The Fourier Valley

David Houston

EE 225 Lab 4

Bruce Barnes

ABSTRACT

This lab aimed at analyzing the process of converting temporal functions in to Fourier space via the Fast Fourier Transform. We took and created a variety of Fourier series functions and examined them via a MATLAB program to distinguish how they operated in the frequency domain. From there we took and evaluated several different piecewise linear function in terms of their Fourier Transforms and graphed their responses in both the temporal and frequency domains.

From there we took our previously gain knowledge on Fourier series expansion and applied it to several different wave forms and expressed them in terms of increasing frequency terms. After this, it was ten determined to analyze an Op-Amp circuit in order to determine its constituent Fourier series components.

INTRODUCTION

The following processes were performed via MATLAB and LT Spice in order to further solidify the concepts of Fourier with respect to its series and transform representations in frequency space.

PROCEDURE

Fourier series

To first analyze the Fourier series we manipulated a GUI file in order to see how the series representation of a temporal function responded.

The first analysis that was taken was to look at the response of a square wave function. It can see from the below graph that as the number of coefficients increases the accuracy of the wave shape does also. For the square wave it seems as though you need at least 15 coefficients in order for the wave shape to have a relatively close shape.

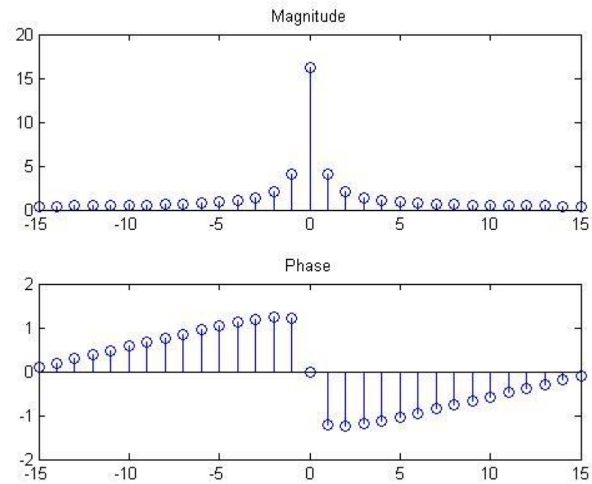
However, when you analyze the triangle wave form it can be seen that only need 4 coefficients in order to produce a similar wave shape, making this particular signal easier to produce.

The Sawtooth (or ramp function) wave needs a high amount of coefficients in order to produce a wave which models the desired form due to its non-sinusoidal shape.

In comparison, when examining the Half-Rectified Sine wave it can be seen that due to its relative phase and shape it is much easier to reproduce this function using the Fourier series as opposed to a square wave.

Now, the Full-Rectified Sine Wave, unlike its ugly step-sister, is much more complicated and unique therefore requiring more coefficients in order to full produce the desired waveform.

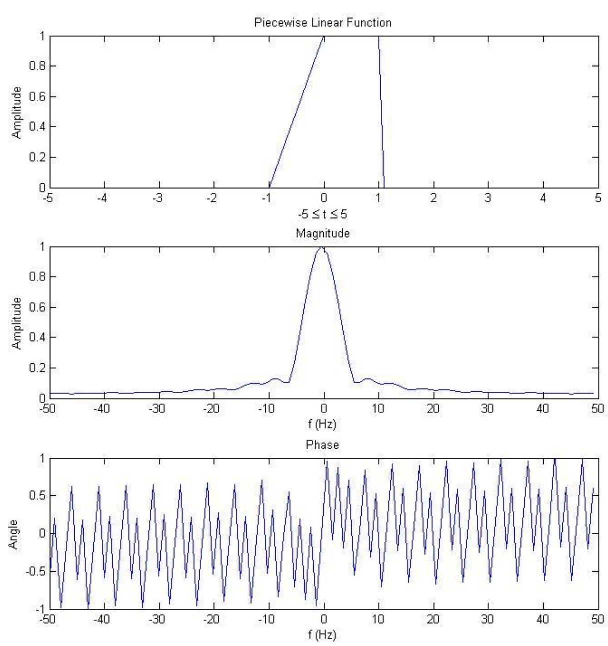
After fully analyzing the effects of the Fourier series of different temporal response systems, we created a MATLAB file to analyze an exponential time dependent response. The phase and magnitude plots were as follows.



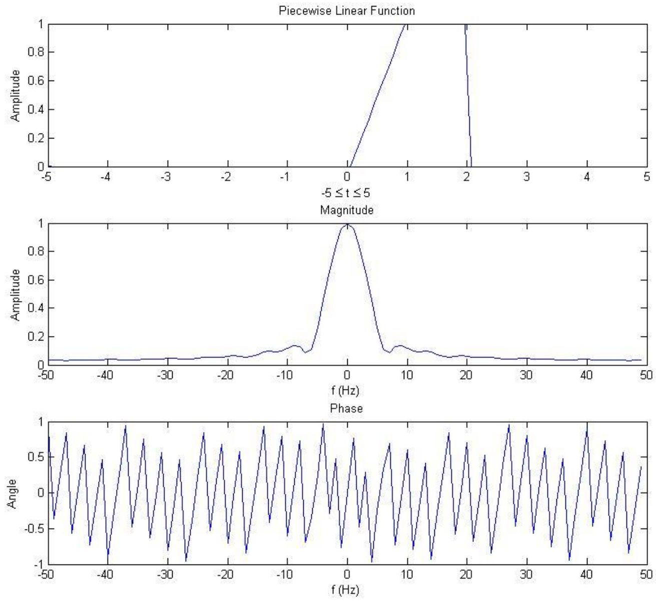
Fourier Transform

In order to fully understand how to utilize the Fast Fourier Transform in MATLAB we analyzed four piecewise linear functions and their transforms.

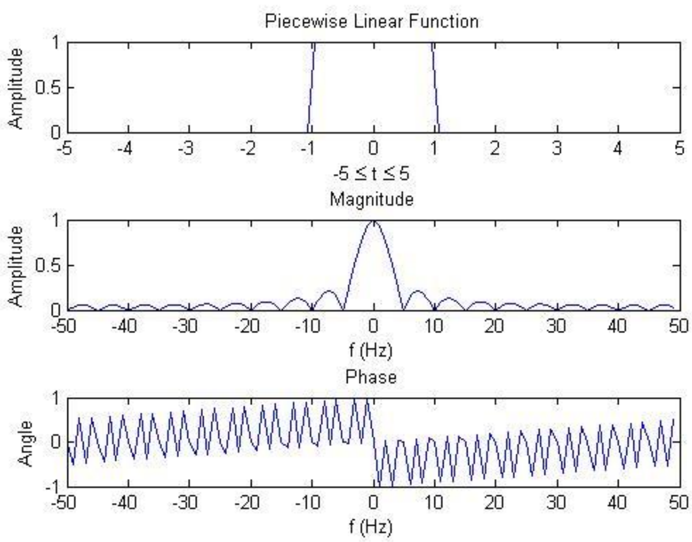
The first piecewise function we analyzed consisted of a ramp function that leveled off at t=0 and then turned off at t=1. The Fourier transform reveled the following phase and magnitude graphs.

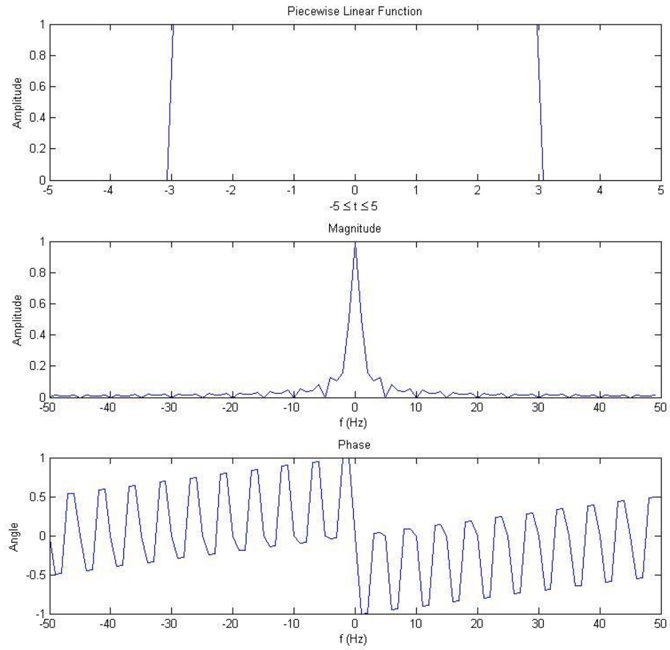


The second piecewise function is just like the first one except it is shifted to the right by a 1 time step. This justifiably results in a change in the phase of the system.



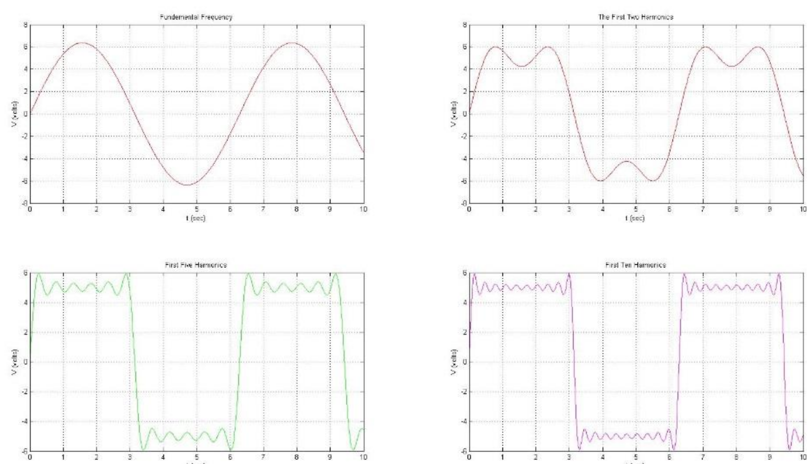
The next two piecewise functions are based off of a modification of the unitary step function. It can be seen that both the magnitude and phase change dramatically when the interval over which the system is active changes.



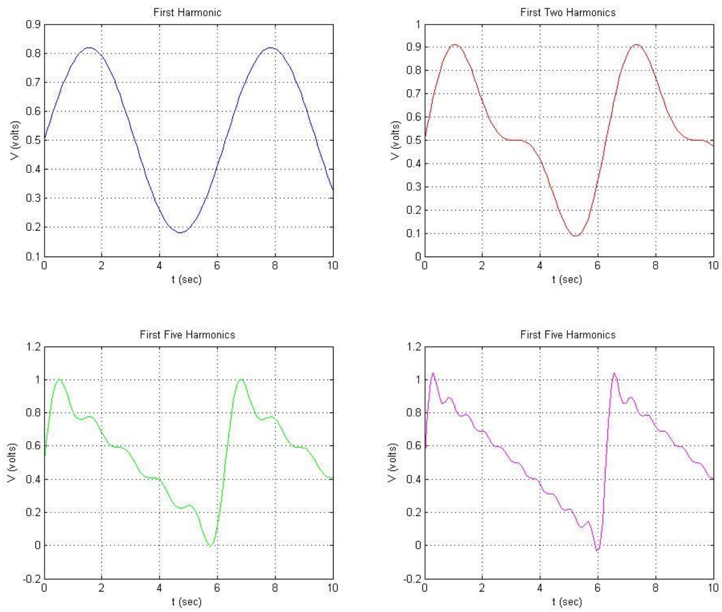


Circuit Analysis

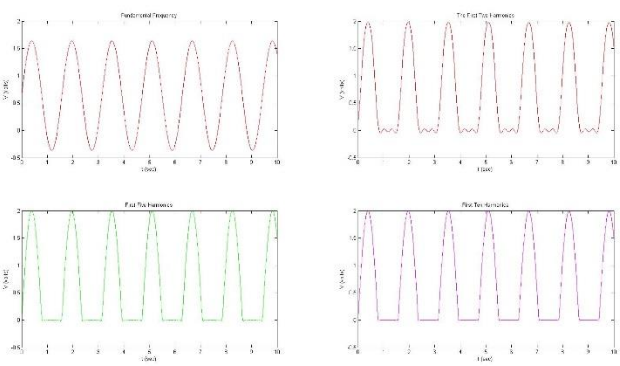
Finally we applied the knowledge which we gained from theoretically experimentation with the Fourier series analysis and applied it to relativistic function. First we analyzed the square wave and broke it up into its constituent harmonics, in order to justify how to produce such a wave.

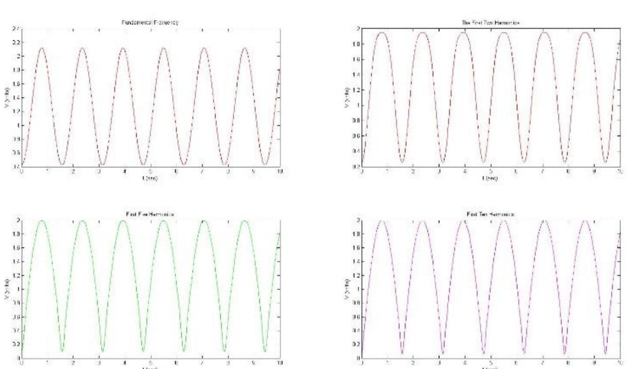


After completing a detailed graph of the square wave, we moved on to bigger fish. We analyzed the saw-tooth wave and saw that it took quite an array of harmonics to produce something relatively close to the desired ramp function.



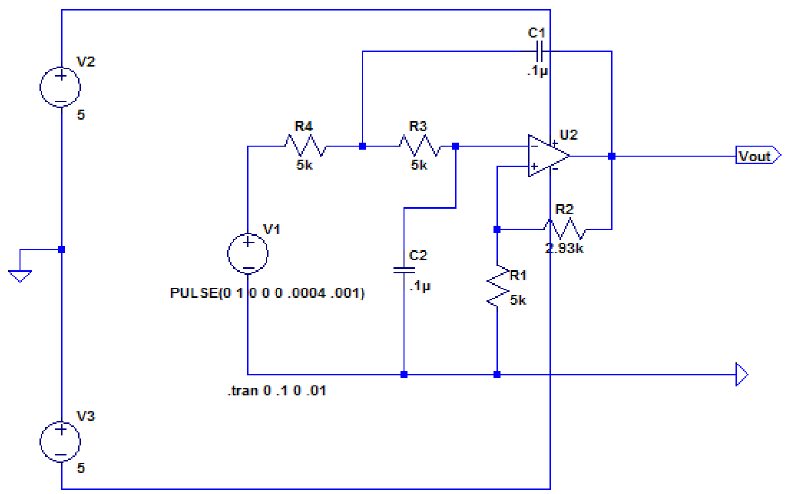
Concurrently, we examined both the half- and full-rectified sine wave forms in increasing number of mixed harmonics in order to determine their easy to reproduce using a series representation.

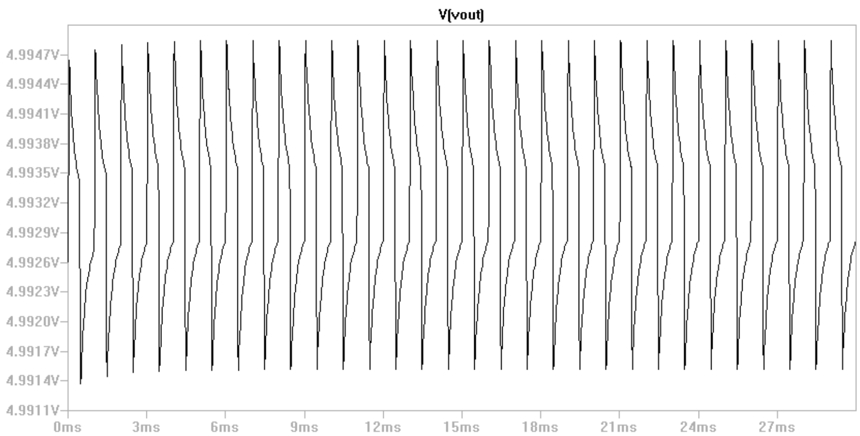




As it can be seen above, the full-rectified sine wave is easier to produce than the half-rectified one due to how closely it matches an original sine wave.

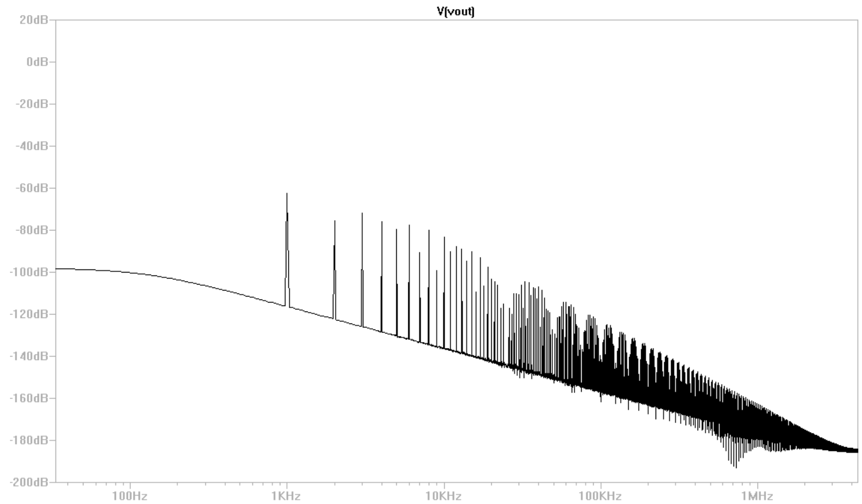
Finally, we analyzed an Op-Amp circuit (see below) to test it for its Fourier series representation of the output.





Based on the form of the temporal representation of the output, it could be assumed that the Fourier series representation of this response would only have a few harmonics with very limited magnitude corresponding to a large DC offset of frequency 0 Hz.

As it can be seen on the graph below, this is just the case. It has one frequency that has a significant magnitude at roughly 1 kHz and a bunch of very small ones out in the higher frequencies.



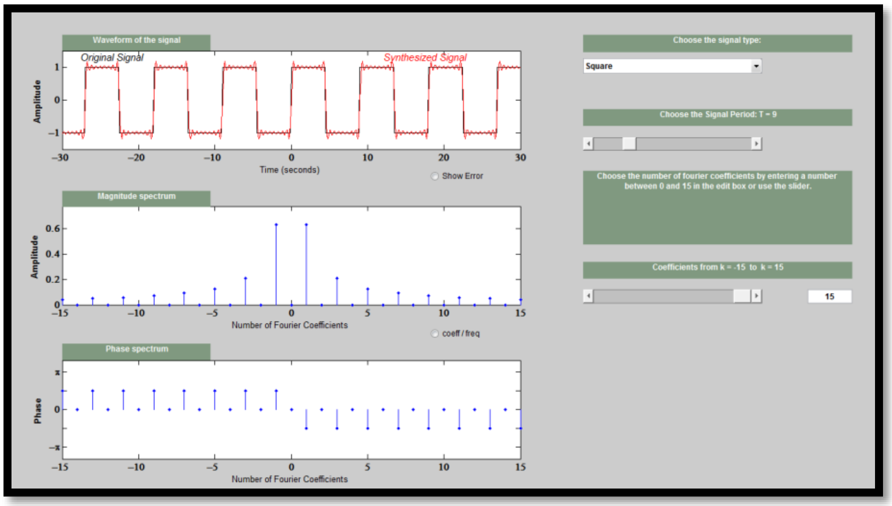
CONCLUSION

This lab helped solidify the concepts of the Fourier series and transform along with its various uses and applications. In order to fully analyze said wave, it must be analyzed by its appropriated Fourier series in order to distinguish the variety of frequencies that can be hidden in one single signal.

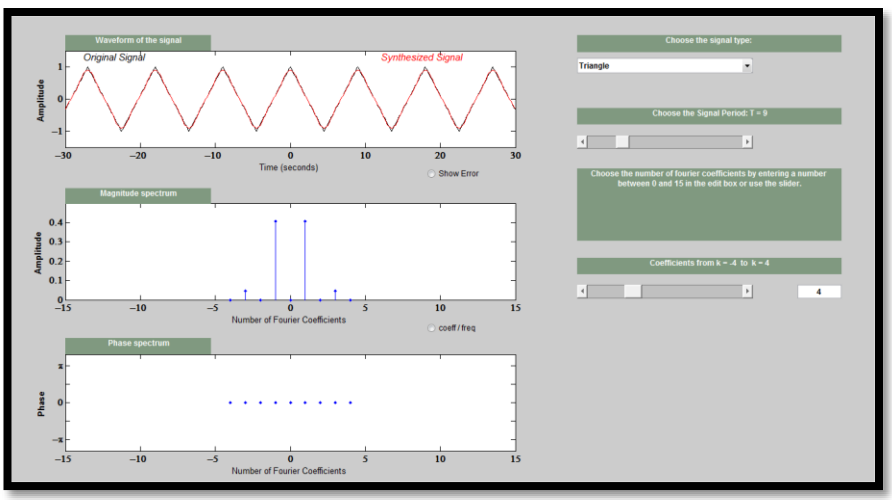
This lab has prepared my understanding for the various concepts that signal and communication theory have in store. I can’t wait to use my new found knowledge in all of the ways that Fourier analysis has to offer.

APPENDIX A

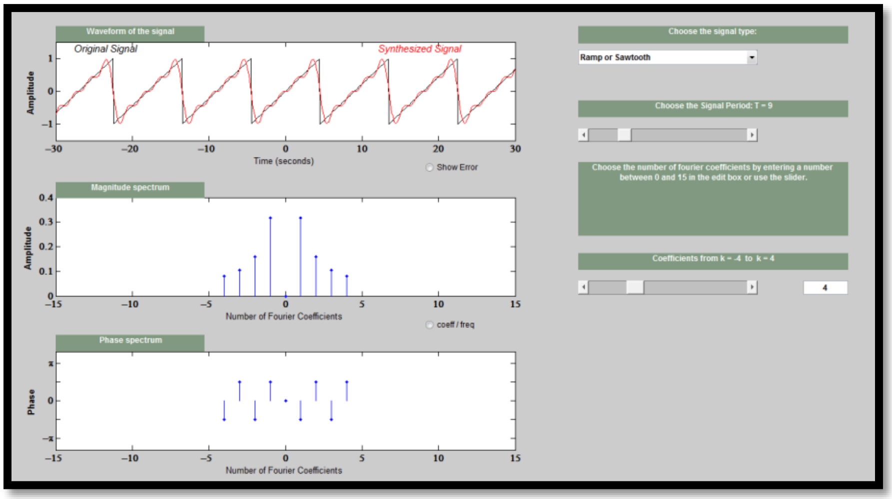
Square Wave

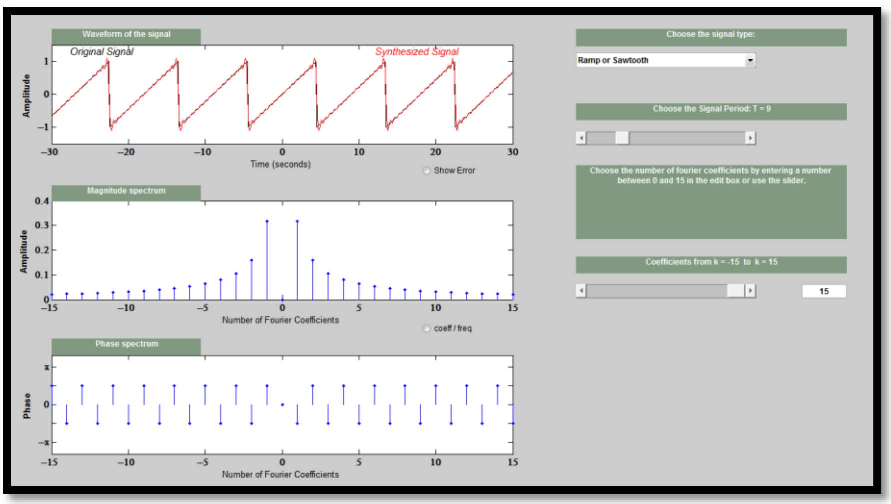


Triangle Wave

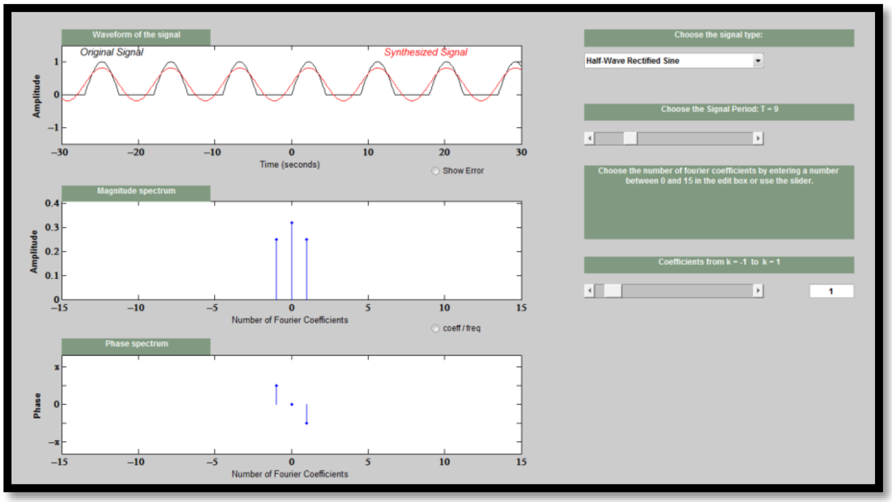


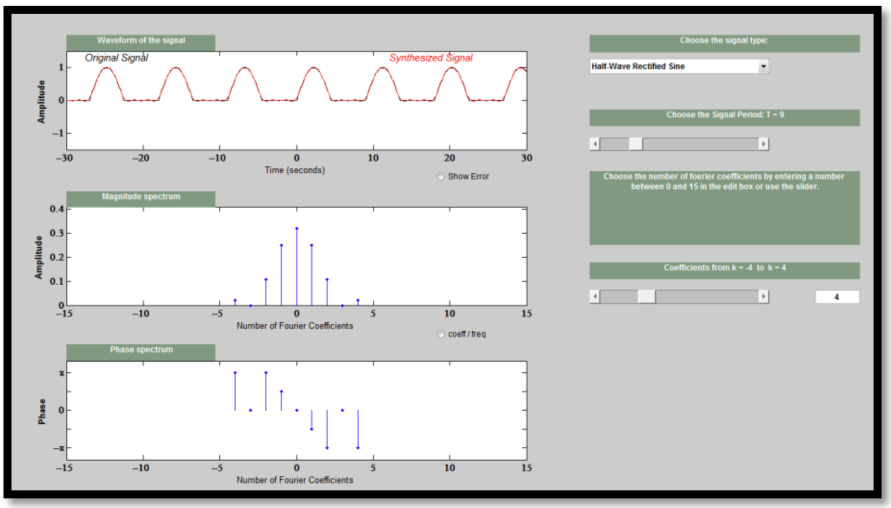
Sawtooth Wave



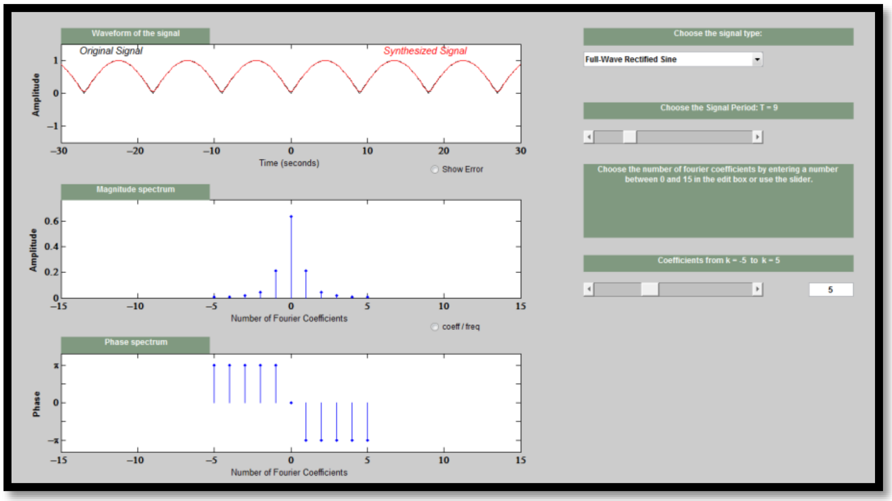


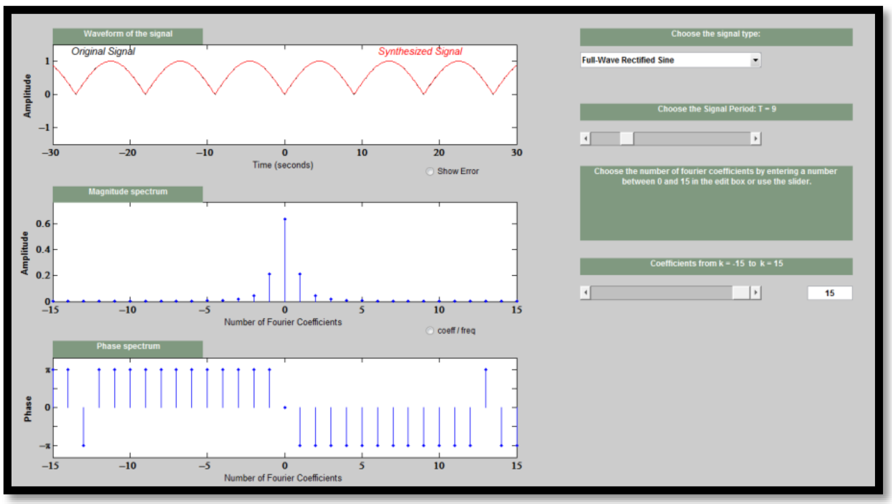
Half-Rectified Sine Wave





Full-Rectified Sine Wave





Appendix B

```
%% EE 225 Lab 4

%% Fourier Series
clear all
format SHORT

No = 32;
To = pi;
n = linspace(0,pi,No);
x = exp(-n/2);

X_f = fft(x,No);
X_f = [conj(X_f(No:-1:2)),X_f];
X_fmag = abs(X_f);
X_fangle = angle(X_f);
k = -No/2+1:No/2-1;

subplot(211)
stem(k,X_fmag(No/2+1:length(X_f)-No/2))
title('Magnitude')

subplot(212)
stem(k,X_fangle(No/2+1:length(X_f)-No/2))
title('Phase')

%% Fourier Transform

%% Part I

clear all
format short

N_pnts = 100; % t goes from -5 to 5

t0 = linspace(-5,(-1-1/10),N_pnts/2.5);
t1 = linspace(-1,0,11);
t2 = linspace((0+1/10),1,N_pnts/10);
t3 = linspace((1+1/10),5,N_pnts/2.5);
t = linspace(-5,5,101);

x_1 = [0*t0 (t1+1) 1 1 1 1 1 1 1 1 1 1 0*t3];

subplot(311)
plot(t,x_1)
title('Piecewise Linear Function')
xlabel('-5 \leq t \leq 5')
ylabel('Amplitude')

X1 = fft(x_1);
X1 = fftshift(X1);
X1mag = abs(X1);
X1mag = X1mag/max(X1mag); %Normalization
```

```
X1angle = angle(X1);
X1angle = X1angle/max(X1angle);
F1 = [-length(X1)/2:(length(X1)/2)-1]*100/length(X1);

subplot(312)
plot(F1,X1mag)
xlabel('f (Hz)')
ylabel('Amplitude')

subplot(313)
plot(F1,X1angle)
title('Phase')
xlabel('f (Hz)')
ylabel('Angle')

%% Part II

clear all
format short

N_pnts = 100; % t goes from -5 to 5

t0 = linspace(-5,(0-1/10),N_pnts/2);
t1 = linspace(0,1,N_pnts/10);
t2 = linspace(1,2,N_pnts/10);
t3 = linspace((2+1/10),5,N_pnts/(10/3));
t = linspace(-5,5,100);

x_2 = [0*t0 (t1) 1 1 1 1 1 1 1 1 1 1 0*t3];

subplot(311)
plot(t,x_2)
title('Piecewise Linear Function')
xlabel('-5 \leq t \leq 5')
ylabel('Amplitude')

X2 = fft(x_2);
X2 = fftshift(X2);
X2mag = abs(X2);
X2mag = X2mag/max(X2mag); %Normalization
X2angle = angle(X2);
X2angle = X2angle/max(X2angle);
F2 = [-length(X2)/2:(length(X2)/2)-1]*100/length(X2);

subplot(312)
plot(F2,X2mag)
xlabel('f (Hz)')
ylabel('Amplitude')

subplot(313)
plot(F2,X2angle)
title('Phase')
xlabel('f (Hz)')
ylabel('Angle')
```

```
%% Part III

clear all
format short

N_pnts = 100; % t goes from -5 to 5

t0 = linspace(-5,(-3-1/10),N_pnts/5);
t1 = linspace(-3,3,N_pnts/(5/3));
t2 = linspace(3+1/10,5,N_pnts/5);
t = linspace(-5,5,100);


x_3 = [0*t0 ones(1,60) 0*t2];

subplot(311)
plot(t,x_3)
title('Piecewise Linear Function')
xlabel('-5 \leq t \leq 5')
ylabel('Amplitude')

X3 = fft(x_3);
X3 = fftshift(X3);
X3mag = abs(X3);
X3mag = X3mag/max(X3mag); %Normalization
X3angle = angle(X3);
X3angle = X3angle/max(X3angle);
F3 = [-length(X3)/2:(length(X3)/2)-1]*100/length(X3);

subplot(312)
plot(F3,X3mag)
xlabel('f (Hz)')
ylabel('Amplitude')

subplot(313)
plot(F3,X3angle)
title('Phase')
xlabel('f (Hz)')
ylabel('Angle')


%% Part IV
clear all
format short

N_pnts = 100; % t goes from -5 to 5

t0 = linspace(-5,(-1-1/10),N_pnts/2.5);
t1 = linspace(-1,1,N_pnts/5);
t2 = linspace(1+1/10,5,N_pnts/2.5);
t = linspace(-5,5,100);
```

```
x_4 = [0*t0 ones(1,20) 0*t2];

subplot(311)
plot(t,x_4)
title('Piecewise Linear Function')
xlabel('-5 \leq t \leq 5')
ylabel('Amplitude')

X4 = fft(x_4);
X4 = fftshift(X4);
X4mag = abs(X4);
X4mag = X4mag/max(X4mag); %Normalization
X4angle = angle(X4);
X4angle = X4angle/max(X4angle);
F4 = [-length(X4)/2:(length(X4)/2)-1]*100/length(X4);

subplot(312)
plot(F4,X4mag)
xlabel('f (Hz)')
ylabel('Amplitude')

subplot(313)
plot(F4,X4angle)
title('Phase')
xlabel('f (Hz)')
ylabel('Angle')

%% Circuit Anlaysis

%% Sawtooth Wave
clear all
format SHORTE
t = linspace(0,10,100);
sin_val = sin(t);
yDC = 0.5 + sin(t)/pi;
yh = zeros(10,100);
yh(1,:) = yDC;

for k=2:10
    yh(k,:) = sin(k*t)/(k*pi);
end

%The Fundemental Frequency
subplot(221)
plot(t,yDC,'b')
title('First Harmonic')
xlabel('t (sec)')
ylabel('V (volts)')
grid on

%The first two harmonics

y2 = yDC + yh(2,:);

subplot(222)
```

```
plot(t,y2,'r')
title('First Two Harmonics')
xlabel('t (sec)')
ylabel('V (volts)')
grid on

%The first five harmonics
y5 = yDC;
for k=2:5
    y5 = y5 + yh(k,:);
end
subplot(223)
plot(t,y5,'g')
title('First Five Harmonics')
xlabel('t (sec)')
ylabel('V (volts)')
grid on

%The first ten harmonics
y10 = yDC;
for k=2:10
    y10 = y10 + yh(k,:);
end
subplot(224)
plot(t,y10,'m')
title('First Ten Harmonics')
xlabel('t (sec)')
ylabel('V (volts)')
grid on

%% Square Wave
t = linspace(0,10,1000);
Amp = 5;
zero = zeros(1,1000);
yh = zeros(10,1000);

for k=1:10
    yh(k,:) = (4*Amp/pi)*(1/((2*k)-1))*sin((2*k-
1)*t);
end

%The Fundemental Frequency

y = yh(1,:);

subplot(221)
plot(t,y,'r')
title('Fundemental Frequency')
xlabel('t (sec)')
ylabel('V (volts)')

%The first two harmonics
y2 = zero;
for k = 1:2
 y2 = y2 + yh(k,:);
```

```
end

subplot(222)
plot(t,y2,'r')
title('The First Two Harmonics')
xlabel('t (sec)')
ylabel('V (volts)')

%The first five harmonics
y5 = zero;
for k=1:6
    y5 = y5 + yh(k,:);
end
subplot(223)
plot(t,y5,'g')
title('First Five Harmonics')
xlabel('t (sec)')
ylabel('V (volts)')

%The first ten harmonics
y10 = zero;
for k=1:10
    y10 = y10 + yh(k,:);
end
subplot(224)
plot(t,y10,'m')
title('First Ten Harmonics')
xlabel('t (sec)')
ylabel('V (volts)')

%% Full-wave Rectified Sine
t = linspace(0,10,1000);
Amp = 2;
yh = zeros(10,1000);
yDC = 2*Amp/pi;

for k=1:10
    yh(k,:) = -(4*Amp/pi)*(1/((4*k^2)-1))*cos(4*k*t);
end

%The Fundemental Frequency

y = yDC + yh(1,:);

subplot(221)
plot(t,y,'r')
title('Fundemental Frequency')
xlabel('t (sec)')
ylabel('V (volts)')


%The first two harmonics
y2 = yDC;
for k = 1:2
 y2 = y2 + yh(k,:);
```

```
end

subplot(222)
plot(t,y2,'r')
title('The First Two Harmonics')
xlabel('t (sec)')
ylabel('V (volts)')

%The first five harmonics
y5 = yDC;
for k=1:6
    y5 = y5 + yh(k,:);
end
subplot(223)
plot(t,y5,'g')
title('First Five Harmonics')
xlabel('t (sec)')
ylabel('V (volts)')

%The first ten harmonics
y10 = yDC;
for k=1:10
    y10 = y10 + yh(k,:);
end
subplot(224)
plot(t,y10,'m')
title('First Ten Harmonics')
xlabel('t (sec)')
ylabel('V (volts)')

%% Full-wave Rectified Sine
t = linspace(0,10,1000);
Amp = 2;
yh = zeros(10,1000);
y1 = Amp/pi + Amp*sin(4*t)/2;

for k=1:10
    yh(k,:) = -(2*Amp/pi)*(1/((4*k^2)-
1))*cos(2*4*k*t);
end

%The Fundemental Frequency

y = y1;

subplot(221)
plot(t,y,'r')
title('Fundemental Frequency')
xlabel('t (sec)')
ylabel('V (volts)')



%The first two harmonics
y2 = y;
for k = 1:2
```

```
 y2 = y2 + yh(k,:);
end

subplot(222)
plot(t,y2,'r')
title('The First Two Harmonics')
xlabel('t (sec)')
ylabel('V (volts)')

%The first five harmonics
y5 = y;
for k=1:6
    y5 = y5 + yh(k,:);
end
subplot(223)
plot(t,y5,'g')
title('First Five Harmonics')
xlabel('t (sec)')
ylabel('V (volts)')

%The first ten harmonics
y10 = y;
for k=1:10
    y10 = y10 + yh(k,:);
end
subplot(224)
plot(t,y10,'m')
title('First Ten Harmonics')
xlabel('t (sec)')
ylabel('V (volts)')
```