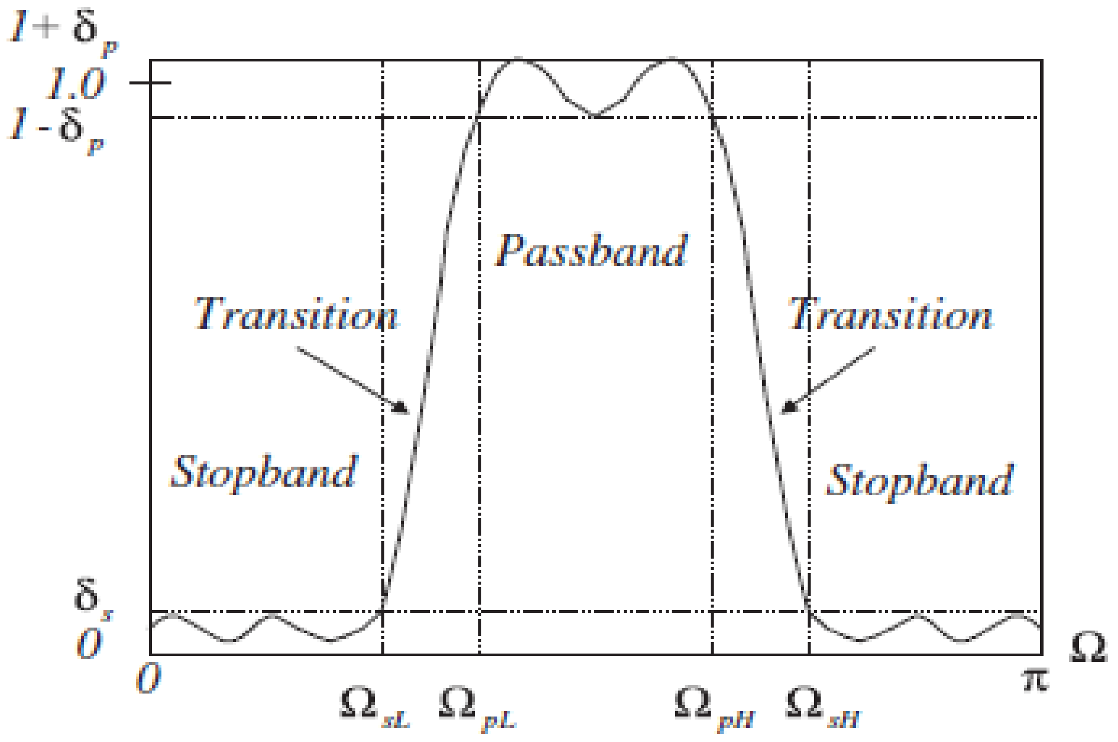
[h, w] = freqz(B, A, N)

where the parameters are defined as follows:

h = an output vector containing frequency response
w = an output vector containing normalized frequency values distributed in the range from 0 to π radians
B = an input vector for numerator coefficients
A = an input vector for denominator coefficients
N = the number of normalized frequency points used for calculating the frequency response

Digital Filter: Frequency Response -contd.

BASIC TYPES OF FILTERING



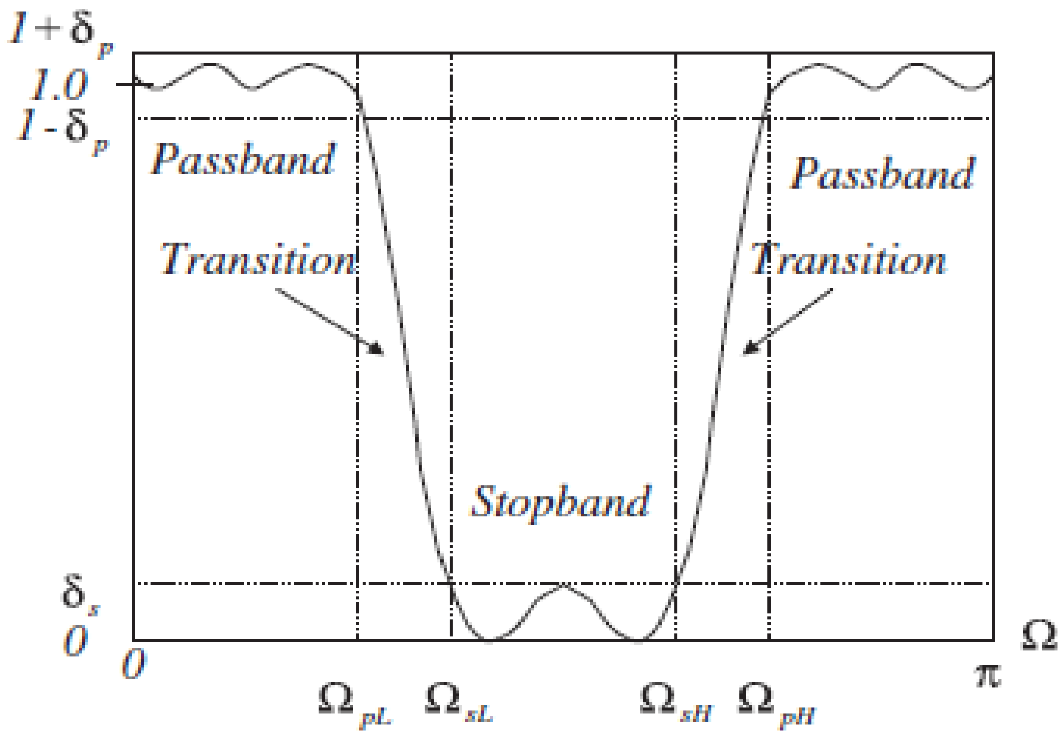
Lower stop-band cut-off frequency

Lower Pass-band cut-off frequency

Higher pass-band Cut-off frequency

Higher stop-band Cut-off frequency

Band-pass filter (BPF)



Band-stop filter (BSF)