Plot the VCSEL spectrum

```
clear all;close all;clc;
GratingNumber = 125;
NGbot= GratingNumber;
NGtop = GratingNumber;
lambdaV=linspace(1450e-9,1650e-9,100001);
count=1;
for lambda=lambdaV
    [S]=Stack(NGbot,NGtop,lambda);
    R(count)=abs(S(2,1)).^2;
    T(count)=abs(S(1,1).*S(2,2));
    Rdb(count) = 10*log10(R(count));
    Tdb(count) = 10*log10(T(count));
    count=count+1;
end

figure
plot(lambdaV*1e9,R,lambdaV*1e9,T,lambdaV*1e9,R+T,'LineWidth',1.5)
grid on;
grid minor;
xlabel ('lambda (nm)', 'Fontsize', 12);
ylabel ('Transmission and Reflection', 'Fontsize', 12);
xlim([min(lambdaV) max(lambdaV)]*1e9)
legend('R','T','R+T')
ylim([0 1])

figure;
plot(lambdaV*1e9, Rdb,lambdaV*1e9, Tdb, 'LineWidth', 2);
set(gca, 'FontSize', 14);
xlabel('Wavelength (nm)','FontSize', 12);
ylabel('Response Transmission in dB','FontSize', 12);
legend('Reflection','Transmission');


alpha= 10^(1.5/10)*1e2; %Loss in m-1
Q=(2*pi*GratingNumber)/(1550e-9*alpha) %Quality factor of Bragg
 grating w specified loss
```

VCSEL Model

```
% starts at low, ends at low, and the cavity is in low
function [S]=Stack(NGbot,NGtop,lambda)
% parameters
theta=0; % normal incidence
mode='TE'; % TE mode assumed (doesn't make difference here bc normal
 incidence)
Bragg=1550e-9; % Bragg wavelength
alpha= 10^(1.5/10)*1e2; % DBR losses (m?-1)
n_eff = calneff(lambda); % given average effective index
delta_n=0.08; % index contrast between n1 and n2
Period=Bragg/(2*n_eff); % period of the gratings combined
```

```
n1=n_eff-delta_n/2; % index of small grating
n2=n_eff+delta_n/2; % index of large grating
d1=Bragg/(4*n1); % width of small grating
d2=Bragg/(4*n2); % width of large grating
% NGbot=41; % number of gratings in the bottom mirror
% phase shifted parameters
dG=300e-9; % length of gain
% dcav=((d1+d1)-dG)/2;
dcav=(Bragg/(2*n1)-dG)/2;
nG=n1;
nCav=n1;
gain=0; %cavity gain
% convert the scattering to transfer matrices for cascading
Mprop1=StoM(Sprop(n1,theta,lambda,d1,alpha));
Mprop2=StoM(Sprop(n2,theta,lambda,d2,alpha));
MpropG=StoM(Sprop(nG,theta,lambda,dG,gain));
MpropCav=StoM(Sprop(nCav,theta,lambda,dcav,alpha));
M1to2=StoM(Sboundary(n1,n2,theta,mode));
M2to1=StoM(Sboundary(n2,n1,theta,mode));
% M1toG=StoM(Sboundary(n1,nG,theta,mode));
% MGto1=StoM(Sboundary(nG,n1,theta,mode));
% M2toG=StoM(Sboundary(n2,nG,theta,mode));
% MGto2=StoM(Sboundary(nG,n2,theta,mode));

% top DBR
Mtop=(M2to1*Mprop2*M1to2*Mprop1)^NGtop;
% gain
Mgain=MpropCav*MpropG*MpropCav;
% bottom DBR
Mbot=(Mprop1*M2to1*Mprop2*M1to2)^NGbot;
% multiply the transfer matrices and convert the result to scattering
M = Mbot*Mgain*Mtop;
S = MtoS(M);

end

function n_eff = calneff(lambda)
    n_eff = 2.38-1.1886*(lambda-1.55)-0.31201*(lambda-1.55)^2;
end

function [S,theta2]=Sboundary(n1,n2,theta,mode) % boundary scattering
 matrix from n1 ... to n2
theta1=theta;
theta2=asin(n1)*sin(theta1)/n2; % Snell's law
if mode=='TE'
    n1bar=n1*cos(theta1);
    n2bar=n2*cos(theta2);
    a12=1;
    a21=1;
end
if mode=='TM'
    n1bar=n1*sec(theta1);
    n2bar=n2*sec(theta2);
    a12=cos(theta1)/cos(theta2);
```

```
    a21=1/a12;
end
S = (1/(n1bar+n2bar))*[2*a12*n1bar n2bar-n1bar;n1bar-n2bar
 2*a21*n2bar];
end

function [S]=Sprop(n,theta,lambda,d,alpha) % propagation scattering
 matrix
k = 2*pi./lambda;
phi = n*k*d*cos(theta);
S = [exp(-i*phi-alpha*d/2) 0;0 exp(-i*phi-alpha*d/2)];
end

function [M]=StoM(S)
A = S(1,1)*S(2,2)-S(1,2)*S(2,1);
B = S(1,2);
C = -S(2,1);
D=1;
M = (1/S(2,2))*[A B;C D];
end

function [S]=MtoS(M)
t1N= M(1,1)*M(2,2)-M(1,2)*M(2,1);
rN1 = M(1,2);
r1N = -M(2,1);
tN1 = 1;
S = (1/M(2,2))*[t1N rN1;r1N tN1];
end
```

```
Q =

   3.5872e+06
```

Published with MATLAB® R2016b