Lab Report 4

ESE – 3014

EMBEDDED SYSTEMS COMMUNICATION PROTOCOLS AND SECURITY

Submitted to :

Linchan Wang

Submitted by:

Gurvinder Singh(748418)

**1.Simulate all operations performed in the transmitter of a PCM system include steps below. You can choose the techniques we introduced in the course, andplease show your code in text form and screenshots of each steps.**

**Solutoin:**

>>

clc;

clear all;

function signal=get\_signal(frequency,axis)

signal=cos(2\*pi\*frequency\*axis);

endfunction

%Coded Signal

dec2bin(coded\_signal)

pulse\_modulated\_signal=interp1(n1,quantized\_signal,t,"nearest");

n=7;

input\_frequency=3e3;

t=[0:1/F:n/input\_frequency];

input\_signal=get\_signal(input\_frequency,t);

plot(t,input\_signal);

hold on;

%Sampling of signal

sampling\_frequency=44e2;

n1=[0:1/sampling\_frequency:n/input\_frequency];

sampled\_signal=get\_signal(input\_frequency,n1);

stem(n1,sampled\_signal,'1');

Z=legend ("Blue- Analog Signal","Red- Sampled Signal");

legend(Z, 'location', 'northeastoutside');

figure();

plot(t,input\_signal,"r",t,pulse\_modulated\_signal,"g");

legend("input\_signal","pulse\_modulated\_signal");

F=200e5;

dynamic\_range=2;

N=3;

number\_of\_levels=2^N; %8 bit quantizer for N=3

resolution=dynamic\_range/(number\_of\_levels-1);

levels=[-1:resolution:1];

boundaries=[-1+(resolution/2):resolution:1-(resolution/2)];

codes=[0:number\_of\_levels-1];

quantized\_signal=[];

coded\_signal=[];

%Quantization and coding

for sample=sampled\_signal

i=1;

if(sample>=boundaries(end))

coded\_signal(end+1)=codes(end);

dec2bin(codes(end),N)

quantized\_signal(end+1)=levels(end);

else

for boundary=boundaries

if(sample<=boundary)

coded\_signal(end+1)=codes(i);

dec2bin(codes(i),N)

quantized\_signal(end+1)=levels(i);

break;

endif

i=i+1;

endfor

endif

endfor

**% bipolar return to zero**

>>

clc;

clear all;

close all;

b=[1 0 0 0 0 1 1 1 1 0 0];

l=length(b);

b(l+1)=0;

n=1;

while n<=l

t=(n-1):.001:n;

if b(n)==1

if b(n+1)==b(n)

y=(t<(n-0.5))+(t==n);

else

y=(t<(n-0.5));

end

else

if b(n+1)==1

y=0\*(t<(n-0.5))+(t==n);

else

y=0\*(t<(n-0.5));

end

end

n=n+1;

plot(t,y)

hold on;

axis([0 l -1.5 1.5]);

end

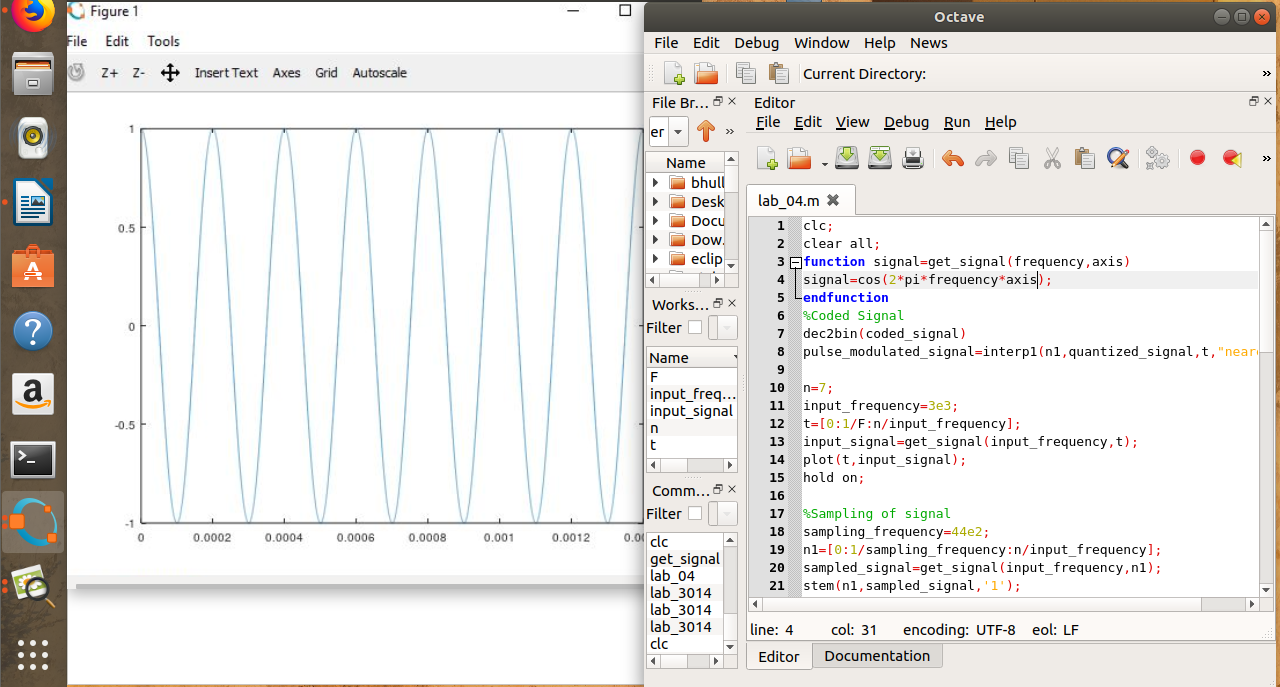
title('RZ');

xlabel('Time');

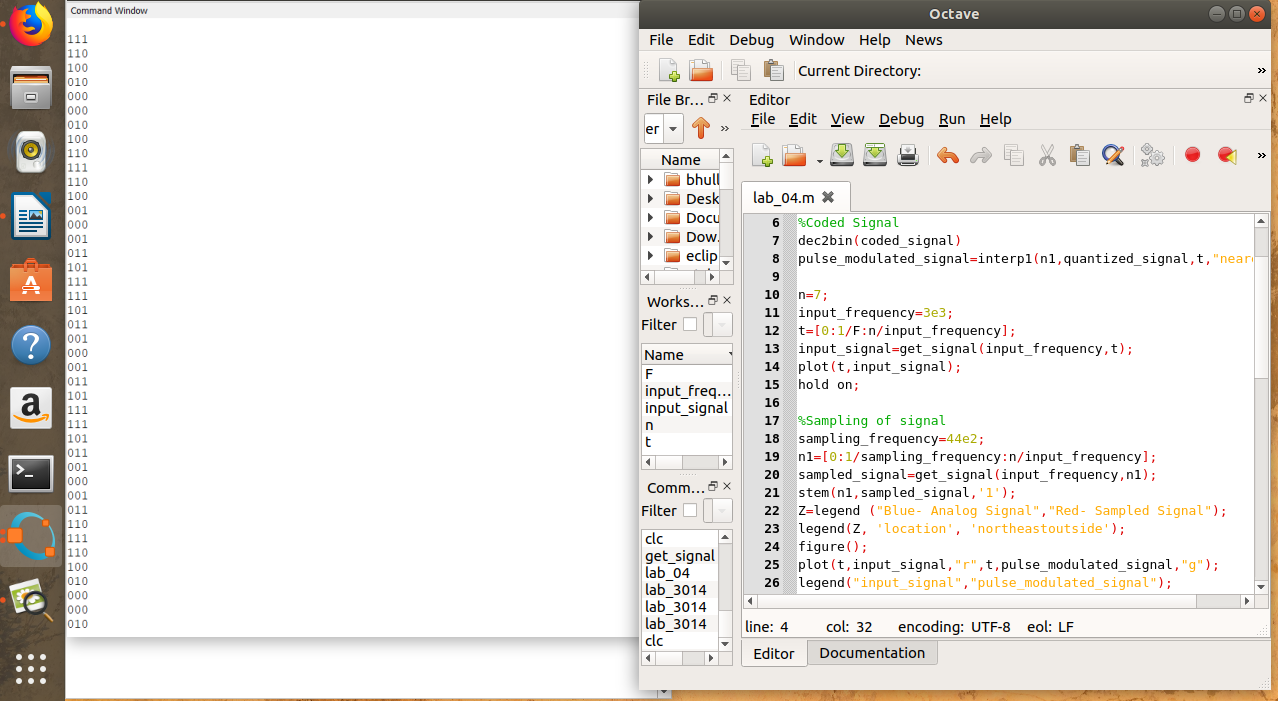
ylabel('Amplitude');

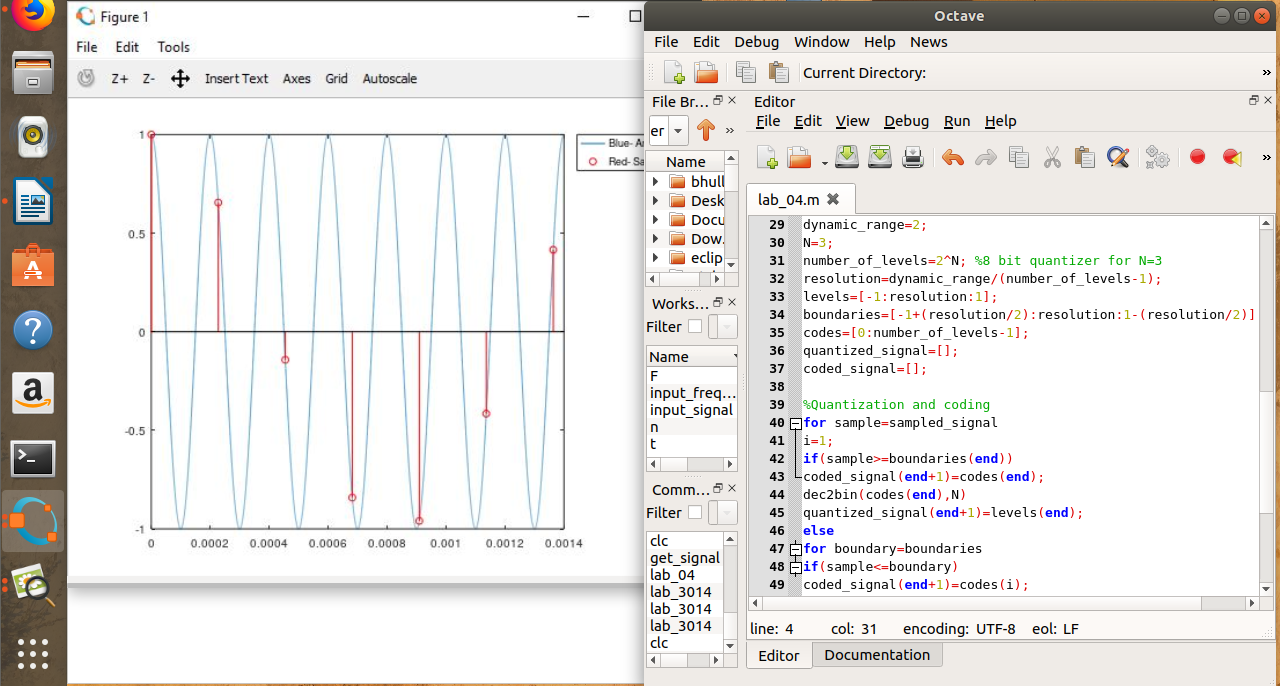
**In Octave:**

**1. Sampled Signal**

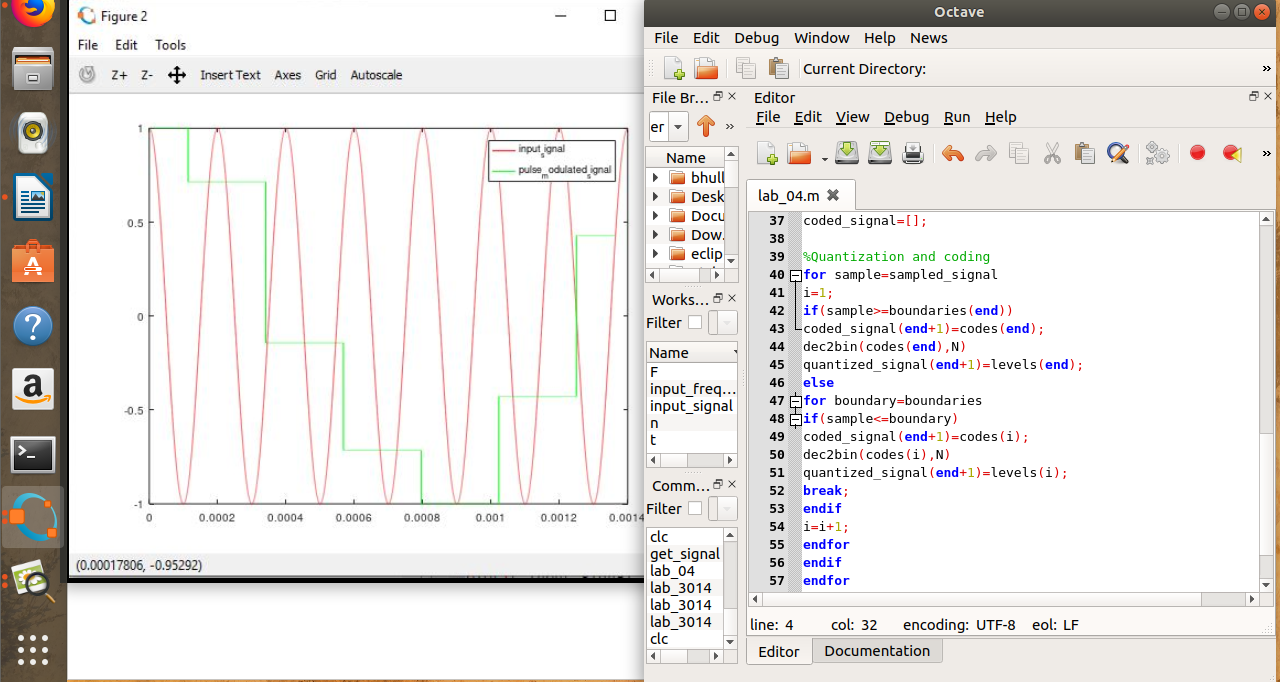


**2. Encoded binary Signal**

**3. Sampling**



**4. Quantized Signal**



**5.Bipolar Return to zero(BRZ)**

