Scilab Manual for ELECTRONICS DESIGN AUTOMATION LAB

by Prof Shanu N
Electronics Engineering
College of engineering Attingal[1]

Solutions provided by
Prof Shanu N
Electronics Engineering
College of engineering Attingal

May 2, 2024

[1]Funded by a grant from the National Mission on Education through ICT, http://spoken-tutorial.org/NMEICT-Intro. This Scilab Manual and Scilab codes written in it can be downloaded from the "Migrated Labs" section at the website http://scilab.in

Contents

List of Experiments

List of Figures

Experiment: 1

Generate and plot various signals like sine square, pulse in same window

Scilab code Solution 1.1 1

```
clear; clc;
//Scilab 5.5.2, Ubuntu 16.04 LTS
//Generate and plot various signals like pulse,
   square,sine in same window
t=[1:240];//x axis
y=[zeros(1,100),ones(1,40),zeros(1,100)]// y-axis
subplot(311)//figure divided to 3 plots
plot2d(t,y,rect=[1,0,220,2])
xlabel('time')
ylabel('Amplitude')
title ('Square Pulse')
t=(0:0.1:6*%pi)';
subplot(312)
plot2d(t,squarewave(t));
xlabel('time')
```

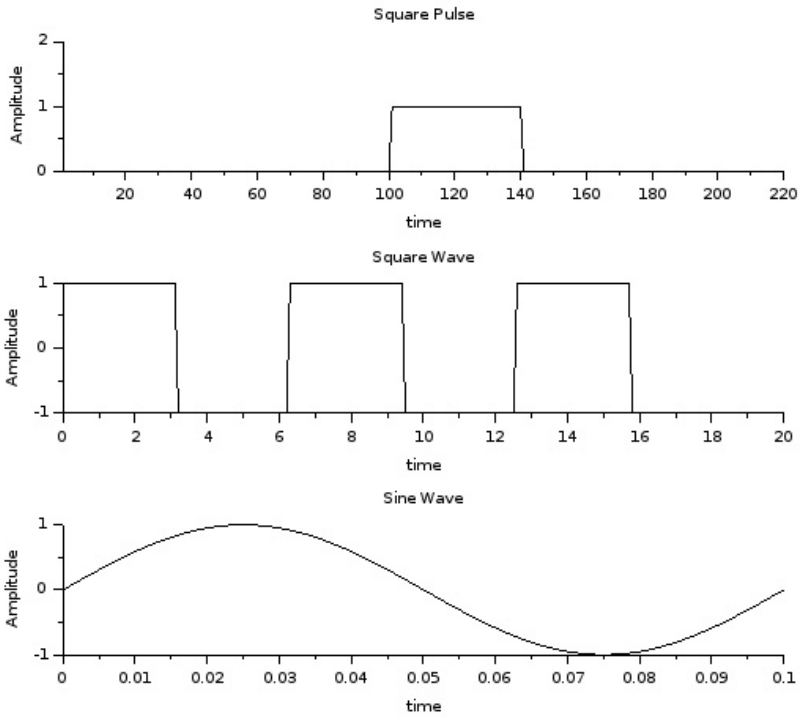
Square Pulse
Amplitude
time
Square Wave
Amplitude
time
Sine Wave
Amplitude
time

Figure 1.1: 1

```
15 ylabel('Amplitude')
16 title ('Square Wave')
17 Fs=8000;
18 t=[0:0.1*Fs]/Fs;
19 y1=sin(2*%pi*10*t);
20 subplot(313);
21 plot2d(t,y1);
22 xlabel('time')
23 ylabel('Amplitude')
24 title ('Sine Wave')
```

Experiment: 2

Plot the diode/transistor characteristics

Scilab code Solution 2.2 2

```
clear; clc;
//Scilab 5.5.2, Ubuntu 16.04 LTS
//Plot the diode/transistor characteristics
Io = 2e-8;//Reverse saturation current
Id = [0:0.1:5]' * 10^(-3);
Vd = 0.052*log(Id/Io + 1);//Diode  equation
plot2d(Vd,Id*10^3)
xlabel('Vd(V)');
ylabel('Id (mA)')
title('Diode Characteristics')
```

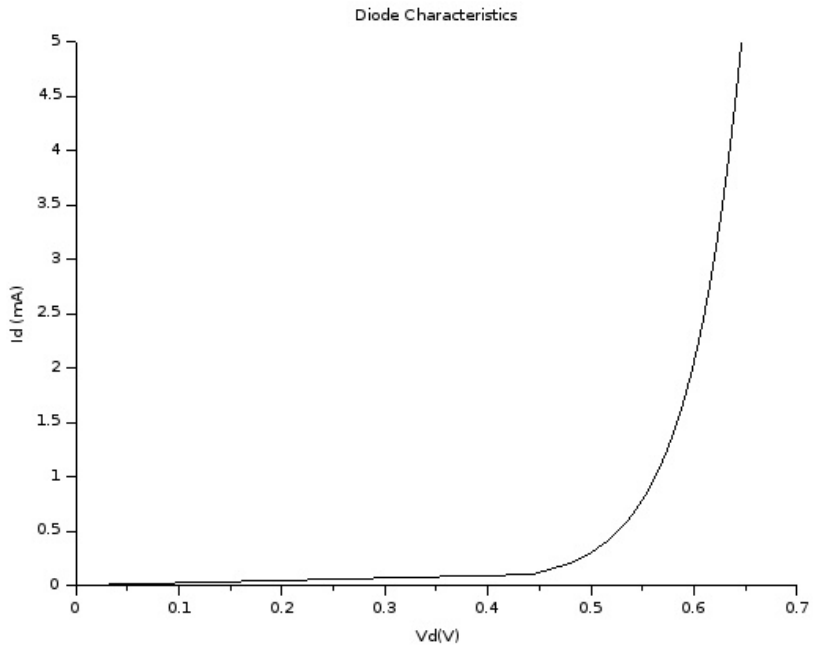
Diode Characteristics
Id (mA)
Vd(V)
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7

Figure 2.1: 2

Experiment: 3

Find the poles and zeros hence plot the transfer functions/polynomials

Scilab code Solution 3.3 3

```
1  clear; clc;
2  //Scilab 5.5.2, Ubuntu 16.04 LTS
3  //Find the poles and zeros hence plot the transfer
       functions/polynomials
4  s=poly(0,'s');
5  h=1/real((s+2*%pi*(15+100*%i))*(s+2*%pi*(15-100*%i))
       );//Transfer function
6
7  //h=(1+2*%s+5*%s^2)/poly(1:3,'s','coeff')//
       another example
8  h=syslin('c',h);//continuous system
9  bode(h,10,1000,.01);//bode plot
10
11 figure();//new figure window
12
```

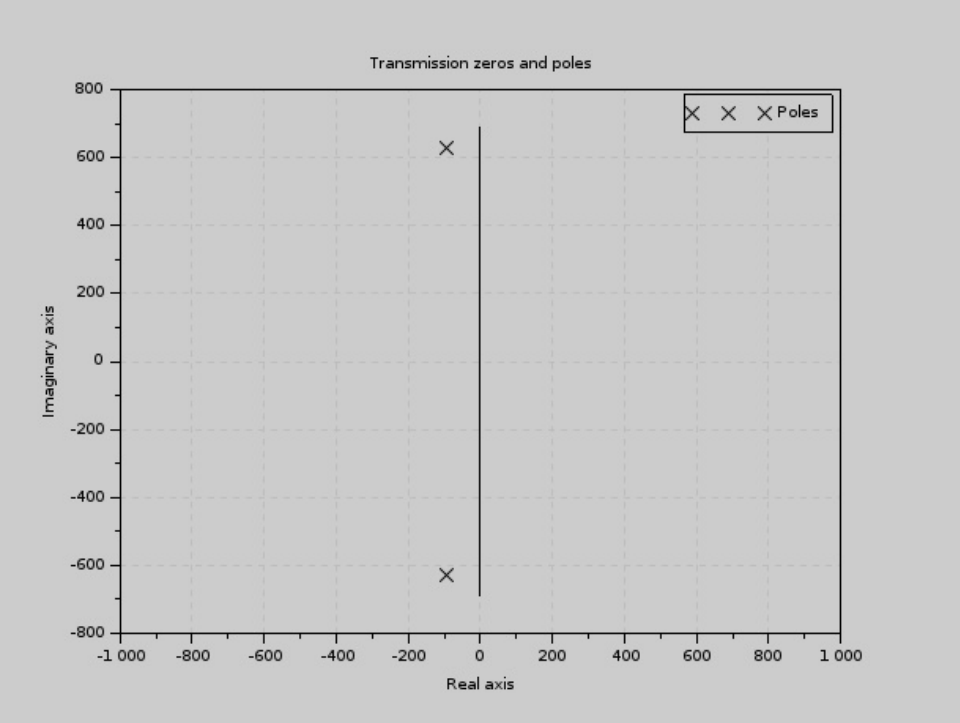
Transmission zeros and poles
Poles
Imaginary axis
Real axis
800
600
400
200
0
-200
-400
-600
-800
-1 000
-800
-600
-400
-200
0
200
400
600
800
1 000

Figure 3.1: 3

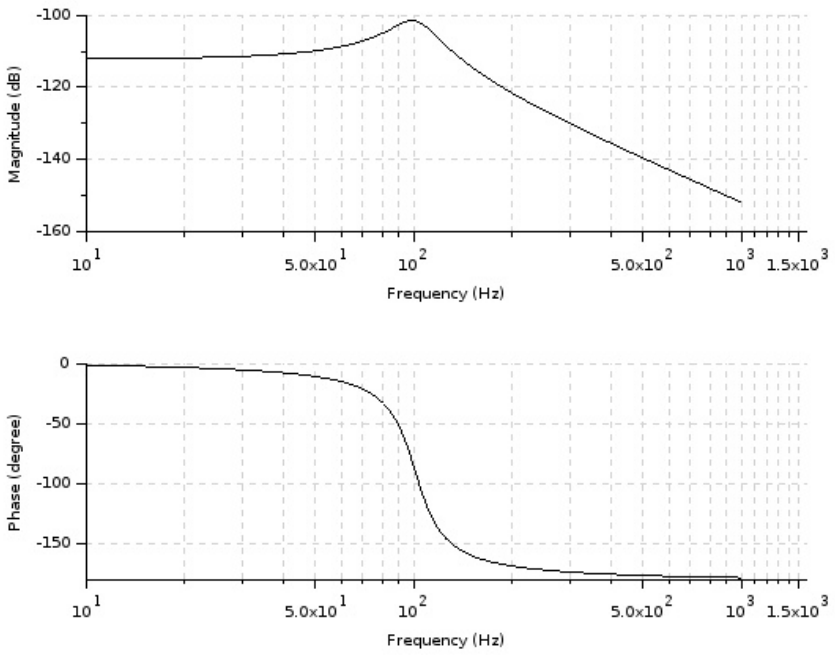

Figure 3.2: 3

```
13 plzr(h);//pole zero plot
```

Experiment: 4

Plot a full wave rectified waveform using Fourier series

Scilab code Solution 4.4 4

```
clear; clc;
//Scilab 5.5.2, Ubuntu 16.04 LTS
//Plot a full wave rectified waveform using Fourier series
dt=0.05;A=5;w0=1;//Amplitude=5V
t=20.0/dt +1;//no. of points on x-axis
//regeneration from Fourier coefficients of FWR signal
for n=-10:10
    for m=1:t
        sig1(n+11,m)=((2*A/(%pi*(1-4*n^2)))*exp(-1*%i*n*w0*dt*(m-1)));
end
end
for m=1:t
    sig2=sig1(:,m);
    sig3(m)=sum(sig2);
```

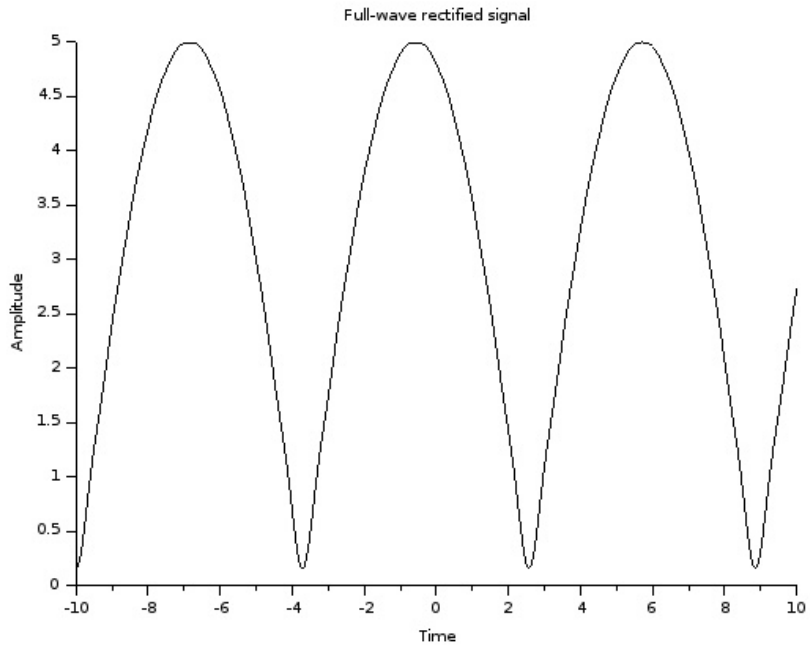
Full-wave rectified signal
Amplitude
Time
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
-10
-8
-6
-4
-2
0
2
4
6
8
10

Figure 4.1: 4

```
15 end
16 sig=sig3';
17 t=-10:0.05:10.0;// x-axis defined
18 plot2d(t,sig)
19 xlabel('Time')
20 ylabel('Amplitude')
21 title('Full-wave rectified signal')
```