Phase 2

Topics

- Overview of discrete time signals and systems
- Generation and synthesis of basic sequences
- Convolution
- Response from system difference equation
- Correlation

Correlation:

- **Autocorrelation** is a mathematical tool used frequently in signal processing for analysing functions or series of values, such as time domain signals.
- Informally, it is a measure of how well a signal matches a time-shifted version of itself, as a function of the amount of time shift. More precisely, it is the cross-correlation of a signal with itself.
- Autocorrelation is useful for finding repeating patterns in a signal, such as determining the presence of a periodic signal which has been buried under noise, or identifying the missing fundamental frequency in a signal implied by its harmonic frequencies.

Correlation:

- The discrete-time autocorrelation R at lag j for a discrete signal $x(n)$ [real/complex] is

$$\boxed{R_{xx}(j) = \sum_{n} x(n)x^{*}(n - j)}$$

✓ The above definitions work for signals that are square integrable, or square summable, that is, of finite energy.

- Signals that "last forever" are treated instead as random processes, in which case different definitions are needed, based on expected values. For wss random processes , the autocorrelations are defined as

$$\boxed{\begin{aligned} R_{xx}(j) &= E[x(n)x^{*}(n - j)] \\ &= \underset{N \to \infty}{Lt} \frac{1}{2N + 1} \sum_{n=-N}^{N} x(n)x^{*}(n - j) \end{aligned}}$$

Correlation:

- For processes that are also ergodic, the expectation can be replaced by the limit of a time average. The autocorrelation of an ergodic process is sometimes defined as or equated to

$$\boxed{R_{xx}(j) = \underset{N \to \infty}{Lt} \frac{1}{N} \sum_{N} x(n) x^*(n - j)}$$

- The central value of an autocorrelation function equals the mean square value of the sequence, and is therefore a measure of its total power.

$$\boxed{R_{xx}(0) = E[x(n) x^*(n)] = E[x^2(n)]}$$

Correlation:

- Similarly the cross-correlation function (CCF) of two sequences x[n] and y[n] can be given as

$$R_{yx}(j) = E[y(n)x^*(n-j)]$$

$$= \underset{N \to \infty}{Lt} \frac{1}{2N+1} \sum_{n=-N}^{N} y(n)x^*(n-j)$$

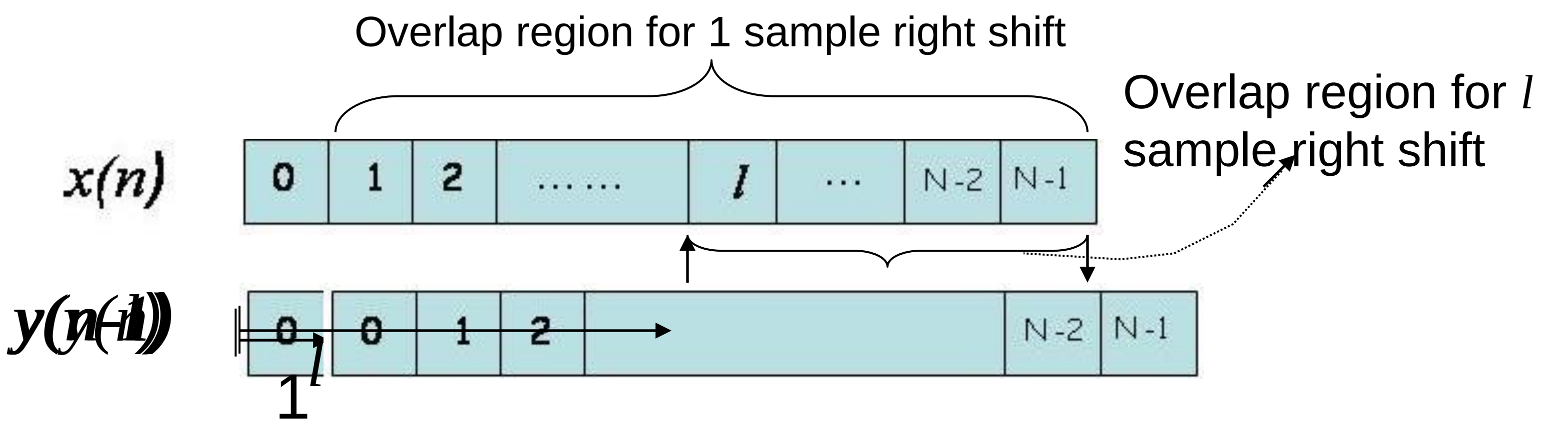
- If x(n)=y(n)=0 for n<0 and n>N-1, above equation turns into

$$\boxed{r_{xy}(l) = \sum_{n=i}^{N-|k|-1} x(n)y(n-l)}$$

Clarification is next

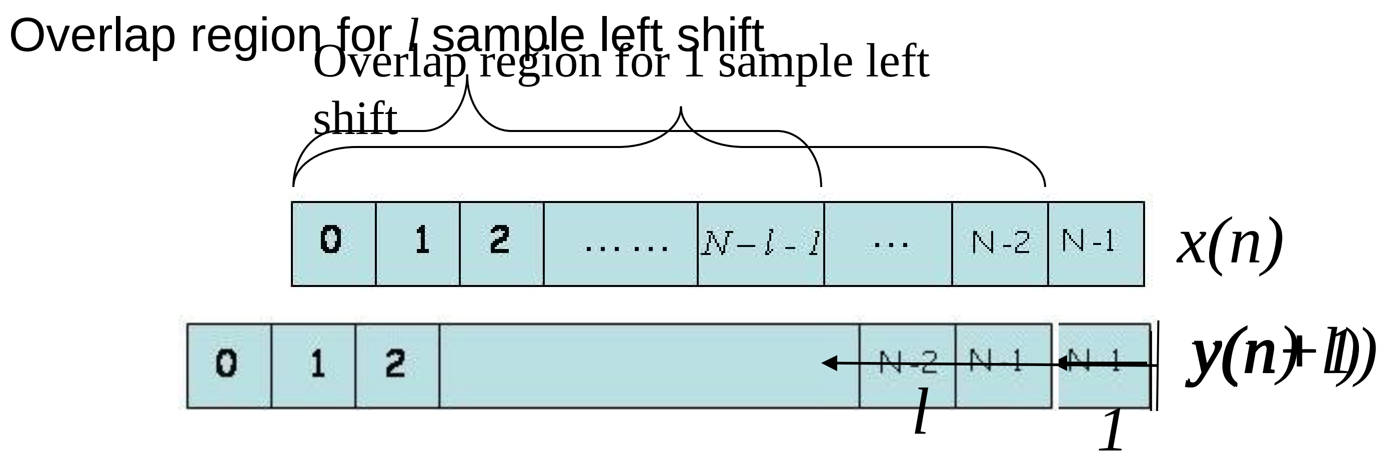
- where i = l, k = 0 for l ≥ 0 and i = 0 ,k = l for l < 0

Correlation:



$$r_{xy}(l) = \sum_{n=i}^{N-|k|-1} x(n)y(n-l) \qquad \text{where } i = l,\ k = 0 \text{ for } l \geq 0$$

Correlation:



$$r_{xy}(l) = \sum_{n=i}^{N-|k|-1} x(n)\,y(n-l) \qquad \text{where } i=0,\ k=l \text{ for } l<0$$

Correlation:

The steps of cross-correlation is summarized as:

1. Shift one of the signals along the time axis by one sample.
2. Multiply the corresponding values of the two signals.
3. Summate the products from step 2 to get one point of the correlation sequence.
4. Repeat steps 1-3 to obtain the total correlation sequence at all times that the signals overlap.

Correlation:

- **Convolution and Correlation are same except for the flip**
- CONVOLUTION:

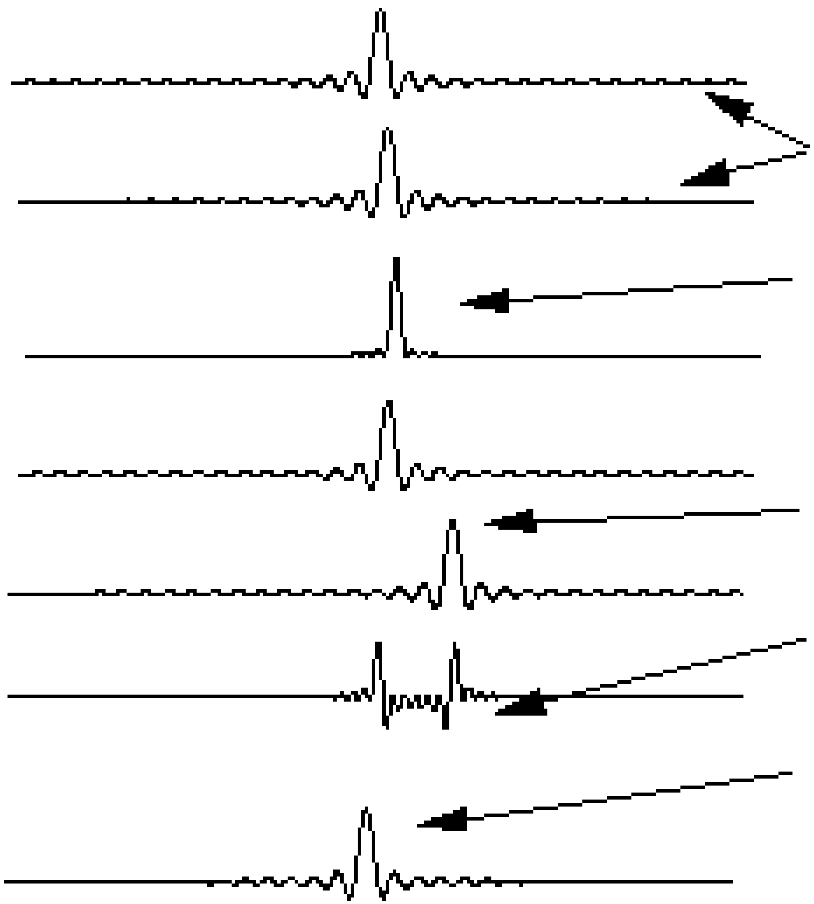
$$x(n) * y(n) = \sum_{n=-\infty}^{\infty} x(n)y(k-n)$$

- CORRELATION:

$$r_{xy}(k) = \sum_{n=-\infty}^{\infty} x(n)y(n-k) = x(n) * y(-n)$$

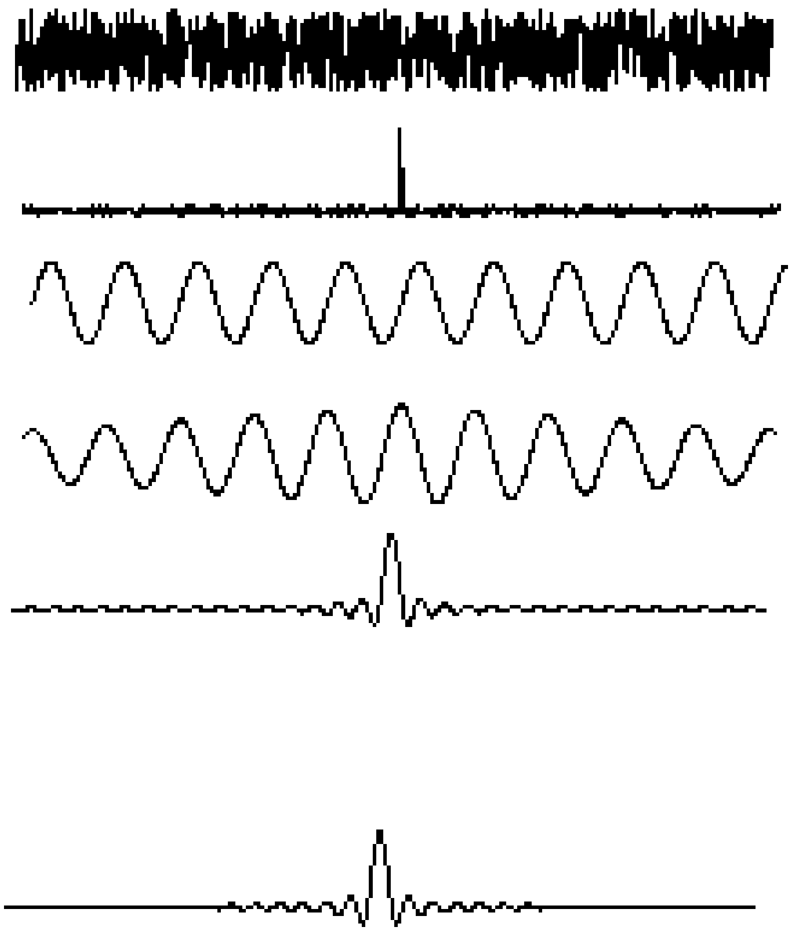
- we can compute Cross-correlation by **conv**() function too
- r_{xy}=conv(x,fliplr(y));
- % Here **fliplr**() function folds y.

Correlation:



- If two signals are similar and unshifted...
- their product is all positive.
- But as the shift increases...
- parts of it become negative...
- and the correlation function shows where the signals are similar and unshifted

Correlation:



- Random noise is similar to itself, and in phase, only with no time shift at all
- so its correlation function is a spike
- periodic signals go in and out of phase as they are time shifted
- so their correlation functions are periodic
- signals that last only a short while are only similar while they last
- so their correlation functions are short

Correlation:

- **Signals in noise**

One of the most important topics in digital signal processing concerns the extraction of wanted signals from unwanted noise. When a real signal, contaminated by noise, is to be recovered or detected, a useful way of detecting it is by using autocorrelation.

$$y(n) = s(n) + q(n)$$

$$R_{yy}(m) = E\{[s(n) + q(n)][s(n - m) + q(n - m)]\}$$

$$= E\{s(n)s(n - m)\} + E\{s(n)q(n - m)\} + E\{q(n)s(n - m)$$

$$+ E\{q(n)q(n - m)\}$$

Correlation:

- **Signals in noise**

The periodic signal s[n] and noise q[n] are completely uncorrelated to each other.

$$E\{s(n)q(n-m)\} = 0$$

Hence, $$R_{yy}(m) = R_{ss}(m) + R_{qq}(m)$$

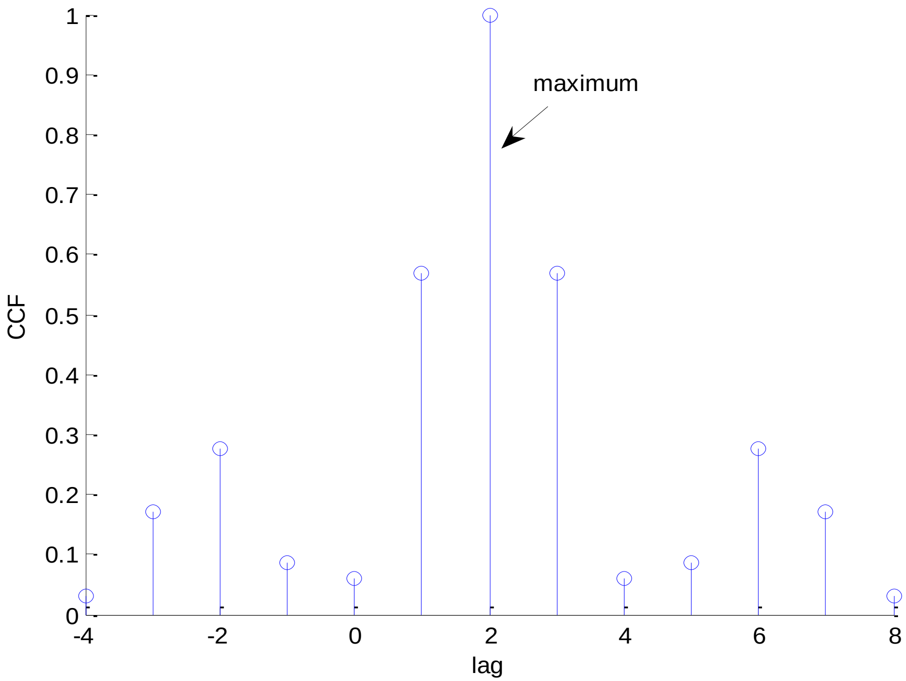
This is the Principle of Superposition that states the ACF is composed of the individual ACF's of both the signal and noise, *providing that signal and noise are uncorrelated*. This is an extremely important relationship, which is often used to detect the signal from the unwanted noise.

Correlation:

- **PART-D : EXAMPLE**

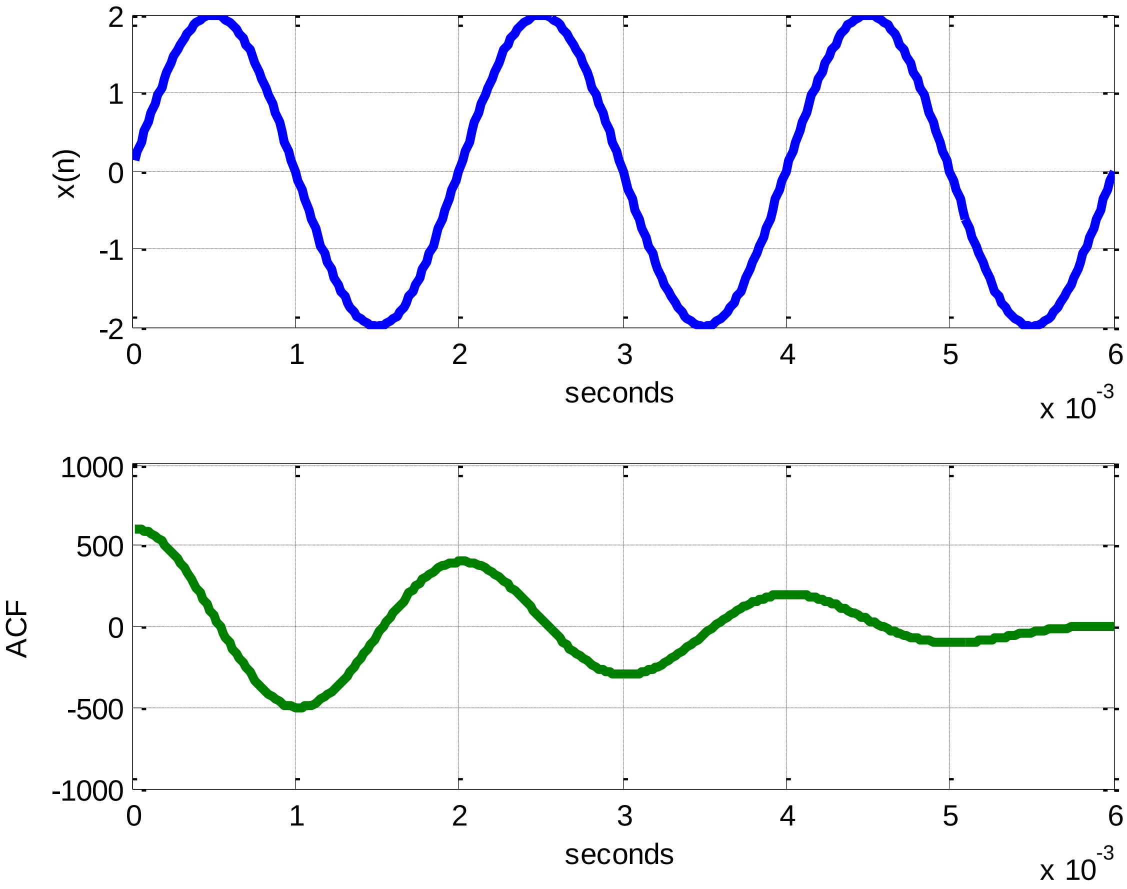
$y(n) = x(n-2)$, where $x(n)=[3\ 11\ \underset{\uparrow}{7}\ 0\ -1\ 4\ 2]$.

cross-correlation result:



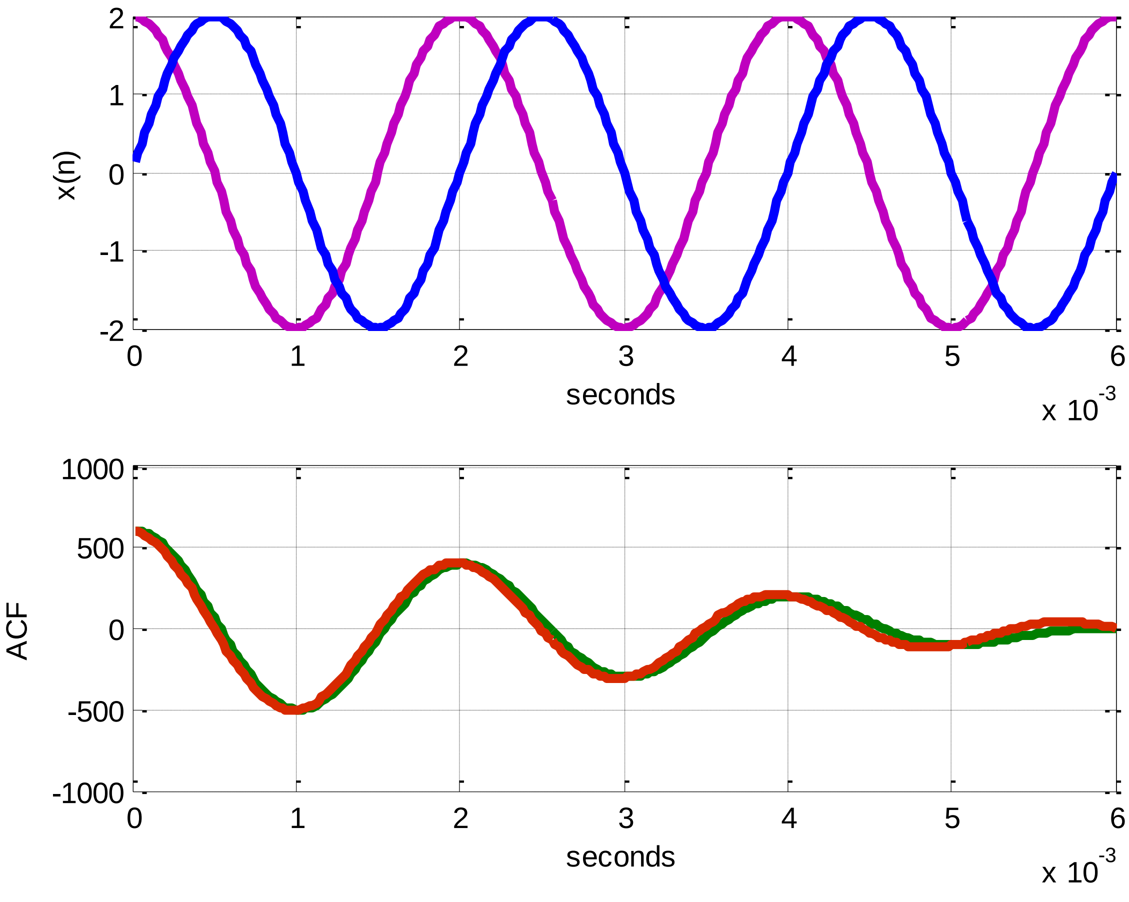
Correlation:

- What about phase information?



Correlation:

- What about phase information?



Correlation:

- Applications of ACF/CCF:

(1) Detecting a periodic input corrupted by additive white Gaussian noise:

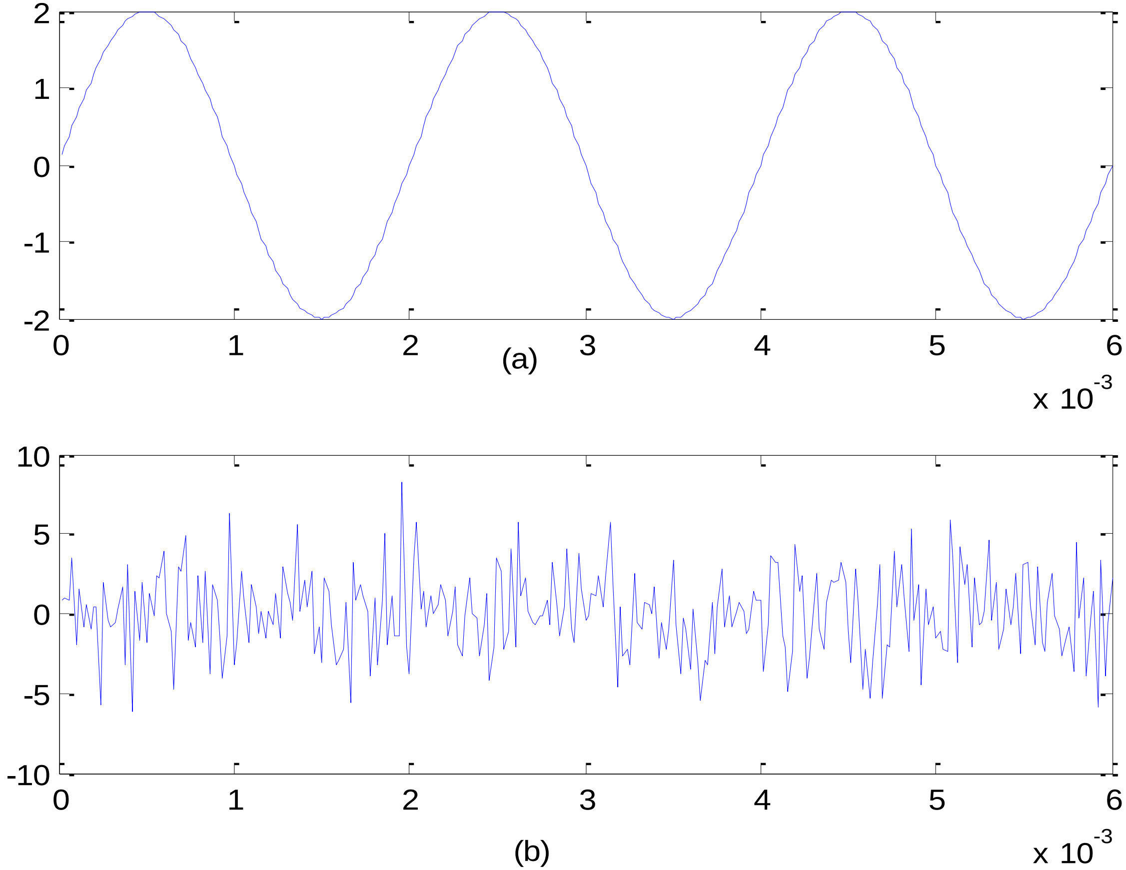


Figure: (a) Input periodic wave (b) Input corrupted by AWGN for -5dB SNR

Correlation:

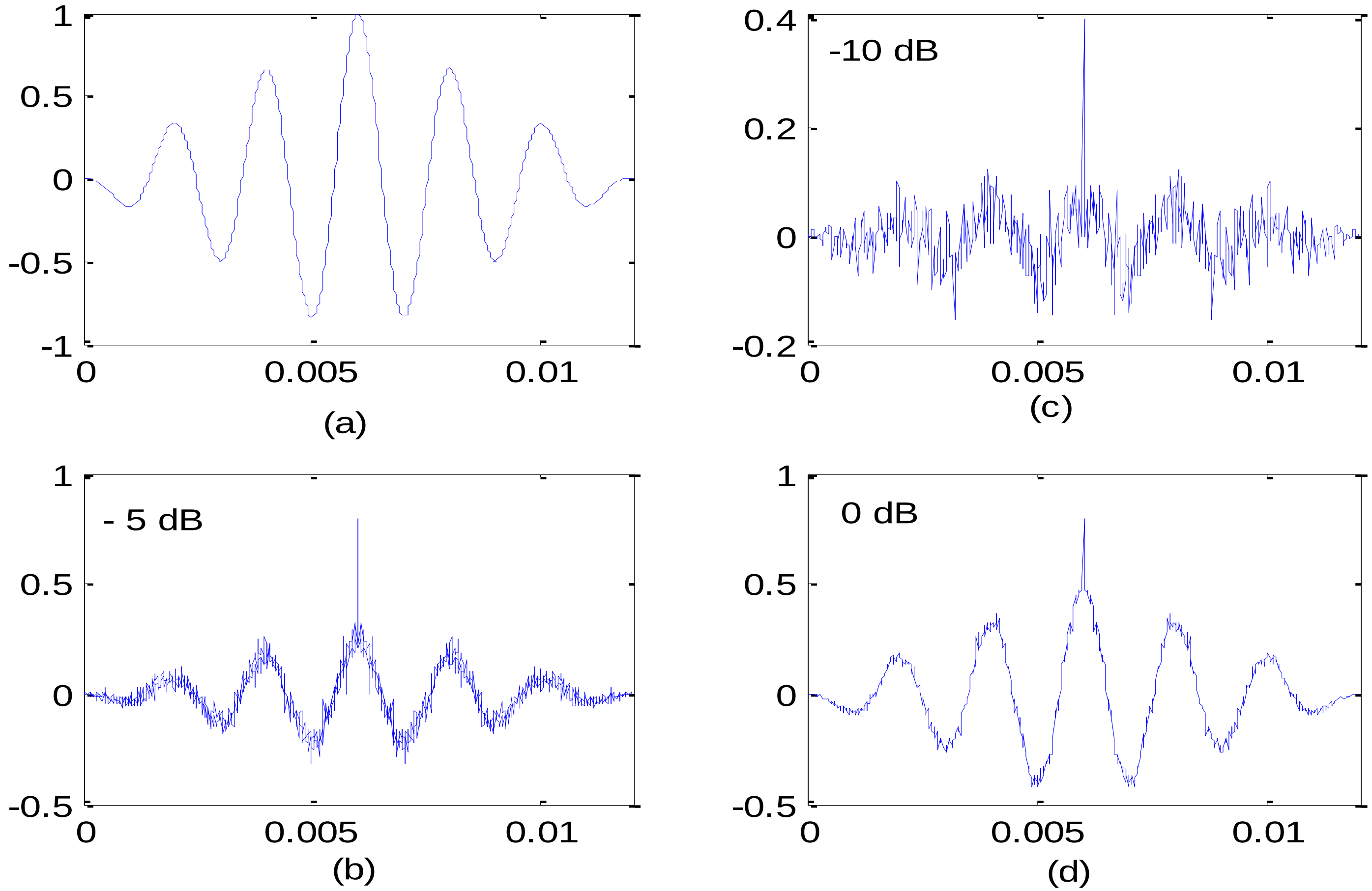


Figure: (a) ACF of input (b,c,d) ACF of noisy sequence.

Correlation:

(2) Estimation of impulse response:

$$R_{yx}(k) = y(n) * x(-n) = [x(n) * h(n)] * x(-n)$$
$$= x(n) * x(-n) * h(n) = R_{xx}(k) * h(n)$$

Where, $x(n)$ is white noise input,

$R_{xx}(k)$ is the auto-correlation of the noise input

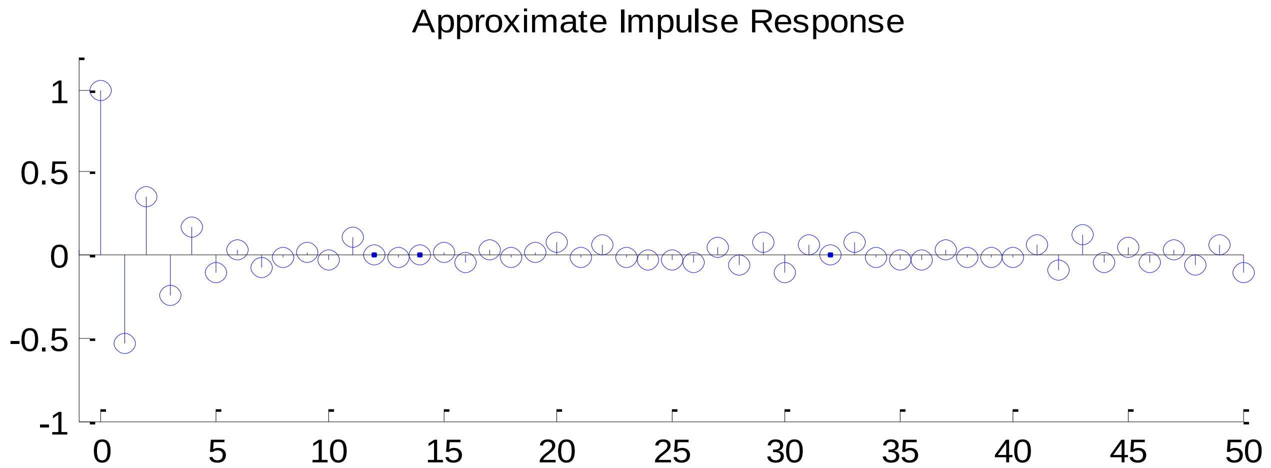
- Auto-correlation of white noise sequence is like impulse sequence.

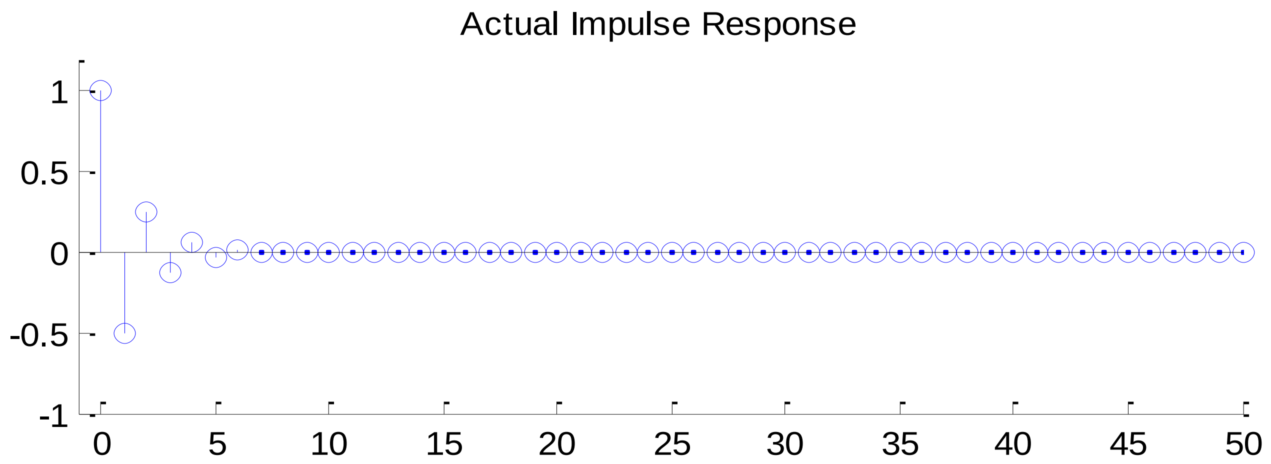
- Convolution of an impulse and DT sequence results in the sequence itself.

$$\boxed{R_{yx}(k) \approx h(n)}$$

Correlation:

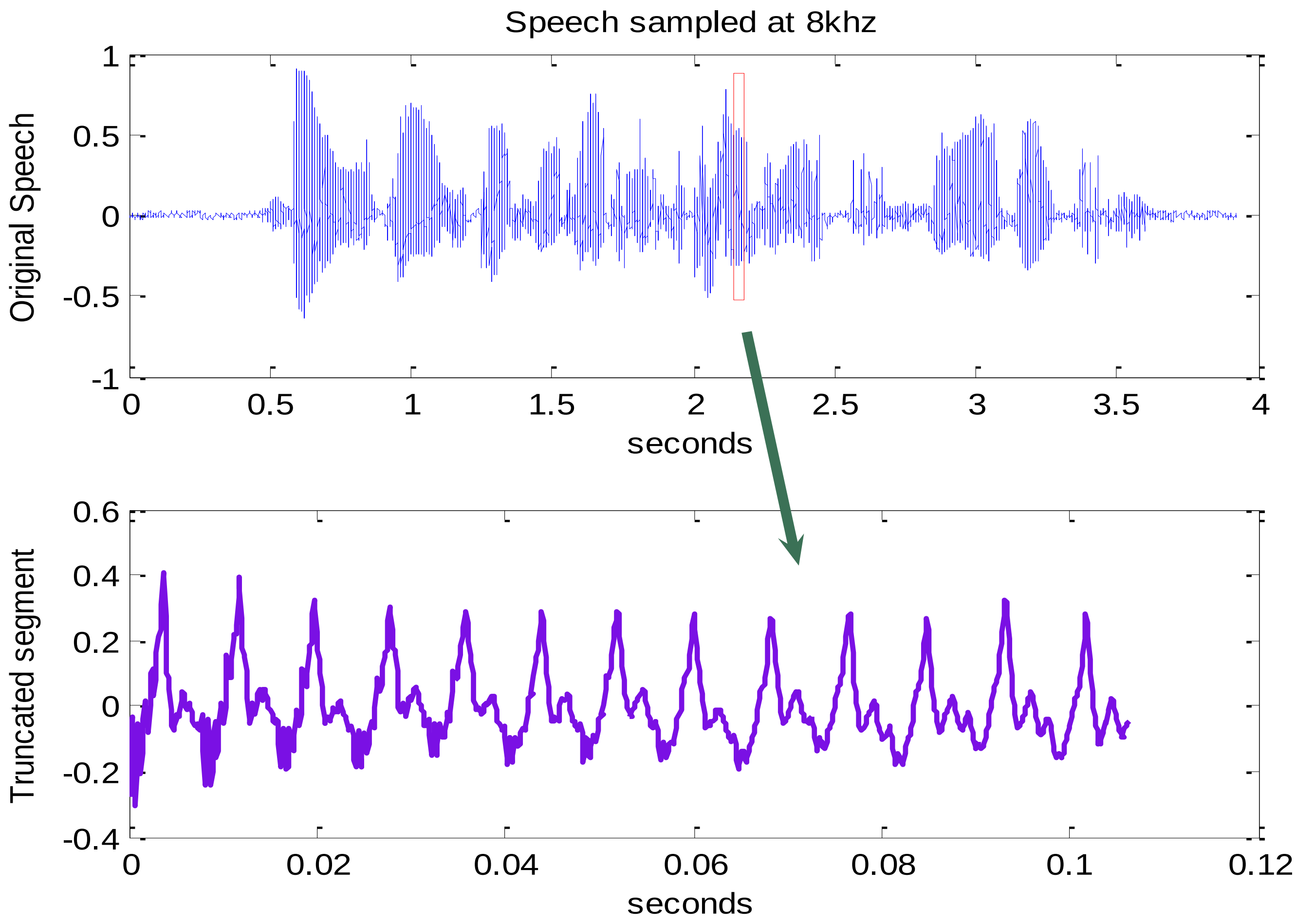
Consider a system: $$y(n) + 0.6y(n-1) = x(n)$$





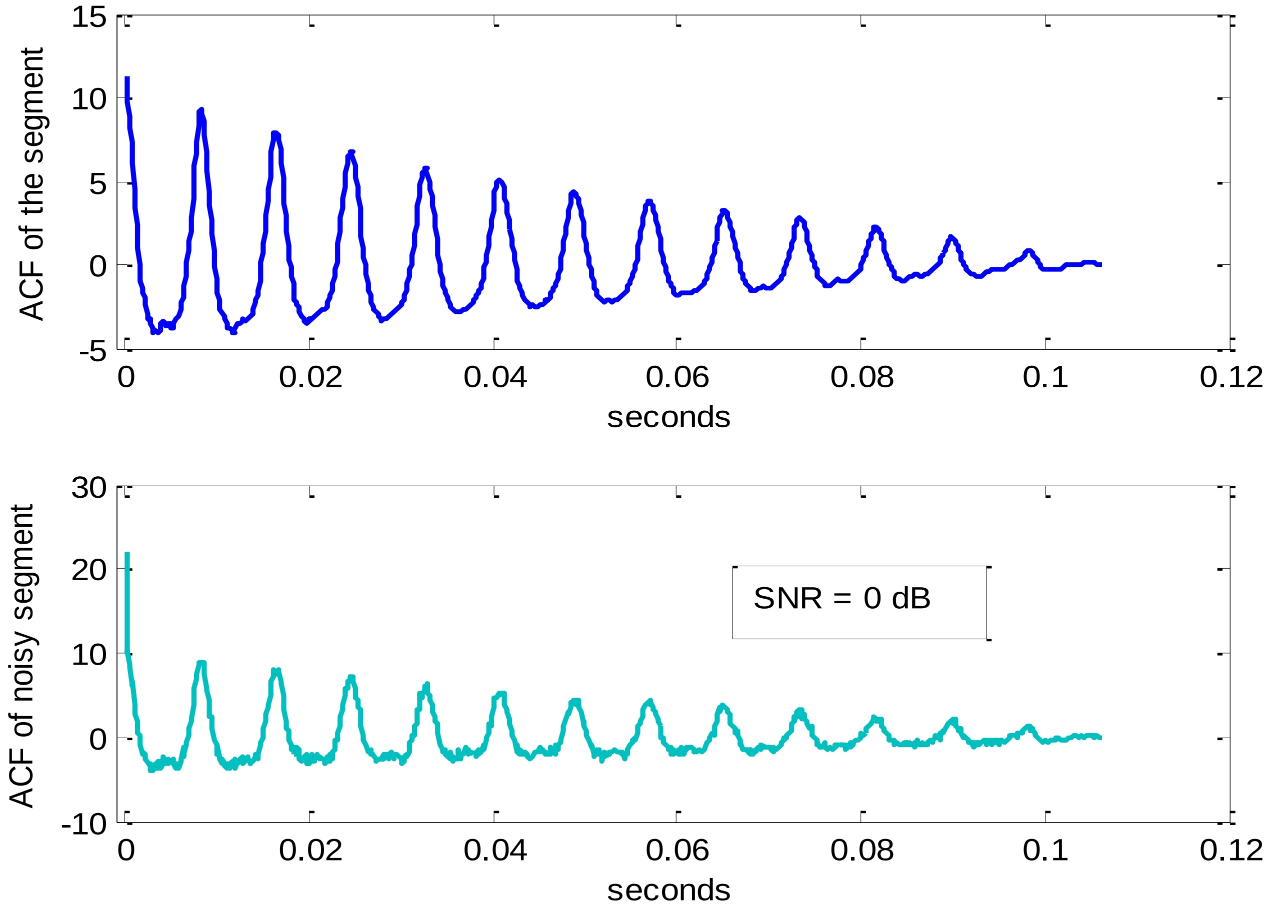
Correlation:

- Pitch Estimation of a noisy speech



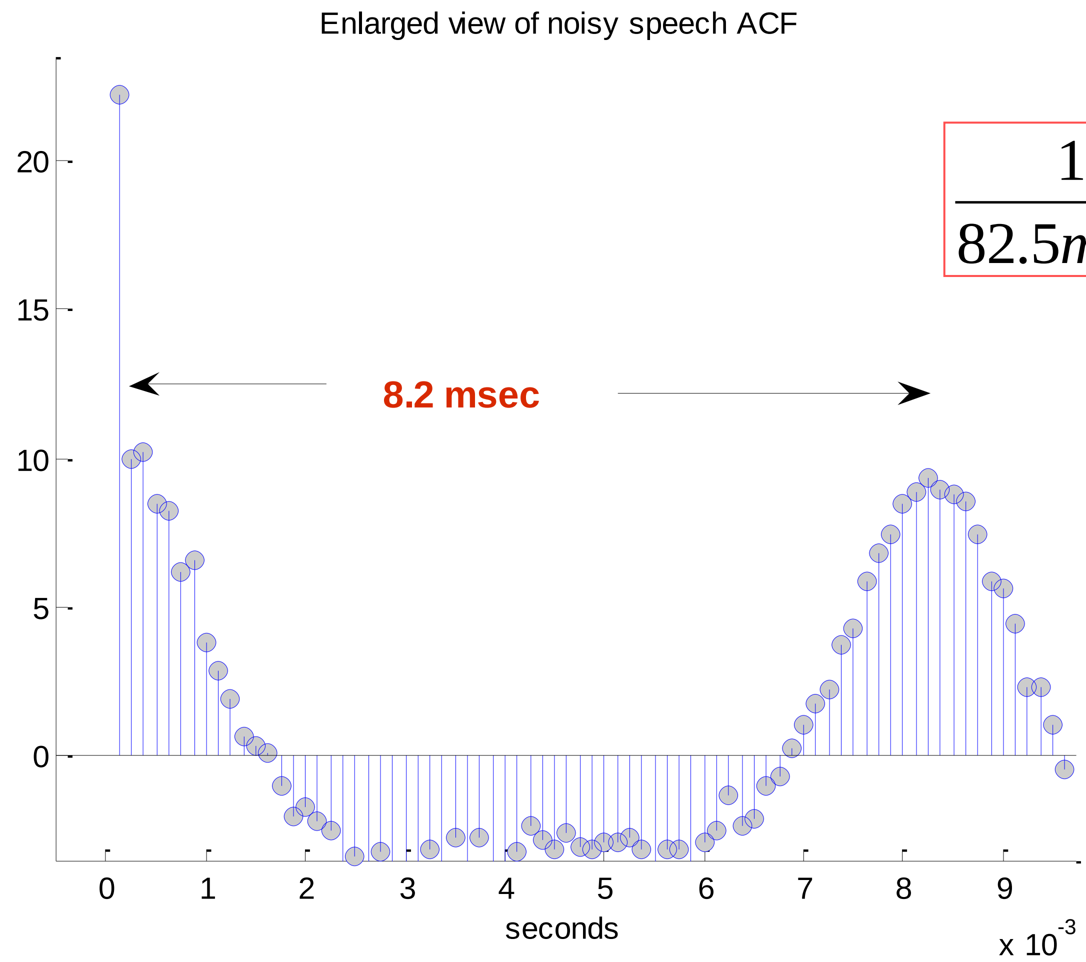
Correlation:

- Pitch Estimation of a noisy speech



Correlation:

- Pitch Estimation of a noisy speech



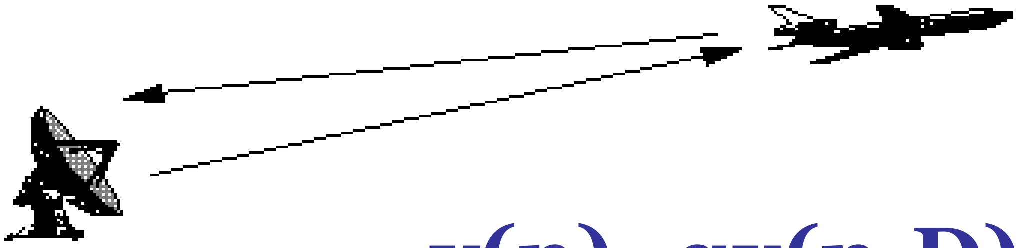
$$\frac{1}{82.5 m\sec} \cong 122\ Hz$$

Correlation:

- **Detection of signals in noise by auto-correlation**

PROBLEM:

A radar transmits a short tone burst of EM energy and receives a weak echo from a distant target. In the absence of noise weak echo can be amplified and there is no problem detecting it. If there is background of noise whose amplitude exceeds that of the echo, the echo will be masked and not detectable . As we know noise is suppressed in ACF domain, then correlation detection can be used to locate the echo.



$$y(n)=\alpha x(n-D)+w(n)$$

Correlation:

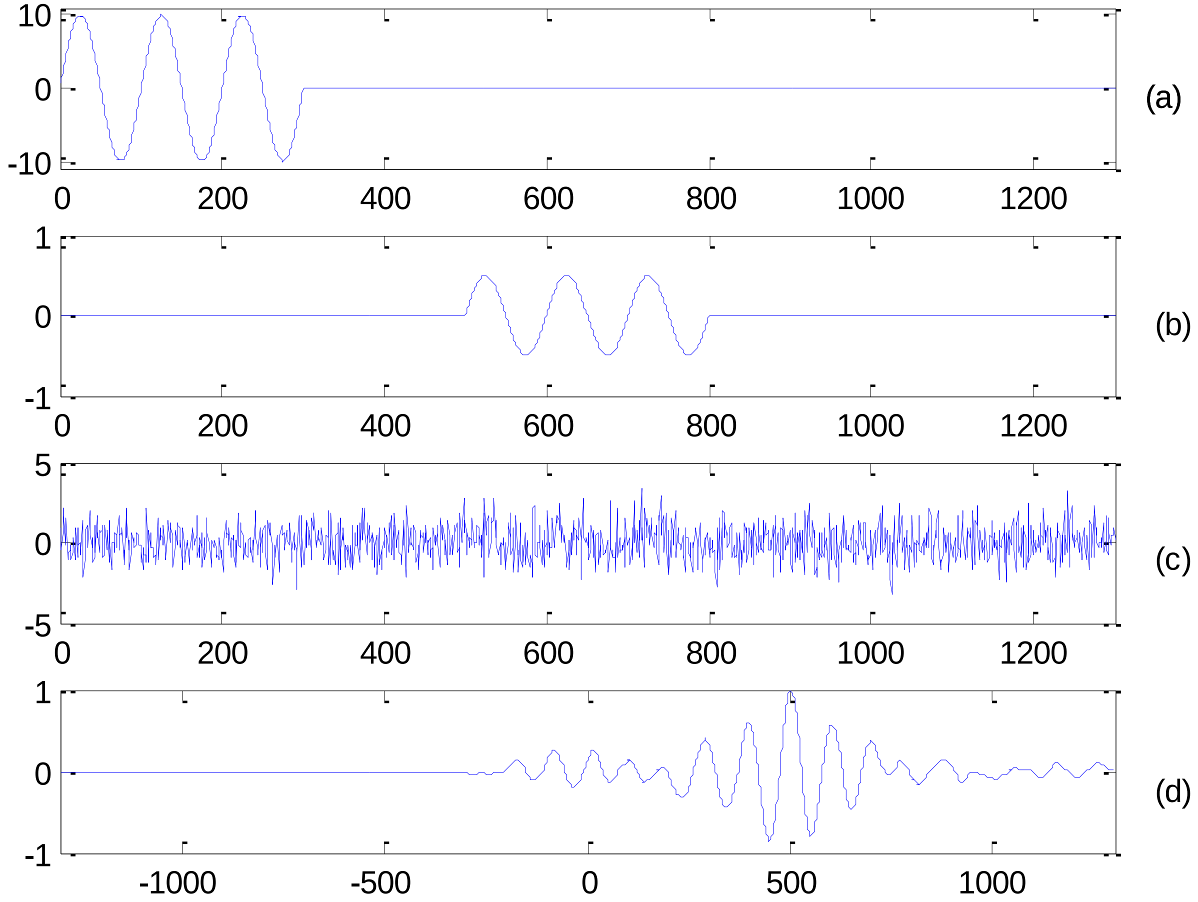
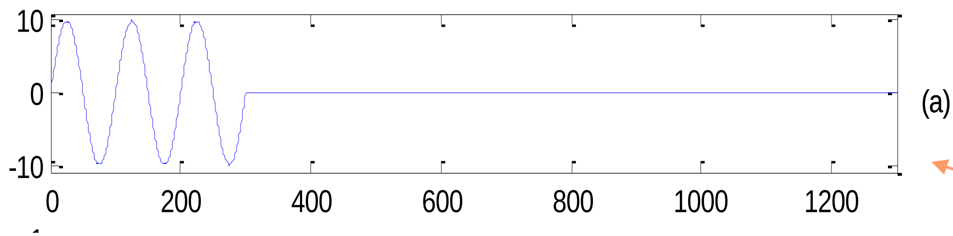


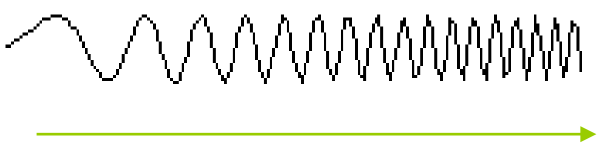
Figure:(a) Transmitted tone burst (b) Received weak echo (c)Received echo buried into background noise(d)CCF between (a) and (c) to locate a weak echo. It shows that after 500 units delay an echo arrives (location of the peak).

Lab Task : 1

- If the tone burst has a gradual increase in frequency, what will happen? Is it advantageous for detection? Compare the results.



Burst with fixed frequency



- A radar or sonar 'chirp' signal...

Gradual increase in frequency

Lab Task : 2

Detection of a transmitted sequence

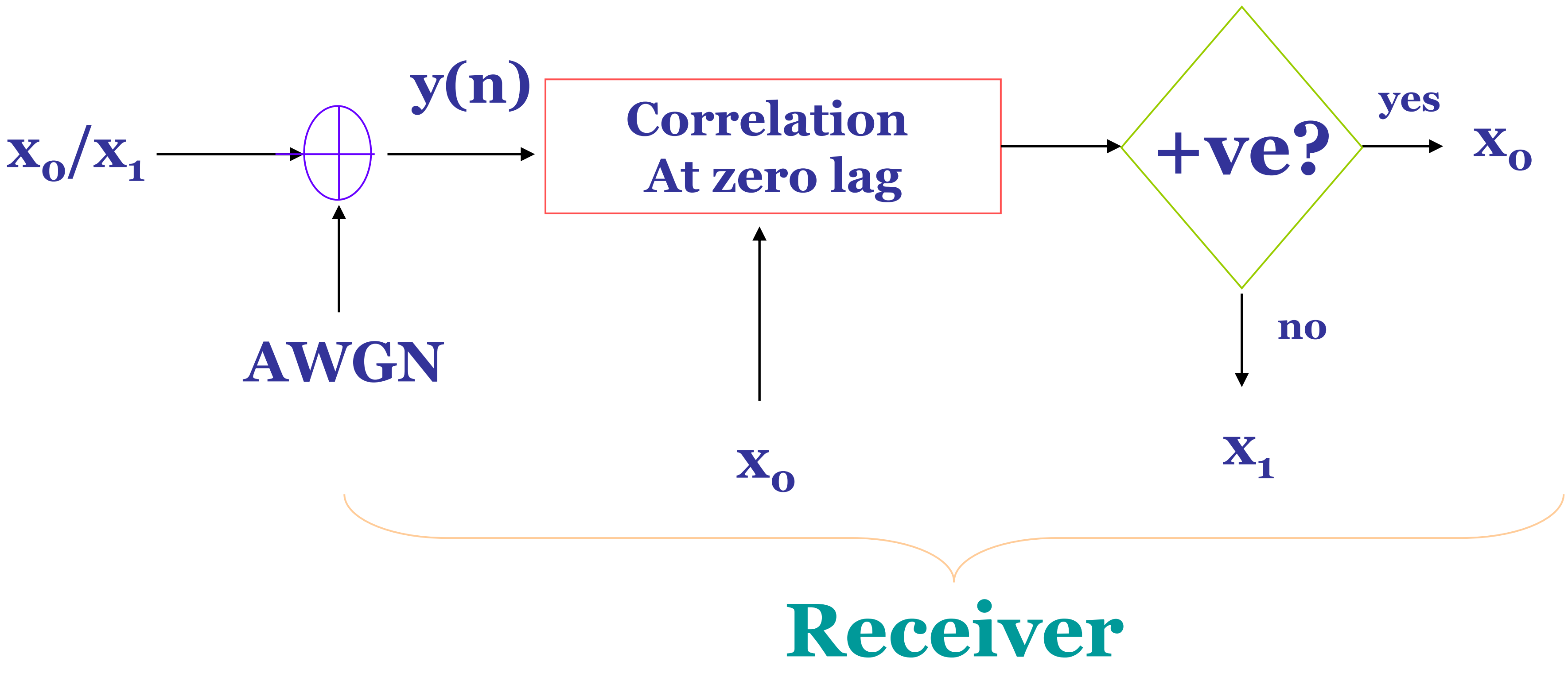
- Let in the transmitter, to transmit zero (0) we send $x_0(n)$ for $0 \le n \le L-1$ and to transmit one (1) we send $x_1(n)$ for $0 \le n \le L-1$ where $x_1(n) = -x_0(n)$. The signal received by the receiver

$$y(n) = x_i(n) + w(n) \quad i = 0,1 \text{ and } 0 \le n \le L-1$$

$w(n)$ is additive white noise.

Present a technique to detect the transmitted sequence from $y(n)$. Assume that particular receiver knows $x_0(n)$ and $x_1(n)$. Write a general MATLAB program for this purpose.

Lab Task : 2


x_o/x_1
y(n)
Correlation
At zero lag
+ve?
yes
x_o
no
x_1
AWGN
x_o
Receiver

Report

- Lab Exercise D.1
- Lab Exercise D.2
- Estimate your vocal pitch !!

References

1) Proakis & Manolakis,"Digital Signal Processing:Principles,Algorithms and Applications.",Chapter 2, 3rd Edition , Prentice Hall Ltd.

2) Mitra," Digital Signal Processing: A Computer Based Approach" Chapter 2, Edition 1998, Tata McGraw-Hill Co. Ltd.

3) Denbigh, "System Analysis and Signal Processing" Chapter (2,16), Edition 1998, Addison-Wesley.

4) Elali, "Discrete Systems and Digital Signal Processing with MATLAB® " Chapter 2,Edition 2004, CRC Press

5) Ingle & Proakis, "Digital Signal Processing using *MATLAB®*" Chapter 2,Edition 2000 Thomson-Brooks/Cole Co Ltd

MATLAB CODES

MATLAB CODES:

```
% generating sequences
n1=-3;
n2=3;
n=n1:n2;
no=-1;
x1=[(n-no)==0];%impulse
x2=[(n-no)>=0];%step
x3=[x1(1:find(x1)-1)
0:length(n)-find(x1)];%ramp
xe=0.5*(x3+fliplr(x3));
xo=0.5*(x3-fliplr(x3));
x4=xe+xo;
subplot(211),stem(n,xe)
subplot(212),stem(n,xo)
```

```
% upsampling
w=0.36;
n =1:52;
L = 3;
x=sin(w*n);
y=zeros(1,length(n));
for n=1:52
    if(rem(n,L)==0)
        y(n)=x(n/L);
    end
end
subplot(211),stem(1:n,x)
xlabel('n')
subplot(212),stem(1:n,y)
xlabel('n')
```

MATLAB CODES:

- PART-2:
- Example-1: CONVOLUTION

```
x1=[4 2 6 3 8 1 5];
n1=[-2:4];
x2=[3 8 6 9 6 7];
n2=[-4:1];
kmin=n1(1)+n2(1);
kmax=n1(end)+n2(end);
y=conv(x1,x2);
k=kmin:kmax;
subplot(311),stem(n1,x1)
subplot(312),stem(n2,x2)
subplot(313),stem(k,y)
```

MATLAB CODES:

- PART-2:
- Exercise-2.1: CONVOLUTION

```
%extension of the previous program

k1=min([n1(1) n2(1) kmin]);
k2=max([n1(end) n2(end) kmax]);
x11=[zeros(1,n1(1)-k1) x1];
x22=[zeros(1,n2(1)-k1) x2];
kc=k1:k2;
x11(length(kc))=0;
x22(length(kc))=0;
subplot(311),stem(kc,x11)
subplot(312),stem(kc,x22)
subplot(313),stem(kc,y)
```

MATLAB CODES:

- PART-2:
- Exercise-2.1: CONVOLUTION

m-file which returns both convolved result and index :

```
function [y ny]=conv_m(x,nx,h,nh)
nyb=nx(1)+nh(1);
nye=nx(length(x))+nh(length(h));
ny=[nyb:nye];
y=conv(x,h)
```

MATLAB CODES:

- Exercise-3.1: System Response from difference equations

```
n=[-10:20];
x1=[(n-0)==0];
x2=[(n-0)>=0];
x3=.5*sin(n);
b=[1];
a=[1 0.6];
impulse=filter(b,a,x1);
step=filter(b,a,x2);
sinusoidal=filter(b,a,x3);
subplot(131),stem(n,impulse)
axis([-10 20 -.7 1])
xlabel('n')
title('Impulse response')
subplot(132),stem(n,step)
axis([-10 20 -.7 1])
xlabel('n')
title('Step response')
subplot(133),stem(n,sinusoidal)
axis([-10 20 -.7 1])
xlabel('n')
title('Sinusoidal response')
```

MATLAB CODES:

- PART-4:
- Example :CORRELATION

```
x=[3,11,7,0,-1,4,2];
n=-3:3;
[y,ny]=sigshift(x,n,2)
[x,nx]=sigfold(x,n);
[rxy,nxy]=conv_m(x,nx,y,ny);
stem(nxy,rxy/max(rxy))
```

MATLAB CODES:

- Applications of ACF/CCF:

```
>> T=2e-3; % period=2ms
>> tstep=T/100;
>>t=tstep:tstep:3*T; % taking time index upto 3 periods
>>x=2*sin(2*pi*t/T); % Input
>>Px=sum(x.^2)/length(x); % Input power
>>SNR= -10; % in dB
>>Py=Px/10^(SNR/10);
>>n=sqrt(Py)*randn(1,length(t));% generate white noise
>>y=x+n; % Corrupted input
>>ACF_x=normalize(xcorr(x)); % Normalizing the peak to 1
>>ACF_n=normalize(xcorr(n));
>>ACF_y=normalize(xcorr(y));
```

```
>>ACF_y(length(x))=.4*max(ACF_y); >>figure(1)
>>subplot(211),plot(t,x)
>>subplot(212),plot(t,n)
>>figure(2)
>>subplot(221),plot(tstep*(1:length(ACF_x)),ACF_x)% showing ACF w.r.t. time
>>subplot(222),plot(tstep*(1:length(ACF_y)),ACF_y)% showing ACF w.r.t. time
% hold on
```

MATLAB CODES:

- Estimation of impulse response:

```
>>N=500;
>>nr=0:499;
>>ny=nr;
>>r=randn(1,N);
>>y=zeros(size(r));
>>for n=2:500
          y(n)=r(n)-0.6*y(n-1);
  end
>>rr=fliplr(r);
>>nrr=-fliplr(nr);
>>Ryr=conv(y,rr);
>>kmin=ny(1)+nrr(1);
```

```
>>kmax=ny(length(ny))+nrr(length(nrr));
>>k=kmin:kmax;
>>subplot(211),stem(k,Ryr/Ryr(N))
>>title('Approximate Impulse Response');
>>num=[1 0];
>>den=[1 0.5];
>>n=0:499;
>>x=zeros(size(n));
>>x(1)=1;
>>yy=filter(num,den,x);
>>subplot(212),stem(n,yy)
>>title('Actual Impulse Response')
```

MATLAB CODES:

- Detection of signals in noise by auto-correlation:

```
>>T=2e-3; % period=2ms
>>tstep=T/100;
>>t=tstep:tstep:3*T; % taking time index upto 3 periods
>>s=2*sin(2*pi*t/T); % Input
>>x=5*[s zeros(1,1000)];
>>e=[zeros(1,500) .25*s zeros(1,500)];
>>n=randn(1,length(x));
>>r=e+n;
>>n=-(length(n)-1):(length(n)-1);
>>Rxr=xcorr(r,x);
>>subplot(411),plot(x)
>>subplot(412),plot(e)
>>subplot(413),plot(r)
>>subplot(414),plot(n,Rxr/max(Rxr))
```

MATLAB CODES:

- Signal smoothing by a moving average (MA) system

```
>> R=50;
>> d=rand(1,R)-0.5;
>> m=0:R-1;
>> x=2*m.*(0.9).^m;
>> stem(x)
>> y=x+d;
>> subplot(121),plot(m,x,'r.-')
>> hold on
>> subplot(121),plot(m,y,'b-')
>> M=3;
>> b=ones(1,M)/M;
>> z=filter(b,1,y);
>> subplot(122),plot(m,x,'b.-')
>> hold on
>> subplot(122),plot(m,z,'r-')
>> legend('x(n)','z(n)')
```

```
R=50;
d=rand(1,R)-0.5;
m=0:R-1;
x=2*m.*(0.9).^m;
%stem(x)
y=x+d;
M=1:10;
avg_ne=[];
for i=1:10
   b=ones(1,M(i))/M(i);
   z=filter(b,1,y);
   e=abs(x-z);
   ne=e(2:end)./(x(2:end));
   avg_ne=[avg_ne mean(ne)];
end
plot(M,avg_ne)
```

Thank You

Laboratory manual for this experiment is prepared by
TOUFIQUL ISLAM
Lecturer, Dept of EEE, BUET
and supervised by
Dr. Md. KAMRUL HASAN
Professor, Dept of EEE, BUET

Presentation for this experiment is prepared by
SHANKHANAAD MALLICK
Lecturer, Dept of EEE, BUET
and
TOUFIQUL ISLAM
Lecturer, Dept of EEE, BUET