# Modeling of Neuronal RC Circuit and Exploration of HH model

A dissertation presented
by

Baihan Lin

to
NBIO 303: Computational Neuroscience,
Department of Physiology and Biophysics

in partial fulfillment of the requirements
for the credits of
Computational Neuroscience
in the subject of
Neurobiology

University of Washington
Seattle, Washington, USA

Februrary 2014

**Mentors:**
Prof. Adrienne Fairhall
Dr. Alison Weber

**Author:**
Baihan Lin

**Abstract:**

This thesis is to explore the utilization of MATLAB into the modeling and visualization of neuron RC circuits. The problem discussed and examined here is Filtering of Input (what matters in driving the membrane response), Summation of simultaneous impulses (does conductance "help or hurt"), and HH model (Exploration of parameters in sinusoidal model). In general, the objective of the thesis is to better understanding the relationship between voltage, spikes, current and conductance, through a series of illustrative graphs I made, including a reminiscence of my previous project on tuning curve, here discussing firing-rate—current tuning curve.

**Keyword:**

Matlab, RC circuits, HH model, firing-rate-current tuning curve

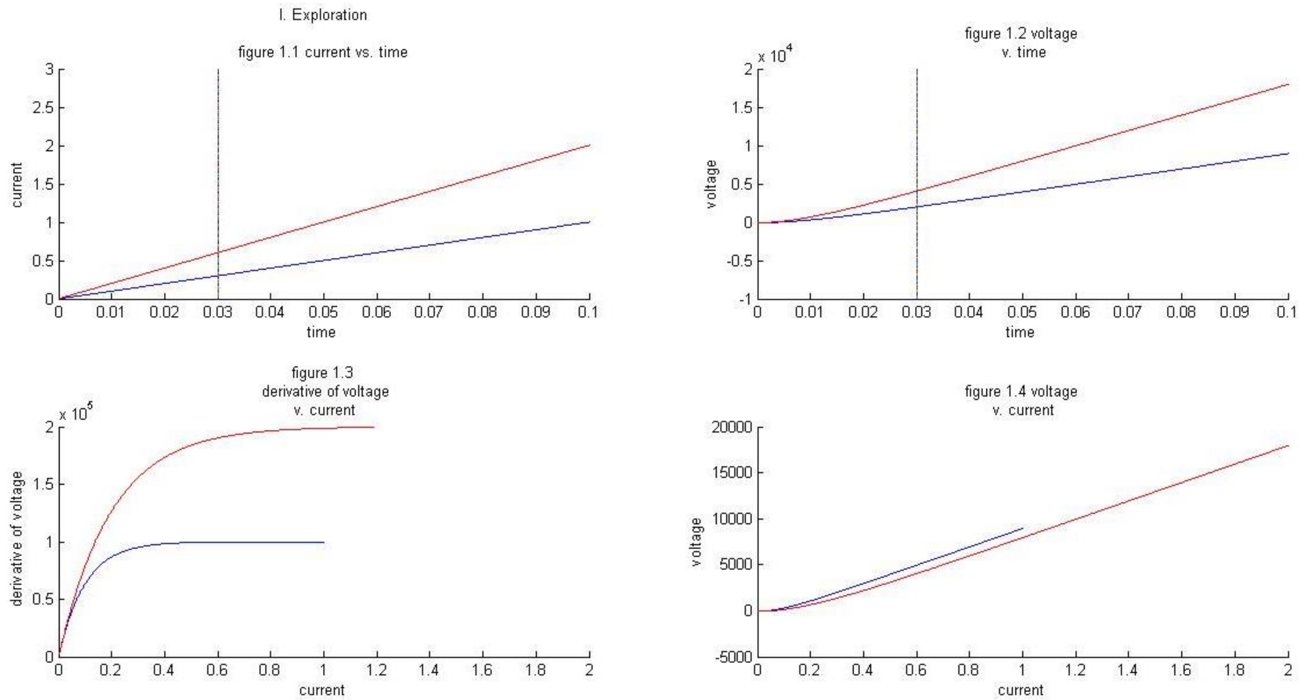
# Contents:

# 1.Filtering of Input

## (what matters in driving the membrane response)

```
% Question 1: Filtering of Inputs
% plotting the relevant voltage and current traces

% ================== Here goes the simplistic attempt ====================
% I. Exploration:
% Try using two simplistic linear models for I (current) and explore its
% relationship with drvtV (derivative of V) and V (voltage), this part is
% a great starting point because not only did it help find out the pattern
% of voltage changing with current, it standardized a method to test,
% compare and readapt models.

C = 10^-6;  % 1 uF (per cm^2)
R = 10^4;   % 10 kOhms (per cm^2)

tMax = .1;
dt = .0001;
tVec = 0:dt:tMax;
nTimeSteps = length(tVec);

Vrest = -70;

V = Vrest*ones(2,nTimeSteps);
I = zeros(2,nTimeSteps);
drvtV = ones(2,nTimeSteps);

for k = 1:2  % try two different step sizes
    for t = 1:nTimeSteps
        I(k,t) = 10^-3*k*t;
        dV = (I(k,t)-(V(k,t)-Vrest)/R)/C * dt;
        drvtV(k,t) = dV/dt;
        if t < nTimeSteps
            V(k,t+1) = V(k,t) + dV;
        end
    end
end
```

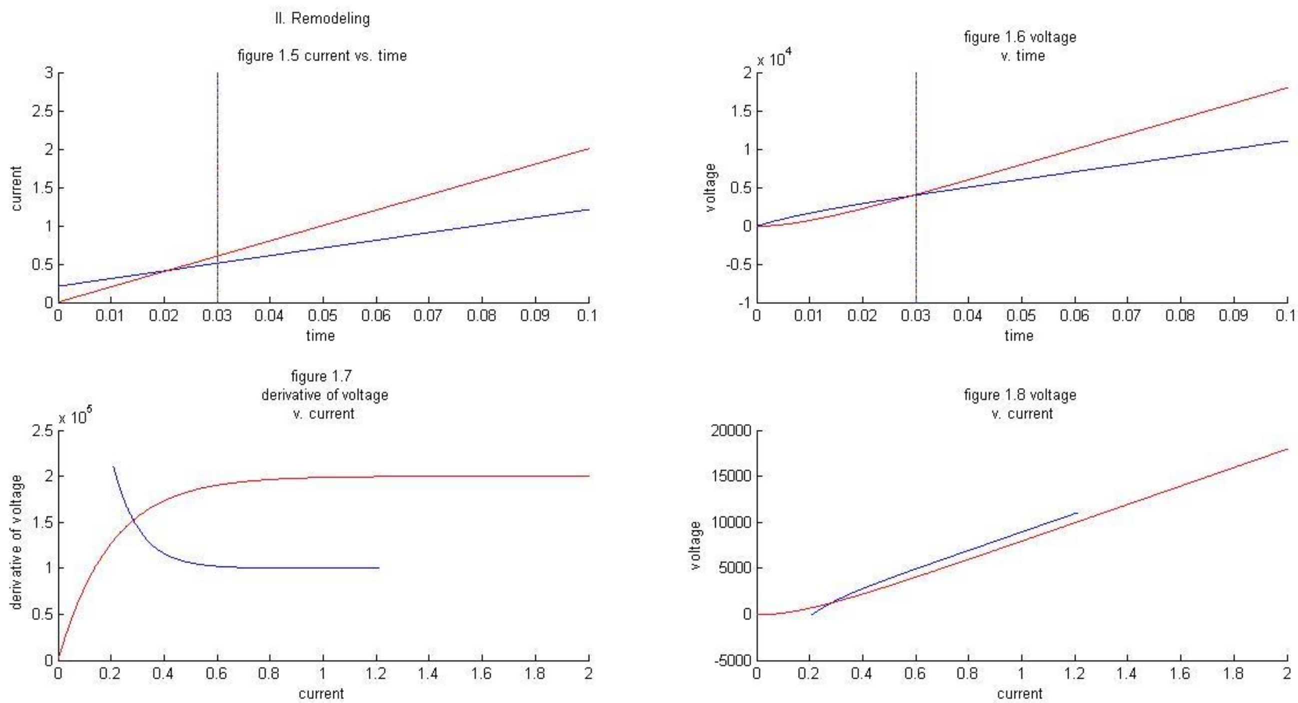
```
figure;

subplot(2,2,1); hold on;
plot(tVec,I(1,:))
plot(tVec,I(2,:),'r')
plot(0.03, 0:0.001:3)
xlabel('time')
ylabel('current')
str1=sprintf('I. Exploration\n\nfigure 1.1 current vs. time');
title(str1)

subplot(2,2,2); hold on;
plot(tVec,V(1,:))
plot(tVec,V(2,:),'r')
plot(0.03, -10^4:100:2*10^4)
xlabel('time')
ylabel('voltage')
str2=sprintf('figure 1.2 voltage\n v. time');
title(str2)

subplot(2,2,3); hold on;
plot(I(1,:),drvtV(1,:))
plot(I(2,:),drvtV(2,:),'r')
xlabel('current')
ylabel('derivative of voltage')
str3=sprintf('figure 1.3\n derivative of voltage\n v. current');
title(str3)
xlim([0 2])

subplot(2,2,4); hold on;
plot(I(1,:),V(1,:))
plot(I(2,:),V(2,:),'r')
xlabel('current')
ylabel('voltage')
str4=sprintf('figure 1.4 voltage\n v. current');
title(str4)
xlim([0 2])
```




```
% Conclusion about relationship.
% From the graph we draw, we can find out that along with current increase,
% voltage increases, but the accelaration (derivative of voltage) is
% slowing down.

%%
% II. Remodeling
% Then try to find out two models that gives the same voltage at tnow=30ms.
% Thus, the fastest way to do, we decide to shift the blue model above left
% to intesect with the red one at tnow=30ms.
% From command window, input V(2,300), it gives out ans = 4.0281e+03.
% So we need to find what current in blue model fit the voltage 4.0281e+03.
% Thus, we run a small program below to calculate:

for t = 1:nTimeSteps
    if (V(1,t+1)>V(2,300))&&(V(1,t-1)<V(2,300))
        tshift = t;
        break;
    end
end
```

```
% Now let's modify the fomula and data of blue model from above:

for t = 1:nTimeSteps
    I(1,t) = (t+tshift-300)*10^-3; % shift the blue model to the left
    dV = (I(1,t)-(V(1,t)-Vrest)/R)/C * dt;
    drvtV(1,t) = dV/dt;
    if t < nTimeSteps
        V(1,t+1) = V(1,t) + dV;
    end
end

% Now we get the new model, let's test it!

figure;

subplot(2,2,1); hold on;
plot(tVec,I(1,:))
plot(tVec,I(2,:),'r')
plot(0.03, 0:0.001:3)
xlabel('time')
ylabel('current')
str5=sprintf('II. Remodeling\n\nfigure 1.5 current vs. time');
title(str5)

subplot(2,2,2); hold on;
plot(tVec,V(1,:))
plot(tVec,V(2,:),'r')
plot(0.03, -10^4:100:2*10^4)
xlabel('time')
ylabel('voltage')
str6=sprintf('figure 1.6 voltage\n v. time');
title(str6)

subplot(2,2,3); hold on;
plot(I(1,:),drvtV(1,:))
plot(I(2,:),drvtV(2,:),'r')
xlabel('current')
ylabel('derivative of voltage')
str7=sprintf('figure 1.7\n derivative of voltage\n v. current');
title(str7)
xlim([0 2])
```

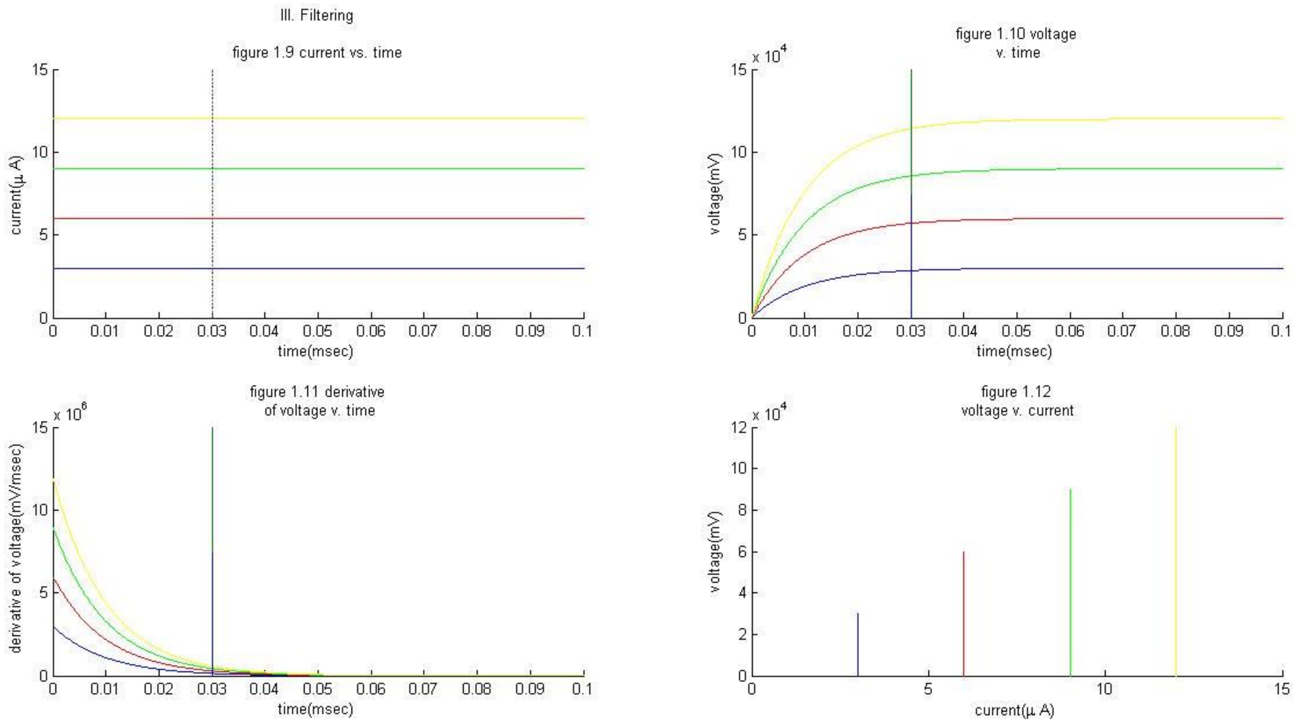
```
subplot(2,2,4); hold on;
plot(I(1,:),V(1,:))
plot(I(2,:),V(2,:),'r')
xlabel('current')
ylabel('voltage')
str8=sprintf('figure 1.8 voltage\n v. current');
title(str8)
xlim([0 2])
```




```
%%
% ================ Here goes the real challenge! =======================
% III. Filtering
% Now the models we are going to test are:
% a. I = constant (that stablize before 30ms).
% b. I = sinusoidal form of constant.


% First test a. I = constant (that stablize before 30ms).
% We chose four different constants

C = 1e-6; % 1 uF (per cm^2)
```

```
R = 1e4; % 10 kOhms (per cm^2)
dt = 0.0001;
tFinal = 0.1;
t = 0:dt:tFinal;

I = ones(4, length(t)); % in nanoamps
V = zeros(4, length(I));
drvtV = zeros(4, length(V));
tau = R*C; % 10 msec = 0.01 sec
V(1) = 0;   % Initialize voltage to zero.

for k = 1:4  % try four different constants
    I(k,1:length(t))= 3*k;
    for i=1:length(t)
        dV = dt * (I(k,i) - V(k,i)/R) / C;
        drvtV(k,i) = dV/dt;
        if i < length(t)
            V(k,i+1) = V(k,i) + dV;
        end
    end
end

figure;

subplot(2,2,1); hold on;
plot(t,I(1,:),'b')
plot(t,I(2,:),'r')
plot(t,I(3,:),'g')
plot(t,I(4,:),'y')
plot(0.03, 0:0.1:15)
xlabel('time(msec)')
ylabel('current(\mu A)')
str9=sprintf('III. Filtering\n\nfigure 1.9 current vs. time');
title(str9)

subplot(2,2,2); hold on;
plot(t,V(1,:),'b')
plot(t,V(2,:),'r')
plot(t,V(3,:),'g')
plot(t,V(4,:),'y')
plot(0.03,0:100:15*1e4)
```

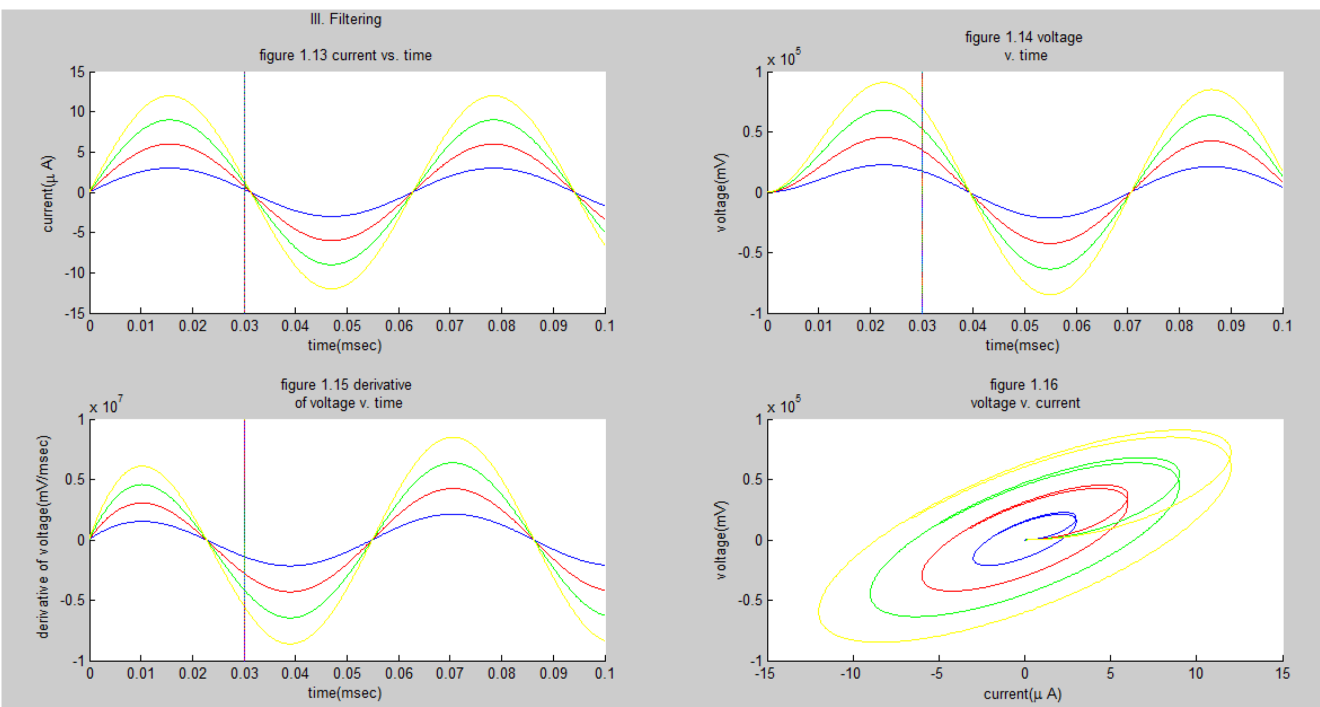
```
xlabel('time(msec)')
ylabel('voltage(mV)')
ylim([0, 15]*1e4)
str10=sprintf('figure 1.10 voltage\n v. time');
title(str10)

subplot(2,2,3); hold on;
plot(t,drvtV(1,:),'b')
plot(t,drvtV(2,:),'r')
plot(t,drvtV(3,:),'g')
plot(t,drvtV(4,:),'y')
plot(0.03,0:10000:15*1e6)
xlabel('time(msec)')
ylabel('derivative of voltage(mV/msec)')
str11=sprintf('figure 1.11 derivative\n of voltage v. time');
title(str11)

subplot(2,2,4); hold on;
plot(I(1,:),V(1,:),'b')
plot(I(2,:),V(2,:),'r')
plot(I(3,:),V(3,:),'g')
plot(I(4,:),V(4,:),'y')
xlabel('current(\mu A)')
ylabel('voltage(mV)')
xlim([0 15])
str12=sprintf('figure 1.12\n voltage v. current');
title(str12)
```



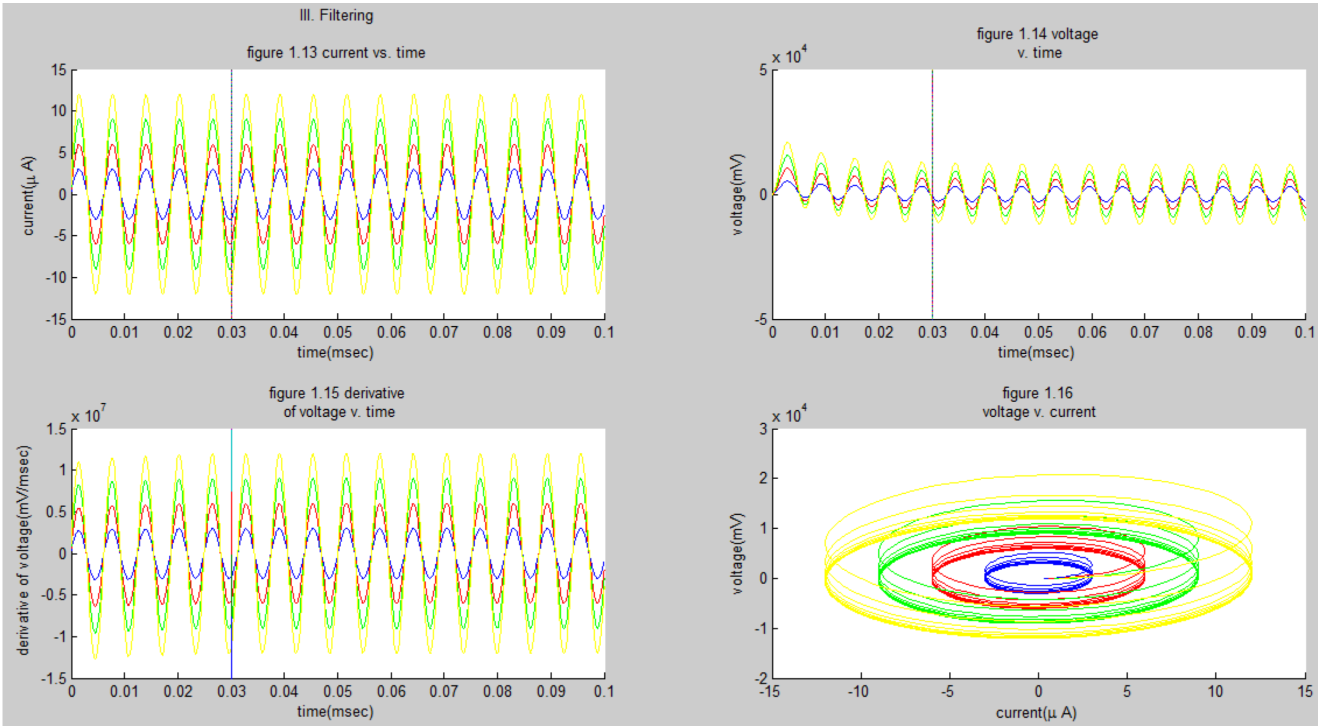

```
% From the graph we draw, we find that:
% 1. All voltage reach to a equilibrium finally injected by a constant I.
% 2. the bigger constant I is, the bigger equilibrium V is.
% 3. the drvtV shows that all accelaration decrease to 0 throughout time.

%%
% Next, test % b. I = sinusoidal form of constant.
% We chose four different constants to change the amplitude of waves.

C = 1e-6; % 1 uF (per cm^2)
R = 1e4; % 10 kOhms (per cm^2)
dt = 0.0001;
tFinal = 0.1;
t = 0:dt:tFinal;

I = ones(4, length(t)); % in nanoamps
V = zeros(4, length(I));
drvtV = zeros(4, length(V));
tau = R*C; % 10 msec = 0.01 sec
V(1) = 0;   % Initialize voltage to zero.
```

```
for k = 1:4  % try four different constants
    for i=1:length(t)
        I(k,i)= sin(i/100)*3*k;
        dV = dt * (I(k,i) - V(k,i)/R) / C;
        drvtV(k,i) = dV/dt;
        if i < length(t)
            V(k,i+1) = V(k,i) + dV;
        end
    end
end

figure;

subplot(2,2,1); hold on;
plot(t,I(1,:),'b')
plot(t,I(2,:),'r')
plot(t,I(3,:),'g')
plot(t,I(4,:),'y')
plot(0.03, -15:0.1:15)
xlabel('time(msec)')
ylabel('current(\mu A)')
str13=sprintf('III. Filtering\n\nfigure 1.13 current vs. time');
title(str13)

subplot(2,2,2); hold on;
plot(t,V(1,:),'b')
plot(t,V(2,:),'r')
plot(t,V(3,:),'g')
plot(t,V(4,:),'y')
plot(0.03,-100000:1:100000)
xlabel('time(msec)')
ylabel('voltage(mV)')
str14=sprintf('figure 1.14 voltage\n v. time');
title(str14)

subplot(2,2,3); hold on;
plot(t,drvtV(1,:),'b')
plot(t,drvtV(2,:),'r')
plot(t,drvtV(3,:),'g')
plot(t,drvtV(4,:),'y')
plot(0.03,-1*10^7:10000:1*1e7)
```

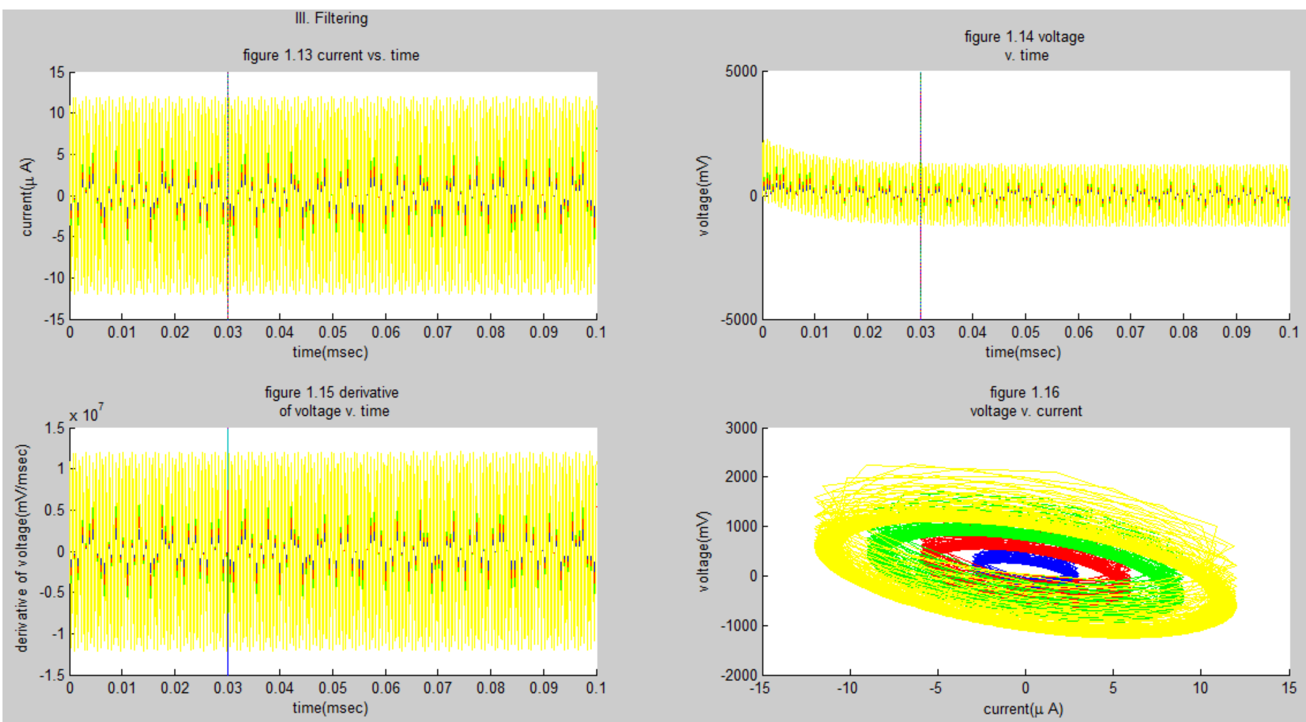
```
xlabel('time(msec)')
ylabel('derivative of voltage(mV/msec)')
str15=sprintf('figure 1.15 derivative\n of voltage v. time');
title(str15)

subplot(2,2,4); hold on;
plot(I(1,:),V(1,:),'b')
plot(I(2,:),V(2,:),'r')
plot(I(3,:),V(3,:),'g')
plot(I(4,:),V(4,:),'y')
xlabel('current(\mu A)')
ylabel('voltage(mV)')
%xlim([0 15])
str16=sprintf('figure 1.16\n voltage v. current');
title(str16)

% We can change angular frequency omega from 1/100 to 1/10 to 1 (as shown
% in graphs attached.
% become more compact, the overall tendency becomes more obvious:
% 1. the voltage decreases with in sinuosidal spiral, regardless of I.
% 2. the bigger constant I amplitude is, the bigger variation of V is.
% 3. the drvtV shows that all accelarations goes in sinuosidal waves.
```




**W=1/100**


III. Filtering
figure 1.13 current vs. time
current(μ A)
time(msec)
figure 1.14 voltage v. time
voltage(mV)
x 10^4
figure 1.15 derivative of voltage v. time
derivative of voltage(mV/msec)
x 10^7
figure 1.16 voltage v. current
current(μ A)

**W=1/10**

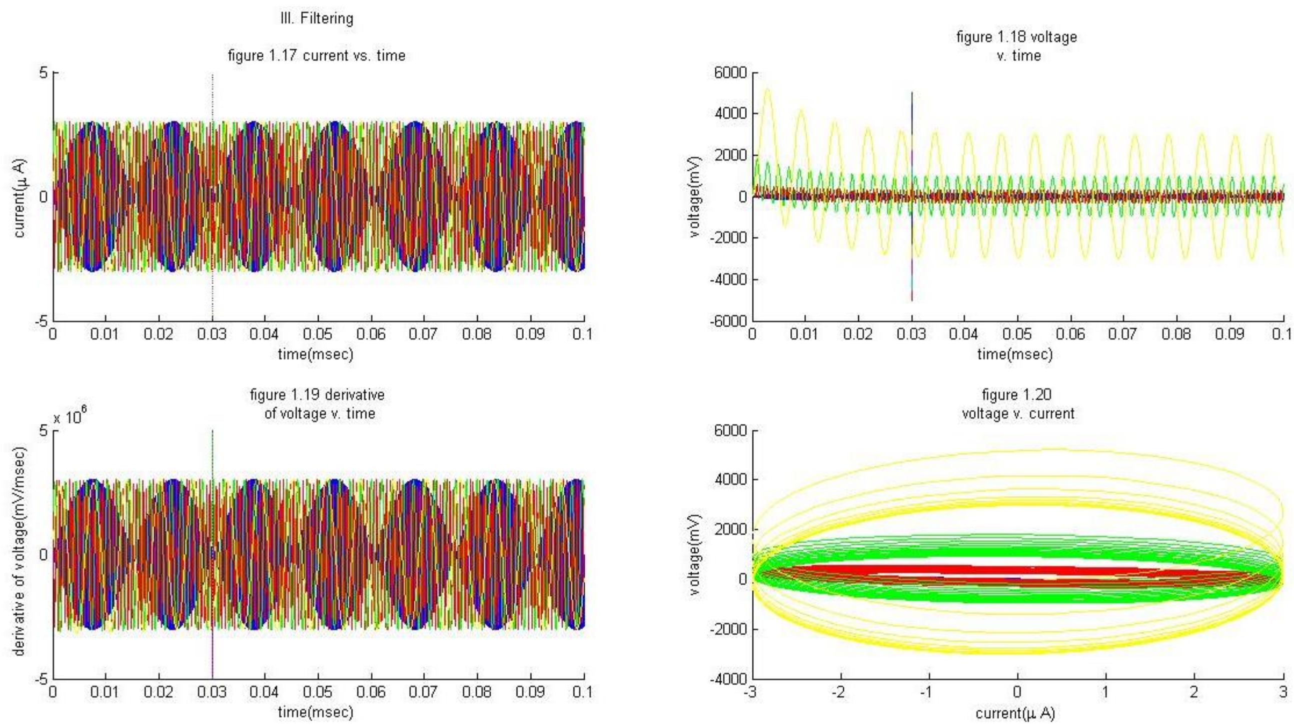

III. Filtering
figure 1.13 current vs. time
current(μ A)
time(msec)
figure 1.14 voltage v. time
voltage(mV)
figure 1.15 derivative of voltage v. time
derivative of voltage(mV/msec)
x 10^7
figure 1.16 voltage v. current
current(μ A)

**W=1**

%%

```
% We can also try choosing four different angular frequencies like above,
% only this time align them on the same graph.

C = 1e-6; % 1 uF (per cm^2)
R = 1e4; % 10 kOhms (per cm^2)
dt = 0.0001;
tFinal = 0.1;
t = 0:dt:tFinal;

I = ones(4, length(t)); % in nanoamps
V = zeros(4, length(I));
drvtV = zeros(4, length(V));
tau = R*C; % 10 msec = 0.01 sec
V(1) = 0;   % Initialize voltage to zero.

for k = 1:4  % try four different constants
    for i=1:length(t)
        I(k,i)= sin(i/10^(k/2-1))*3;
        dV = dt * (I(k,i) - V(k,i)/R) / C;
        drvtV(k,i) = dV/dt;
        if i < length(t)
            V(k,i+1) = V(k,i) + dV;
        end
    end
end

figure;

subplot(2,2,1); hold on;
plot(t,I(1,:),'b')
plot(t,I(2,:),'r')
plot(t,I(3,:),'g')
plot(t,I(4,:),'y')
plot(0.03, -5:0.1:5)
xlabel('time(msec)')
ylabel('current(\mu A)')
str17=sprintf('III. Filtering\n\nfigure 1.17 current vs. time');
title(str17)

subplot(2,2,2); hold on;
plot(t,V(1,:),'b')
```

```
plot(t,V(2,:),'r')
plot(t,V(3,:),'g')
plot(t,V(4,:),'y')
plot(0.03,-5000:1:5000)
xlabel('time(msec)')
ylabel('voltage(mV)')
str18=sprintf('figure 1.18 voltage\n v. time');
title(str18)

subplot(2,2,3); hold on;
plot(t,drvtV(1,:),'b')
plot(t,drvtV(2,:),'r')
plot(t,drvtV(3,:),'g')
plot(t,drvtV(4,:),'y')
plot(0.03,-5*10^6:10000:5*1e6)
xlabel('time(msec)')
ylabel('derivative of voltage(mV/msec)')
str19=sprintf('figure 1.19 derivative\n of voltage v. time');
title(str19)

subplot(2,2,4); hold on;
plot(I(1,:),V(1,:),'b')
plot(I(2,:),V(2,:),'r')
plot(I(3,:),V(3,:),'g')
plot(I(4,:),V(4,:),'y')
xlabel('current(\mu A)')
ylabel('voltage(mV)')
%xlim([0 15])
str20=sprintf('figure 1.20\n voltage v. current');
title(str20)
```

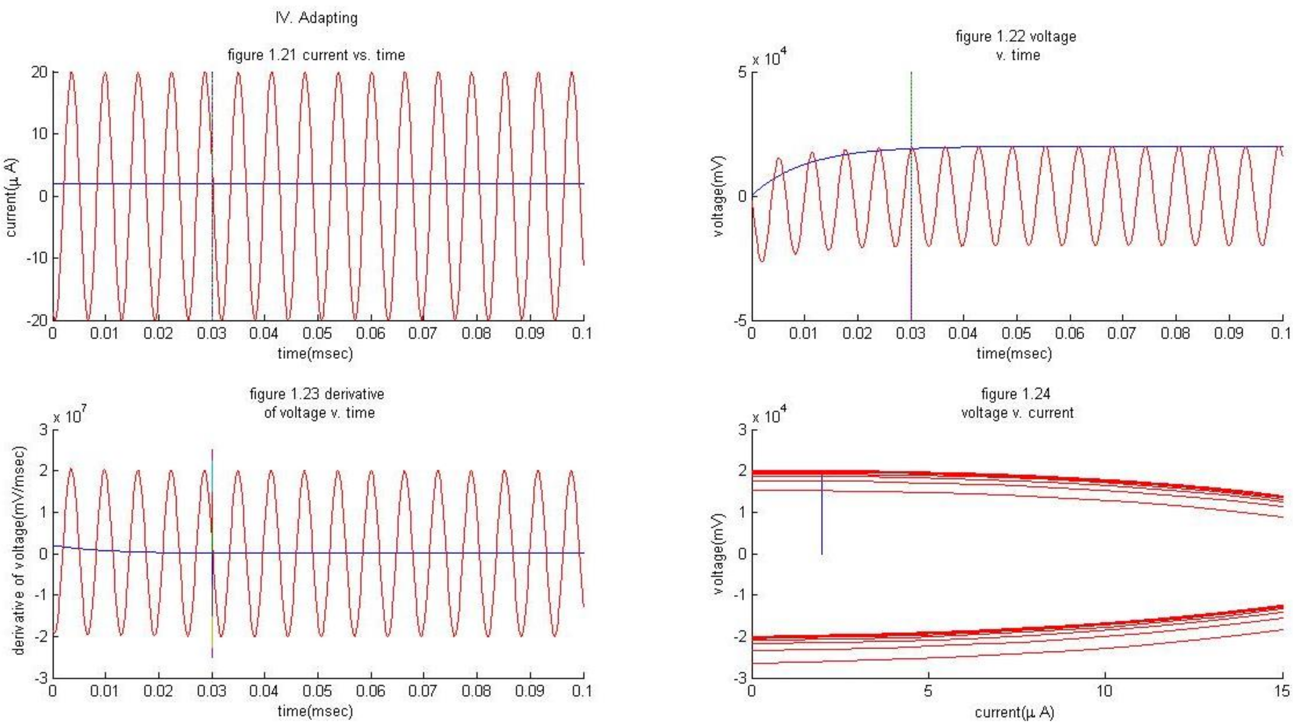



```
% Here the graph is cool but I don't think we can draw any significant
% conclusion from it. So we will move on.

%%
% IV. Adapting
% Now I am going to make the sinusiodal model adapt to the constant model.
% We need a program similar to "Remodeling" to train the program.

C = 1e-6; % 1 uF (per cm^2)
R = 1e4; % 10 kOhms (per cm^2)
dt = 0.0001;
tFinal = 0.1;
t = 0:dt:tFinal;

I = ones(2, length(t)); % in nanoamps
V = zeros(2, length(I));
drvtV = zeros(2, length(V));
tau = R*C; % 10 msec = 0.01 sec
V(1) = 0;   % Initialize voltage to zero.

iangfrq = 1;
iphase = 0;
```

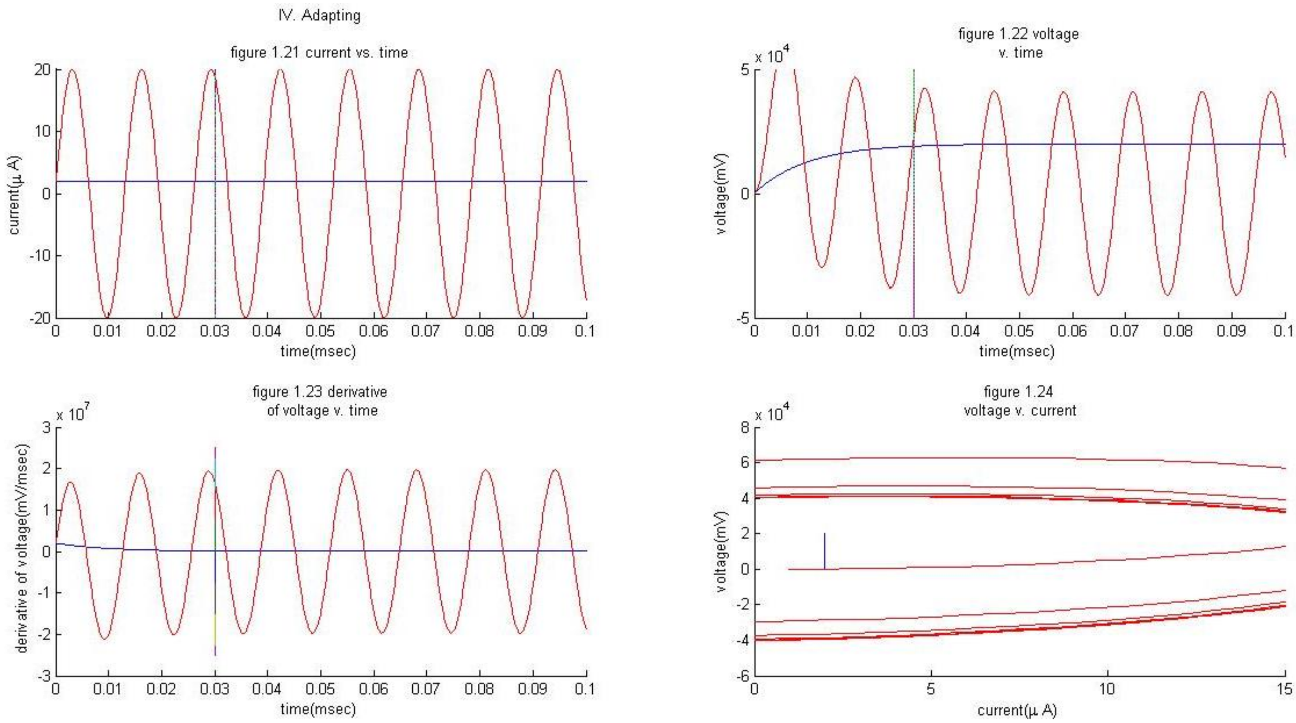
```
for k = 1:2
    for i=1:length(t)
        I(1,i)= 2;
        I(2,i)= sin(iangfrq*i/10+iphase)*20;
        dV = dt * (I(k,i) - V(k,i)/R) / C;
        drvtV(k,i) = dV/dt;
        if i < length(t)
            V(k,i+1) = V(k,i) + dV;
        end
    end
end

% We have two ways to choose: change the angular frequency iangfrq, or
% change the phase iphase.

choice = input('Want to change phase or angular frequency? 1 or 2?\n')

switch choice
    case 1
        disp('Good choice. This is more efficient...');
        for iphase = 1:1000
            if ((V(2,301+iphase)-V(1,300))*(V(2,299+iphase)-V(1,300)) <= 0)
                iphase = iphase*10;
                break;
            end
        end
    case 2
        disp('Would be my 2nd choice. a little slower. But not much.');
        for iangfrq = 0.001:0.01:1000
            for i=1:300
                I(2,i)= sin(iangfrq*i/10+iphase)*20;
                dV = dt * (I(2,i) - V(2,i)/R) / C;
                V(2,i+1) = V(2,i) + dV;
            end
            if ((V(2,301)-V(1,300))*(V(2,299)-V(1,300)) <= 0)
                break;
            end
        end
    otherwise
        disp('Behave! Please make the right choice, alright?')
end
```

```
for i=1:length(t)
    I(2,i)= sin(iangfrq*i/10+iphase)*20;
    dV = dt * (I(2,i) - V(2,i)/R) / C;
    drvtV(2,i) = dV/dt;
    if i < length(t)
        V(2,i+1) = V(2,i) + dV;
    end
end

figure;

subplot(2,2,1); hold on;
plot(t,I(1,:),'b')
plot(t,I(2,:),'r')
plot(0.03, -20:0.1:20)
xlabel('time(msec)')
ylabel('current(\mu A)')
str21=sprintf('IV. Adapting\n\nfigure 1.21 current vs. time');
title(str21)

subplot(2,2,2); hold on;
plot(t,V(1,:),'b')
plot(t,V(2,:),'r')
plot(0.03,-5*1e4:100:5*1e4)
xlabel('time(msec)')
ylabel('voltage(mV)')
ylim([-5, 5]*1e4)
str22=sprintf('figure 1.22 voltage\n v. time');
title(str22)

subplot(2,2,3); hold on;
plot(t,drvtV(1,:),'b')
plot(t,drvtV(2,:),'r')
plot(0.03,-25*1e6:10000:25*1e6)
xlabel('time(msec)')
ylabel('derivative of voltage(mV/msec)')
str23=sprintf('figure 1.23 derivative\n of voltage v. time');
title(str23)

subplot(2,2,4); hold on;
```

```
plot(I(1,:),V(1,:),'b')
plot(I(2,:),V(2,:),'r')
xlabel('current(\mu A)')
ylabel('voltage(mV)')
xlim([0 15])
str24=sprintf('figure 1.24\n voltage v. current');
title(str24)
```




## Method 1


IV. Adapting
figure 1.21 current vs. time
current(μ A)
time(msec)
figure 1.22 voltage v. time
voltage(mV)
x 10^4
time(msec)
figure 1.23 derivative of voltage v. time
derivative of voltage(mV/msec)
x 10^7
time(msec)
figure 1.24 voltage v. current
voltage(mV)
x 10^4
current(μ A)

**Method 2**

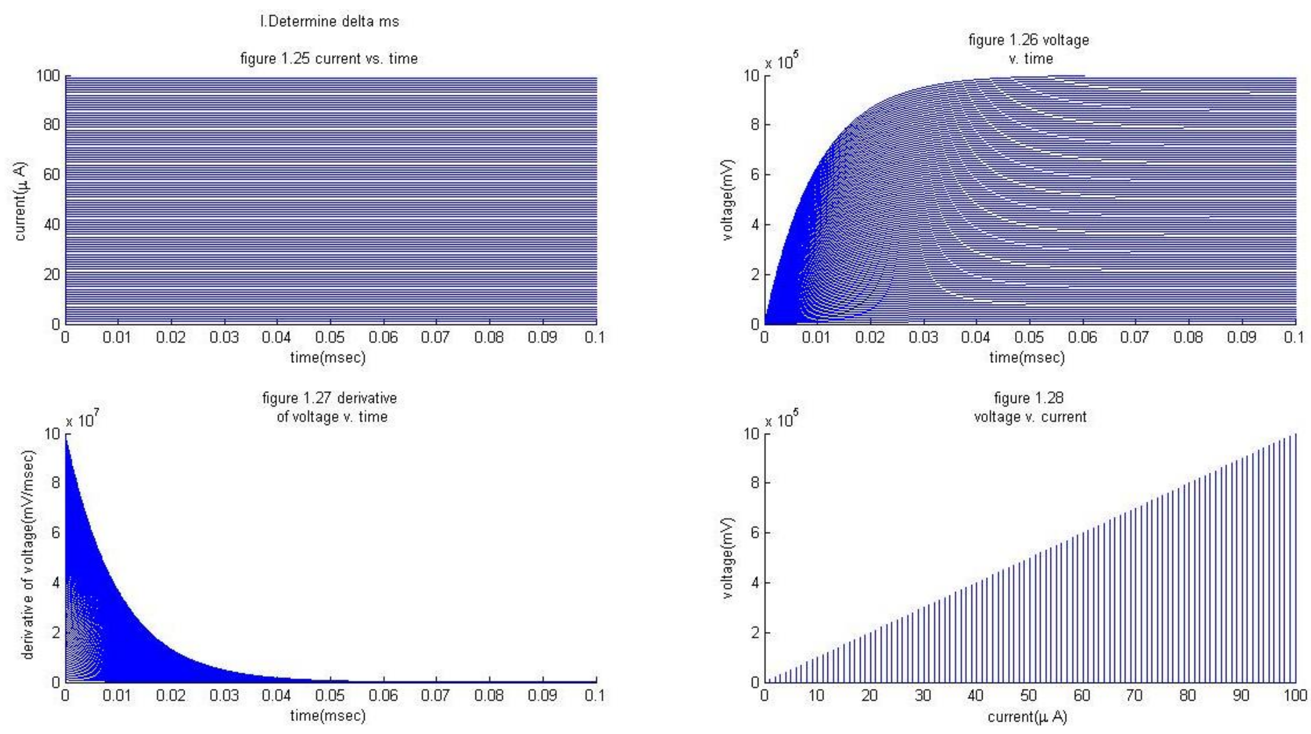
# 2.Summation of simultaneous impulses

## (does conductance “help or hurt”)

```
%%
% Question 2: Summation of Simulataneous Impulses
% Peak Voltage?
% Fraction of way to threshold?
% Lowest value N to drive voltage over threshold?
% How are f and N related?
% Make a conductance-based model?

% From our findings above we know that the equlibrium, i.e.
% the peak in the question 2, increases with the injected I-bar.
% Therefore my thought of this is to plot a graph of voltage peak v.
% injected current to find out the exact current to make the peak at
% 10 mV, therefore solving the N problem easily. Then plot N v. f to
% to find out the relationships.
% Notes: considering the I bar (average) notation used in the problem
% set question 2, I will use constant model instead of sinuosiodal model.

% First, let's revise the codes in our III.Filtering constant model:

% I. Determine delta ms

C = 1e-6; % 1 uF (per cm^2)
R = 1e4; % 10 kOhms (per cm^2)
dt = 0.0001;
tFinal = 0.1;
t = 0:dt:tFinal;

I = ones(100, length(t)); % in nanoamps
V = zeros(100, length(I));
drvtV = zeros(100, length(V));
%drvtV0 = zeros(4);
tau = R*C; % 10 msec = 0.01 sec
V(1) = 0;   % Initialize voltage to zero.

for k = 1:100  % try four different constants
    I(k,1:length(t))= k;
```

```
    for i=1:length(t)
        dV = dt * (I(k,i) - V(k,i)/R) / C;
        drvtV(k,i) = dV/dt;
%        if (drvtV0(k) == 0)&&(drvtV(k,i) == 0)
%            drvtV0(k) =  i;
 %       end
        if i < length(t)
            V(k,i+1) = V(k,i) + dV;
        end
    end
end

figure;

subplot(2,2,1); hold on;
for k = 1:100
    plot(t,I(k,:))
end
%plot(drvtV0(1)/10000,I(1,drvtV0(1))*0.9:0.1:I(1,drvtV0(1))*1.1,'b')
%plot(drvtV0(2)/10000,I(2,drvtV0(2))*0.9:0.1:I(1,drvtV0(2))*1.1,'r')
%plot(drvtV0(3)/10000,I(3,drvtV0(3))*0.9:0.1:I(1,drvtV0(3))*1.1,'g')
%plot(drvtV0(4)/10000,I(4,drvtV0(4))*0.9:0.1:I(1,drvtV0(4))*1.1,'y')
xlabel('time(msec)')
ylabel('current(\mu A)')
str25=sprintf('I.Determine delta ms\n\nfigure 1.25 current vs. time');
title(str25)

subplot(2,2,2); hold on;
for k = 1:100
    plot(t,V(k,:))
end
%plot(drvtV0(1)/10000,I(1,drvtV0(1))*0.9:0.1:I(1,drvtV0(1))*1.1,'b')
%plot(drvtV0(2)/10000,I(2,drvtV0(2))*0.9:0.1:I(1,drvtV0(2))*1.1,'r')
%plot(drvtV0(3)/10000,I(3,drvtV0(3))*0.9:0.1:I(1,drvtV0(3))*1.1,'g')
%plot(drvtV0(4)/10000,I(4,drvtV0(4))*0.9:0.1:I(1,drvtV0(4))*1.1,'y')
xlabel('time(msec)')
ylabel('voltage(mV)')
str26=sprintf('figure 1.26 voltage\n v. time');
title(str26)

subplot(2,2,3); hold on;
```

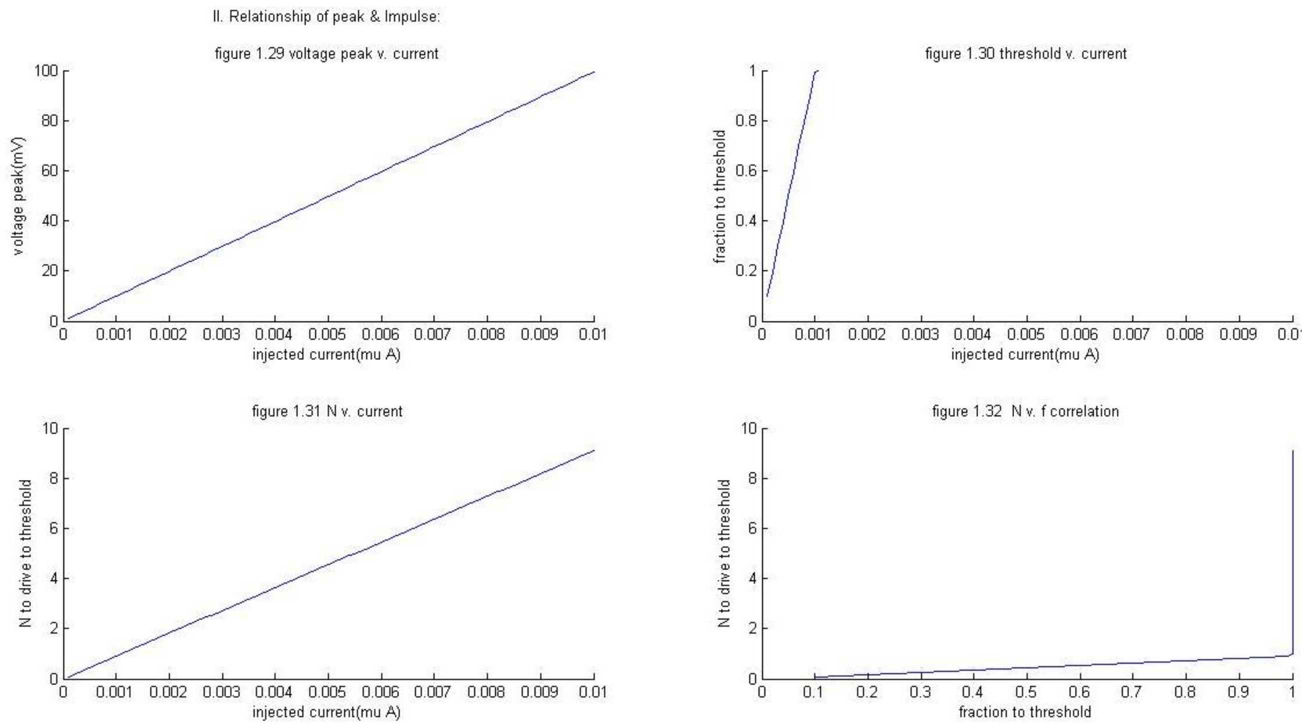
```
for k = 1:100
    plot(t,drvtV(k,:))
end
%plot(drvtV0(1)/10000,I(1,drvtV0(1))*0.9:0.1:I(1,drvtV0(1))*1.1,'b')
%plot(drvtV0(2)/10000,I(2,drvtV0(2))*0.9:0.1:I(1,drvtV0(2))*1.1,'r')
%plot(drvtV0(3)/10000,I(3,drvtV0(3))*0.9:0.1:I(1,drvtV0(3))*1.1,'g')
%plot(drvtV0(4)/10000,I(4,drvtV0(4))*0.9:0.1:I(1,drvtV0(4))*1.1,'y')
xlabel('time(msec)')
ylabel('derivative of voltage(mV/msec)')
str27=sprintf('figure 1.27 derivative\n of voltage v. time');
title(str27)

subplot(2,2,4); hold on;
for k = 1:100
    plot(I(k,:),V(k,:))
end
xlabel('current(\mu A)')
ylabel('voltage(mV)')
str28=sprintf('figure 1.28\n voltage v. current');
title(str28)
```



```
% In my code's comments, you may see a failed attempt to label out
% a small flag on the point derivative of voltage reaches zero
% But it turns out to be too far away (t>0.1s) and drvtV0(k) becomes
% zero and cannot serve as a index for the plotting.
% What this taught me is that we only need to think like an engineer,
% find out the pragmatic range to determine "the true zero," instead
% finding a precise zero scientifically. So that's why I increase the
% number of currents tested here trying to find out the proper delta ms.

% Conclusion:
% 0.05 msec is a reasonable interval to allow voltages reach peaks.

%%
% II. Relationship of peak & Impulse:
% For efficiency, from the maintrend in graphs above (the approximate
% ratio of current(muA) to voltage (mV) peak is like 1:10^4). Thus the
% range of injected input will be chosen around 10^-3 --> 10^-4 to 10^-2

C = 1e-6; % 1 uF (per cm^2)
R = 1e4; % 10 kOhms (per cm^2)
dt = 0.0001;
tFinal = 0.05;
t = 0:dt:tFinal;

I = ones(100, length(t)); % in nanoamps
V = zeros(100, length(I));
drvtV = zeros(100, length(V));
tau = R*C; % 10 msec = 0.01 sec
V(1) = 0;   % Initialize voltage to zero.
f = ones(100);
N = ones(100);
firstpeak = 0;

for k = 1:100  % try four different constants
    I(k,1:length(t))= k*10^-4;
    for i=1:length(t)
        dV = dt * (I(k,i) - V(k,i)/R) / C;
        drvtV(k,i) = dV/dt;
        if i < length(t)
            V(k,i+1) = V(k,i) + dV;
        end
```

```
    end
    if (V(k,500) < 10)
        f(k) = V(k,500)/10;
    else
        if (V(k-1,500) < 10)
            firstpeak = k;
        end
    end
end

for k=1:100
    N(k) = I(k,500)/I(firstpeak,500);
end

figure;

subplot(2,2,1); hold on;
plot(I(1:100,500),V(1:100,500))
xlabel('injected current(mu A)')
ylabel('voltage peak(mV)')
str29=sprintf('II. Relationship of peak & Impulse:\n\nfigure 1.29 voltage
peak v. current');
title(str29)

subplot(2,2,2); hold on;
plot(I(1:100,500),f(1:100))
xlabel('injected current(mu A)')
ylabel('fraction to threshold')
str30=sprintf('figure 1.30 threshold v. current');
title(str30)

subplot(2,2,3); hold on;
plot(I(1:100,500),N(1:100))
xlabel('injected current(mu A)')
ylabel('N to drive to threshold')
str31=sprintf('figure 1.31 N v. current');
title(str31)

subplot(2,2,4); hold on;
plot(f(1:100),N(1:100))
xlabel('fraction to threshold')
```

```
ylabel('N to drive to threshold')
str32=sprintf('figure 1.32  N v. f correlation');
title(str32)
```

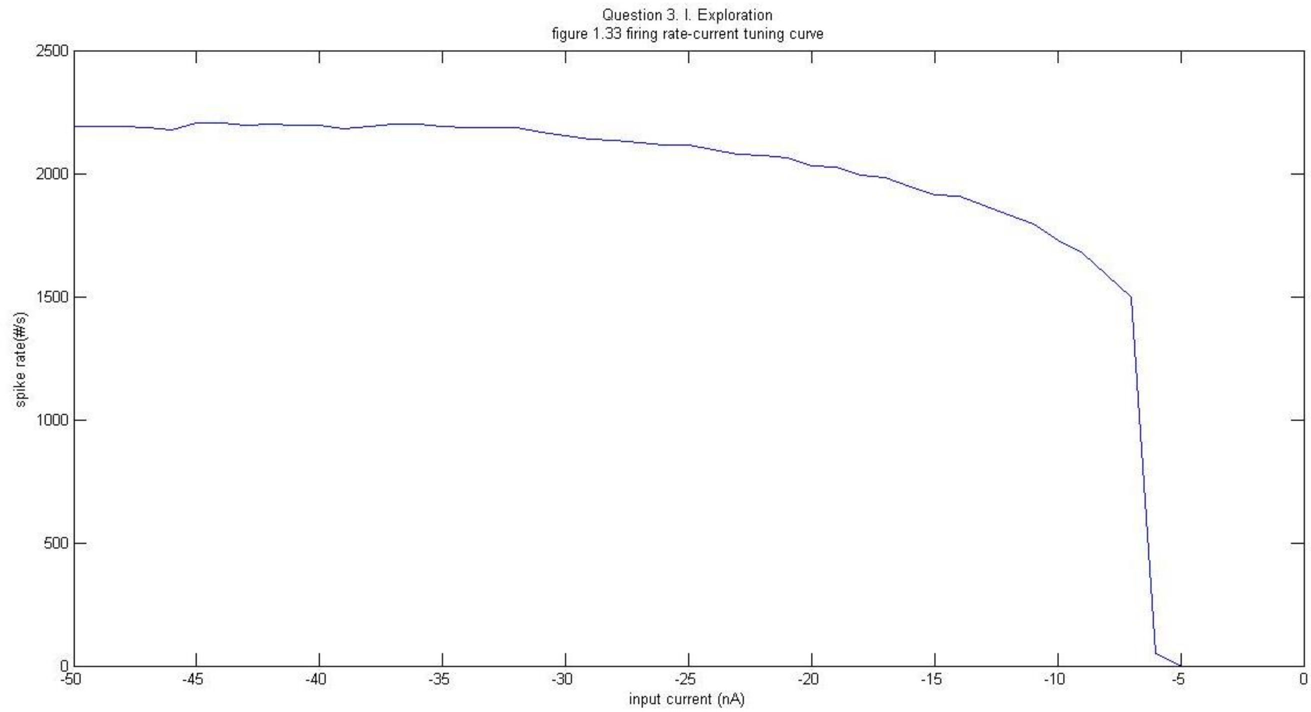


# 3.HH model

## (Exploration of parameters in sinusoidal model)

```
%%
% Question 3: HH model
% Plot the firing rate-current tuning curve
% Plot firing frequency as a function of I for applied current
% Ia(t)=I+esin(2pi*t*w)
% How that change firing rate-current tuning curve
% Quantitative explanation?
% How well each stimulus? How depend?

% I. Exploration
% First try simple linear model: current as constant.
% Let's draw the tuning curve!

dt = 0.05;
tFinal = 1000;
t = 0:dt:tFinal;

clear V;
m = zeros(50,length(t));   % Initialize everything to zero.
n = zeros(50,length(t));
h = zeros(50,length(t));
V = zeros(50,length(t));
n(:,1) = 0.6;
h(:,1) = 0.3;
spike = zeros(50);

gkmax = 36; %*0.01 * 1000 ; % cm--> 10^7 g --> 10^-2 dt --> 10^3
gnamax = 120;
gl = 0.3;
Vk = 12;
Vna = -115;
Vl = -10.613;
C = 1.0;

Iext = zeros(50,length(t));
for k =1:50
```

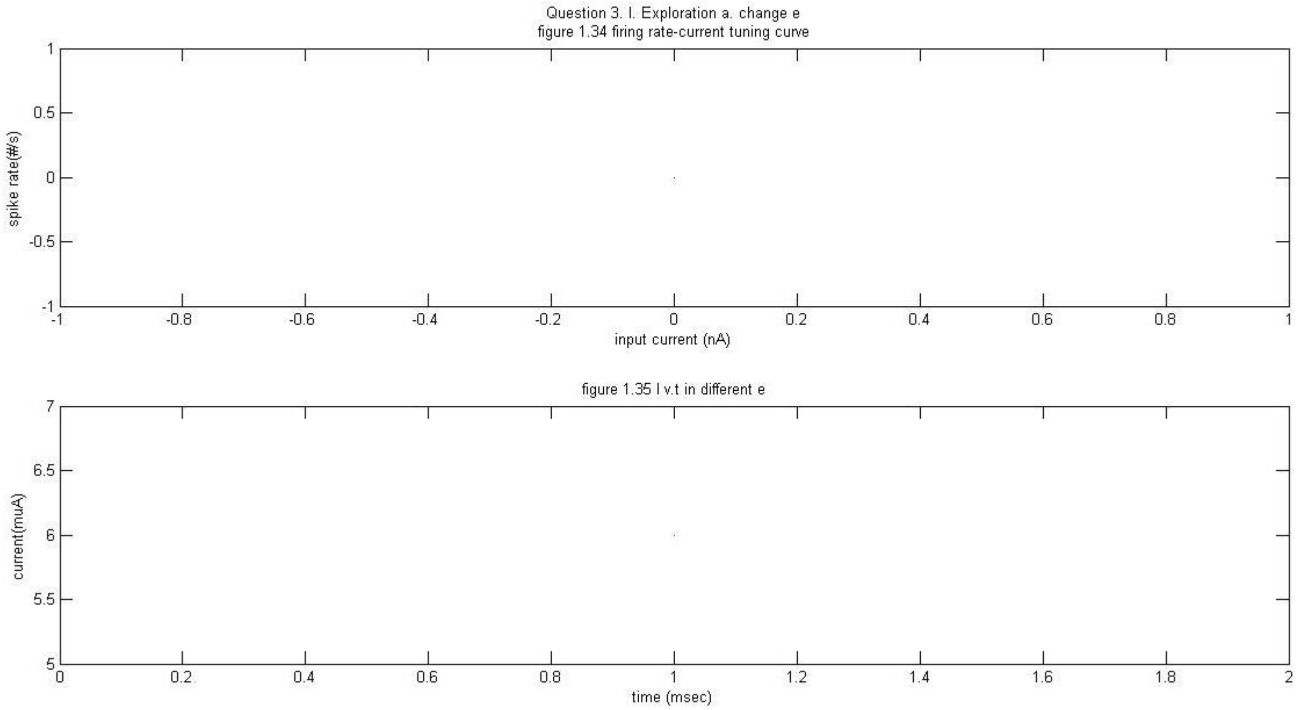
```
    Iext(k,:) = -k;   % don't forget to make the current negative...
    for i=1:(length(t)-1)
        alpham = 0.1*(V(k,i)+25)/(exp((V(k,i)+25)/10)-1);
        betam = 4*exp(V(k,i)/18);
        alphan = 0.01*(V(k,i)+10)/(exp((V(k,i)+10)/10)-1);
        betan = 0.125*exp(V(k,i)/80);
        alphah = 0.07*exp(V(k,i)/20);
        betah = 1./(exp((V(k,i)+30)/10)+1);
        dm = dt * (-m(k,i)*(alpham + betam) + alpham);
        m(k,i+1) = m(k,i) + dm;
        dn = dt * (-n(k,i)*(alphan + betan) + alphan);
        n(k,i+1) = n(k,i) + dn;
        dh = dt * (-h(k,i)*(alphah + betah) + alphah);
        h(k,i+1) = h(k,i) + dh;
        Ik = gkmax*n(k,i)^4*(V(k,i) - Vk);
        Ina = gnamax*m(k,i)^3*h(k,i)*(V(k,i)-Vna);
        Il = gl*(V(k,i) - Vl);
        V(k,i+1) = V(k,i) + dt*(Iext(k,i) - Ik - Ina - Il)/C;
        if (V(k,i) < -50)
            spike(k) = spike(k)+1;
        end
    end
end

figure;

plot(Iext(1:50,1),spike(1:50));
xlabel('input current (nA)')
ylabel('spike rate(#/s)')
str33=sprintf('Question 3. I. Exploration\nfigure 1.33 firing rate-current
tuning curve')
title(str33)
```




```
% From the tuning curve we can see the tuning curve resembles our
% stimulus tuning curve in Problem Set 1, as long as the current is
% taken absolute value and reverse the x-axis.

%%

% II. Remodeling
% Then try IA(t)= I + e*sin(2*pi*t*w)
% Let's draw the tuning curve!

% First, change e.

dt = 0.05;
tFinal = 1000;
t = 0:dt:tFinal;

clear V;
m = zeros(50,length(t));   % Initialize everything to zero.
n = zeros(50,length(t));
h = zeros(50,length(t));
V = zeros(50,length(t));
```

```
n(:,1) = 0.6;
h(:,1) = 0.3;
spike = zeros(50);

gkmax = 36; %*0.01 * 1000 ; % cm--> 10^7 g --> 10^-2 dt --> 10^3
gnamax = 120;
gl = 0.3;
Vk = 12;
Vna = -115;
Vl = -10.613;
C = 1.0;
e=1;
t=1;
w=1;

Iext = zeros(50,length(t));
for k =1:50
    for i=1:(length(t)-1)
        Iext(k,:) = -1e200*k*sin(2*pi*i*w);
        alpham = 0.1*(V(k,i)+25)/(exp((V(k,i)+25)/10)-1);
        betam = 4*exp(V(k,i)/18);
        alphan = 0.01*(V(k,i)+10)/(exp((V(k,i)+10)/10)-1);
        betan = 0.125*exp(V(k,i)/80);
        alphah = 0.07*exp(V(k,i)/20);
        betah = 1./(exp((V(k,i)+30)/10)+1);
        dm = dt * (-m(k,i)*(alpham + betam) + alpham);
        m(k,i+1) = m(k,i) + dm;
        dn = dt * (-n(k,i)*(alphan + betan) + alphan);
        n(k,i+1) = n(k,i) + dn;
        dh = dt * (-h(k,i)*(alphah + betah) + alphah);
        h(k,i+1) = h(k,i) + dh;
        Ik = gkmax*n(k,i)^4*(V(k,i) - Vk);
        Ina = gnamax*m(k,i)^3*h(k,i)*(V(k,i)-Vna);
        Il = gl*(V(k,i) - Vl);
        V(k,i+1) = V(k,i) + dt*(Iext(k,i) - Ik - Ina - Il)/C;
        if (V(k,i) < -50)
            spike(k) = spike(k)+1;
        end
    end
end
```

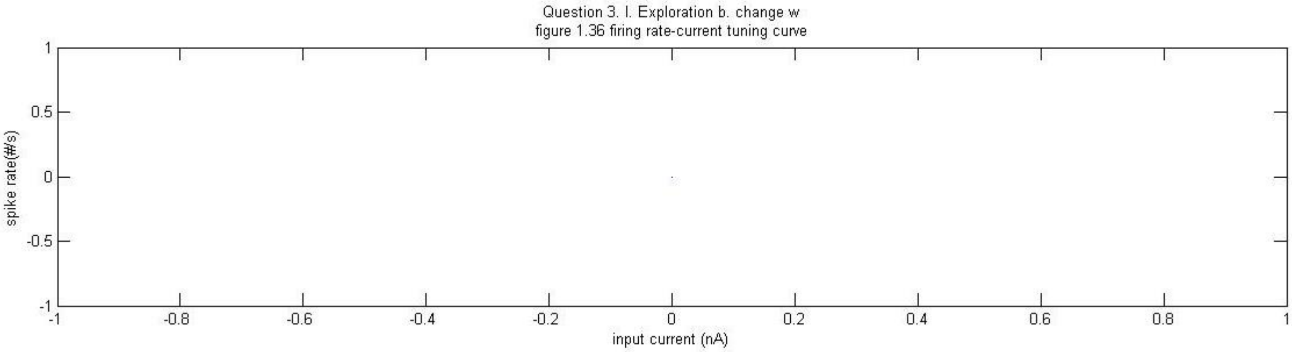
```
figure;

subplot(2,1,1)
plot(Iext(1:50,1),spike(1:50));
xlabel('input current (nA)')
ylabel('spike rate(#/s)')
str34=sprintf('Question 3. I. Exploration a. change e\nfigure 1.34 firing
rate-current tuning curve')
title(str34)

subplot(2,1,2)
plot(t, I(1,t),'r')
%for k = 1:50
%    plot(t,I(k,t));
%end
xlabel('time (msec)')
ylabel('current(muA)')
str35=sprintf('figure 1.35 I v.t in different e')
title(str35)
```




```
% From the graph we can see e makes a neglible effect.
```

```
%%
% First, change w.

dt = 0.05;
tFinal = 1000;
t = 0:dt:tFinal;

clear V;
m = zeros(50,length(t));   % Initialize everything to zero.
n = zeros(50,length(t));
h = zeros(50,length(t));
V = zeros(50,length(t));
n(:,1) = 0.6;
h(:,1) = 0.3;
spike = zeros(50);

gkmax = 36; %*0.01 * 1000 ; % cm--> 10^7 g --> 10^-2 dt --> 10^3
gnamax = 120;
gl = 0.3;
Vk = 12;
Vna = -115;
Vl = -10.613;
C = 1.0;
e=1;
t=1;
w=1;

Iext = zeros(50,length(t));
for k =1:50
    for i=1:(length(t)-1)
        Iext(k,:) = -50*sin(2*pi*i*k);
        alpham = 0.1*(V(k,i)+25)/(exp((V(k,i)+25)/10)-1);
        betam = 4*exp(V(k,i)/18);
        alphan = 0.01*(V(k,i)+10)/(exp((V(k,i)+10)/10)-1);
        betan = 0.125*exp(V(k,i)/80);
        alphah = 0.07*exp(V(k,i)/20);
        betah = 1./(exp((V(k,i)+30)/10)+1);
        dm = dt * (-m(k,i)*(alpham + betam) + alpham);
        m(k,i+1) = m(k,i) + dm;
        dn = dt * (-n(k,i)*(alphan + betan) + alphan);
```

```
        n(k,i+1) = n(k,i) + dn;
        dh = dt * (-h(k,i)*(alphah + betah) + alphah);
        h(k,i+1) = h(k,i) + dh;
        Ik = gkmax*n(k,i)^4*(V(k,i) - Vk);
        Ina = gnamax*m(k,i)^3*h(k,i)*(V(k,i)-Vna);
        Il = gl*(V(k,i) - Vl);
        V(k,i+1) = V(k,i) + dt*(Iext(k,i) - Ik - Ina - Il)/C;
        if (V(k,i) < -50)
            spike(k) = spike(k)+1;
        end
    end
end

figure;

subplot(2,1,1)
plot(Iext(1:50,1),spike(1:50));
xlabel('input current (nA)')
ylabel('spike rate(#/s)')
str34=sprintf('Question 3. I. Exploration b. change w\nfigure 1.36 firing
rate-current tuning curve')
title(str34)

subplot(2,1,2)
plot(t, I(1,t),'r')
%for k = 1:50
%    plot(t,I(k,t));
%end
xlabel('time (msec)')
ylabel('current(muA)')
str35=sprintf('figure 1.37 I v.t in different w')
title(str35)

% From the graph we can see w is also neglible.
```


Question 3. I. Exploration b. change w
figure 1.36 firing rate-current tuning curve
spike rate(#/s)
input current (nA)
1
0.5
0
-0.5
-1
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1

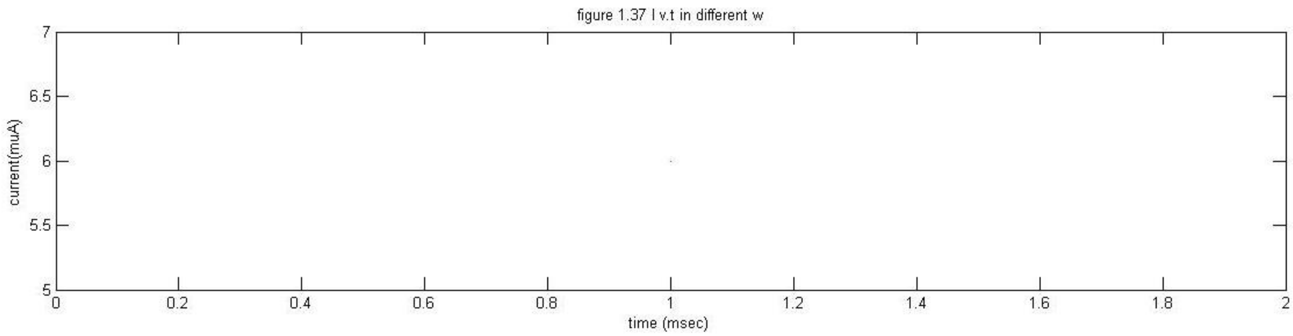


figure 1.37 I v.t in different w
current(muA)
time (msec)
7
6.5
6
5.5
5
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2

# 4. Graph Summary

I. Exploration

figure 1.1 current vs. time

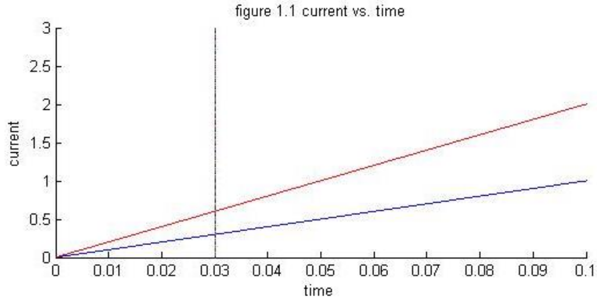



figure 1.2 voltage v. time

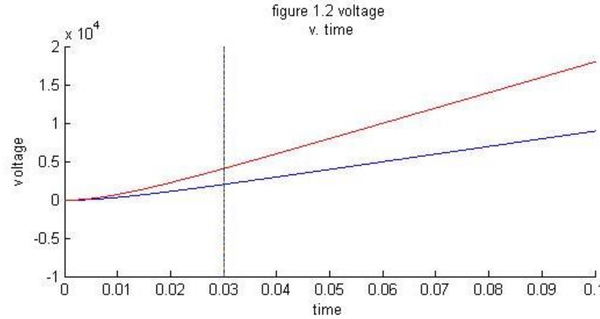



figure 1.3 derivative of voltage v. current

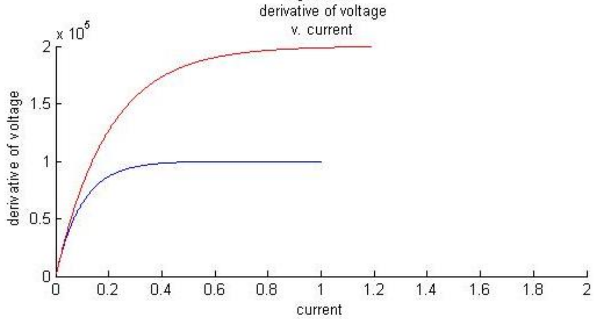
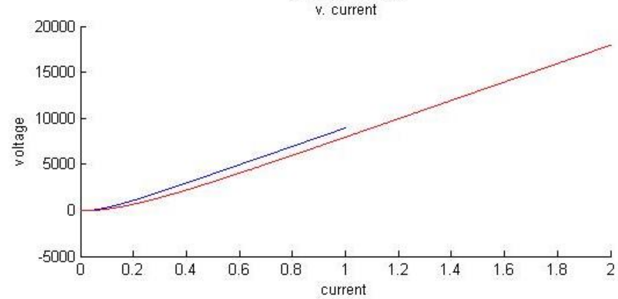



figure 1.4 voltage v. current




II. Remodeling

figure 1.5 current vs. time

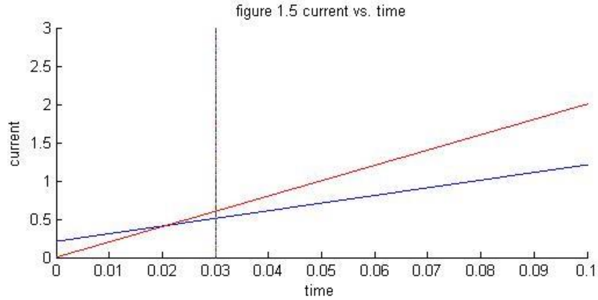



figure 1.6 voltage v. time

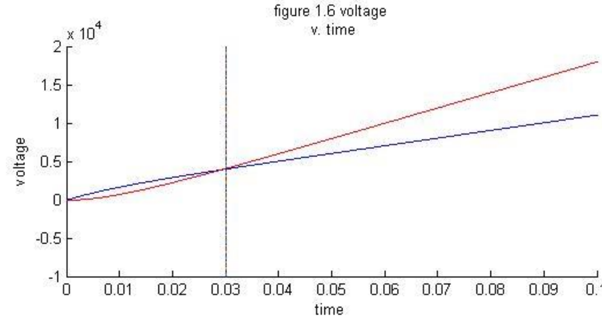



figure 1.7 derivative of voltage v. current

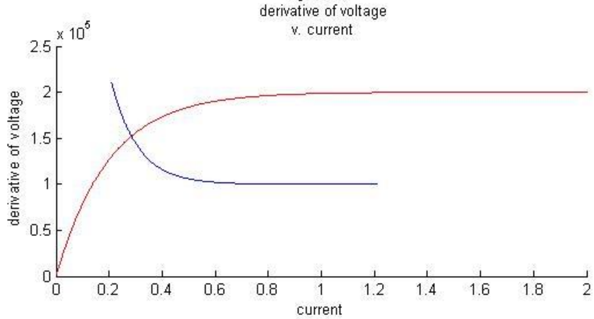
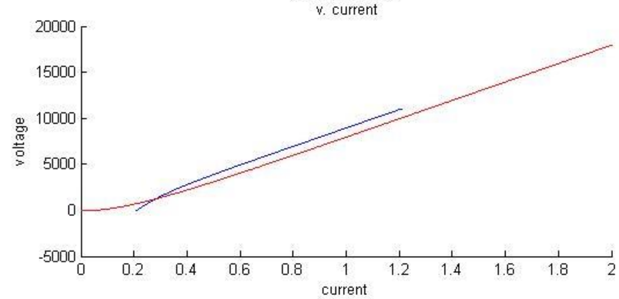
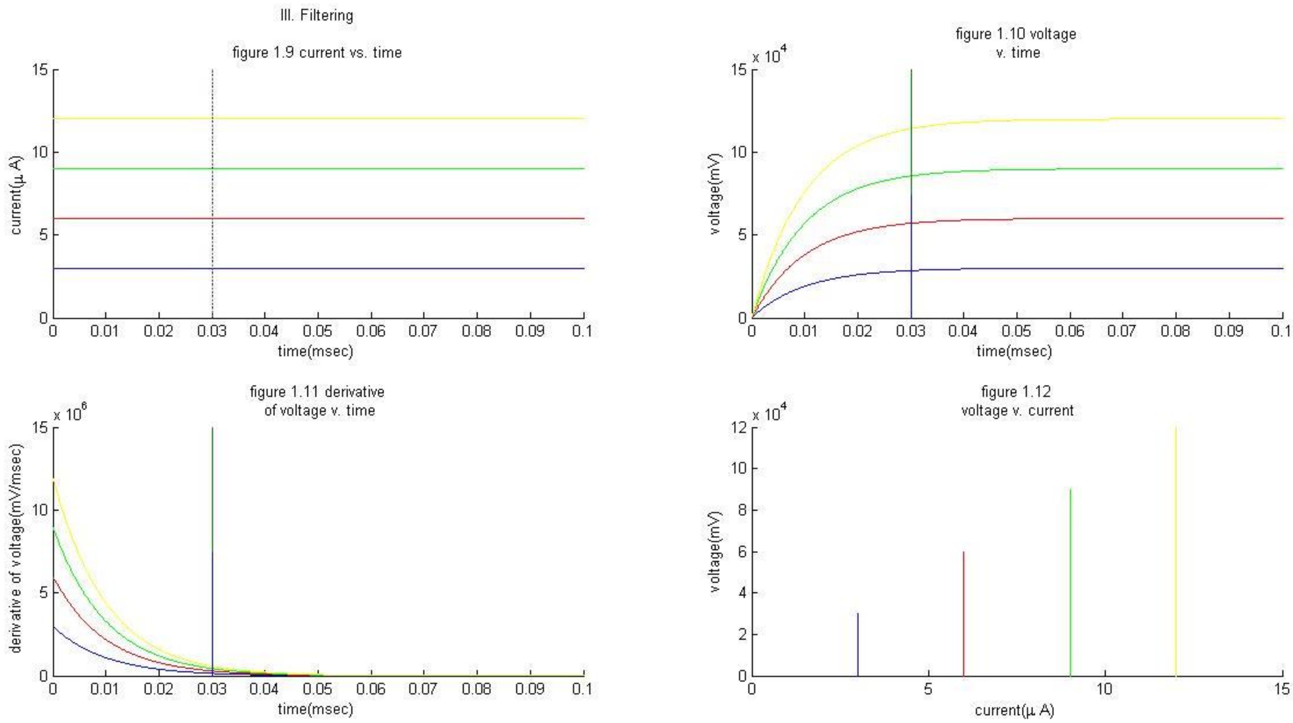
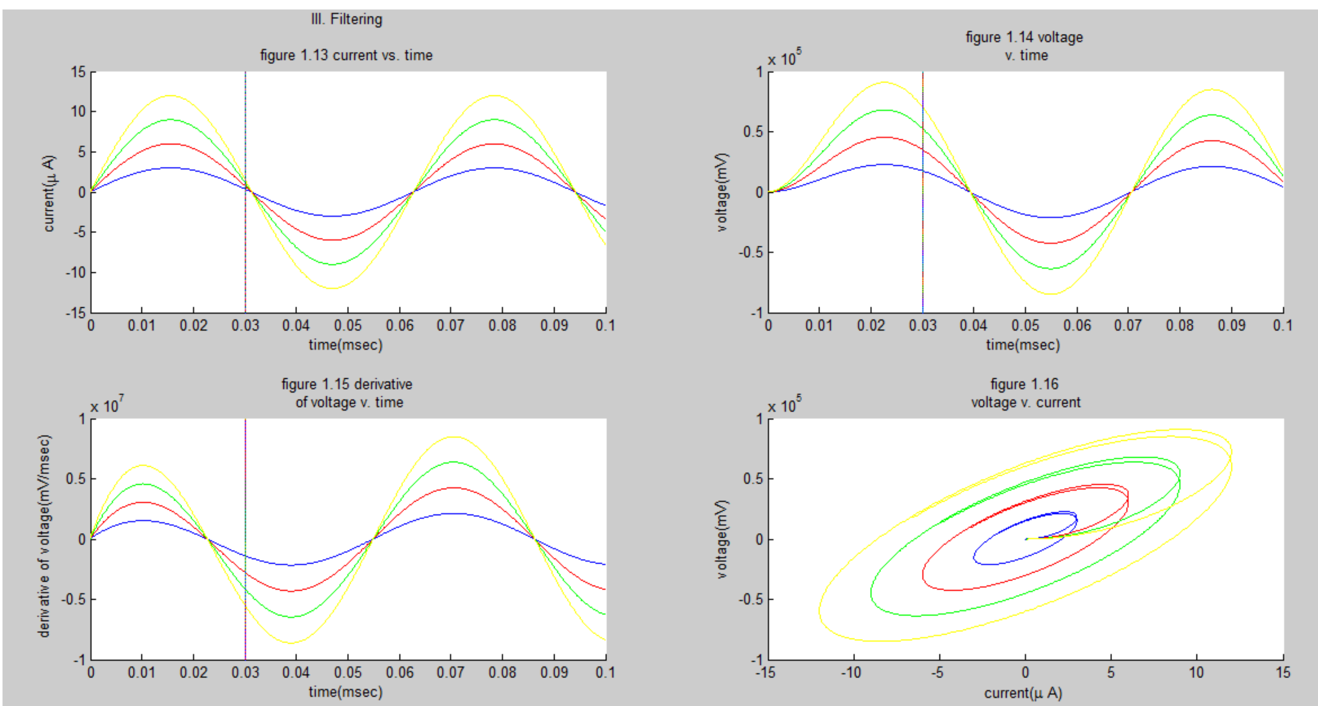


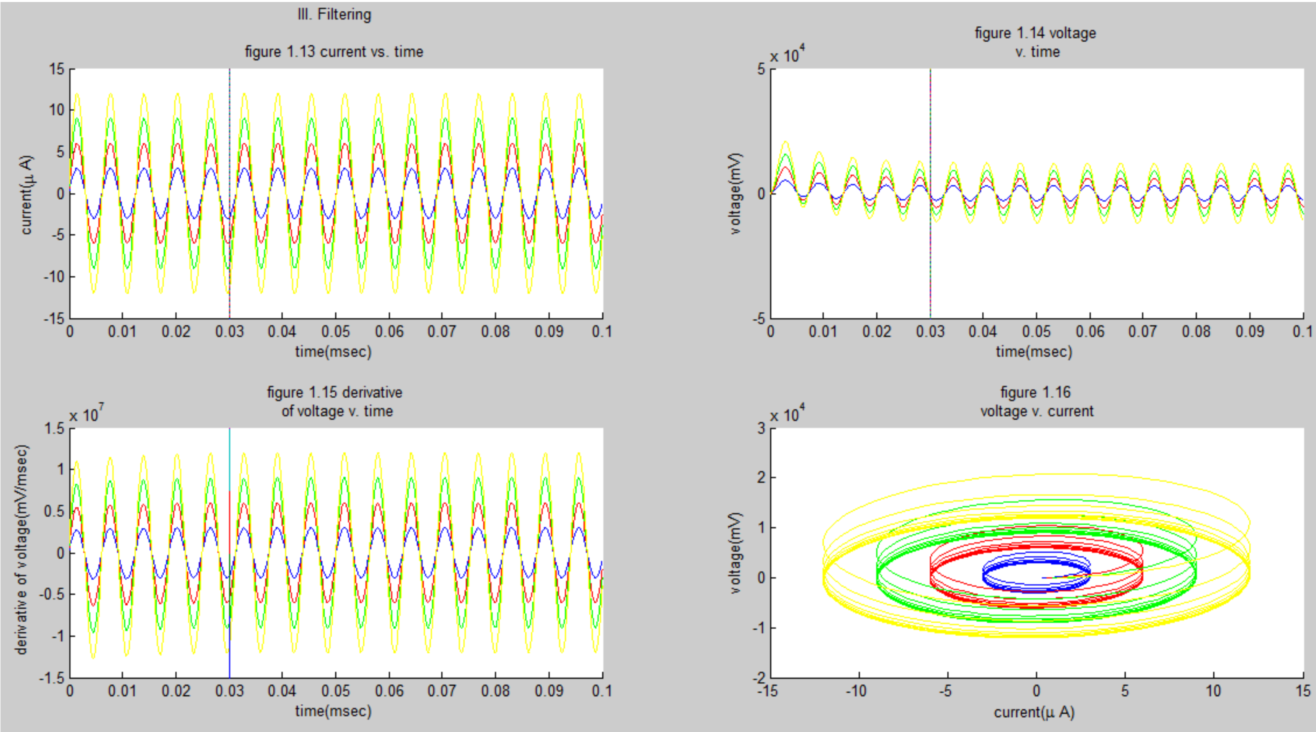

figure 1.8 voltage v. current



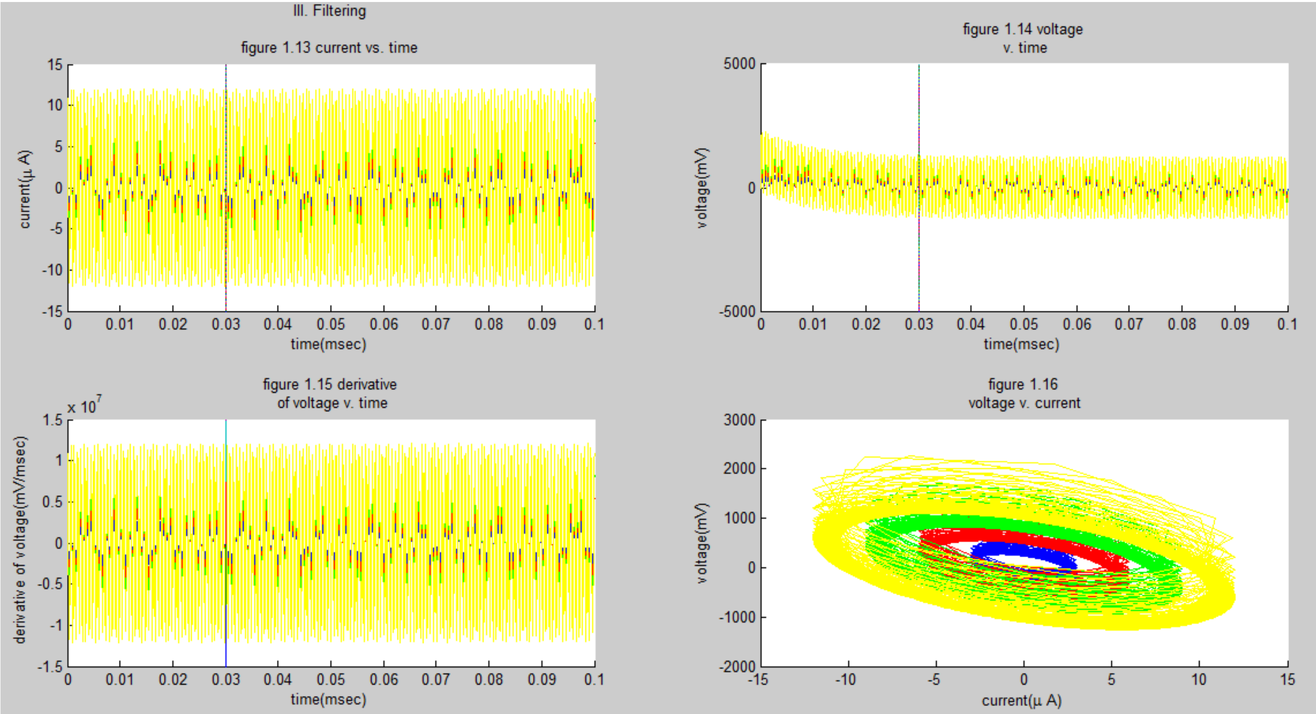

III. Filtering
figure 1.9 current vs. time
current(µ A)
time(msec)
figure 1.10 voltage v. time
voltage(mV)
time(msec)
figure 1.11 derivative of voltage v. time
derivative of voltage(mV/msec)
time(msec)
figure 1.12 voltage v. current
voltage(mV)
current(µ A)


III. Filtering
figure 1.13 current vs. time
current(µ A)
time(msec)
figure 1.14 voltage v. time
voltage(mV)
time(msec)
figure 1.15 derivative of voltage v. time
derivative of voltage(mV/msec)
time(msec)
figure 1.16 voltage v. current
voltage(mV)
current(µ A)

**W=1/100**

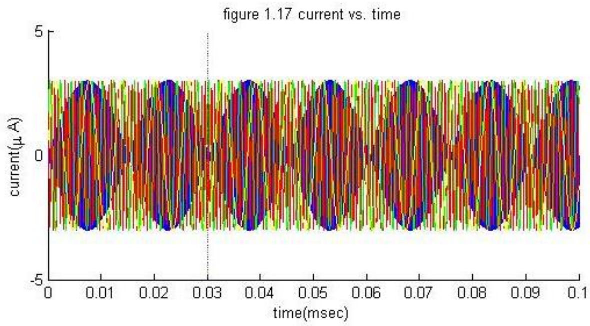

III. Filtering
figure 1.13 current vs. time
current(μ A)
time(msec)
figure 1.14 voltage v. time
voltage(mV)
time(msec)
figure 1.15 derivative of voltage v. time
derivative of voltage(mV/msec)
time(msec)
figure 1.16 voltage v. current
voltage(mV)
current(μ A)

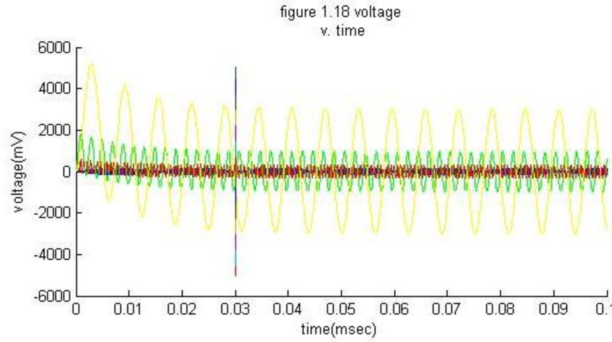
**W=1/10**

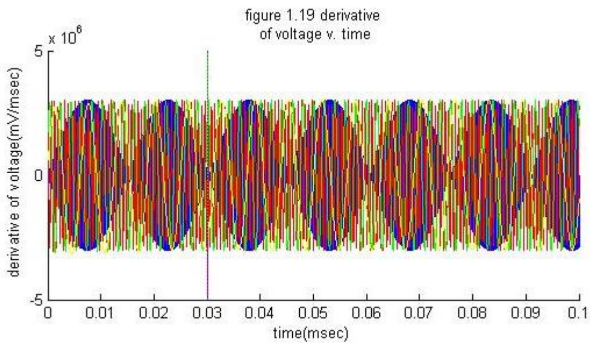

III. Filtering
figure 1.13 current vs. time
current(μ A)
time(msec)
figure 1.14 voltage v. time
voltage(mV)
time(msec)
figure 1.15 derivative of voltage v. time
derivative of voltage(mV/msec)
time(msec)
figure 1.16 voltage v. current
voltage(mV)
current(μ A)

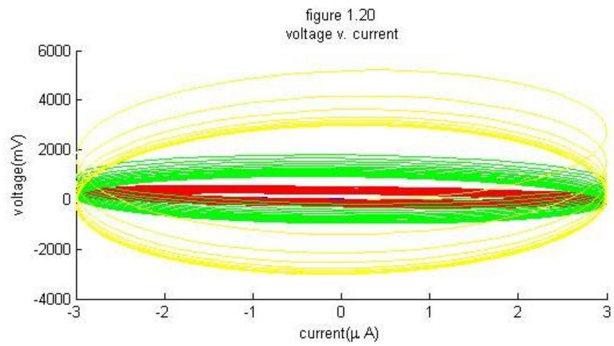
**W=1**

III. Filtering

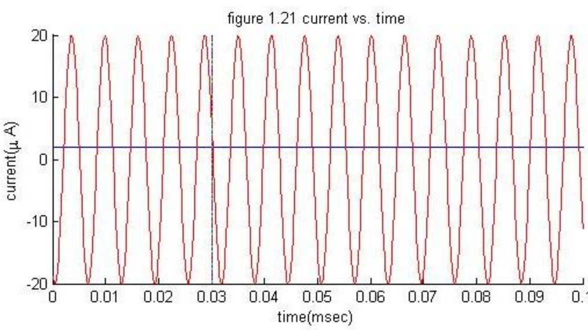


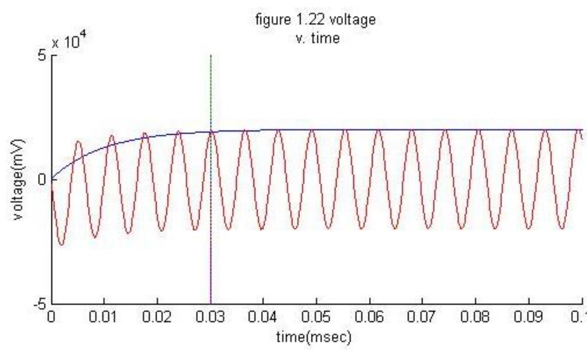



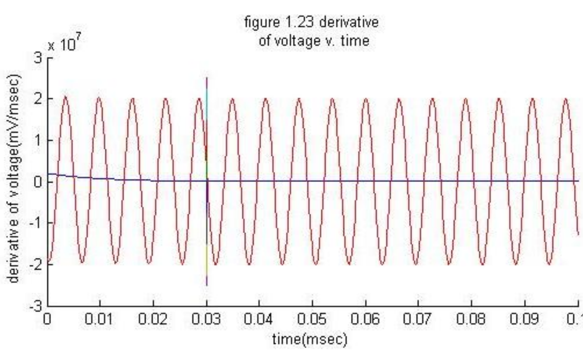



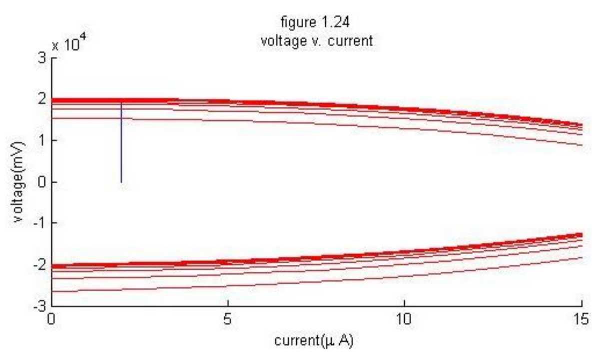



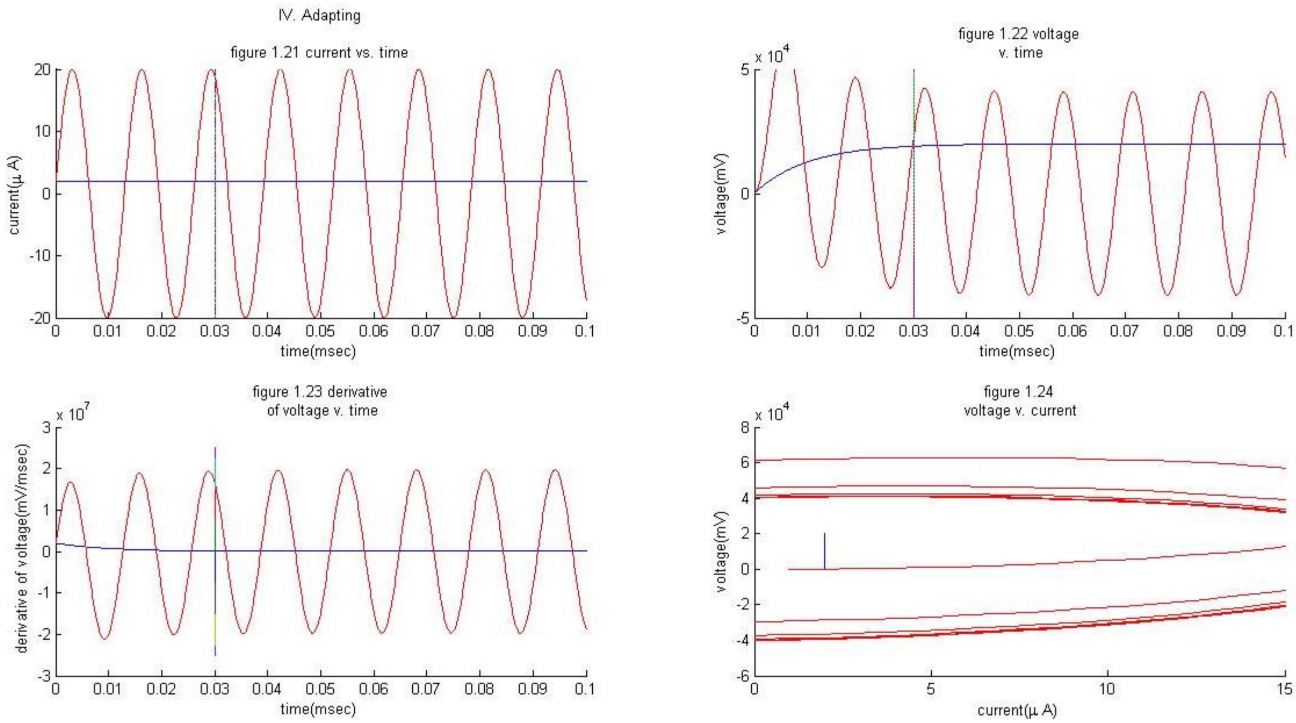

IV. Adapting

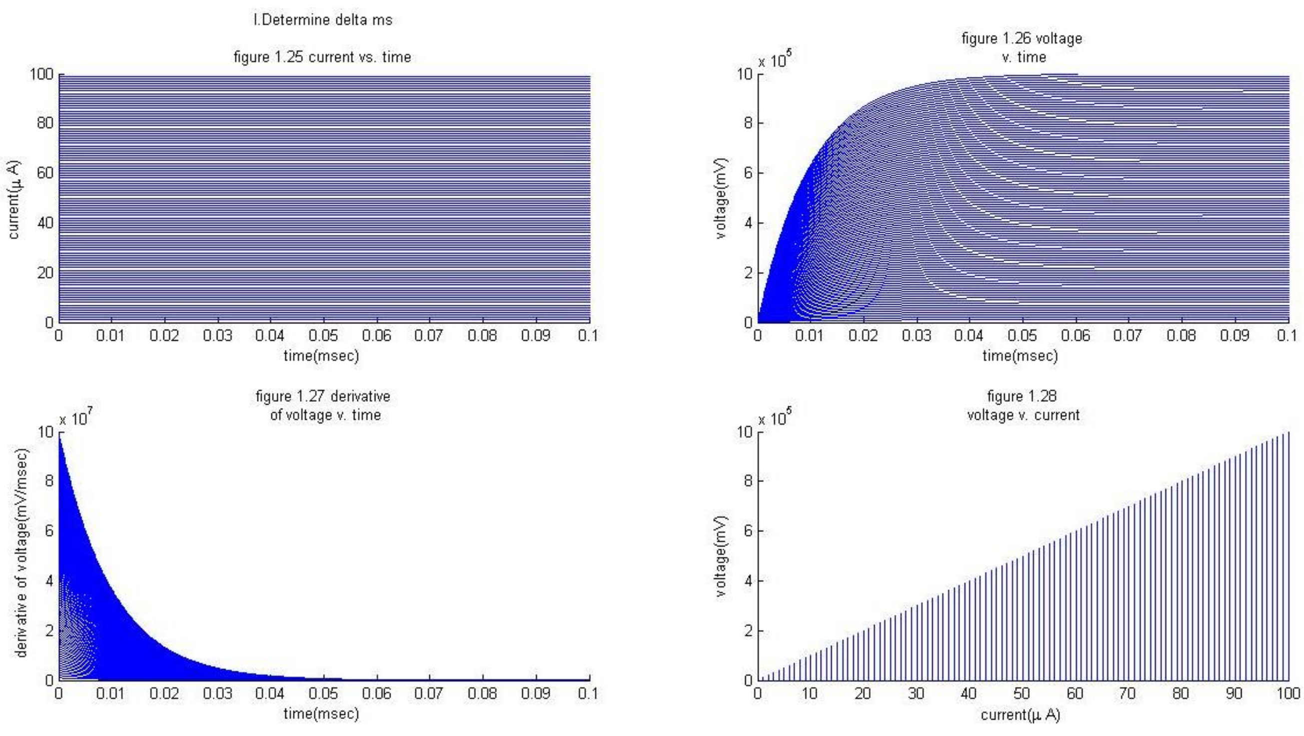


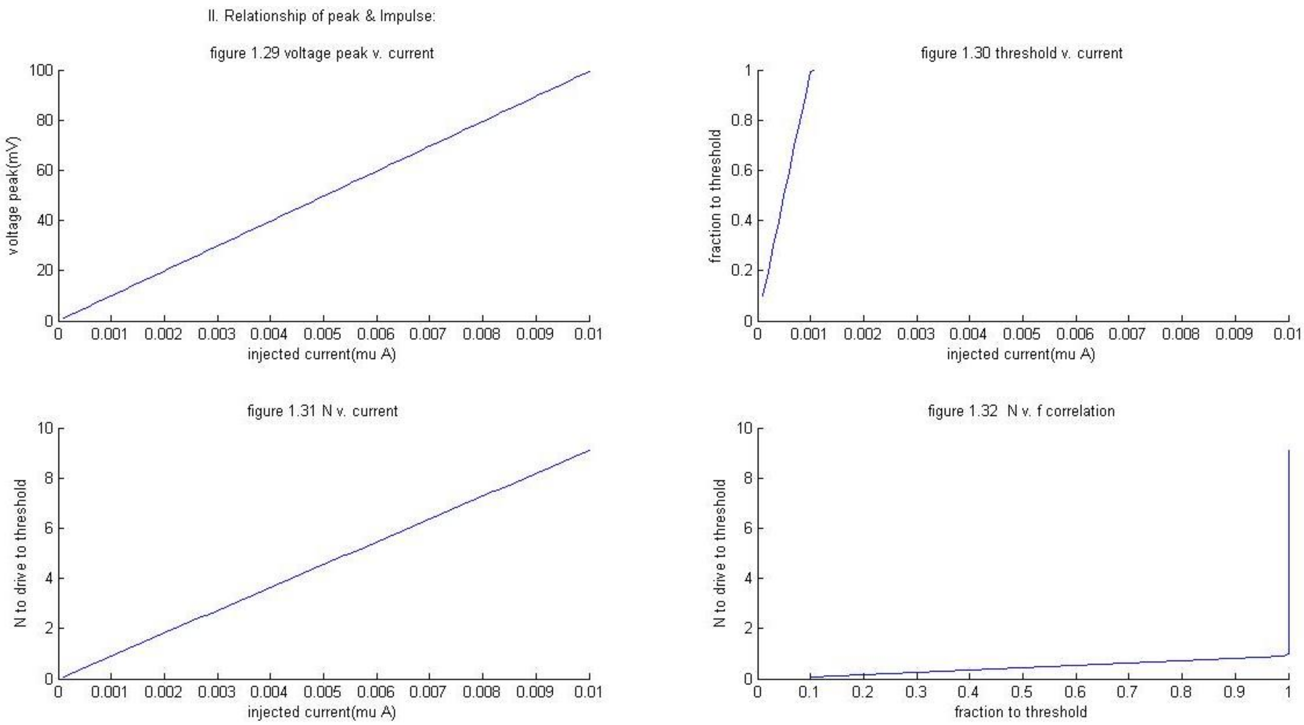



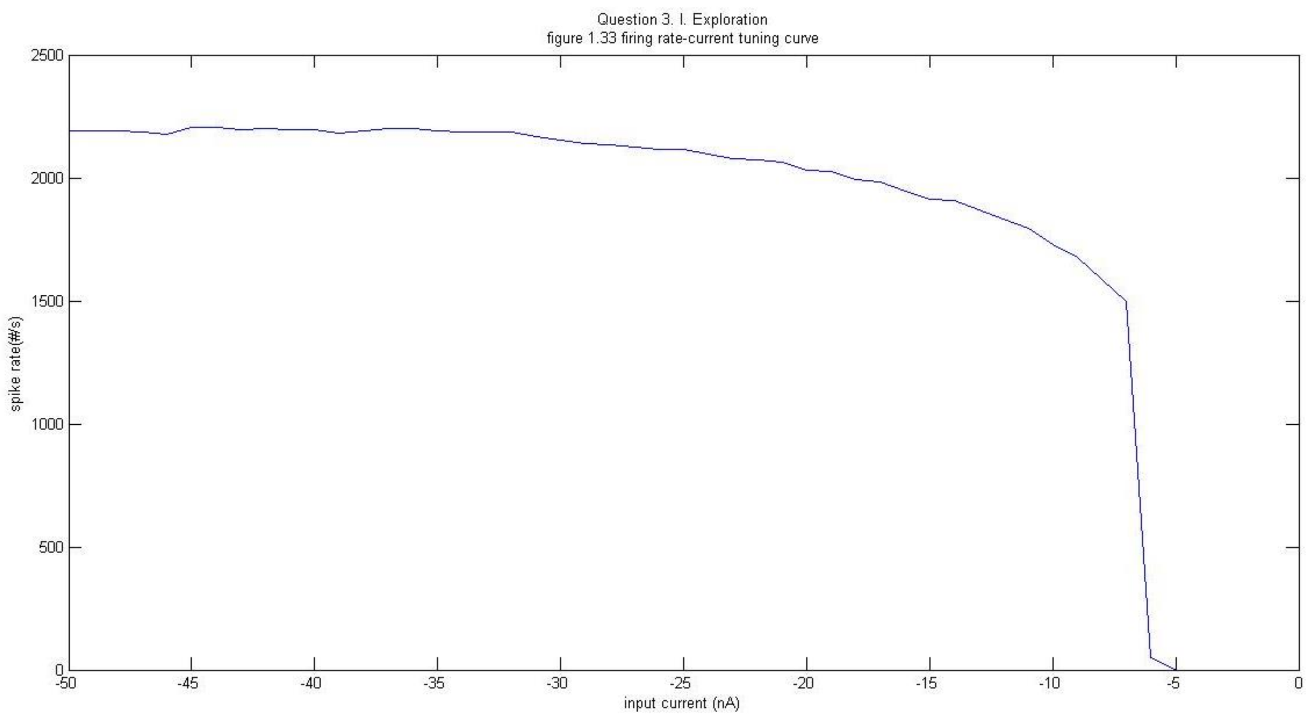



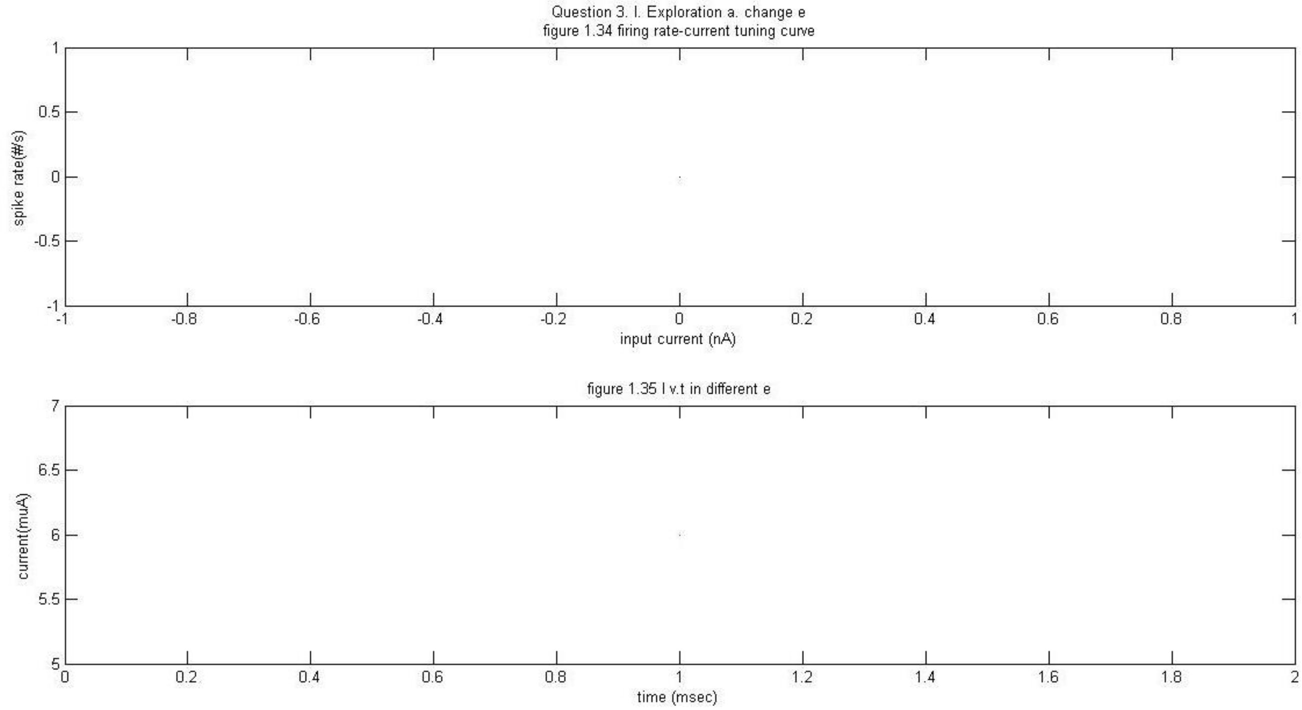



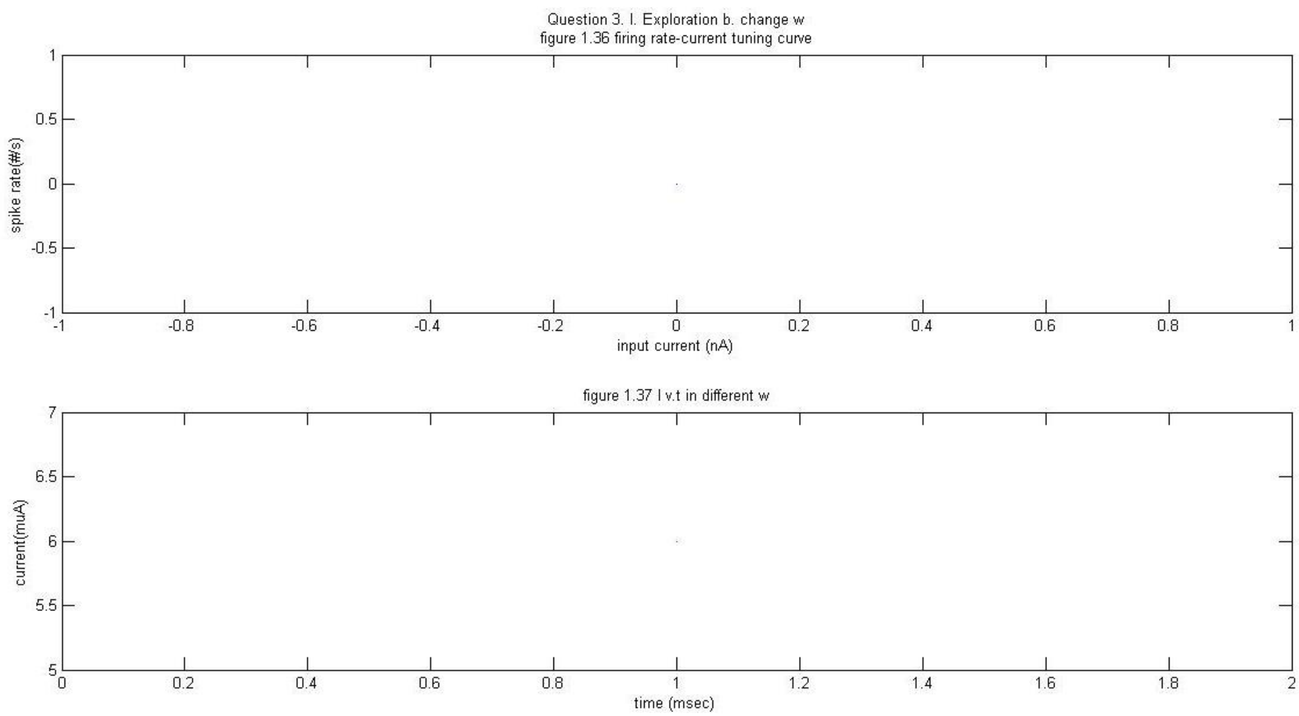

IV. Adapting
figure 1.21 current vs. time
current(μ A)
time(msec)
figure 1.22 voltage v. time
voltage(mV)
time(msec)
figure 1.23 derivative of voltage v. time
derivative of voltage(mV/msec)
time(msec)
figure 1.24 voltage v. current
voltage(mV)
current(μ A)


I.Determine delta ms
figure 1.25 current vs. time
current(μ A)
time(msec)
figure 1.26 voltage v. time
voltage(mV)
time(msec)
figure 1.27 derivative of voltage v. time
derivative of voltage(mV/msec)
time(msec)
figure 1.28 voltage v. current
voltage(mV)
current(μ A)


II. Relationship of peak & Impulse:
figure 1.29 voltage peak v. current
voltage peak(mV)
injected current(mu A)
figure 1.30 threshold v. current
fraction to threshold
injected current(mu A)
figure 1.31 N v. current
N to drive to threshold
injected current(mu A)
figure 1.32 N v. f correlation
N to drive to threshold
fraction to threshold


Question 3. I. Exploration
figure 1.33 firing rate-current tuning curve
spike rate(#/s)
input current (nA)


Question 3. I. Exploration a. change e
figure 1.34 firing rate-current tuning curve
spike rate(#/s)
input current (nA)
figure 1.35 I v.t in different e
current(muA)
time (msec)


Question 3. I. Exploration b. change w
figure 1.36 firing rate-current tuning curve
spike rate(#/s)
input current (nA)
figure 1.37 I v.t in different w
current(muA)
time (msec)

# Bibliography:

Driscoll, T. A. (2009). *Learning MATLAB*. Philadelphia, PA: Society for Industrial and Applied Mathematics.

Pärt-Enander, E. (1996). *The MATLAB handbook*. Harlow, England: Reading, Mass.

Gonzalez, R. C., Woods, R. E., & Eddins, S. L. (2004). *Digital Image processing using MATLAB*. Upper Saddle River, N.J: Pearson Prentice Hall.

Feng, J. (2004). *Computational neuroscience: Comprehensive approach*. Boca Raton: Chapman & Hall/CRC.

# Acknowledgement:

Thank Prof. Adrienne Fairhall for giving us insightful lectures about the dynamic field of computational neuroscience and setting challenging questions for us to explore!

Thank Dr. Alison Weber for clarifying the concepts vividly and giving me instrumental suggestions and guidance throughout the problem solving selflessly with her busy time!

Thank my friends in NBIO 303 who support me as a freshman without MATLAB experience with encouragement and tolerance!

Thank University of Washington for giving us the platform to scientifically explore problems and subjects!

I will continue the voyage of exploring the infinite realm of computational neuroscience in my academic career fearlessly.

Baihan Lin
January 2014