Experiment No: 08

Name of the Experiment:

Study and Simulation of Frequency Division Multiple Access (FDMA)

Objective:

1. To understand the concept of FDMA and how multiple users share the frequency spectrum.
2. To simulate FDMA by allocating different frequency bands to multiple users.
3. To analyze the effect of guard bands on signal separation and interference.
4. To visualize the frequency spectrum allocation using MATLAB.

Theory:

Frequency Division Multiple Access (FDMA) is a channel access method where the total available bandwidth is divided into frequency bands, each assigned to a different user. Users transmit simultaneously, but on separate frequencies, minimizing interference.

- Each user has a unique frequency band.
- Guard bands are placed between channels to avoid interference.
- FDMA is used in 1G cellular systems and traditional radio communications.

Experiment Setup & Procedure:

1. Define total bandwidth available (e.g., 100 MHz).
2. Divide bandwidth into smaller frequency bands for multiple users (e.g., 5 users × 18 MHz each + 1 MHz guard bands).
3. Simulate simple sinusoidal signals for each user at assigned center frequencies.
4. Plot the frequency spectrum of the combined signal to observe separation.
5. Adjust guard band width and observe overlap/interference between user signals.

MATLAB Source Code:

```
clc; clear; close all;

% Parameters
totalBW = 100e6;        % Total bandwidth (100 MHz)
numUsers = 5;           % Number of users
guardBW = 1e6;          % Guard band between users (1 MHz)
userBW = (totalBW - (numUsers-1)*guardBW)/numUsers; % Bandwidth per user

Fs = 1e9;               % Sampling frequency for simulation (1 GHz)
t = 0:1/Fs:1e-6;        % Time vector (1 microsecond)
```

```
% Generate signals for each user at different center frequencies
signals = zeros(numUsers, length(t));
centerFreqs = zeros(1,numUsers);

for k = 1:numUsers
    % Calculate center frequency of kth user
    centerFreqs(k) = (k-1)*(userBW + guardBW) + userBW/2;
    % Generate sinusoidal signal at center frequency
    signals(k,:) = sin(2*pi*centerFreqs(k)*t);
end

% Sum all user signals (combined signal)
combinedSignal = sum(signals, 1);

% FFT to visualize frequency spectrum
N = length(combinedSignal);
f = (-N/2:N/2-1)*(Fs/N)/1e6; % Frequency axis in MHz
Y = fftshift(abs(fft(combinedSignal))/N);

% Plot frequency spectrum
figure;
plot(f, Y);
xlabel('Frequency (MHz)');
ylabel('Amplitude');
title('Frequency Spectrum of Combined FDMA Signal');
grid on;
xlim([0 totalBW/1e6]); % Focus on total bandwidth

% Display frequency allocation
disp('User Frequency Allocations (MHz):');
for k = 1:numUsers
    fprintf('User %d: %.2f MHz to %.2f MHz\n', k, ...
        (k-1)*(userBW + guardBW)/1e6, (k-1)*(userBW + guardBW)/1e6 +
userBW/1e6);
end
```

Expected Output:

- A plot showing **5 distinct peaks** in the frequency domain, each corresponding to a user's frequency band separated by guard bands.
- Frequency allocations printed to the command window showing the exact bands assigned to each user.

Result and Discussion:

The plot shows a clear separation of user signals in the frequency domain due to FDMA allocation. Guard bands prevent overlapping of adjacent user signals, minimizing interference. Increasing or decreasing guard bands will affect the overlap and interference between channels.FDMA allows simultaneous transmission but requires careful frequency planning. In real systems, filters and more complex modulation schemes are used for better separation.

Experiment No: 09

Name of the Experiment:

Study and Simulation of Code Division Multiple Access (CDMA)

Objective:

1. To understand the principle of CDMA and how multiple users share the same frequency spectrum using unique codes.
2. To simulate CDMA by assigning orthogonal codes to multiple users.
3. To demonstrate how signals can be separated at the receiver using correlation.
4. To visualize the process of spreading and despreading signals in MATLAB.

Theory:

Code Division Multiple Access (CDMA) is a multiple access technique where all users share the same frequency band simultaneously but are separated by unique spreading codes. Each user's signal is multiplied (spread) by a unique code sequence, enabling multiple signals to coexist with minimal interference.

- Each user has a unique pseudo-random code (PN code).
- Signals are spread over a wide bandwidth, increasing robustness to interference and multipath fading.
- At the receiver, correlation with the user's code despreads the desired signal while other signals appear as noise.
- Used in 3G cellular systems (e.g., IS-95).

Experiment Setup & Procedure:

1. Define spreading codes (e.g., Walsh codes) for multiple users.
2. Generate data bits for each user.
3. Spread data bits by multiplying with the user's code.
4. Sum all spread signals to simulate the combined transmitted signal.
5. At the receiver, correlate combined signal with a user's code to recover original data.
6. Visualize the signals before spreading, after spreading, and after despreading.

MATLAB Source Code:

```
clc; clear; close all;

% Number of users
numUsers = 3;
```

```
% Define Walsh codes (orthogonal codes)
codes = [1  1  1  1;
         1 -1  1 -1;
         1  1 -1 -1];

% User data bits (1 or -1)
data = [1 -1 1];

% Spreading length
N = size(codes, 2);

% Spread signals for each user
spreadSignals = zeros(numUsers, N);
for k = 1:numUsers
    spreadSignals(k, :) = data(k) * codes(k, :);
end

% Transmitted signal (sum of all users)
txSignal = sum(spreadSignals, 1);

% Receiver: Despreading for User 2
userIndex = 2; % User to recover
despreadSignal = txSignal .* codes(userIndex, :);
recoveredBit = sign(sum(despreadSignal));

% Display results
fprintf('Original bit for User %d: %d\n', userIndex, data(userIndex));
fprintf('Recovered bit for User %d after despreading: %d\n', userIndex,
recoveredBit);

% Plot signals
figure;
subplot(4,1,1);
stem(data, 'filled');
title('Original Data Bits of Users');
xlabel('User');
ylabel('Bit');

subplot(4,1,2);
imagesc(spreadSignals);
title('Spread Signals (Users × Code Length)');
xlabel('Code Chip Index');
ylabel('User');
colorbar;

subplot(4,1,3);
stem(txSignal, 'filled');
title('Combined Transmitted Signal');
xlabel('Code Chip Index');
ylabel('Amplitude');

subplot(4,1,4);
stem(despreadSignal, 'filled');
title(['Despread Signal for User ', num2str(userIndex)]);
xlabel('Code Chip Index');
ylabel('Amplitude');
```

Expected Output:

- Display of original and recovered bit for User 2 (should match).
- Plots showing original bits, spread signals, combined transmitted signal, and despread signal for the selected user.

Result and Discussion:

The Walsh codes are orthogonal, allowing separation of user signals despite simultaneous transmission. After despreading, the desired user’s data bit is recovered successfully by correlation. Other users’ signals appear as noise due to orthogonality of codes. CDMA is robust against interference and supports multiple users over the same bandwidth. Real systems use longer PN codes and advanced error correction for better performance.

Experiment No: 10

Name of the Experiment:

Study and Simulation of Time Division Multiple Access (TDMA)

Objective:

1. To understand the concept of TDMA and how multiple users share the same frequency by dividing access in time slots.
2. To simulate TDMA by assigning distinct time slots to multiple users.
3. To analyze how signals are transmitted sequentially in their time slots without interference.
4. To visualize time slot allocation and user data transmission using MATLAB.

Theory:

Time Division Multiple Access (TDMA) is a channel access method where the total bandwidth is shared among users by allocating unique time slots to each user on the same frequency channel. Each user transmits in rapid succession, one after another, so their transmissions do not overlap in time.

- All users use the same frequency but transmit in different time intervals (time slots).
- Efficient for bursty data traffic since slots can be allocated dynamically.
- Used in 2G GSM cellular systems.

Experiment Setup & Procedure:

1. Define the number of users and total time frame length.
2. Assign unique time slots to each user within the time frame.
3. Generate user data signals (e.g., simple pulse signals).
4. Arrange the signals sequentially in assigned time slots to form the composite TDMA signal.
5. Plot the time domain signals showing individual user transmissions and combined TDMA frame.

MATLAB Source Code:

```
clc; clear; close all;
```

```
% Parameters
numUsers = 4;           % Number of users
slotLength = 50;        % Number of samples per time slot
totalLength = numUsers * slotLength;

% Generate user data signals (rectangular pulses)
userSignals = zeros(numUsers, slotLength);
for k = 1:numUsers
    % Simple pulse with amplitude = user index
    userSignals(k, :) = k * ones(1, slotLength);
end

% Construct TDMA frame by placing user signals in assigned time slots
tdmaSignal = zeros(1, totalLength);
for k = 1:numUsers
    startIdx = (k-1)*slotLength + 1;
    endIdx = k*slotLength;
    tdmaSignal(startIdx:endIdx) = userSignals(k, :);
end

% Plot user signals
figure;
subplot(numUsers+1,1,1);
plot(userSignals(1,:));
title('User 1 Signal');
ylim([0 numUsers+1]);

for k = 2:numUsers
    subplot(numUsers+1,1,k);
    plot(userSignals(k,:));
    title(['User ', num2str(k), ' Signal']);
    ylim([0 numUsers+1]);
end

% Plot TDMA signal
subplot(numUsers+1,1,numUsers+1);
plot(tdmaSignal, 'k', 'LineWidth', 1.5);
title('Combined TDMA Signal');
ylim([0 numUsers+1]);
xlabel('Sample Number');
ylabel('Amplitude');
grid on;

% Display time slot allocation
disp('Time Slot Allocation (Samples):');
for k = 1:numUsers
    fprintf('User %d: Samples %d to %d\n', k, (k-1)*slotLength+1,
k*slotLength);
end
```

Expected Output:

- Separate plots of each user’s pulse signal representing data in their time slot.

- A combined plot showing the TDMA frame where each user transmits sequentially in assigned slots.
- Printed allocation of sample indices corresponding to each user’s time slot.

Result and Discussion:

Each user transmits only in their assigned time slot, avoiding overlap and interference. The composite TDMA signal is formed by concatenating user signals in time domain. TDMA efficiently shares bandwidth by multiplexing users in time rather than frequency. Time synchronization is critical for correct slot alignment. TDMA is suitable for digital systems with bursty data and is widely used in GSM.