UNIVERSITY
OF WOLLONGONG
IN DUBAI

Assignment Cover Sheet

Student Name	Student number
Taha Yaseen Parker	8243578

Subject code and name	ECTE213 – Engineering Electromagnetics
Lab Instructor	Mr. Mahmoud Alkakuri
Title of Assignment	Lab 3
Date and time due	04 February 2025, 23.55
Lab Number	3

Student declaration and acknowledgment

By submitting this assignment online, the submitting student declares on behalf of the team that:

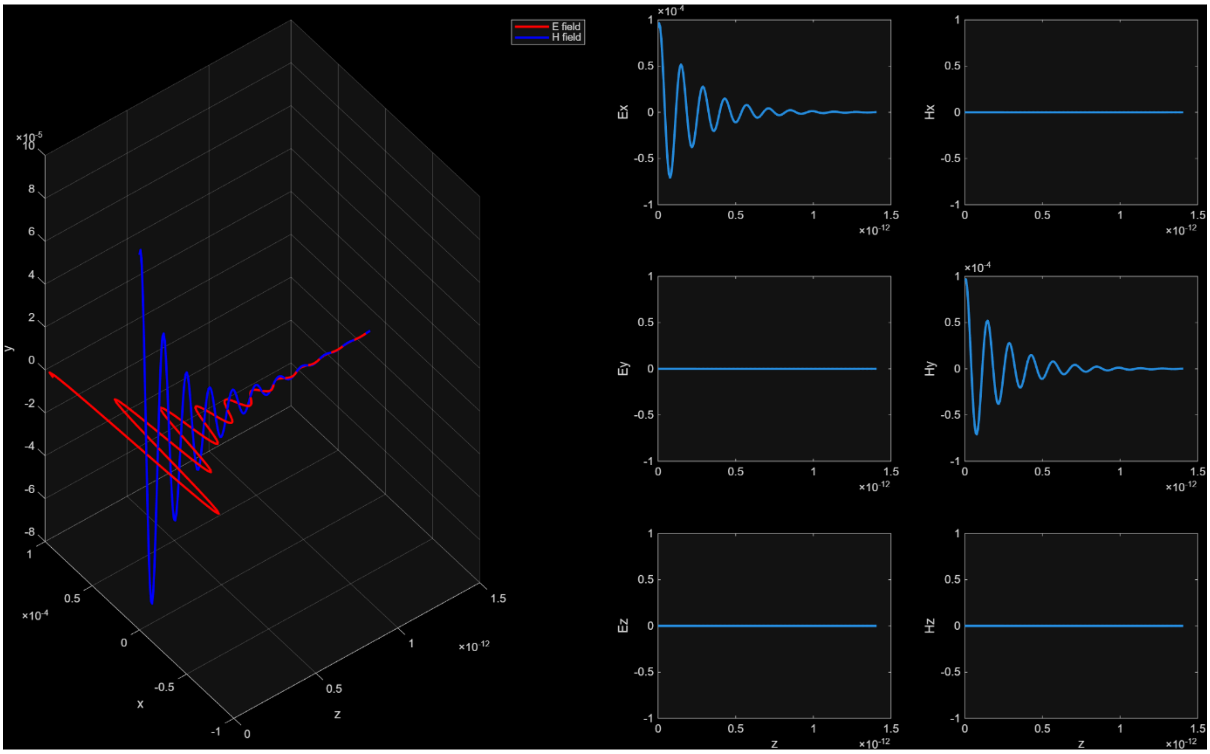
1. All team members have read the subject outline for this subject, and this assessment item meets the requirements of the subject detailed therein.
2. This assessment is entirely our work, except where we have included fully documented references to the work of others. The material in this assessment item has yet to be submitted for assessment.
3. Acknowledgement of source information is by the guidelines or referencing style specified in the subject outline.
4. All team members know the late submission policy and penalty.
5. The submitting student undertakes to communicate all feedback with the other team members.

Lab 3

Task 3.1 a.

```
function TEM_Lossy()
% Function: Demonstrate the TEM wave is a lossy medium
% Parameters:
%          f: frequency in Hz
%          epsilon_r: relative permittivity
%          mu_r: relative permeability
%          t: time of interest
%          E0: peak amplitude of the electric field intensity

%% Set the frequency and medium parameters
f = 100^6;
omega = 2*pi*f;
T = 2*pi/omega;
t = T/15;
epsilon_r =50-10j;
mu_r = 1;
E0 = 0.0001;

%% Compute wave parameters
c = 3*10^8;
k = omega*sqrt(mu_r*epsilon_r);
alpha = -imag(k);
beta = real(k);
V_p = c/sqrt(mu_r*epsilon_r);
lambda = 2*pi/beta;
eta = 377*sqrt(mu_r/epsilon_r);
scaleH = abs(eta);

%% Compute field distributions
z = [0:lambda/20:10*lambda]';
Ex=zeros(size(z)); Ey=zeros(size(z)); Ez=zeros(size(z));
Hx=zeros(size(z)); Hy=zeros(size(z)); Hz=zeros(size(z));
Sx=zeros(size(z)); Sy=zeros(size(z)); Sz=zeros(size(z));
Ex = E0*exp(-alpha*z).*cos(omega*t-beta*z);
Hy= E0./abs(eta)*exp(-alpha*z).*cos(omega*t-beta*z-angle(eta));
Sz = 1/eta*Ex.^2;

figure('units','normalized','outerposition',[0 0 1 1])
subplot(3,4,[1,2,5,6,9,10]);
plot3(z, Ex, Ey,'r', 'linewidth',2)
hold on
plot3(z, scaleH*Hx, scaleH*Hy,'b', 'linewidth',2)
hold on
% arrow3([z zeros(size(z)) zeros(size(z)) ], [z Ex Ey ])
% arrow3([z zeros(size(z)) zeros(size(z)) ], [z scaleH*Hx scaleH*Hy ])
legend('E field', 'H field'); grid on; xlabel('z'); ylabel('x'); zlabel('y')
subplot(3,4,3);plot(z, Ex, 'linewidth',2); ylabel('Ex')
subplot(3,4,7);plot(z, Ey, 'linewidth',2); ylabel('Ey')
```

```
subplot(3,4,11);plot(z, Ez, 'linewidth',2); xlabel('z'); ylabel('Ez')
subplot(3,4,4);plot(z, scaleH*Hx, 'linewidth',2); ylabel('Hx');
subplot(3,4,8);plot(z, scaleH*Hy, 'linewidth',2); ylabel('Hy');
subplot(3,4,12);plot(z, scaleH*Hz, 'linewidth',2); xlabel('z'); ylabel('Hz')
```



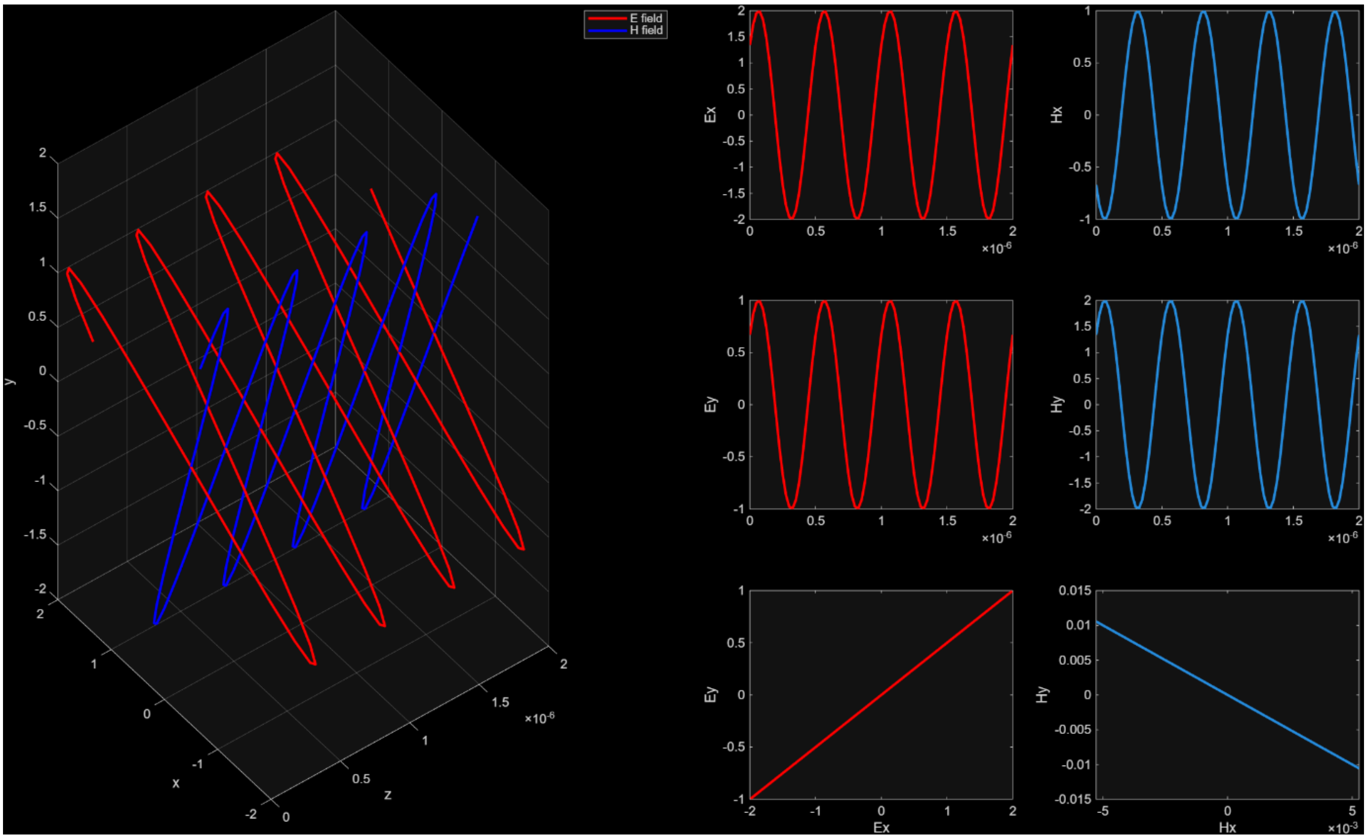
Task 3.1 b.

a.

```
function TEM_Polarized( )

%% Function: Simulate polarized TEM waves
%  Parameters: f: frequency in Hz
%              Ex0: amplitude of Ex component
%              Ey0: amplitude of Ey component
%              phi: phase difference between the Ex and Ey components
%              mu_r, epsilon_r: relative permittivity and permeability of the material
%              t: time of interest

%% Initialize parameters
Ex0 = 2;
Ey0 = 1;
phi = 0;

% The type is linear

f = 10^6;
omega = 2*pi*f;
T = 2*pi/omega;
t = 2*T/15;

epsilon_r = 4;
mu_r = 1;

%% Wave parameters
c = 3*10^8;
k = omega*sqrt(mu_r*epsilon_r);      % Propagation constant
alpha = -imag(k);                               % Attenuation constant
beta = real(k);                                    % Phase constant
lambda = 2*pi/beta;
V_p = c/sqrt(mu_r*epsilon_r);        % Phase velocity
eta=377*sqrt(mu_r/epsilon_r);        % Wave impedance
scaleH = abs(eta);                             % Only for visualization

%% Compute and plot the field components
z=[0:lambda/20:4*lambda]';                                                 % Grid points
on z axis
Ex=zeros(size(z)); Ey=zeros(size(z)); Ez=zeros(size(z));    % Initialization
Sx=zeros(size(z)); Sy=zeros(size(z)); Sz=zeros(size(z));
Ex = Ex0*exp(-alpha*z).*cos(omega*t-beta*z);                   % Ex component
Hy = Ex0/eta*exp(-alpha*z).*cos(omega*t-beta*z);            % Hy component
Ey = Ey0*exp(-alpha*z).*cos(omega*t-beta*z+phi);           % Ey component
Hx = -Ey0/eta*exp(-alpha*z).*cos(omega*t-beta*z+phi);    % Hx component

Sz = 1/eta*(abs(Ex.^2)+abs(Ey.^2));                               % Power density
figure('units','normalized','outerposition',[0 0 1 1])

subplot(3,4,[1,2,5,6,9,10]);
```

```
plot3(z, Ex, Ey,'r', 'linewidth',2)                                    % Plot the E field using plot3
hold on
plot3(z, scaleH*Hx, scaleH*Hy,'b', 'linewidth',2)           % Plot the H field using plot3 with scaling
hold on
% arrow3([z zeros(size(z)) zeros(size(z)) ], [z Ex Ey ])       % Plot the E field using arrow3
% arrow3([z zeros(size(z)) zeros(size(z)) ], [z scaleH*Hx scaleH*Hy ]) % Plot the H field using arrow3 with scaling
legend('E field', 'H field'); grid on; xlabel('z'); ylabel('x'); zlabel('y')

subplot(3,4,3);plot(z, Ex, 'r', 'linewidth',2); ylabel('Ex')                % Plot Ex only
subplot(3,4,7);plot(z, Ey, 'r','linewidth',2); ylabel('Ey')                 % Plot Ey only
subplot(3,4,11);plot(Ex, Ey, 'r', 'linewidth',2); xlabel('Ex'); ylabel('Ey') % Plot (Ex, Ey) to see the polarization type
subplot(3,4,4);plot(z, scaleH*Hx, 'linewidth',2); ylabel('Hx');       % Plot Hx only with scaling
subplot(3,4,8);plot(z, scaleH*Hy, 'linewidth',2); ylabel('Hy');      % Plot Hy only with scaling
subplot(3,4,12);plot(Hx, Hy, 'linewidth',2); xlabel('Hx'); ylabel('Hy')% Plot (Hx, Hy) to see the polarization type
```

The type of polarization is linear.

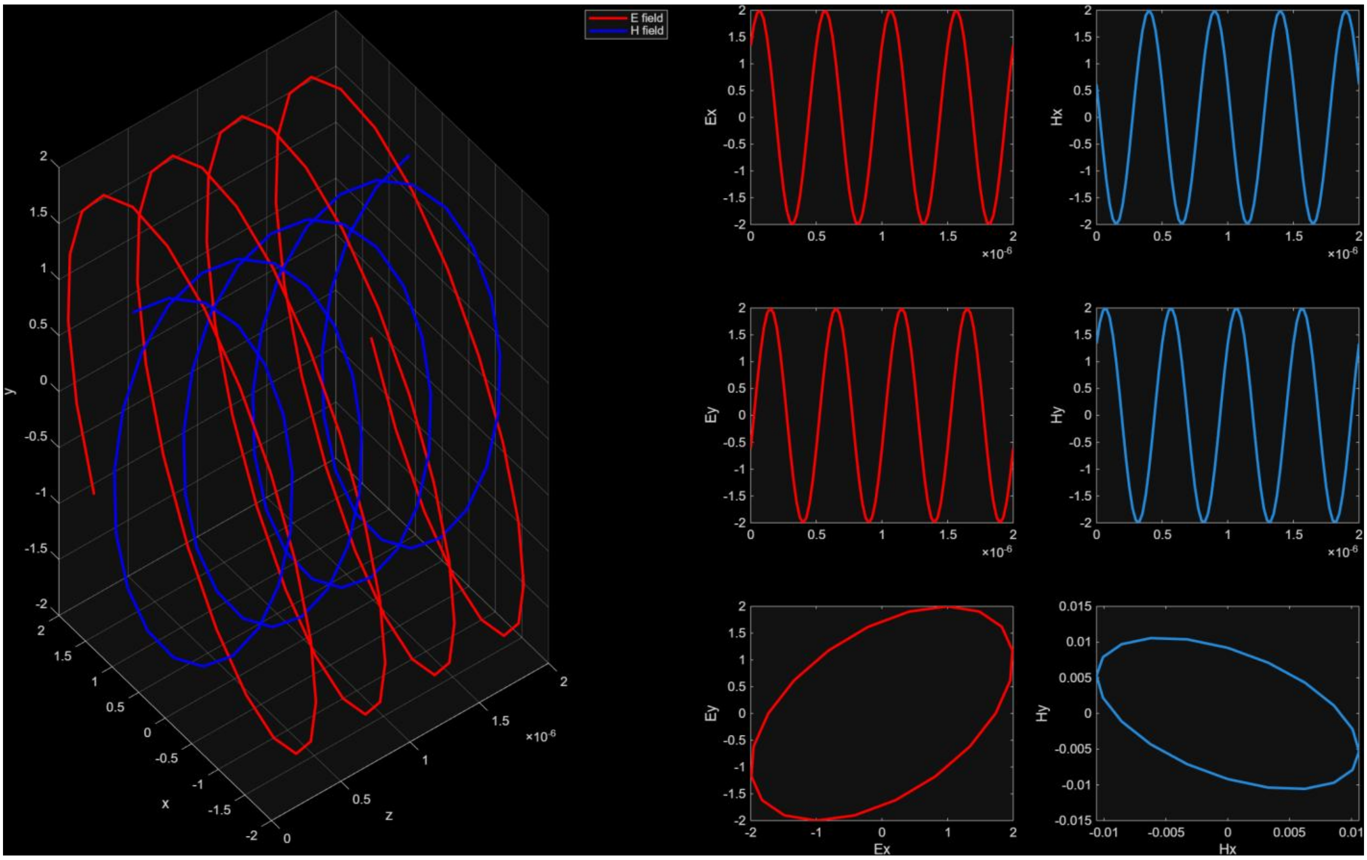


b.

```
function TEM_Polarized( )

%% Function: Simulate polarized TEM waves
%  Parameters: f: frequency in Hz
%              Ex0: amplitude of Ex component
%              Ey0: amplitude of Ey component
%              phi: phase difference between the Ex and Ey components
%              mu_r, epsilon_r: relative permittivity and permeability of the material
%              t: time of interest

%% Initialize parameters
Ex0 = 2;
Ey0 = 2;
phi = pi/3;

% The type is elliptical

f = 10^6;
omega = 2*pi*f;
T = 2*pi/omega;
t = 2*T/15;

epsilon_r = 4;
mu_r = 1;

%% Wave parameters
c = 3*10^8;
k = omega*sqrt(mu_r*epsilon_r);      % Propagation constant
alpha = -imag(k);                              % Attenuation constant
beta = real(k);                                   % Phase constant
lambda = 2*pi/beta;
V_p = c/sqrt(mu_r*epsilon_r);        % Phase velocity
eta=377*sqrt(mu_r/epsilon_r);        % Wave impedance
scaleH = abs(eta);                             % Only for visualization

%% Compute and plot the field components
z=[0:lambda/20:4*lambda]';                                                   % Grid points
on z axis
Ex=zeros(size(z)); Ey=zeros(size(z)); Ez=zeros(size(z));    % Initialization
Sx=zeros(size(z)); Sy=zeros(size(z)); Sz=zeros(size(z));
Ex = Ex0*exp(-alpha*z).*cos(omega*t-beta*z);                      % Ex component
Hy = Ex0/eta*exp(-alpha*z).*cos(omega*t-beta*z);               % Hy component
Ey = Ey0*exp(-alpha*z).*cos(omega*t-beta*z+phi);              % Ey component
Hx = -Ey0/eta*exp(-alpha*z).*cos(omega*t-beta*z+phi);     % Hx component

Sz = 1/eta*(abs(Ex.^2)+abs(Ey.^2));                                      % Power density
figure('units','normalized','outerposition',[0 0 1 1])

subplot(3,4,[1,2,5,6,9,10]);
plot3(z, Ex, Ey,'r', 'linewidth',2)                                        % Plot the E
field using plot3
```

```
hold on
plot3(z, scaleH*Hx, scaleH*Hy,'b', 'linewidth',2)          % Plot the H field using plot3 with scaling
hold on
% arrow3([z zeros(size(z)) zeros(size(z)) ], [z Ex Ey ])       % Plot the E field using arrow3
% arrow3([z zeros(size(z)) zeros(size(z)) ], [z scaleH*Hx scaleH*Hy ]) % Plot the H field using arrow3 with scaling
legend('E field', 'H field'); grid on; xlabel('z'); ylabel('x'); zlabel('y')

subplot(3,4,3);plot(z, Ex, 'r', 'linewidth',2); ylabel('Ex')                  % Plot Ex only
subplot(3,4,7);plot(z, Ey, 'r','linewidth',2); ylabel('Ey')                    % Plot Ey only
subplot(3,4,11);plot(Ex, Ey, 'r', 'linewidth',2); xlabel('Ex'); ylabel('Ey') % Plot (Ex, Ey) to see the polarization type
subplot(3,4,4);plot(z, scaleH*Hx, 'linewidth',2); ylabel('Hx');       % Plot Hx only with scaling
subplot(3,4,8);plot(z, scaleH*Hy, 'linewidth',2); ylabel('Hy');       % Plot Hy only with scaling
subplot(3,4,12);plot(Hx, Hy, 'linewidth',2); xlabel('Hx'); ylabel('Hy')% Plot (Hx, Hy) to see the polarization type
```

The type of polarization is elliptical.

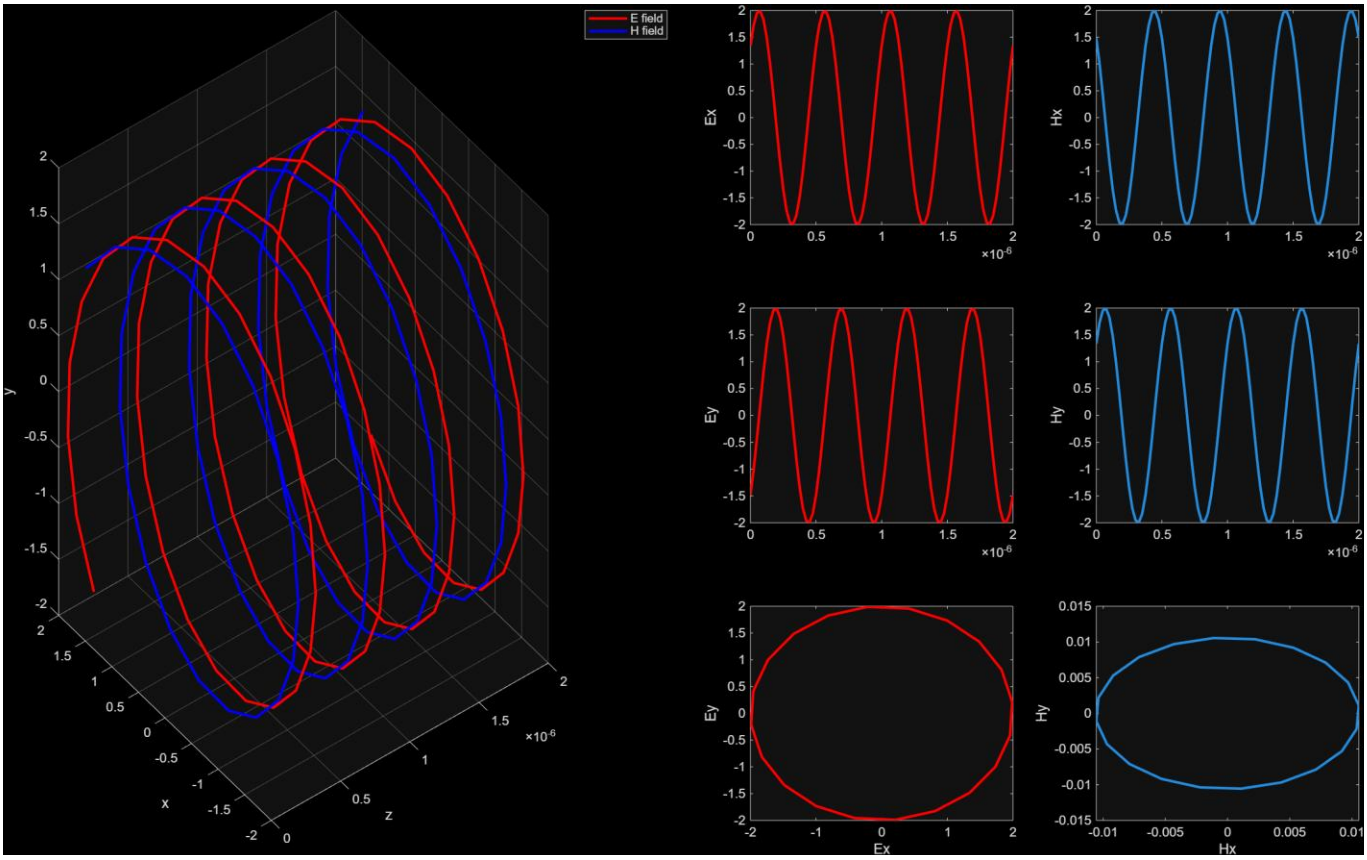


c.

```
function TEM_Polarized( )

%% Function: Simulate polarized TEM waves
%  Parameters: f: frequency in Hz
%              Ex0: amplitude of Ex component
%              Ey0: amplitude of Ey component
%              phi: phase difference between the Ex and Ey components
%              mu_r, epsilon_r: relative permittivity and permeability of the material
%              t: time of interest

%% Initialize parameters
Ex0 = 2;
Ey0 = 2;
phi = pi/2;

% The type is circular

f = 10^6;
omega = 2*pi*f;
T = 2*pi/omega;
t = 2*T/15;

epsilon_r = 4;
mu_r = 1;

%% Wave parameters
c = 3*10^8;
k = omega*sqrt(mu_r*epsilon_r);      % Propagation constant
alpha = -imag(k);                              % Attenuation constant
beta = real(k);                                   % Phase constant
lambda = 2*pi/beta;
V_p = c/sqrt(mu_r*epsilon_r);         % Phase velocity
eta=377*sqrt(mu_r/epsilon_r);         % Wave impedance
scaleH = abs(eta);                             % Only for visualization

%% Compute and plot the field components
z=[0:lambda/20:4*lambda]';                                                 % Grid points on
z axis
Ex=zeros(size(z)); Ey=zeros(size(z)); Ez=zeros(size(z));    % Initialization
Sx=zeros(size(z)); Sy=zeros(size(z)); Sz=zeros(size(z));
Ex = Ex0*exp(-alpha*z).*cos(omega*t-beta*z);                       % Ex component
Hy = Ex0/eta*exp(-alpha*z).*cos(omega*t-beta*z);                % Hy component
Ey = Ey0*exp(-alpha*z).*cos(omega*t-beta*z+phi);              % Ey component
Hx = -Ey0/eta*exp(-alpha*z).*cos(omega*t-beta*z+phi);     % Hx component

Sz = 1/eta*(abs(Ex.^2)+abs(Ey.^2));                                  % Power density
figure('units','normalized','outerposition',[0 0 1 1])

subplot(3,4,[1,2,5,6,9,10]);
plot3(z, Ex, Ey,'r', 'linewidth',2)                                        % Plot the E field
using plot3
```

```
hold on
plot3(z, scaleH*Hx, scaleH*Hy,'b', 'linewidth',2)          % Plot the H field using plot3
with scaling
hold on
% arrow3([z zeros(size(z)) zeros(size(z)) ], [z Ex Ey ])       % Plot the E field using
arrow3
% arrow3([z zeros(size(z)) zeros(size(z)) ], [z scaleH*Hx scaleH*Hy ]) % Plot the H field
using arrow3 with scaling
legend('E field', 'H field'); grid on; xlabel('z'); ylabel('x'); zlabel('y')

subplot(3,4,3);plot(z, Ex, 'r', 'linewidth',2); ylabel('Ex')                % Plot Ex
only
subplot(3,4,7);plot(z, Ey, 'r','linewidth',2); ylabel('Ey')                 % Plot Ey only
subplot(3,4,11);plot(Ex, Ey, 'r', 'linewidth',2); xlabel('Ex'); ylabel('Ey') % Plot (Ex,
Ey) to see the polarization type
subplot(3,4,4);plot(z, scaleH*Hx, 'linewidth',2); ylabel('Hx');        % Plot Hx only with
scaling
subplot(3,4,8);plot(z, scaleH*Hy, 'linewidth',2); ylabel('Hy');        % Plot Hy only with
scaling
subplot(3,4,12);plot(Hx, Hy, 'linewidth',2); xlabel('Hx'); ylabel('Hy')% Plot (Hx, Hy) to
see the polarization type
```

The type of polarization is circular.

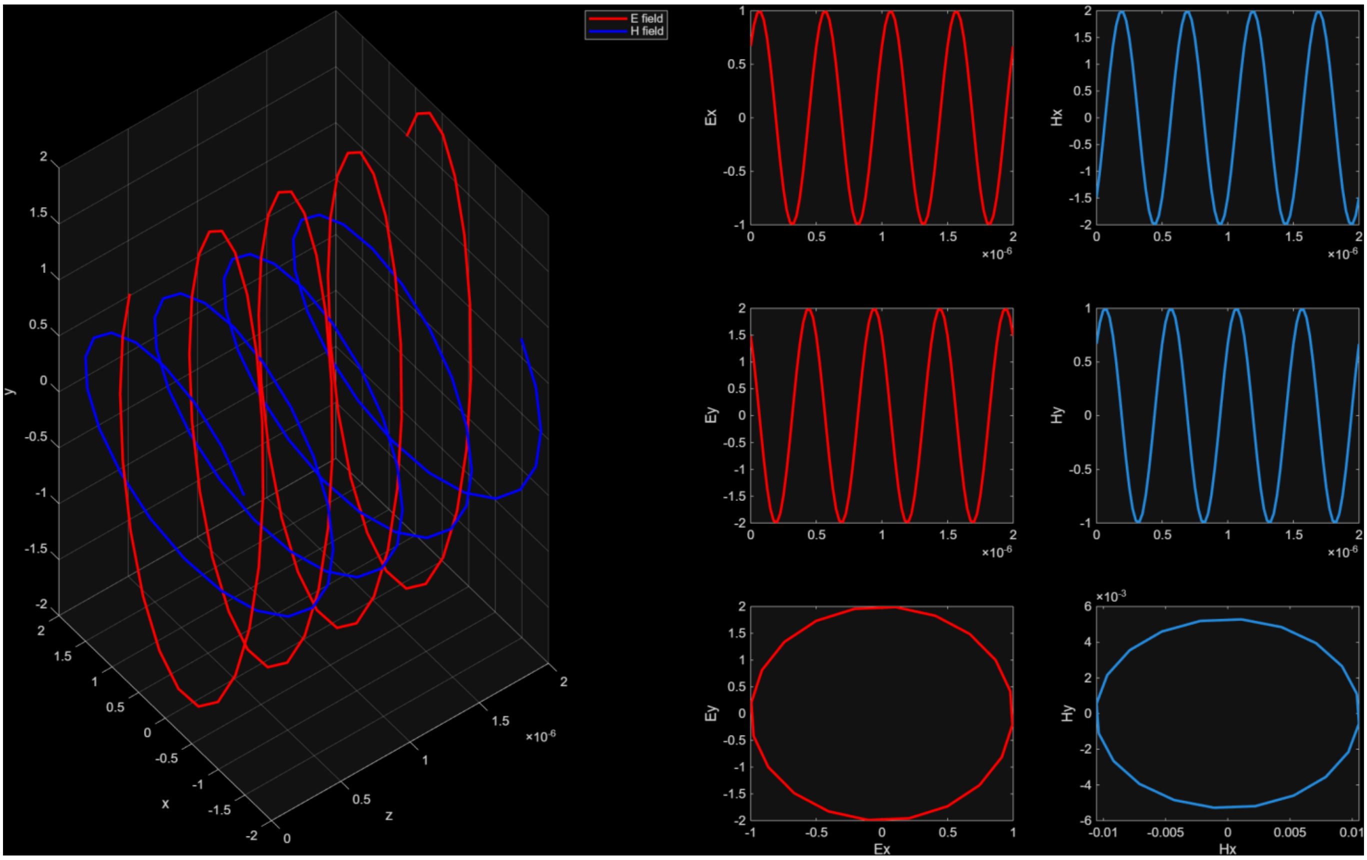


d.

```
function TEM_Polarized( )

%% Function: Simulate polarized TEM waves
%  Parameters: f: frequency in Hz
%              Ex0: amplitude of Ex component
%              Ey0: amplitude of Ey component
%              phi: phase difference between the Ex and Ey components
%              mu_r, epsilon_r: relative permittivity and permeability of the material
%              t: time of interest

%% Initialize parameters
Ex0 = 1;
Ey0 = 2;
phi = -pi/2;

% The type is elliptical

f = 10^6;
omega = 2*pi*f;
T = 2*pi/omega;
t = 2*T/15;

epsilon_r = 4;
mu_r = 1;

%% Wave parameters
c = 3*10^8;
k = omega*sqrt(mu_r*epsilon_r);      % Propagation constant
alpha = -imag(k);                              % Attenuation constant
beta = real(k);                                   % Phase constant
lambda = 2*pi/beta;
V_p = c/sqrt(mu_r*epsilon_r);          % Phase velocity
eta=377*sqrt(mu_r/epsilon_r);          % Wave impedance
scaleH = abs(eta);                             % Only for visualization

%% Compute and plot the field components
z=[0:lambda/20:4*lambda]';                                                   % Grid points on
z axis
Ex=zeros(size(z)); Ey=zeros(size(z)); Ez=zeros(size(z));    % Initialization
Sx=zeros(size(z)); Sy=zeros(size(z)); Sz=zeros(size(z));
Ex = Ex0*exp(-alpha*z).*cos(omega*t-beta*z);                    % Ex component
Hy = Ex0/eta*exp(-alpha*z).*cos(omega*t-beta*z);             % Hy component
Ey = Ey0*exp(-alpha*z).*cos(omega*t-beta*z+phi);            % Ey component
Hx = -Ey0/eta*exp(-alpha*z).*cos(omega*t-beta*z+phi);     % Hx component

Sz = 1/eta*(abs(Ex.^2)+abs(Ey.^2));                                    % Power density
figure('units','normalized','outerposition',[0 0 1 1])

subplot(3,4,[1,2,5,6,9,10]);
plot3(z, Ex, Ey,'r', 'linewidth',2)                                          % Plot the E field
using plot3
```

```
hold on
plot3(z, scaleH*Hx, scaleH*Hy,'b', 'linewidth',2)          % Plot the H field using plot3
with scaling
hold on
% arrow3([z zeros(size(z)) zeros(size(z)) ], [z Ex Ey ])       % Plot the E field using
arrow3
% arrow3([z zeros(size(z)) zeros(size(z)) ], [z scaleH*Hx scaleH*Hy ]) % Plot the H field
using arrow3 with scaling
legend('E field', 'H field'); grid on; xlabel('z'); ylabel('x'); zlabel('y')

subplot(3,4,3);plot(z, Ex, 'r', 'linewidth',2); ylabel('Ex')                % Plot Ex
only
subplot(3,4,7);plot(z, Ey, 'r','linewidth',2); ylabel('Ey')                 % Plot Ey only
subplot(3,4,11);plot(Ex, Ey, 'r', 'linewidth',2); xlabel('Ex'); ylabel('Ey') % Plot (Ex,
Ey) to see the polarization type
subplot(3,4,4);plot(z, scaleH*Hx, 'linewidth',2); ylabel('Hx');       % Plot Hx only with
scaling
subplot(3,4,8);plot(z, scaleH*Hy, 'linewidth',2); ylabel('Hy');       % Plot Hy only with
scaling
subplot(3,4,12);plot(Hx, Hy, 'linewidth',2); xlabel('Hx'); ylabel('Hy')% Plot (Hx, Hy) to
see the polarization type
```

The type of polarization is elliptical.

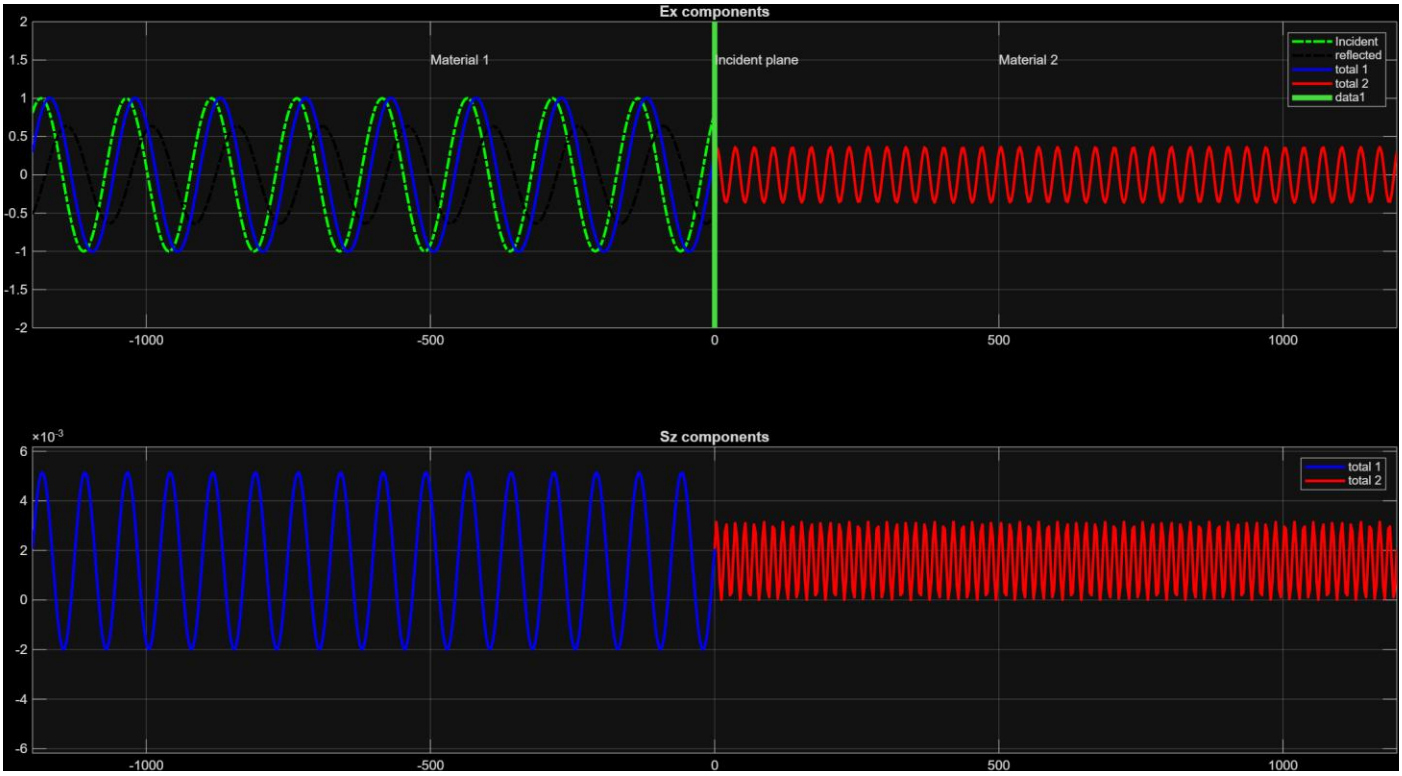


Task 3.2 a.

a.

```
function TEM_Reflection( )
%% Function: Simulate the waves with a TEM wave incident on a boundary between two
materials
%           Assuming a TEM wave polarized in the x direction
%  Parameters: f: frequency in Hz
%              Ex10_incid: amplitude of the incident wave
%              mu_r_1, epsilon_r_1, mu_r_2, epsilon_r_2: relative permittivity and
permeability of the materials; assumed real-valued or infinity

%% Set parameters
f = 1*10^6;
Ex10_incid = 1;

% Region 1
mu_r_1 = 1;
epsilon_r_1 = 4;

% Region 2 (set eta_2=0 for perfect conductors)
mu_r_2 = 1;
epsilon_r_2 = 81;

%% Find wave parameters
c = 3*10^8;
lambda_0 = c/f;
[omega,beta_1,lambda_1,T0,eta_1] = losslesspropagation(f,mu_r_1,epsilon_r_1);
[omega,beta_2,lambda_2,T0,eta_2] = losslesspropagation(f,mu_r_2,epsilon_r_2);

%% Find reflection and transmission coefficients
Gamma = (eta_2-eta_1)/(eta_2+eta_1);
tau = Gamma + 1;

%% Calculate and visualize the Ex components of the waves in the two regions
dz = lambda_0/100;
z_1 = [-4*lambda_0:dz:0]';
z_2 = [0:dz:4*lambda_0]';
T_sim = [0:T0/100:T0/10];

for n_sim = 1:1:length(T_sim)
    t = T_sim(n_sim);
    % Total wave in Region 1
    E_x1 = Ex10_incid*cos(omega*t-
beta_1*z_1)+abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma));
    % Total wave in Region 2
    E_x2 = abs(tau)*Ex10_incid*cos(omega*t-beta_2*z_2+angle(tau));

    % Plot Ex field components
    subplot(2,1,1)
    plot(z_1,Ex10_incid*cos(omega*t-beta_1*z_1), 'g-.', 'linewidth', 2 );  % Incident
wave, Region 1
```

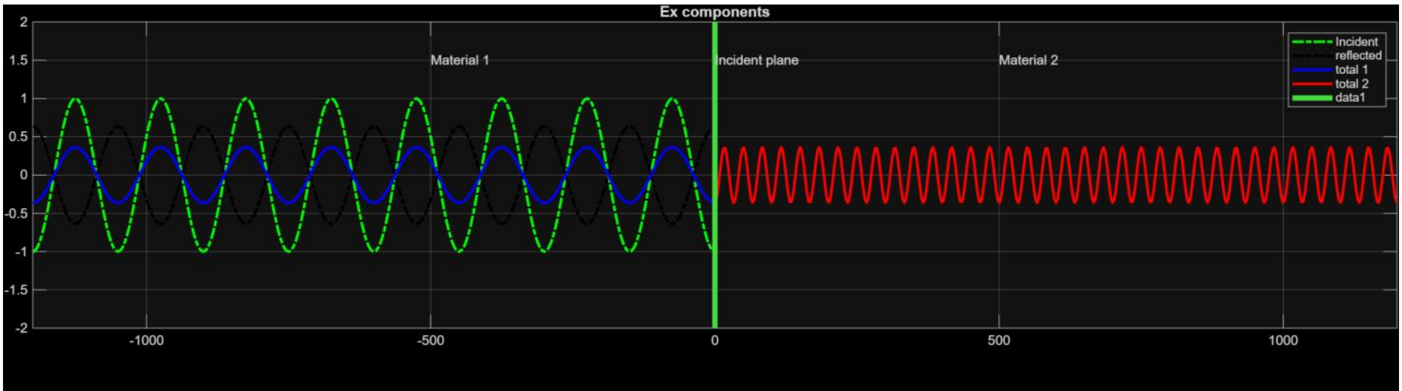
```
    hold on
    plot(z_1,abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma)), 'k-.',
'linewidth', 2 );   % Reflected wave, Region 1
    plot(z_1,E_x1,'b', 'linewidth', 2 );    % Total wave, Region 1
    plot(z_2,E_x2, 'r', 'linewidth', 2 ); grid on; axis([min(z_1), max(z_2) -2 2]);  %
Total wave, Region 2
    legend('Incident', 'reflected', 'total 1', 'total 2')

    title('Ex components')
    plot([0 0], [-2 2], 'linewidth', 4)
    text(-500,1.5,'Material 1');
    text(500,1.5,'Material 2');
    text(0,1.5,'Incident plane');
    hold off

    % Calculate and plot z-direction component of the Poynting vectors
    % First find all the E and H field components
    H_y1 = Ex10_incid/eta_1*cos(omega*t-beta_1*z_1)-
abs(Gamma)*Ex10_incid/eta_1*cos(omega*t+beta_1*z_1+angle(Gamma));
    E_y1 = zeros(size(z_1));     E_z1 = zeros(size(z_1));
    H_x1 = zeros(size(z_1));     H_z1 = zeros(size(z_1));

    H_y2 = Ex10_incid*abs(tau)/eta_2*cos(omega*t-beta_2*z_2+angle(tau));
    E_y2 = zeros(size(z_2));     E_z2 = zeros(size(z_2));
    H_x2 = zeros(size(z_2));     H_z2 = zeros(size(z_2));

    % Compute the Poynting vector according to definition
    S_z1=zeros(size(z_1)); S_z2=zeros(size(z_2));
    subplot(2,1,2)
    for m=1:1:length(z_1)
        s = cross([E_x1(m) E_y1(m) E_z1(m)], [H_x1(m) H_y1(m) H_z1(m)]);
        S_z1(m) = s(3);
    end
    for m=1:1:length(z_2)
        s = cross([E_x2(m) E_y2(m) E_z2(m)], [H_x2(m) H_y2(m) H_z2(m)]);
        S_z2(m) = s(3);
    end
    plot(z_1,S_z1, 'b', 'linewidth', 2 ); grid on;
    hold on
    plot(z_2,S_z2, 'r', 'linewidth', 2 );
    title('Sz components')
    legend('total 1', 'total 2')
    axis([min(z_1), max(z_2) -1.2*max(max(S_z1,S_z2))  1.2*max(max(S_z1,S_z2))]) ;
    hold off

    pause(0.1)
end
```


Ex components
Material 1
Incident plane
Material 2
Incident
reflected
total 1
total 2
data1
Sz components
×10⁻³
total 1
total 2

b.

```
function TEM_Reflection( )
%% Function: Simulate the waves with a TEM wave incident on a boundary between two
materials
%            Assuming a TEM wave polarized in the x direction
%  Parameters: f: frequency in Hz
%              Ex10_incid: amplitude of the incident wave
%              mu_r_1, epsilon_r_1, mu_r_2, epsilon_r_2: relative permittivity and
permeability of the materials; assumed real-valued or infinity

%% Set parameters
f = 1*10^6;
Ex10_incid = 1;

% Region 1
mu_r_1 = 1;
epsilon_r_1 = 4;

% Region 2 (set eta_2=0 for perfect conductors)
mu_r_2 = 1;
epsilon_r_2 = 81;

%% Find wave parameters
c = 3*10^8;
lambda_0 = c/f;
[omega,beta_1,lambda_1,T0,eta_1] = losslesspropagation(f,mu_r_1,epsilon_r_1);
[omega,beta_2,lambda_2,T0,eta_2] = losslesspropagation(f,mu_r_2,epsilon_r_2);

%% Find reflection and transmission coefficients
Gamma = (eta_2-eta_1)/(eta_2+eta_1);
tau = Gamma + 1;

%% Calculate and visualize the Ex components of the waves in the two regions
dz = lambda_0/100;
z_1 = [-4*lambda_0:dz:0]';
z_2 = [0:dz:4*lambda_0]';
T_sim = [0:T0/100:T0/2];

for n_sim = 1:1:length(T_sim)
    t = T_sim(n_sim);
    % Total wave in Region 1
    E_x1 = Ex10_incid*cos(omega*t-
beta_1*z_1)+abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma));
    % Total wave in Region 2
    E_x2 = abs(tau)*Ex10_incid*cos(omega*t-beta_2*z_2+angle(tau));

    % Plot Ex field components
    subplot(2,1,1)
    plot(z_1,Ex10_incid*cos(omega*t-beta_1*z_1), 'g-.', 'linewidth', 2 );  % Incident
wave, Region 1
    hold on
```

```
    plot(z_1,abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma)), 'k-.',
'linewidth', 2 );   % Reflected wave, Region 1
    plot(z_1,E_x1,'b', 'linewidth', 2 );    % Total wave, Region 1
    plot(z_2,E_x2, 'r', 'linewidth', 2 ); grid on; axis([min(z_1), max(z_2) -2 2]);  %
Total wave, Region 2
    legend('Incident', 'reflected', 'total 1', 'total 2')

    title('Ex components')
    plot([0 0], [-2 2], 'linewidth', 4)
    text(-500,1.5,'Material 1');
    text(500,1.5,'Material 2');
    text(0,1.5,'Incident plane');
    hold off

    % Calculate and plot z-direction component of the Poynting vectors
    % First find all the E and H field components
    H_y1 = Ex10_incid/eta_1*cos(omega*t-beta_1*z_1)-
abs(Gamma)*Ex10_incid/eta_1*cos(omega*t+beta_1*z_1+angle(Gamma));
    E_y1 = zeros(size(z_1));     E_z1 = zeros(size(z_1));
    H_x1 = zeros(size(z_1));     H_z1 = zeros(size(z_1));

    H_y2 = Ex10_incid*abs(tau)/eta_2*cos(omega*t-beta_2*z_2+angle(tau));
    E_y2 = zeros(size(z_2));     E_z2 = zeros(size(z_2));
    H_x2 = zeros(size(z_2));     H_z2 = zeros(size(z_2));

    % Compute the Poynting vector according to definition
    S_z1=zeros(size(z_1)); S_z2=zeros(size(z_2));
    subplot(2,1,2)
    for m=1:1:length(z_1)
        s = cross([E_x1(m) E_y1(m) E_z1(m)], [H_x1(m) H_y1(m) H_z1(m)]);
        S_z1(m) = s(3);
    end
    for m=1:1:length(z_2)
        s = cross([E_x2(m) E_y2(m) E_z2(m)], [H_x2(m) H_y2(m) H_z2(m)]);
        S_z2(m) = s(3);
    end
    plot(z_1,S_z1, 'b', 'linewidth', 2 ); grid on;
    hold on
    plot(z_2,S_z2, 'r', 'linewidth', 2 );
    title('Sz components')
    legend('total 1', 'total 2')
    axis([min(z_1), max(z_2) -1.2*max(max(S_z1,S_z2))  1.2*max(max(S_z1,S_z2))]) ;
    hold off

    pause(0.1)
end
```



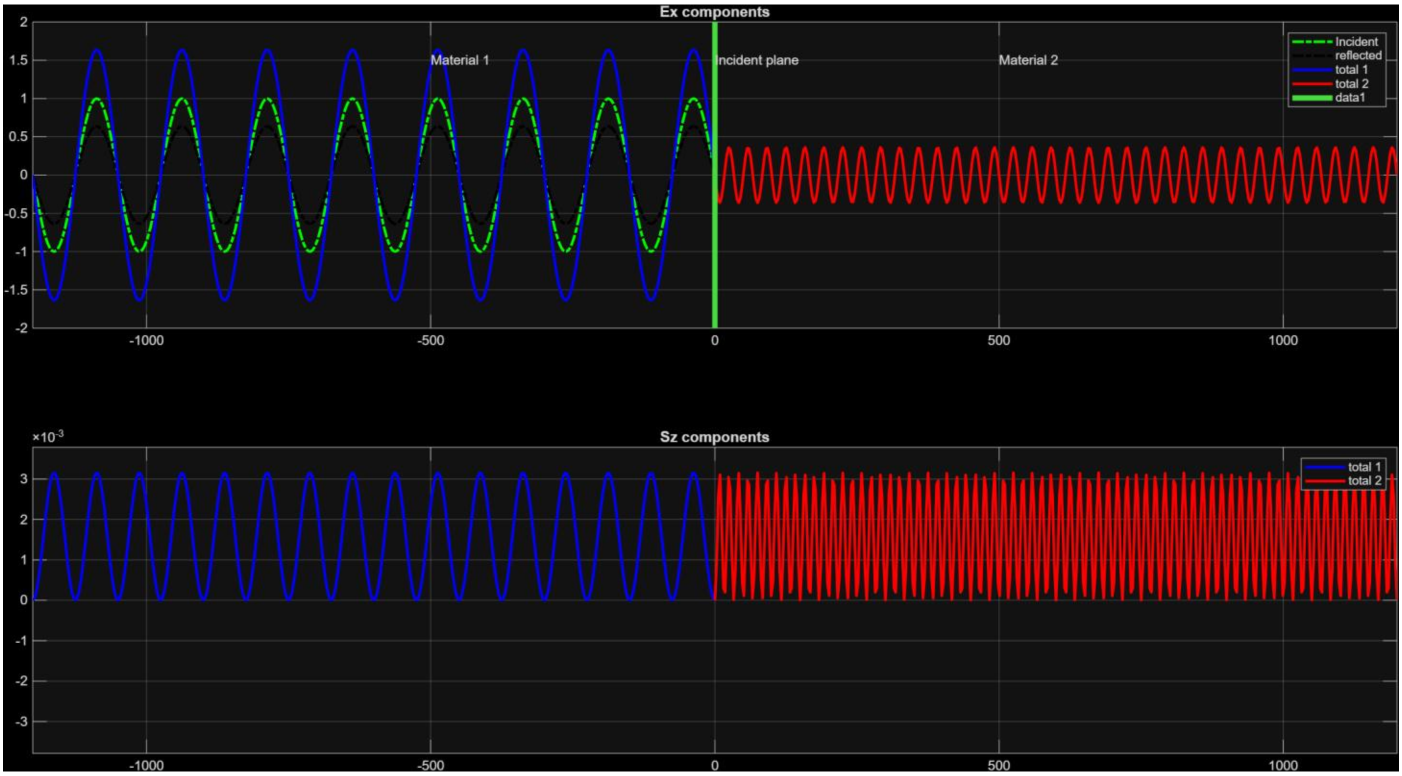
Ex components
Incident
reflected
total 1
total 2
data1
Material 1
Incident plane
Material 2
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1000
-500
0
500
1000

Sz components
×10-3
total 1
total 2
3
2
1
0
-1
-2
-3
-1000
-500
0
500
1000

c.

```
function TEM_Reflection( )
%% Function: Simulate the waves with a TEM wave incident on a boundary between two
materials
%             Assuming a TEM wave polarized in the x direction
%  Parameters: f: frequency in Hz
%               Ex10_incid: amplitude of the incident wave
%               mu_r_1, epsilon_r_1, mu_r_2, epsilon_r_2: relative permittivity and
permeability of the materials; assumed real-valued or infinity

%% Set parameters
f = 1*10^6;
Ex10_incid = 1;

% Region 1
mu_r_1 = 1;
epsilon_r_1 = 4;

% Region 2 (set eta_2=0 for perfect conductors)
mu_r_2 = 1;
epsilon_r_2 = 81;

%% Find wave parameters
c = 3*10^8;
lambda_0 = c/f;
[omega,beta_1,lambda_1,T0,eta_1] = losslesspropagation(f,mu_r_1,epsilon_r_1);
[omega,beta_2,lambda_2,T0,eta_2] = losslesspropagation(f,mu_r_2,epsilon_r_2);

%% Find reflection and transmission coefficients
Gamma = (eta_2-eta_1)/(eta_2+eta_1);
tau = Gamma + 1;

%% Calculate and visualize the Ex components of the waves in the two regions
dz = lambda_0/100;
z_1 = [-4*lambda_0:dz:0]';
z_2 = [0:dz:4*lambda_0]';
T_sim = [0:T0/100:3*T0/4];

for n_sim = 1:1:length(T_sim)
    t = T_sim(n_sim);
    % Total wave in Region 1
    E_x1 = Ex10_incid*cos(omega*t-
beta_1*z_1)+abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma));
    % Total wave in Region 2
    E_x2 = abs(tau)*Ex10_incid*cos(omega*t-beta_2*z_2+angle(tau));

    % Plot Ex field components
    subplot(2,1,1)
    plot(z_1,Ex10_incid*cos(omega*t-beta_1*z_1), 'g-.', 'linewidth', 2 );  % Incident
wave, Region 1
    hold on
```

```
    plot(z_1,abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma)), 'k-.',
'linewidth', 2 );   % Reflected wave, Region 1
    plot(z_1,E_x1,'b', 'linewidth', 2 );    % Total wave, Region 1
    plot(z_2,E_x2, 'r', 'linewidth', 2 ); grid on; axis([min(z_1), max(z_2) -2 2]);  %
Total wave, Region 2
    legend('Incident', 'reflected', 'total 1', 'total 2')

    title('Ex components')
    plot([0 0], [-2 2], 'linewidth', 4)
    text(-500,1.5,'Material 1');
    text(500,1.5,'Material 2');
    text(0,1.5,'Incident plane');
    hold off

    % Calculate and plot z-direction component of the Poynting vectors
    % First find all the E and H field components
    H_y1 = Ex10_incid/eta_1*cos(omega*t-beta_1*z_1)-
abs(Gamma)*Ex10_incid/eta_1*cos(omega*t+beta_1*z_1+angle(Gamma));
    E_y1 = zeros(size(z_1));     E_z1 = zeros(size(z_1));
    H_x1 = zeros(size(z_1));     H_z1 = zeros(size(z_1));

    H_y2 = Ex10_incid*abs(tau)/eta_2*cos(omega*t-beta_2*z_2+angle(tau));
    E_y2 = zeros(size(z_2));     E_z2 = zeros(size(z_2));
    H_x2 = zeros(size(z_2));     H_z2 = zeros(size(z_2));

    % Compute the Poynting vector according to definition
    S_z1=zeros(size(z_1)); S_z2=zeros(size(z_2));
    subplot(2,1,2)
    for m=1:1:length(z_1)
        s = cross([E_x1(m) E_y1(m) E_z1(m)], [H_x1(m) H_y1(m) H_z1(m)]);
        S_z1(m) = s(3);
    end
    for m=1:1:length(z_2)
        s = cross([E_x2(m) E_y2(m) E_z2(m)], [H_x2(m) H_y2(m) H_z2(m)]);
        S_z2(m) = s(3);
    end
    plot(z_1,S_z1, 'b', 'linewidth', 2 ); grid on;
    hold on
    plot(z_2,S_z2, 'r', 'linewidth', 2 );
    title('Sz components')
    legend('total 1', 'total 2')
    axis([min(z_1), max(z_2) -1.2*max(max(S_z1,S_z2))  1.2*max(max(S_z1,S_z2))]) ;
    hold off

    pause(0.1)
end
```


Ex components
Material 1
Incident plane
Material 2
Incident
reflected
total 1
total 2
data1
Sz components
total 1
total 2

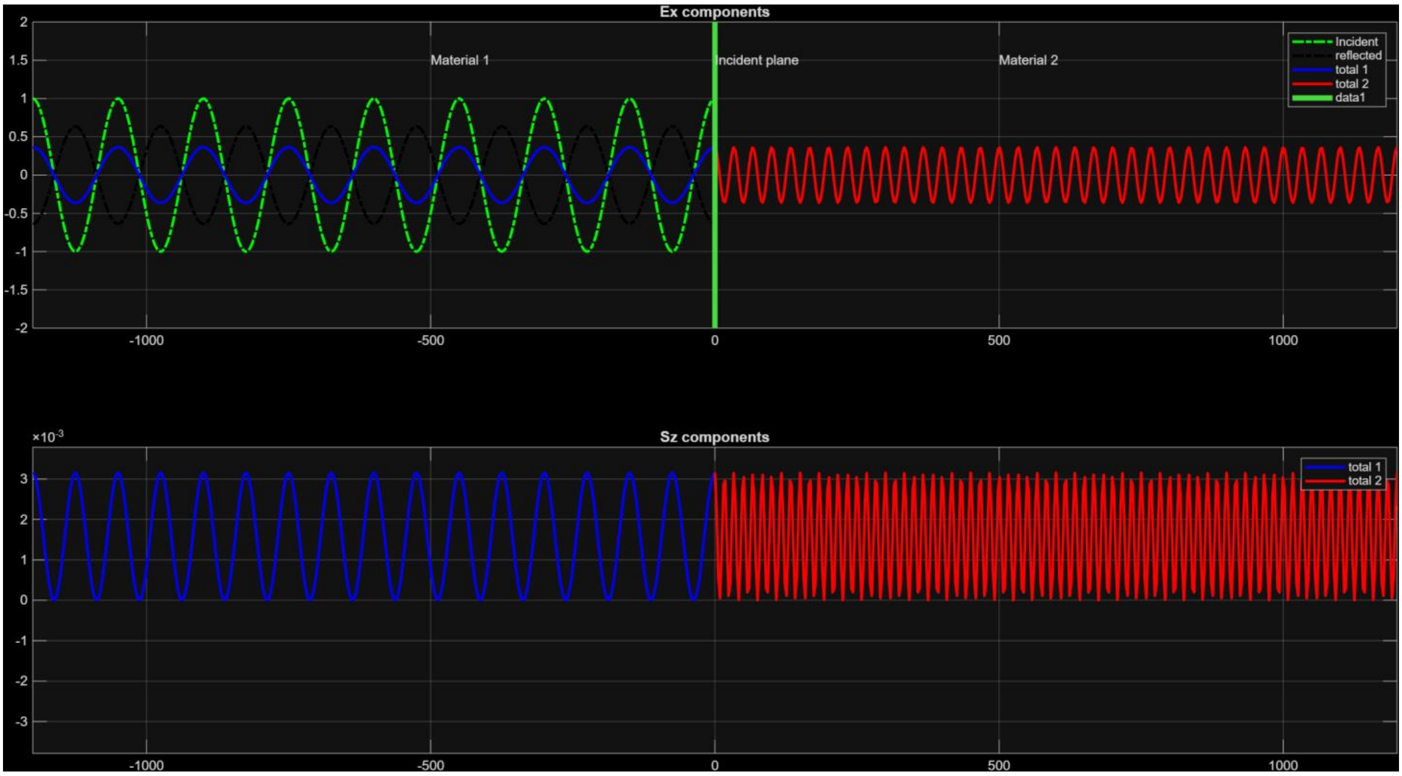
Task 3.2 b.

At t = 0

```
function TEM_Reflection( )
%% Function: Simulate the waves with a TEM wave incident on a boundary between two
materials
%            Assuming a TEM wave polarized in the x direction
%  Parameters: f: frequency in Hz
%              Ex10_incid: amplitude of the incident wave
%              mu_r_1, epsilon_r_1, mu_r_2, epsilon_r_2: relative permittivity and
permeability of the materials; assumed real-valued or infinity

%% Set parameters
f = 1*10^6;
Ex10_incid = 1;

% Region 1
mu_r_1 = 1;
epsilon_r_1 = 4;

% Region 2 (set eta_2=0 for perfect conductors)
mu_r_2 = 1;
epsilon_r_2 = 81;

%% Find wave parameters
c = 3*10^8;
lambda_0 = c/f;
[omega,beta_1,lambda_1,T0,eta_1] = losslesspropagation(f,mu_r_1,epsilon_r_1);
[omega,beta_2,lambda_2,T0,eta_2] = losslesspropagation(f,mu_r_2,epsilon_r_2);

%% Find reflection and transmission coefficients
Gamma = (eta_2-eta_1)/(eta_2+eta_1);
tau = Gamma + 1;

%% Calculate and visualize the Ex components of the waves in the two regions
dz = lambda_0/100;
z_1 = [-4*lambda_0:dz:0]';
z_2 = [0:dz:4*lambda_0]';
T_sim = 0;

for n_sim = 1:1:length(T_sim)
    t = T_sim(n_sim);
    % Total wave in Region 1
    E_x1 = Ex10_incid*cos(omega*t-
beta_1*z_1)+abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma));
    % Total wave in Region 2
    E_x2 = abs(tau)*Ex10_incid*cos(omega*t-beta_2*z_2+angle(tau));

    % After calculating E_x1 and E_x2, add:
    max_E_x1 = max(abs(E_x1));
    max_E_x2 = max(abs(E_x2));
    max_E_total = max(max_E_x1, max_E_x2);
```

```
    fprintf('Maximum electric field magnitude: %f V/m\n', max_E_total);

    % Plot Ex field components
    subplot(2,1,1)
    plot(z_1,Ex10_incid*cos(omega*t-beta_1*z_1), 'g-.', 'linewidth', 2 );  % Incident
wave, Region 1
    hold on
    plot(z_1,abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma)), 'k-.',
'linewidth', 2 );   % Reflected wave, Region 1
    plot(z_1,E_x1,'b', 'linewidth', 2 );    % Total wave, Region 1
    plot(z_2,E_x2, 'r', 'linewidth', 2 ); grid on; axis([min(z_1), max(z_2) -2 2]);  %
Total wave, Region 2
    legend('Incident', 'reflected', 'total 1', 'total 2')

    title('Ex components')
    plot([0 0], [-2 2], 'linewidth', 4)
    text(-500,1.5,'Material 1');
    text(500,1.5,'Material 2');
    text(0,1.5,'Incident plane');
    hold off

    % Calculate and plot z-direction component of the Poynting vectors
    % First find all the E and H field components
    H_y1 = Ex10_incid/eta_1*cos(omega*t-beta_1*z_1)-
abs(Gamma)*Ex10_incid/eta_1*cos(omega*t+beta_1*z_1+angle(Gamma));
    E_y1 = zeros(size(z_1));     E_z1 = zeros(size(z_1));
    H_x1 = zeros(size(z_1));     H_z1 = zeros(size(z_1));

    H_y2 = Ex10_incid*abs(tau)/eta_2*cos(omega*t-beta_2*z_2+angle(tau));
    E_y2 = zeros(size(z_2));     E_z2 = zeros(size(z_2));
    H_x2 = zeros(size(z_2));     H_z2 = zeros(size(z_2));

    % Compute the Poynting vector according to definition
    S_z1=zeros(size(z_1)); S_z2=zeros(size(z_2));
    subplot(2,1,2)
    for m=1:1:length(z_1)
        s = cross([E_x1(m) E_y1(m) E_z1(m)], [H_x1(m) H_y1(m) H_z1(m)]);
        S_z1(m) = s(3);
    end
    for m=1:1:length(z_2)
        s = cross([E_x2(m) E_y2(m) E_z2(m)], [H_x2(m) H_y2(m) H_z2(m)]);
        S_z2(m) = s(3);
    end
    plot(z_1,S_z1, 'b', 'linewidth', 2 ); grid on;
    hold on
    plot(z_2,S_z2, 'r', 'linewidth', 2 );
    title('Sz components')
    legend('total 1', 'total 2')
    axis([min(z_1), max(z_2) -1.2*max(max(S_z1,S_z2))  1.2*max(max(S_z1,S_z2))]) ;
    hold off

    pause(0.1)
end
```

Maximum electric field magnitude: 0.363636 V/m



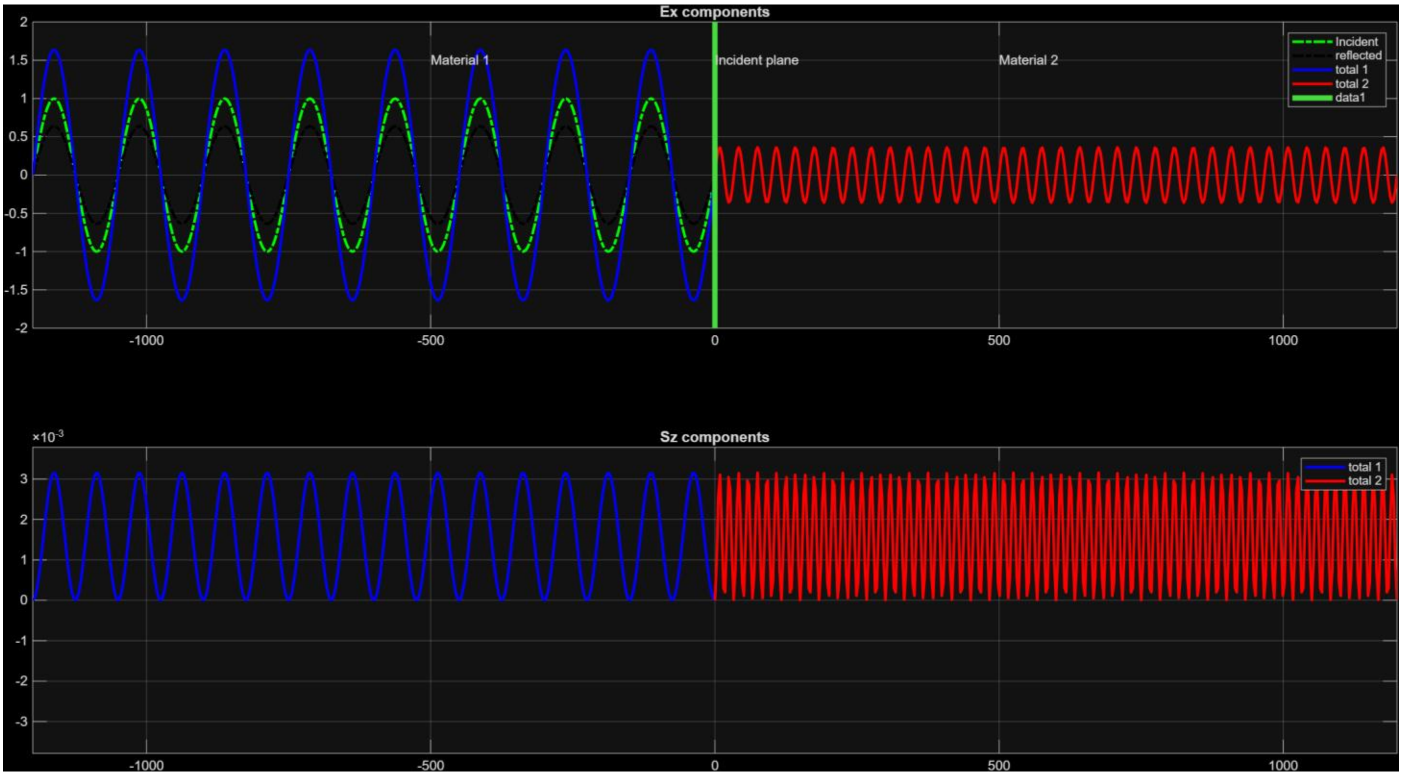
At t = T0/4

```
function TEM_Reflection( )
%% Function: Simulate the waves with a TEM wave incident on a boundary between two
materials
%           Assuming a TEM wave polarized in the x direction
%  Parameters: f: frequency in Hz
%              Ex10_incid: amplitude of the incident wave
%              mu_r_1, epsilon_r_1, mu_r_2, epsilon_r_2: relative permittivity and
permeability of the materials; assumed real-valued or infinity

%% Set parameters
f = 1*10^6;
Ex10_incid = 1;

% Region 1
mu_r_1 = 1;
epsilon_r_1 = 4;

% Region 2 (set eta_2=0 for perfect conductors)
mu_r_2 = 1;
epsilon_r_2 = 81;

%% Find wave parameters
c = 3*10^8;
lambda_0 = c/f;
[omega,beta_1,lambda_1,T0,eta_1] = losslesspropagation(f,mu_r_1,epsilon_r_1);
[omega,beta_2,lambda_2,T0,eta_2] = losslesspropagation(f,mu_r_2,epsilon_r_2);
```

```
%% Find reflection and transmission coefficients
Gamma = (eta_2-eta_1)/(eta_2+eta_1);
tau = Gamma + 1;

%% Calculate and visualize the Ex components of the waves in the two regions
dz = lambda_0/100;
z_1 = [-4*lambda_0:dz:0]';
z_2 = [0:dz:4*lambda_0]';
T_sim = T0/4;

for n_sim = 1:1:length(T_sim)
    t = T_sim(n_sim);
    % Total wave in Region 1
    E_x1 = Ex10_incid*cos(omega*t-
beta_1*z_1)+abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma));
    % Total wave in Region 2
    E_x2 = abs(tau)*Ex10_incid*cos(omega*t-beta_2*z_2+angle(tau));

    % After calculating E_x1 and E_x2, add:
    max_E_x1 = max(abs(E_x1));
    max_E_x2 = max(abs(E_x2));
    max_E_total = max(max_E_x1, max_E_x2);
    fprintf('Maximum electric field magnitude: %f V/m\n', max_E_total);

    % Plot Ex field components
    subplot(2,1,1)
    plot(z_1,Ex10_incid*cos(omega*t-beta_1*z_1), 'g-.', 'linewidth', 2 );  % Incident
wave, Region 1
    hold on
    plot(z_1,abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma)), 'k-.',
'linewidth', 2 );   % Reflected wave, Region 1
    plot(z_1,E_x1,'b', 'linewidth', 2 );    % Total wave, Region 1
    plot(z_2,E_x2, 'r', 'linewidth', 2 ); grid on; axis([min(z_1), max(z_2) -2 2]);  %
Total wave, Region 2
    legend('Incident', 'reflected', 'total 1', 'total 2')

    title('Ex components')
    plot([0 0], [-2 2], 'linewidth', 4)
    text(-500,1.5,'Material 1');
    text(500,1.5,'Material 2');
    text(0,1.5,'Incident plane');
    hold off

    % Calculate and plot z-direction component of the Poynting vectors
    % First find all the E and H field components
    H_y1 = Ex10_incid/eta_1*cos(omega*t-beta_1*z_1)-
abs(Gamma)*Ex10_incid/eta_1*cos(omega*t+beta_1*z_1+angle(Gamma));
    E_y1 = zeros(size(z_1));     E_z1 = zeros(size(z_1));
    H_x1 = zeros(size(z_1));     H_z1 = zeros(size(z_1));

    H_y2 = Ex10_incid*abs(tau)/eta_2*cos(omega*t-beta_2*z_2+angle(tau));
    E_y2 = zeros(size(z_2));     E_z2 = zeros(size(z_2));
```

```
    H_x2 = zeros(size(z_2));      H_z2 = zeros(size(z_2));

    % Compute the Poynting vector according to definition
    S_z1=zeros(size(z_1)); S_z2=zeros(size(z_2));
    subplot(2,1,2)
    for m=1:1:length(z_1)
        s = cross([E_x1(m) E_y1(m) E_z1(m)], [H_x1(m) H_y1(m) H_z1(m)]);
        S_z1(m) = s(3);
    end
    for m=1:1:length(z_2)
        s = cross([E_x2(m) E_y2(m) E_z2(m)], [H_x2(m) H_y2(m) H_z2(m)]);
        S_z2(m) = s(3);
    end
    plot(z_1,S_z1, 'b', 'linewidth', 2 ); grid on;
    hold on
    plot(z_2,S_z2, 'r', 'linewidth', 2 );
    title('Sz components')
    legend('total 1', 'total 2')
    axis([min(z_1), max(z_2) -1.2*max(max(S_z1,S_z2))  1.2*max(max(S_z1,S_z2))]) ;
    hold off

    pause(0.1)
end
```

Maximum electric field magnitude: 1.633135 V/m



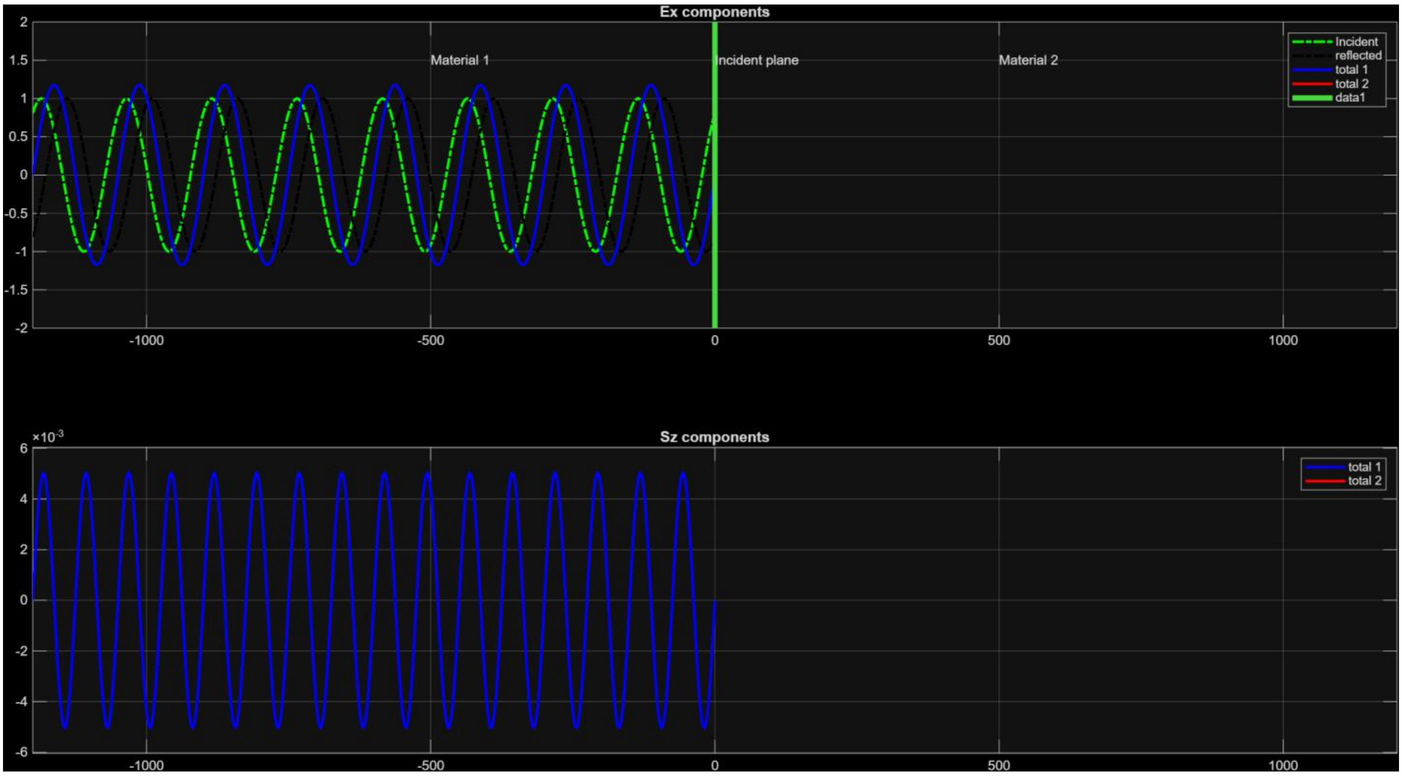
$$\text{Ratio} = \frac{1.633135}{0.363636} = 4.4911$$

Task 3.2 c - a

a.

```
function TEM_Reflection( )
%% Function: Simulate the waves with a TEM wave incident on a boundary between two
materials
%            Assuming a TEM wave polarized in the x direction
%  Parameters: f: frequency in Hz
%              Ex10_incid: amplitude of the incident wave
%              mu_r_1, epsilon_r_1, mu_r_2, epsilon_r_2: relative permittivity and
permeability of the materials; assumed real-valued or infinity

%% Set parameters
f = 1*10^6;
Ex10_incid = 1;

% Region 1
mu_r_1 = 1;
epsilon_r_1 = 4;

% Region 2 (set eta_2=0 for perfect conductors)
mu_r_2 = 1;
epsilon_r_2 = 81-(j*inf);

%% Find wave parameters
c = 3*10^8;
lambda_0 = c/f;
[omega,beta_1,lambda_1,T0,eta_1] = losslesspropagation(f,mu_r_1,epsilon_r_1);
[omega,beta_2,lambda_2,T0,eta_2] = losslesspropagation(f,mu_r_2,epsilon_r_2);

%% Find reflection and transmission coefficients
Gamma = (eta_2-eta_1)/(eta_2+eta_1);
tau = Gamma + 1;

%% Calculate and visualize the Ex components of the waves in the two regions
dz = lambda_0/100;
z_1 = [-4*lambda_0:dz:0]';
z_2 = [0:dz:4*lambda_0]';
T_sim = [0:T0/100:T0/10];

for n_sim = 1:1:length(T_sim)
    t = T_sim(n_sim);
    % Total wave in Region 1
    E_x1 = Ex10_incid*cos(omega*t-
beta_1*z_1)+abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma));
    % Total wave in Region 2
    E_x2 = abs(tau)*Ex10_incid*cos(omega*t-beta_2*z_2+angle(tau));

    % Plot Ex field components
    subplot(2,1,1)
    plot(z_1,Ex10_incid*cos(omega*t-beta_1*z_1), 'g-.', 'linewidth', 2 );  % Incident
wave, Region 1
```

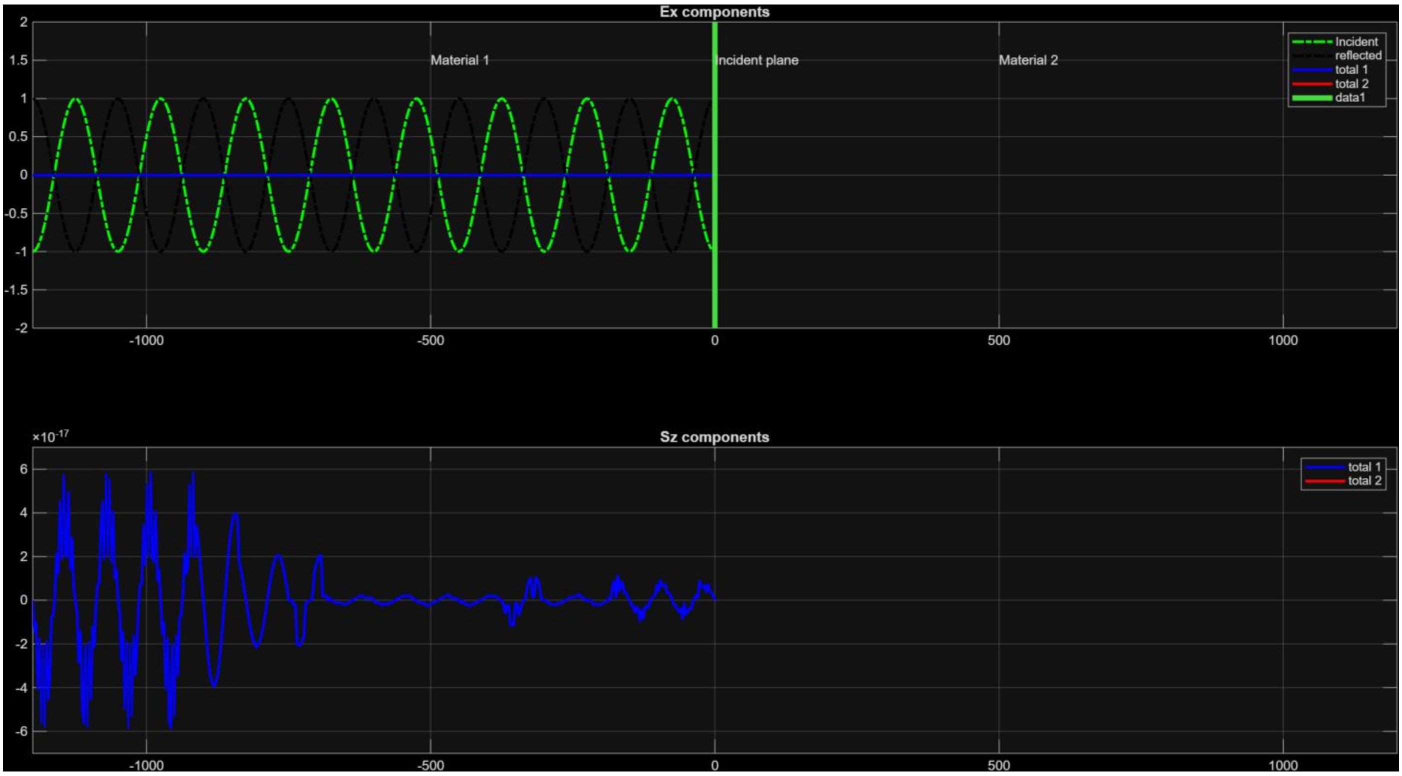
```
    hold on
    plot(z_1,abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma)), 'k-.',
'linewidth', 2 );   % Reflected wave, Region 1
    plot(z_1,E_x1,'b', 'linewidth', 2 );    % Total wave, Region 1
    plot(z_2,E_x2, 'r', 'linewidth', 2 ); grid on; axis([min(z_1), max(z_2) -2 2]);  %
Total wave, Region 2
    legend('Incident', 'reflected', 'total 1', 'total 2')

    title('Ex components')
    plot([0 0], [-2 2], 'linewidth', 4)
    text(-500,1.5,'Material 1');
    text(500,1.5,'Material 2');
    text(0,1.5,'Incident plane');
    hold off

    % Calculate and plot z-direction component of the Poynting vectors
    % First find all the E and H field components
    H_y1 = Ex10_incid/eta_1*cos(omega*t-beta_1*z_1)-
abs(Gamma)*Ex10_incid/eta_1*cos(omega*t+beta_1*z_1+angle(Gamma));
    E_y1 = zeros(size(z_1));     E_z1 = zeros(size(z_1));
    H_x1 = zeros(size(z_1));     H_z1 = zeros(size(z_1));

    H_y2 = Ex10_incid*abs(tau)/eta_2*cos(omega*t-beta_2*z_2+angle(tau));
    E_y2 = zeros(size(z_2));     E_z2 = zeros(size(z_2));
    H_x2 = zeros(size(z_2));     H_z2 = zeros(size(z_2));

    % Compute the Poynting vector according to definition
    S_z1=zeros(size(z_1)); S_z2=zeros(size(z_2));
    subplot(2,1,2)
    for m=1:1:length(z_1)
        s = cross([E_x1(m) E_y1(m) E_z1(m)], [H_x1(m) H_y1(m) H_z1(m)]);
        S_z1(m) = s(3);
    end
    for m=1:1:length(z_2)
        s = cross([E_x2(m) E_y2(m) E_z2(m)], [H_x2(m) H_y2(m) H_z2(m)]);
        S_z2(m) = s(3);
    end
    plot(z_1,S_z1, 'b', 'linewidth', 2 ); grid on;
    hold on
    plot(z_2,S_z2, 'r', 'linewidth', 2 );
    title('Sz components')
    legend('total 1', 'total 2')
    axis([min(z_1), max(z_2) -1.2*max(max(S_z1,S_z2))  1.2*max(max(S_z1,S_z2))]) ;
    hold off

    pause(0.1)
end
```


Ex components
Incident
reflected
total 1
total 2
data1
Material 1
Incident plane
Material 2
Sz components
×10^-3
total 1
total 2

b.

```
function TEM_Reflection( )
%% Function: Simulate the waves with a TEM wave incident on a boundary between two materials
%           Assuming a TEM wave polarized in the x direction
%  Parameters: f: frequency in Hz
%              Ex10_incid: amplitude of the incident wave
%              mu_r_1, epsilon_r_1, mu_r_2, epsilon_r_2: relative permittivity and permeability of the materials; assumed real-valued or infinity

%% Set parameters
f = 1*10^6;
Ex10_incid = 1;

% Region 1
mu_r_1 = 1;
epsilon_r_1 = 4;

% Region 2 (set eta_2=0 for perfect conductors)
mu_r_2 = 1;
epsilon_r_2 = 81-(j*inf);

%% Find wave parameters
c = 3*10^8;
lambda_0 = c/f;
[omega,beta_1,lambda_1,T0,eta_1] = losslesspropagation(f,mu_r_1,epsilon_r_1);
[omega,beta_2,lambda_2,T0,eta_2] = losslesspropagation(f,mu_r_2,epsilon_r_2);

%% Find reflection and transmission coefficients
Gamma = (eta_2-eta_1)/(eta_2+eta_1);
tau = Gamma + 1;

%% Calculate and visualize the Ex components of the waves in the two regions
dz = lambda_0/100;
z_1 = [-4*lambda_0:dz:0]';
z_2 = [0:dz:4*lambda_0]';
T_sim = [0:T0/100:T0/2];

for n_sim = 1:1:length(T_sim)
    t = T_sim(n_sim);
    % Total wave in Region 1
    E_x1 = Ex10_incid*cos(omega*t-beta_1*z_1)+abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma));
    % Total wave in Region 2
    E_x2 = abs(tau)*Ex10_incid*cos(omega*t-beta_2*z_2+angle(tau));

    % Plot Ex field components
    subplot(2,1,1)
    plot(z_1,Ex10_incid*cos(omega*t-beta_1*z_1), 'g-.', 'linewidth', 2 );  % Incident wave, Region 1
    hold on
```

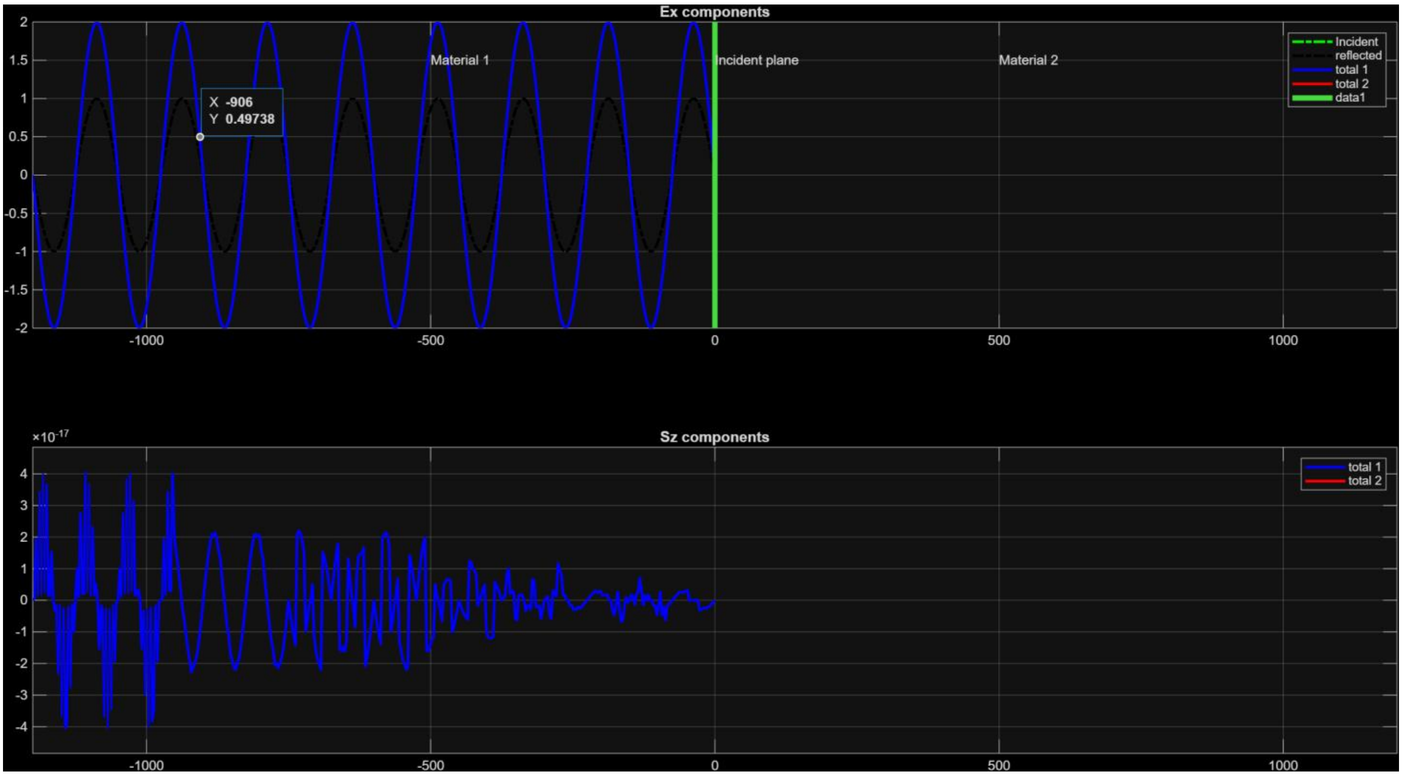
```
    plot(z_1,abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma)), 'k-.',
'linewidth', 2 );   % Reflected wave, Region 1
    plot(z_1,E_x1,'b', 'linewidth', 2 );    % Total wave, Region 1
    plot(z_2,E_x2, 'r', 'linewidth', 2 ); grid on; axis([min(z_1), max(z_2) -2 2]);  %
Total wave, Region 2
    legend('Incident', 'reflected', 'total 1', 'total 2')

    title('Ex components')
    plot([0 0], [-2 2], 'linewidth', 4)
    text(-500,1.5,'Material 1');
    text(500,1.5,'Material 2');
    text(0,1.5,'Incident plane');
    hold off

    % Calculate and plot z-direction component of the Poynting vectors
    % First find all the E and H field components
    H_y1 = Ex10_incid/eta_1*cos(omega*t-beta_1*z_1)-
abs(Gamma)*Ex10_incid/eta_1*cos(omega*t+beta_1*z_1+angle(Gamma));
    E_y1 = zeros(size(z_1));     E_z1 = zeros(size(z_1));
    H_x1 = zeros(size(z_1));     H_z1 = zeros(size(z_1));

    H_y2 = Ex10_incid*abs(tau)/eta_2*cos(omega*t-beta_2*z_2+angle(tau));
    E_y2 = zeros(size(z_2));     E_z2 = zeros(size(z_2));
    H_x2 = zeros(size(z_2));     H_z2 = zeros(size(z_2));

    % Compute the Poynting vector according to definition
    S_z1=zeros(size(z_1)); S_z2=zeros(size(z_2));
    subplot(2,1,2)
    for m=1:1:length(z_1)
        s = cross([E_x1(m) E_y1(m) E_z1(m)], [H_x1(m) H_y1(m) H_z1(m)]);
        S_z1(m) = s(3);
    end
    for m=1:1:length(z_2)
        s = cross([E_x2(m) E_y2(m) E_z2(m)], [H_x2(m) H_y2(m) H_z2(m)]);
        S_z2(m) = s(3);
    end
    plot(z_1,S_z1, 'b', 'linewidth', 2 ); grid on;
    hold on
    plot(z_2,S_z2, 'r', 'linewidth', 2 );
    title('Sz components')
    legend('total 1', 'total 2')
    axis([min(z_1), max(z_2) -1.2*max(max(S_z1,S_z2))  1.2*max(max(S_z1,S_z2))]) ;
    hold off

    pause(0.1)
end
```


Ex components
Material 1
Incident plane
Material 2
Incident
reflected
total 1
total 2
data1
Sz components
×10^-17
total 1
total 2

c.

```
function TEM_Reflection( )
%% Function: Simulate the waves with a TEM wave incident on a boundary between two materials
%           Assuming a TEM wave polarized in the x direction
%  Parameters: f: frequency in Hz
%             Ex10_incid: amplitude of the incident wave
%             mu_r_1, epsilon_r_1, mu_r_2, epsilon_r_2: relative permittivity and permeability of the materials; assumed real-valued or infinity

%% Set parameters
f = 1*10^6;
Ex10_incid = 1;

% Region 1
mu_r_1 = 1;
epsilon_r_1 = 4;

% Region 2 (set eta_2=0 for perfect conductors)
mu_r_2 = 1;
epsilon_r_2 = 81-(j*inf);

%% Find wave parameters
c = 3*10^8;
lambda_0 = c/f;
[omega,beta_1,lambda_1,T0,eta_1] = losslesspropagation(f,mu_r_1,epsilon_r_1);
[omega,beta_2,lambda_2,T0,eta_2] = losslesspropagation(f,mu_r_2,epsilon_r_2);

%% Find reflection and transmission coefficients
Gamma = (eta_2-eta_1)/(eta_2+eta_1);
tau = Gamma + 1;

%% Calculate and visualize the Ex components of the waves in the two regions
dz = lambda_0/100;
z_1 = [-4*lambda_0:dz:0]';
z_2 = [0:dz:4*lambda_0]';
T_sim = [0:T0/100:3*T0/4];

for n_sim = 1:1:length(T_sim)
    t = T_sim(n_sim);
    % Total wave in Region 1
    E_x1 = Ex10_incid*cos(omega*t-beta_1*z_1)+abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma));
    % Total wave in Region 2
    E_x2 = abs(tau)*Ex10_incid*cos(omega*t-beta_2*z_2+angle(tau));

    % Plot Ex field components
    subplot(2,1,1)
    plot(z_1,Ex10_incid*cos(omega*t-beta_1*z_1), 'g-.', 'linewidth', 2 );  % Incident wave, Region 1
    hold on
```

```
    plot(z_1,abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma)), 'k-.',
'linewidth', 2 );   % Reflected wave, Region 1
    plot(z_1,E_x1,'b', 'linewidth', 2 );    % Total wave, Region 1
    plot(z_2,E_x2, 'r', 'linewidth', 2 ); grid on; axis([min(z_1), max(z_2) -2 2]);  %
Total wave, Region 2
    legend('Incident', 'reflected', 'total 1', 'total 2')

    title('Ex components')
    plot([0 0], [-2 2], 'linewidth', 4)
    text(-500,1.5,'Material 1');
    text(500,1.5,'Material 2');
    text(0,1.5,'Incident plane');
    hold off

    % Calculate and plot z-direction component of the Poynting vectors
    % First find all the E and H field components
    H_y1 = Ex10_incid/eta_1*cos(omega*t-beta_1*z_1)-
abs(Gamma)*Ex10_incid/eta_1*cos(omega*t+beta_1*z_1+angle(Gamma));
    E_y1 = zeros(size(z_1));     E_z1 = zeros(size(z_1));
    H_x1 = zeros(size(z_1));     H_z1 = zeros(size(z_1));

    H_y2 = Ex10_incid*abs(tau)/eta_2*cos(omega*t-beta_2*z_2+angle(tau));
    E_y2 = zeros(size(z_2));     E_z2 = zeros(size(z_2));
    H_x2 = zeros(size(z_2));     H_z2 = zeros(size(z_2));

    % Compute the Poynting vector according to definition
    S_z1=zeros(size(z_1)); S_z2=zeros(size(z_2));
    subplot(2,1,2)
    for m=1:1:length(z_1)
        s = cross([E_x1(m) E_y1(m) E_z1(m)], [H_x1(m) H_y1(m) H_z1(m)]);
        S_z1(m) = s(3);
    end
    for m=1:1:length(z_2)
        s = cross([E_x2(m) E_y2(m) E_z2(m)], [H_x2(m) H_y2(m) H_z2(m)]);
        S_z2(m) = s(3);
    end
    plot(z_1,S_z1, 'b', 'linewidth', 2 ); grid on;
    hold on
    plot(z_2,S_z2, 'r', 'linewidth', 2 );
    title('Sz components')
    legend('total 1', 'total 2')
    axis([min(z_1), max(z_2) -1.2*max(max(S_z1,S_z2))  1.2*max(max(S_z1,S_z2))]) ;
    hold off

    pause(0.1)
end
```


Ex components
X -906
Y 0.49738
Material 1
Incident plane
Material 2
Incident
reflected
total 1
total 2
data1
Sz components
×10^-17
total 1
total 2

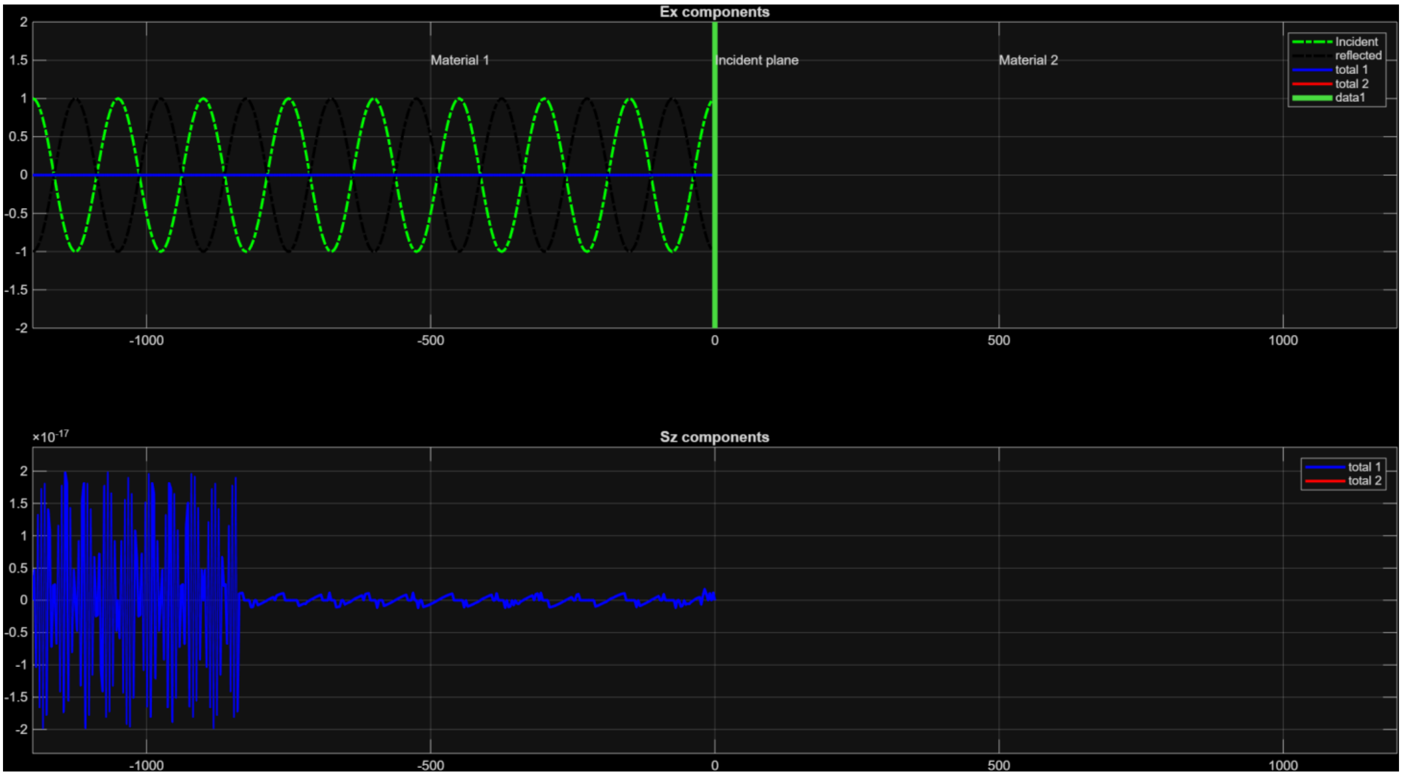
Task 3.2 c - b

At t = 0

```
function TEM_Reflection( )
%% Function: Simulate the waves with a TEM wave incident on a boundary between two
materials
%           Assuming a TEM wave polarized in the x direction
%  Parameters: f: frequency in Hz
%              Ex10_incid: amplitude of the incident wave
%              mu_r_1, epsilon_r_1, mu_r_2, epsilon_r_2: relative permittivity and
permeability of the materials; assumed real-valued or infinity

%% Set parameters
f = 1*10^6;
Ex10_incid = 1;

% Region 1
mu_r_1 = 1;
epsilon_r_1 = 4;

% Region 2 (set eta_2=0 for perfect conductors)
mu_r_2 = 1;
epsilon_r_2 = 81-(j*inf);

%% Find wave parameters
c = 3*10^8;
lambda_0 = c/f;
[omega,beta_1,lambda_1,T0,eta_1] = losslesspropagation(f,mu_r_1,epsilon_r_1);
[omega,beta_2,lambda_2,T0,eta_2] = losslesspropagation(f,mu_r_2,epsilon_r_2);

%% Find reflection and transmission coefficients
Gamma = (eta_2-eta_1)/(eta_2+eta_1);
tau = Gamma + 1;

%% Calculate and visualize the Ex components of the waves in the two regions
dz = lambda_0/100;
z_1 = [-4*lambda_0:dz:0]';
z_2 = [0:dz:4*lambda_0]';
T_sim = 0;

for n_sim = 1:1:length(T_sim)
    t = T_sim(n_sim);
    % Total wave in Region 1
    E_x1 = Ex10_incid*cos(omega*t-
beta_1*z_1)+abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma));
    % Total wave in Region 2
    E_x2 = abs(tau)*Ex10_incid*cos(omega*t-beta_2*z_2+angle(tau));

    % After calculating E_x1 and E_x2, add:
    max_E_x1 = max(abs(E_x1));
    max_E_x2 = max(abs(E_x2));
    max_E_total = max(max_E_x1, max_E_x2);
```

```
    fprintf('Maximum electric field magnitude: %f V/m\n', max_E_total);

    % Plot Ex field components
    subplot(2,1,1)
    plot(z_1,Ex10_incid*cos(omega*t-beta_1*z_1), 'g-.', 'linewidth', 2 );  % Incident
wave, Region 1
    hold on
    plot(z_1,abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma)), 'k-.',
'linewidth', 2 );   % Reflected wave, Region 1
    plot(z_1,E_x1,'b', 'linewidth', 2 );    % Total wave, Region 1
    plot(z_2,E_x2, 'r', 'linewidth', 2 ); grid on; axis([min(z_1), max(z_2) -2 2]);  %
Total wave, Region 2
    legend('Incident', 'reflected', 'total 1', 'total 2')

    title('Ex components')
    plot([0 0], [-2 2], 'linewidth', 4)
    text(-500,1.5,'Material 1');
    text(500,1.5,'Material 2');
    text(0,1.5,'Incident plane');
    hold off

    % Calculate and plot z-direction component of the Poynting vectors
    % First find all the E and H field components
    H_y1 = Ex10_incid/eta_1*cos(omega*t-beta_1*z_1)-
abs(Gamma)*Ex10_incid/eta_1*cos(omega*t+beta_1*z_1+angle(Gamma));
    E_y1 = zeros(size(z_1));     E_z1 = zeros(size(z_1));
    H_x1 = zeros(size(z_1));     H_z1 = zeros(size(z_1));

    H_y2 = Ex10_incid*abs(tau)/eta_2*cos(omega*t-beta_2*z_2+angle(tau));
    E_y2 = zeros(size(z_2));     E_z2 = zeros(size(z_2));
    H_x2 = zeros(size(z_2));     H_z2 = zeros(size(z_2));

    % Compute the Poynting vector according to definition
    S_z1=zeros(size(z_1)); S_z2=zeros(size(z_2));
    subplot(2,1,2)
    for m=1:1:length(z_1)
        s = cross([E_x1(m) E_y1(m) E_z1(m)], [H_x1(m) H_y1(m) H_z1(m)]);
        S_z1(m) = s(3);
    end
    for m=1:1:length(z_2)
        s = cross([E_x2(m) E_y2(m) E_z2(m)], [H_x2(m) H_y2(m) H_z2(m)]);
        S_z2(m) = s(3);
    end
    plot(z_1,S_z1, 'b', 'linewidth', 2 ); grid on;
    hold on
    plot(z_2,S_z2, 'r', 'linewidth', 2 );
    title('Sz components')
    legend('total 1', 'total 2')
    axis([min(z_1), max(z_2) -1.2*max(max(S_z1,S_z2))  1.2*max(max(S_z1,S_z2))]) ;
    hold off

    pause(0.1)
end
```

Maximum electric field magnitude: 0.000000 V/m



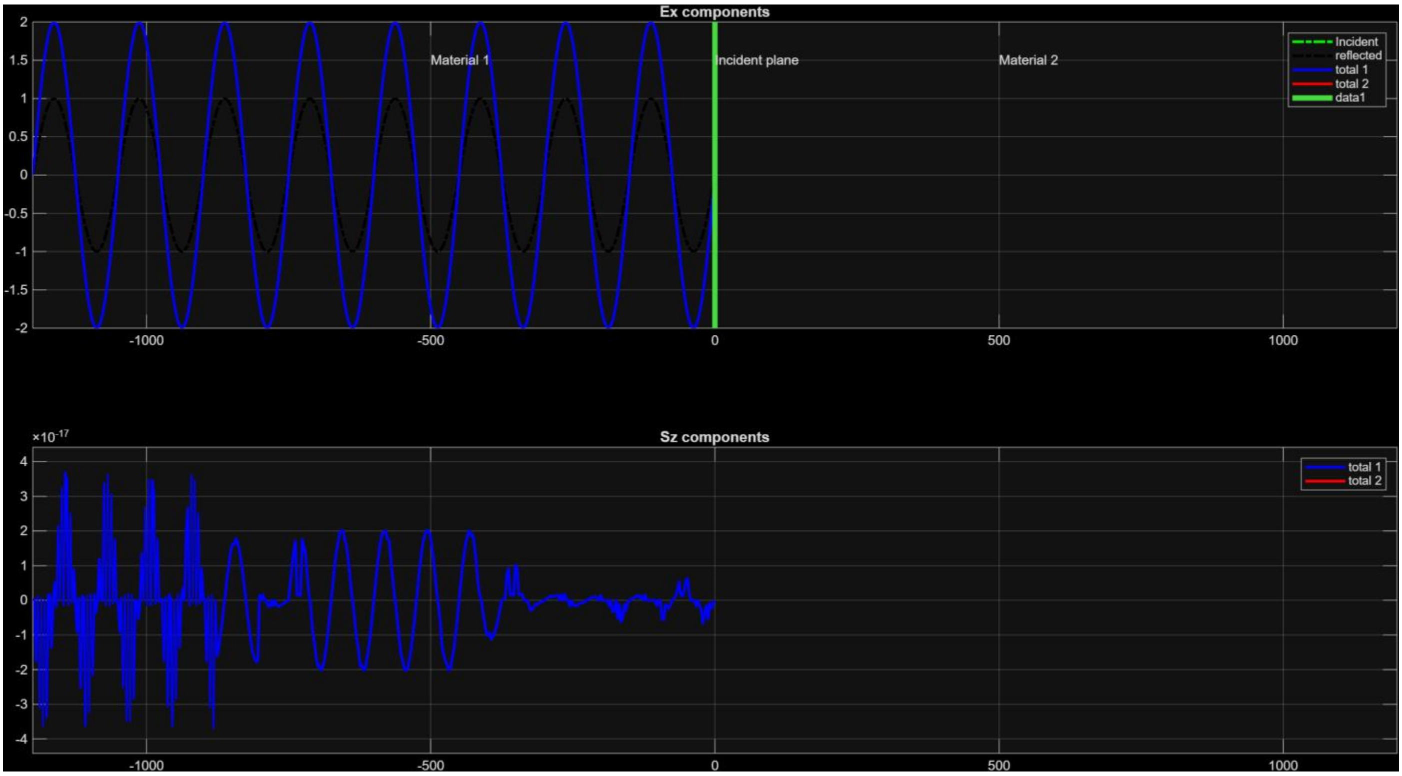
At t = T0/4

```
function TEM_Reflection( )
%% Function: Simulate the waves with a TEM wave incident on a boundary between two materials
%            Assuming a TEM wave polarized in the x direction
%  Parameters: f: frequency in Hz
%              Ex10_incid: amplitude of the incident wave
%              mu_r_1, epsilon_r_1, mu_r_2, epsilon_r_2: relative permittivity and permeability of the materials; assumed real-valued or infinity

%% Set parameters
f = 1*10^6;
Ex10_incid = 1;

% Region 1
mu_r_1 = 1;
epsilon_r_1 = 4;

% Region 2 (set eta_2=0 for perfect conductors)
mu_r_2 = 1;
epsilon_r_2 = 81-(j*inf);

%% Find wave parameters
c = 3*10^8;
lambda_0 = c/f;
[omega,beta_1,lambda_1,T0,eta_1] = losslesspropagation(f,mu_r_1,epsilon_r_1);
[omega,beta_2,lambda_2,T0,eta_2] = losslesspropagation(f,mu_r_2,epsilon_r_2);
```

```
%% Find reflection and transmission coefficients
Gamma = (eta_2-eta_1)/(eta_2+eta_1);
tau = Gamma + 1;

%% Calculate and visualize the Ex components of the waves in the two regions
dz = lambda_0/100;
z_1 = [-4*lambda_0:dz:0]';
z_2 = [0:dz:4*lambda_0]';
T_sim = T0/4;

for n_sim = 1:1:length(T_sim)
    t = T_sim(n_sim);
    % Total wave in Region 1
    E_x1 = Ex10_incid*cos(omega*t-
beta_1*z_1)+abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma));
    % Total wave in Region 2
    E_x2 = abs(tau)*Ex10_incid*cos(omega*t-beta_2*z_2+angle(tau));

    % After calculating E_x1 and E_x2, add:
    max_E_x1 = max(abs(E_x1));
    max_E_x2 = max(abs(E_x2));
    max_E_total = max(max_E_x1, max_E_x2);
    fprintf('Maximum electric field magnitude: %f V/m\n', max_E_total);

    % Plot Ex field components
    subplot(2,1,1)
    plot(z_1,Ex10_incid*cos(omega*t-beta_1*z_1), 'g-.', 'linewidth', 2 );  % Incident
wave, Region 1
    hold on
    plot(z_1,abs(Gamma)*Ex10_incid*cos(omega*t+beta_1*z_1+angle(Gamma)), 'k-.',
'linewidth', 2 );   % Reflected wave, Region 1
    plot(z_1,E_x1,'b', 'linewidth', 2 );    % Total wave, Region 1
    plot(z_2,E_x2, 'r', 'linewidth', 2 ); grid on; axis([min(z_1), max(z_2) -2 2]);  %
Total wave, Region 2
    legend('Incident', 'reflected', 'total 1', 'total 2')

    title('Ex components')
    plot([0 0], [-2 2], 'linewidth', 4)
    text(-500,1.5,'Material 1');
    text(500,1.5,'Material 2');
    text(0,1.5,'Incident plane');
    hold off

    % Calculate and plot z-direction component of the Poynting vectors
    % First find all the E and H field components
    H_y1 = Ex10_incid/eta_1*cos(omega*t-beta_1*z_1)-
abs(Gamma)*Ex10_incid/eta_1*cos(omega*t+beta_1*z_1+angle(Gamma));
    E_y1 = zeros(size(z_1));     E_z1 = zeros(size(z_1));
    H_x1 = zeros(size(z_1));     H_z1 = zeros(size(z_1));

    H_y2 = Ex10_incid*abs(tau)/eta_2*cos(omega*t-beta_2*z_2+angle(tau));
    E_y2 = zeros(size(z_2));     E_z2 = zeros(size(z_2));
```

```
    H_x2 = zeros(size(z_2));      H_z2 = zeros(size(z_2));

    % Compute the Poynting vector according to definition
    S_z1=zeros(size(z_1)); S_z2=zeros(size(z_2));
    subplot(2,1,2)
    for m=1:1:length(z_1)
        s = cross([E_x1(m) E_y1(m) E_z1(m)], [H_x1(m) H_y1(m) H_z1(m)]);
        S_z1(m) = s(3);
    end
    for m=1:1:length(z_2)
        s = cross([E_x2(m) E_y2(m) E_z2(m)], [H_x2(m) H_y2(m) H_z2(m)]);
        S_z2(m) = s(3);
    end
    plot(z_1,S_z1, 'b', 'linewidth', 2 ); grid on;
    hold on
    plot(z_2,S_z2, 'r', 'linewidth', 2 );
    title('Sz components')
    legend('total 1', 'total 2')
    axis([min(z_1), max(z_2) -1.2*max(max(S_z1,S_z2))  1.2*max(max(S_z1,S_z2))]) ;
    hold off

    pause(0.1)
end
```

Maximum electric field magnitude: 1.996053 V/m



$$\text{Ratio} = \frac{1.996053}{0.000000} = \infty$$