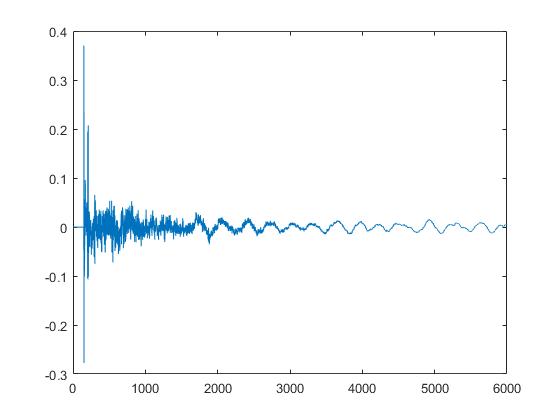
**QUESTION 3 – Commented MatLab code**

****

**Figure 1. Plot of impulse response h(t)**

%%Part (a) and (b)

f = 16000; %%declared playback frequency

h = audioread('h.wav'); %%reading in the impulse response and speech recording

x = audioread('speech.wav');

plot(h); %%plots sample number vs amplitude of the impulse response

sound(x,f); %%listening to original speech file

convhx = conv(h,x); %%convolving the two signals together

sound(convhx);

%%Part (c)

h1 = h; %%Create new matrix of the same size

N = 3000;

for n = N+1:length(h) %%populating the new matrix with time shifted entries

h1(n-N) = h(n);

end;

convh1 = conv(h1,x);

sound(convh1,f); %%listening to the new signal

%%Part (d)

xreverse = flipud(x); %%Use MatLab function to flip the matrix upside down

sound(xreverse,f);

%%Part(e)

sound(x,5000);

sound(x,7000);

sound(x,12000); %%playback at various sampling speeds etc etc;

%%I did not include every single frequency for brevity's sake

%%Part(f)

length = length(x);

sub1 = zeros(67290,1); %%Creating new matrix of half the size for 2:1 aliasing

for n = 0:length

sub1(n) = x(2\*n); %%Insert every second element of x into sub1

end;

sound(sub1,f/2); %%playback sub1 at half of the original playback frequency

%%repeat for the other aliasing rates

%%Part(g)- commented code sample for 16-bit quantization

max = max(x);

min = min(x);

step = (max-min)/(2^16-1); %%Defining max, min and step size

sig = x; % declare new variable sig as x

partition = [min+step:step:max]; % 16 intervals

codebook = [min:step:max]; % 16 intervals, one entry per interval

[index,quants] = quantiz(sig,partition,codebook); % Quantize.

plot(sig,'x',t,quants,'.');