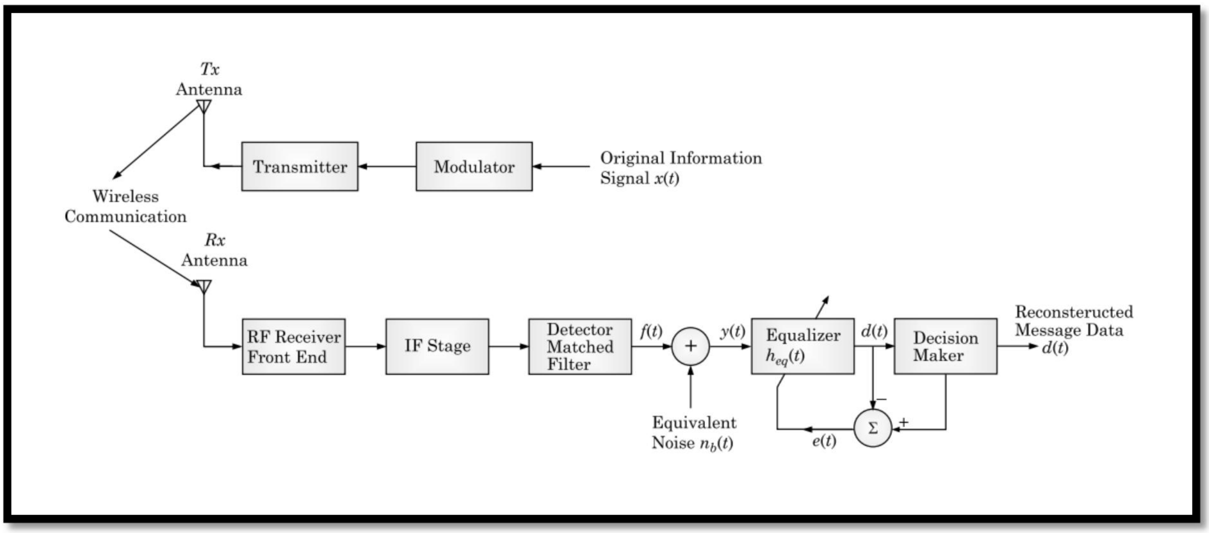
AIM

To Simulate Equalization Techniques using AWGN channel considering input as any random data as well as an Image with the help of MATLAB software. Also compare output of both with and without equalization.

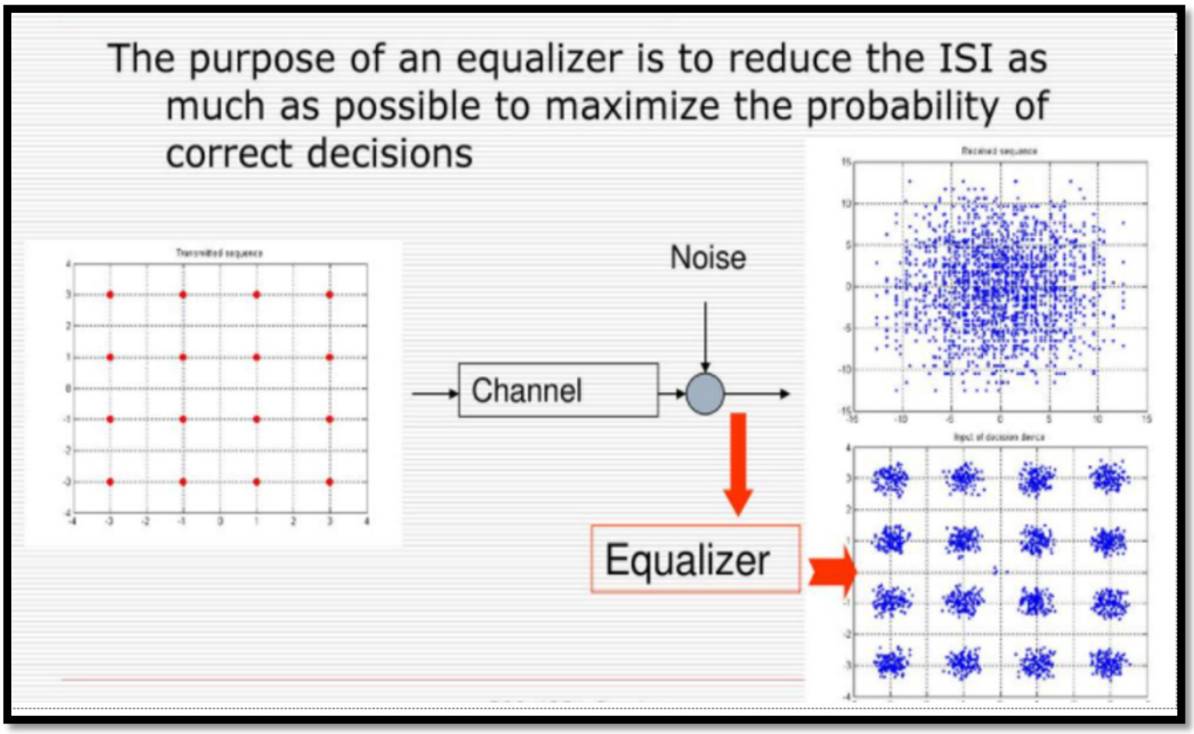
THEORY

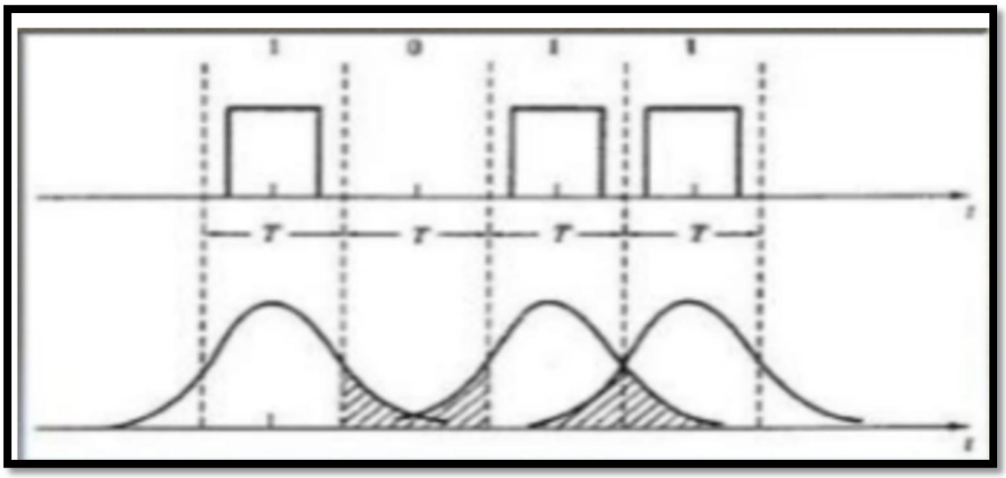
With the growth in wireless communication, there was a requirement to improve the received signal quality because there were a number of channels working at different frequencies which cause signal fading and interference. Thus techniques were evolved to improve received signal of quality. These techniques are 'Equalization and Diversity'. Equalization is used to compensate intersymbol interference (ISI) while Diversity is used to compensate for fading.

An equalizer is usually implemented at base band or at IF stage in a receiver. Adaptive equalizer works on two operating modes, that are training and tracking. A fixed length training sequence is sent by the transmitter so that the receiver's equalizer may receive the signal for minimum bit error rule (BER). This sequence is a pseudo random binary signal followed by the user data.

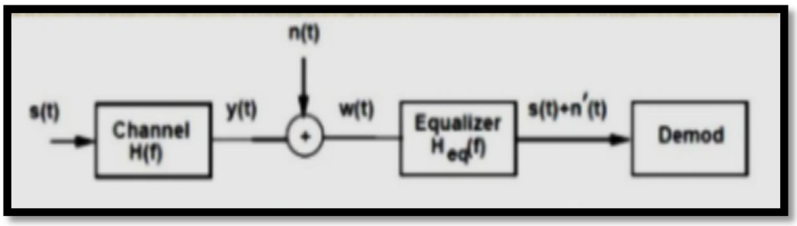


Block diagram of a Communication system using an equalizer





Inter Symbol Interference (ISI)



Matlab Implementation

ALGORITHM:

- Input the random data (Let say 1000 in number)
- Change into required format.
- Modulate it with psk
- Assume SNR
- Add AWGN
- Assume Tau and PdB
- Assume Rayleigh channel
- Realize it with filter function
- Assume LMS as adaptive algorithm object

- Construct linear equalizer object (lineareq)
- Equalize signal using equalizer object (equalize)
- Demodulate the data
- Convert it into required format
- Find BER
- Compare output for both with and without equalizer (Repeat steps 12-14 for without equalizer)
- Repeat the above procedure for input as an image

MATLAB CODE

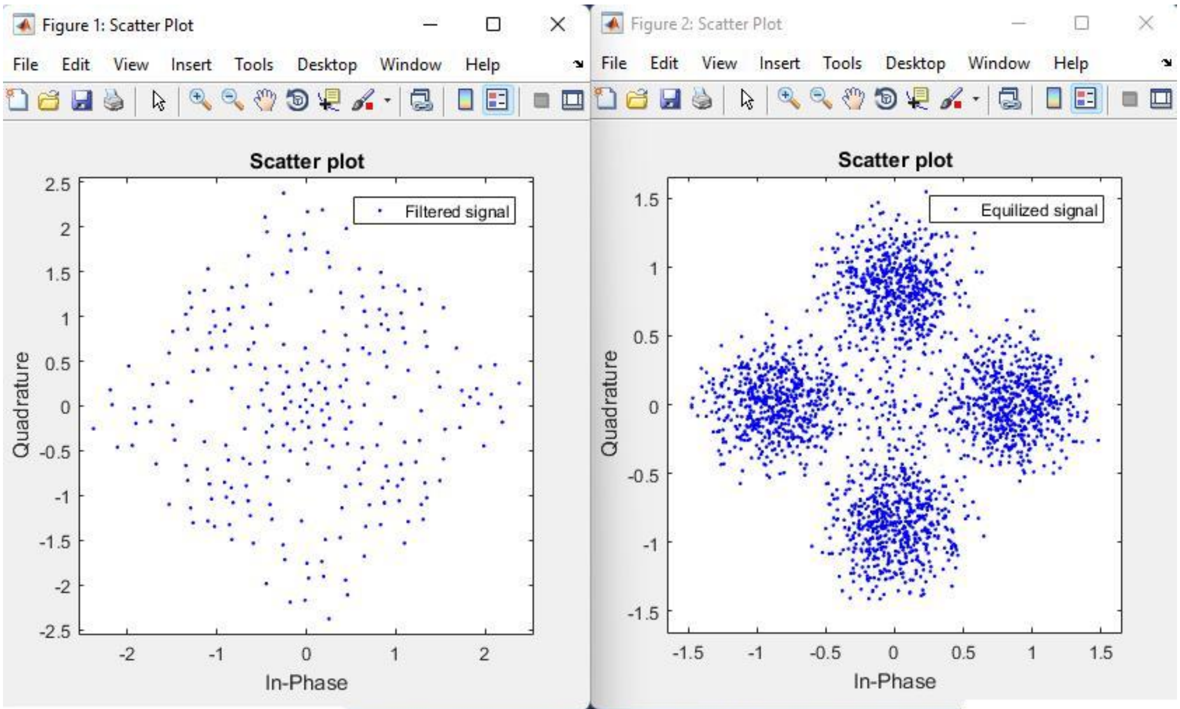
1.

```
% Set up parameters and signals.
M = 4; % Alphabet size for modulation

data = randi([0 1], 5000, 1);

[n, m] = size(data);

Mod=2;

data_vector = reshape(data, [numel(data)/Mod Mod]);

msg = bi2de(data_vector);

hMod = comm.QPSKModulator('PhaseOffset',0);
modmsg = step(hMod,msg); % Modulate using QPSK.
trainlen = 200; % Length of training sequence

Tauj = [0.986 0.845 0.237 0.123];
Beta = [-0.1 0 -0.03 0.31];

chan = rayleighchan(1,0,Tauj, Beta);
chanCoeff = chan.AvgPathGaindB + 1i*chan.PathDelays;
filtmsg = filter(chanCoeff,1, modmsg); % Introduce channel distortion.

% Equalize the received signal.
eq1 = lineareq(8, lms(0.01)); % Create an equalizer object.
eq1.SigConst = step(hMod,(0:M-1)')'; % Set signal constellation.
[symbolest,yd] = equalize(eq1,filtmsg,modmsg(1:trainlen)); % Equalize.


scatterplot(filtmsg)
legend('Filtered signal')
scatterplot(symbolest)
legend('Equilized signal')

% Compute error rates with and without equalization.
hDemod = comm.QPSKDemodulator('PhaseOffset',0);
demodmsg_noeq = step(hDemod,filtmsg); % Demodulate unequalized signal.
demodmsg = step(hDemod,yd); % Demodulate detected signal from equalizer.
hErrorCalc = comm.ErrorRate; % ErrorRate calculator
ser_noEq = step(hErrorCalc, ...
    msg(trainlen+1:end), demodmsg_noeq(trainlen+1:end));
reset(hErrorCalc)
```

```
ser_Eq = step(hErrorCalc, msg(trainlen+1:end),demodmsg(trainlen+1:end));
disp('Symbol error rates with and without equalizer:')
disp([ser_Eq(1) ser_noEq(1)])
```

OUTPUT

1.




```
Symbol error rates with and without equalizer:
     0    0.8865
```

CONCLUSION

In this practical we have implemented using matlab with AWGN channel under Rayleigh fading. It was observed that error rate was reduced after using equalization methods.