Experiment No. 9

Pulse Amplitude Modulation

Objective: To visualize the message modulating the carrier frequency using Pulse Amplitude Modulation.

Pre-requests: Basics of MATLAB and fundamentals of signals & systems.

Useful References:

- Lecture Notes of the course,
- Signal processing & Linear Systems, (B. P. Lathi, ©2004, ISBN: 978-0-19-568583-1).
- Communication Systems, (Simon S. Haykin, © 2000, ISBN: 978-0-47-117869-9).

Theory :

Pulse amplitude modulation (PAM), or sampling, is our gate to the digital communication techniques. In other words, to convert a signal from its continuous time analog signal to digital format, the first step is to do the sampling or PAM. Thus, PAM is the bridge between angle and digital worlds.

Analog signal is sampled every T_S seconds. T_s is referred to as the sampling interval. $f_s = 1/T_s$ is called the sampling rate or sampling frequency. There are **three** sampling methods:

1. **Ideal** - an impulse at each sampling instant
2. **Natural** - a pulse of short width with varying amplitude
3. **Flattop** - sample and hold, like natural but with single amplitude value

*The process is referred to as **pulse amplitude modulation*** **PAM** and the outcome is a signal with analog (non integer) values. With PAM, the amplitude of the pulse in the train will be altered according to the amplitude of the message signal at the sampling time. Note that the period and duration of the pulses are fixed.

Procedure: Implementing the PAM modulation. Use the following MATLAB program to implement the **PAM**-modulation, write the program in your PC and run it.

```
clear all; close all; clc;
%% System Settings! do not change them
% Sampling Frequency of the system do not change it
Fs = 1e6;
% time vector of the system, do not change it
t = 0:1/Fs:(0.1e-2);
%% Experiment starts here
% Natural Sampling PAM
fs = 8000; %sampling frequency
fm = input('Message Frequency [in Hz] Fm = '); %message
frequency
xm = sin(2*pi*fm*t); %message
T = [0:10]*1/fs; % pulse period,
tau = 50e-6; % Pulse width
Ptrain = pulstran(t,T,@rectpuls,tau);% Pulse train
y = Ptrain.*xm; % Pulse Amplitude Modulation

% Flat-Top PAM
yFlat=y;
for i = 2:length(t)
    if Ptrain(i) == 1 && Ptrain(i-1) == 0 %if the rising edge is
detected
        yFlat(i) = Ptrain(i) * xm(i);    %sampling occurs
    elseif Ptrain(i) == 1 && Ptrain(i-1) == 1 %and while the
carrier signal is 1
        yFlat(i) = yFlat(i-1);              %the value of y1
remains constant
    else
        yFlat(i) = 0;                %otherwise, y is zero
    end
end

%% Ploting results for Natural Sampling PAM
figure
subplot(2,2,1);plot(t,xm); % plot message
xlabel('Time');ylabel('Amplitude');title('Message');hold on;
subplot(2,2,3);plot(t,Ptrain); % plot pulse
xlabel('Time');ylabel('Amplitude');title('Switching waveform');
subplot(2,2,2);plot(t,y); % plot PAM naturel
xlabel('Time');ylabel('Amplitude');title('PAM Naturel');
%% Ploting results for Flat-Top PAM
subplot(2,2,4);plot(t,yFlat); % flat-top PAM plot
xlabel('Time');ylabel('Amplitude'); title('PAM Flat-Top');
```

Perform the following steps,

1. Run the program and record all your results, when Fm = 1000 Hz.
2. Change the message frequency to 1500, and record all the results.
3. Change the message signal to $-square(2\pi Wt)$ and repeat steps 1 and 2.
4. Change the Pulse width to 100μs and repeat steps 1 to 3.

Discussion:

1. How to reconstruct the modulated signal?
2. What is the minimum sampling frequency required in steps 1 to 4 of your results? Why?
3. Use your lectures and/or textbooks to explain the theory behind the PAM modulation, supported by mathematical models and block diagrams, for all the types of the PAM modulation techniques.

Good Luck
Dr. Montadar Abas Taher