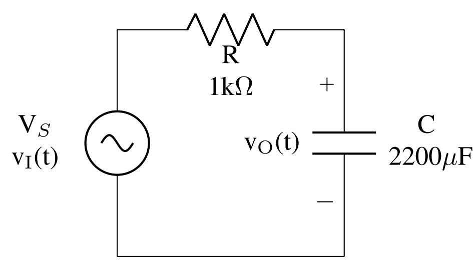
CHAPTER 14

FOURIER SERIES AND CIRCUIT ANALYSIS

Estimate $v_O(t)$ using the complex Fourier series (k=0..100), the function $v_I(t) = sin(\frac{4\pi}{3}t)\,[u(t+3) - u(t-0)]$ on the interval $-4 \leq t \leq 4$, and the circuit below:




```
clear all
close all
clc
format short eng

syms x w t T integrand

n=100; %# of coefficients to find
T0=8; %Frequency of the cosine
w0=(2*pi)/T0; % period that corresponds to w0
x=@(t) sin((4*pi/3).*t).*(heaviside(t+3)-heaviside(t));
t0=-4; t1=4;

tk=t0:T0/1000:t1;
figure(1)
plot(tk,x(tk),'LineWidth',3)

for k=0:n;
```

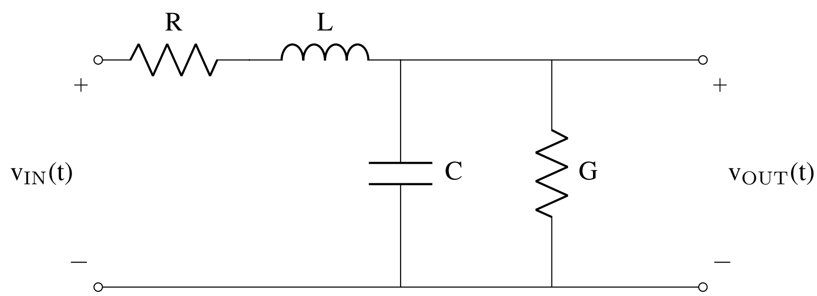
```
    integrand=@(t) (x(t)).*cos(k.*w0.*t);
    a(k+1)=(2/T0)*integral(integrand,t0,t1);

    integrand=@(t) (x(t)).*sin(k.*w0.*t);
    b(k+1)=(2/T0)*integral(integrand,t0,t1);
end

figure(2)
hold on;grid on
plot(0:length(a)-1,a,'x','LineWidth',4,'MarkerSize',10)
plot(0:length(a)-1,b,'ro','LineWidth',4,'MarkerSize',10)

xlabel('Multiple of the fundamental frequency - k')
ylabel('Coefficient')
legend('a_k','b_k')

x_hat=(a(1)/2);
for k=1:n;
    x_hat=x_hat+ a(k+1).*cos(k.*w0.*tk)+ b(k+1).*sin(k.*w0.*tk);
end
set(findall(gcf,'-property','FontSize'),'FontSize',14)

figure(3)
hold on;grid on
plot(tk,x(tk),'ro')
plot(tk,x_hat,'LineWidth',3)

%perform ckt analysis
syms s
Hs=1/(2.2*s+1)

%DC component output
vo_hat=(a(0+1)/2)*(1); %DC input times DC gain

for k=1:n
    H=subs(Hs,s,j*k*w0); % Find gain at multiple of w0
    Vi=a(k+1)-j*b(k+1);
    Vo=Vi*H;
    vo_hat=vo_hat+eval(abs(Vo)).*cos(k.*w0.*tk+eval(angle(Vo)));
end

plot(tk,vo_hat,'g','LineWidth',3)
xlabel('Time (s)')
legend('x(t)','x\_hat(t)')
set(findall(gcf,'-property','FontSize'),'FontSize',14)
```

14.1 Application to Data Transmission

Let's revisit an example from an earlier chapter. We looked at the frequency response of a transmission line when we were first developing transfer functions and Bode plots. Now we have the tools necessary to send a signal down that transmission line and determine what comes out the other end.

EXAMPLE 14.1

Here is the transmission line model we studied earlier



I'll include the parameters again for completeness. I'll also include the values we used for two length of cable and I will add a third. The third length we'll consider is 3280 ft (1000 m), far longer than the limit for an ethernet cable. Often when I'm trying to determin why something is a problem I"ll consider an extreme circumstance. The question we're answering here is why cable length is an issue for digital transmission. We could, and will, study the case at the specified limit for cable length but that may not reveal what happens when we exceed the limit.

Component	Value	50 ft	328 ft	3280 ft
R	51 Ω/1,000 ft	2.55 Ω	16.73 Ω	167.3 Ω
L	160 μH/1,000 ft	8 μH	52.48 μH	524.8 μH
C	13.5 nF/1,000 ft	675 pF	4.43 nF	44.3 nF
G	100 μS/1,000 ft	5 μS	32.8 μS	328 μS

My primary purpose for including this problem is to introduce more complicated circuit analysis to our discussion regarding transfer functions. The study of transmission lines is just a nice practical application. To meet the primary goal of this example I will use mesh analysis to develop the expression for the output voltage. I've labeled the two mesh currents in the schematic above to help up write the two KVL equations needed to analyze this circuit.

$$\begin{array}{ll} \textbf{KVL } \mathbf{I_1} & (R + Z_L + Z_C)I_1 + (-Z_C)I_2 = V_{IN} \\ \textbf{KVL } \mathbf{I_2} & (-Z_C)I_1 + (Z_C + {}^1\!/_G)I_2 = 0 \end{array}$$

We can use MATLAB to solve the system of equations. Once we have a solution for I_1 and I_2 we can find the output voltage with

$$V_{OUT} = ({}^1\!/_G)\, I_2$$

and find the transfer function with

$$H(s) = \frac{V_{OUT}}{V_{IN}}$$

Next I'll use MATLAB to substitute the part values for both lengths of cable and generate the Bode plots for both cables.

```
%% Finding a transfer function with Mesh Analysis
clear all
close all
clc
format short eng

syms R G L C s Vin I1 I2
Zl=s*L;
Zc=inv(s*C);

eqn(1)=(R+Zl+Zc)*I1+(-Zc)*I2==Vin;
eqn(2)=(-Zc)*I1+(Zc+(1/G))*I2==0;
sol=solve(eqn,I1,I2);
Vout=sol.I2*(1/G);

```

```
H=Vout/Vin;% Gain is output over input

H1s=subs(H,[R G L C],[200 5e-6 8e-6 675e-12]); %50 ft values
H2s=subs(H,[R G L C],[267.3 32.8e-6 52.48e-6 4.43e-9]); %328 ft values
H3s=subs(H,[R G L C],[2673 328e-6 524.8e-6 44.3e-9]); %3280 ft values
pretty(vpa(simplify(H1s),3)) %Print the transfer function in the Command Window
pretty(vpa(simplify(H2s),3)) %Print the transfer function in the Command Window
pretty(vpa(simplify(H3s),3)) %Print the transfer function in the Command Window
f=logspace(0,10,1000);
w=2*pi*f;
H1=subs(H1s,s,j*w);
H2=subs(H2s,s,j*w);
H3=subs(H3s,s,j*w);

figure(1)
subplot(2,1,1)
semilogx(f,20*log10(abs(H1)),'LineWidth',2)
hold on
semilogx(f,20*log10(abs(H2)),'LineWidth',2)
semilogx(f,20*log10(abs(H3)),'LineWidth',2)
grid on
ylabel('Gain (dB)')
legend('50 ft','328 ft','3280 ft')

fig=gcf;
set(findall(fig,'-property','FontSize'),'FontSize',18)

subplot(2,1,2)
semilogx(f,angle(H1)*(180/pi),'LineWidth',2)
hold on
semilogx(f,angle(H2)*(180/pi),'LineWidth',2)
semilogx(f,angle(H3)*(180/pi),'LineWidth',2)
grid on
xlabel('Frequency (Hz)')
ylabel('Phase Shift (deg)')

fig=gcf;
set(findall(fig,'-property','FontSize'),'FontSize',18)

%% Break the signal into phasors

syms x w t T integrand

n=100;
A=2.5;
T0=1/20e3;
w0=(2*pi)/T0;
x=@(t) A*sign(sin(w0*(t)))+A;
t0=0;t1=T0;

for k=0:n;
    integrand=@(t) (x(t)).*cos(k.*w0.*t);
    a(k+1)=(2/T0)*integral(integrand,t0,t1);
```

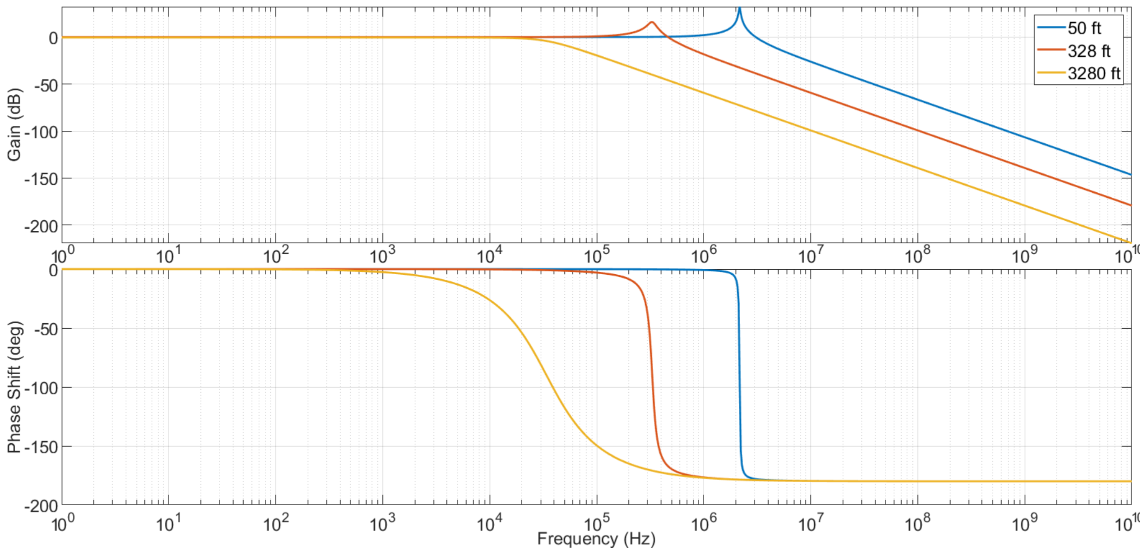
```
    integrand=@(t) (x(t)).*sin(k.*w0.*t);
    b(k+1)=(2/T0)*integral(integrand,t0,t1);
end

figure(2)
hold on;grid on
semilogy(0:length(a)-1,a,'x','LineWidth',4,'MarkerSize',10)
semilogy(0:length(a)-1,b,'ro','LineWidth',4,'MarkerSize',10)

xlabel('Multiple of the fundamental frequency - k')
ylabel('Coefficient')
legend('a_k','b_k')

tk=0:T0/1000:2*T0;
x_hat=(a(1)/2);
for k=1:n;
    x_hat=x_hat+ a(k+1).*cos(k.*w0.*tk)+ b(k+1).*sin(k.*w0.*tk);
end
set(findall(gcf,'-property','FontSize'),'FontSize',14)

figure(3)
hold on;grid on
plot(tk,x(tk),'ro-','LineWidth',3)
plot(tk,x_hat,'LineWidth',3)
xlabel('Time (s)')
legend('x(t)','x\_hat(t)')
set(findall(gcf,'-property','FontSize'),'FontSize',14)


%%
%DC component output
vo_hat=(a(0+1)/2)*(1); %DC input times DC gain

for k=1:n
    H=subs(H1s,s,j*k*w0); % Find gain at multiple of w0
    Vi=a(k+1)-j*b(k+1);
    Vo=Vi*H;
    vo_hat=vo_hat+eval(abs(Vo)).*cos(k.*w0.*tk+eval(angle(Vo)));
end

plot(tk,vo_hat,'g','LineWidth',3)
l1=line([(2/6)*T0,(2/6)*T0],[-1,6]);
l2=line([(5/6)*T0,(5/6)*T0],[-1,6]);
l3=line([(8/6)*T0,(8/6)*T0],[-1,6]);
l4=line([(11/6)*T0,(11/6)*T0],[-1,6]);
set([l1 l2 l3 l4],'LineStyle','--','LineWidth',2,'Color',[0 0 0])

a = text((1.8/6)*T0, 1.5,'RX Sample','Rotation',90);
a = text((4.8/6)*T0, 1.5,'RX Sample','Rotation',90);
a = text((7.8/6)*T0, 1.5,'RX Sample','Rotation',90);
a = text((10.8/6)*T0, 1.5,'RX Sample','Rotation',90);
xlabel('Time (s)')
legend('x(t)','x\_hat(t)')
xlim([0,2*T0])
set(findall(gcf,'-property','FontSize'),'FontSize',14)
```

The transfer function for the 50 foot cable is

$$H(s) = \frac{V_{OUT}}{V_{IN}} = \frac{1}{5.4e{-}15s^2 + 1.76e{-}9s + 1}$$

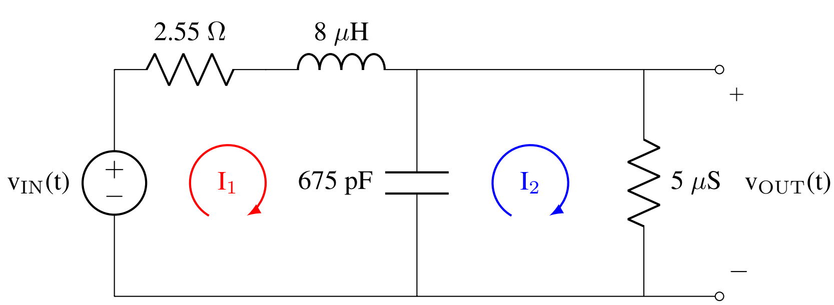
The transfer function for the 328 foot cable is

$$H(s) = \frac{V_{OUT}}{V_{IN}} = \frac{1}{232e{-}15s^2 + 75.8e{-}9s + 1}$$

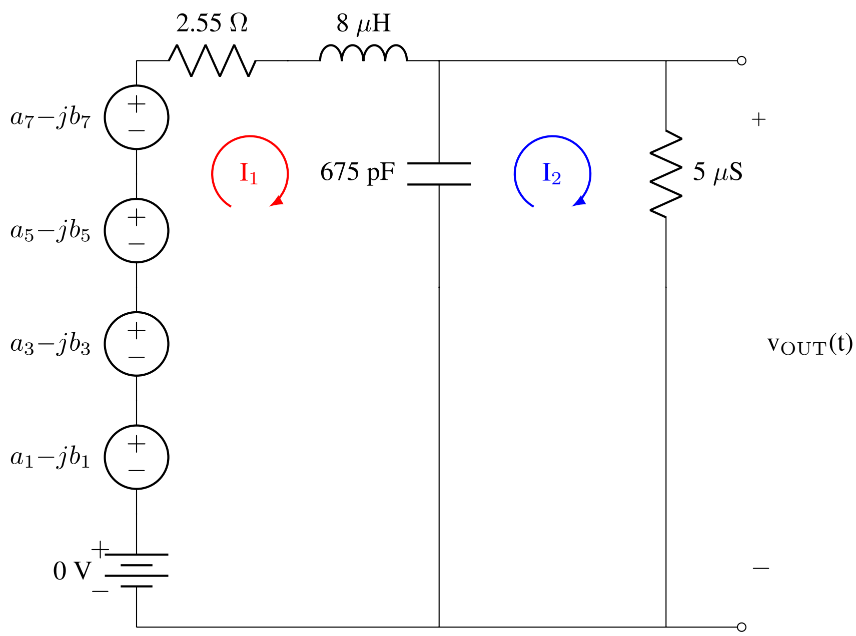


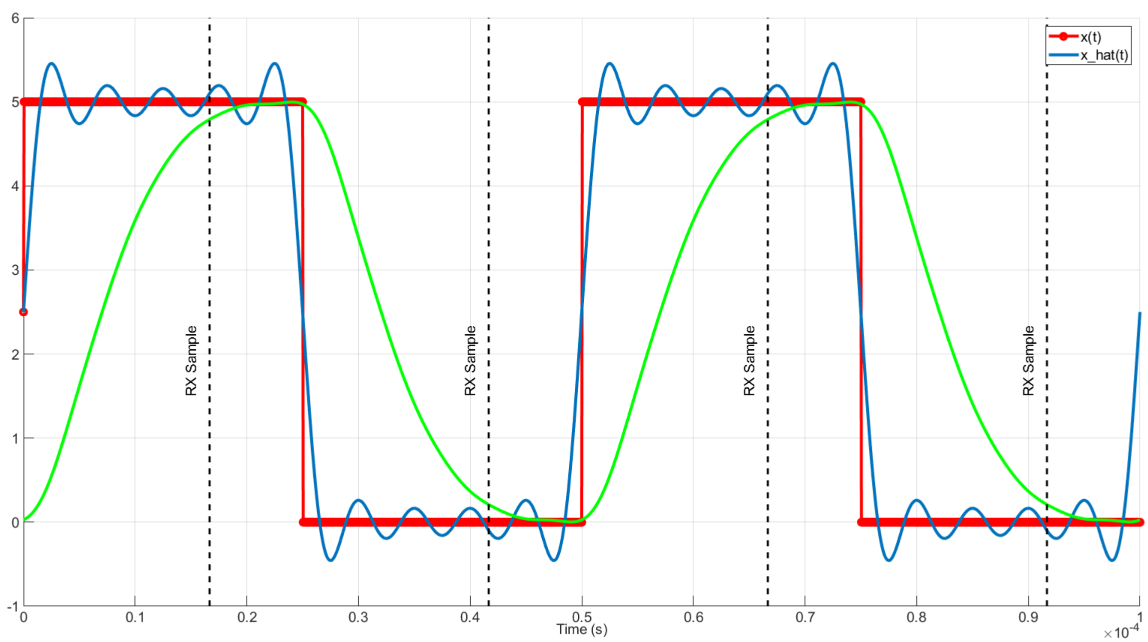
k	a_k	b_k
0	0	0
1	0	6.366
2	0	0
3	0	2.122
4	0	0
5	0	1.273
6	0	0
7	0	0.909

We can now apply the signal to the transmission line using an approximation based on the Fourier series. The voltage is applied to the transmission line as $v_{IN}(t)$ shows in the schematic below



Just as in the previous example we can break the voltage source into a source for each individual frequency. Each source, other than the DC source, will have a value of $a_k - jb_k$ using that values from the previous table. . I've omitted the sources for $k = 2, 4, 6$ as they have values of 0 V. The DC source also has a value of 0 V but I included it as it is treated separately from the AC sources.





14.2 An Aperiodic Example