

REPORT

4th Homework



Subject	Computational Microelectronics
Professor	홍 성 민
Name	조 성 훈
Student Number	20162071
Due Date	2018/10/1 - 8AM

1. Write your own code for the double-gate structure

A. This code is for question 1st, 2nd, 3rd, and 4th

```
1 - clc; clear all;
2
3
4 - %%% Set Parameter
5 - q = 1.602192e-19; % elementary charge, C
6 - eps0 = 8.854187817e-12; % Vacuum permittivity, F/m
7 - k_B = 1.380662e-23; % Boltzmann constant, J/K
8 - T = 300.0; % Temperature, K
9 - Deltax = 0.1e-9; % 0.1 nm spacing
10 - N = 61; % 6 nm thick
11 - x = Deltax*transpose([0:N-1]); % real space, m
12 - interface1 = 6; % At x=0.5 nm
13 - interface2 = 56; % At x=5.5 nm
14 - eps_si = 11.7; eps_ox = 3.9; % Relative permittivity
15 - Nacc = 1e24; % 1e18 /cm^3
16 - ni = 1.075e16; % 1.075e10 /cm^3
17 - work_vac = 4.3; % Work function under vacuum
18 - int_si = 4.63374; % Work function under silicon
19 - v_bias = 1; % v_bias is varied from 0 to 1 with the 0.1 step
20
21
22
23 - %%% Set 'A' Matrix
24 - A = zeros (N,N);
25 - A(1,1) = 1.0;
26 - for ii=2:N-1
27 -     if (ii < interface1) A(ii,ii-1) = eps_ox; A(ii,ii) = -2*eps_ox; A(ii,ii+1) = eps_ox;
28 -     elseif (ii == interface1) A(ii,ii-1) = eps_ox; A(ii,ii) = -eps_ox-eps_si; A(ii,ii+1) = eps_si;
29 -     elseif (ii < interface2) A(ii,ii-1) = eps_si; A(ii,ii) = -2*eps_si; A(ii,ii+1) = eps_si;
30 -     elseif (ii == interface2) A(ii,ii-1) = eps_si; A(ii,ii) = -eps_si-eps_ox; A(ii,ii+1) = eps_ox;
31 -     elseif (ii > interface2) A(ii,ii-1) = eps_ox; A(ii,ii) = -2*eps_ox; A(ii,ii+1) = eps_ox;
32 -     end
33 - end
34 - A(N,N) = 1.0;
35
36
37 - %%% Initial poisson equation
38 - b = zeros(N,1);
39 - b(1,1) = - ( work_vac - int_si ) - v_bias;
40 - for ii = interface1:interface2 % only silicon region is doped
41 -
42 -     if (ii==interface1) b(ii,1) = Deltax*Deltax*q*Nacc/eps0*0.5; % interface doping level
43 -     elseif (ii==interface2) b(ii,1) = Deltax*Deltax*q*Nacc/eps0*0.5; % interface doping level
44 -     else b(ii,1) = Deltax*Deltax*q*Nacc/eps0; % interface doping level
45 -
46 -     end
47 - end
48 - b(N,1) = - ( work_vac - int_si ) - v_bias;
```

```

49
50     %%% Calculation for electrostatic potential
51     phi = A#b;
52
53     elec = zeros(N,1);
54     for ii = interface1:interface2
55         elec(ii,1) = ni * exp(q*phi(ii,1)/(k_B*T)); % only silicon region that is doped is cacluated
56     end
57
58     figure(1)
59     plot(x/1e-9, phi+1); % Plot with scailing the unit
60     title('The electrostatic potential under the depletion')
61     figure(2)
62     stem(x/1e-9, elec*1e-6); % Plot with scailing the unit
63     title('The electrostatic density under the depletion')
64
65
66     %%% Iteration of poisson eqations for electrostatic potential
67     b_2nd = zeros(N,1);
68     b_2nd(1,1) = - ( work_vac - int_si) - v_bias;
69     for ii = interface1:interface2 % only silicon region is doped
70
71         if (ii==interface1) b_2nd(ii,1) = Deltax*Deltax+q*(Nacc+elec(ii,1))/eps0*0.5; % interface doping level
72         elseif (ii==interface2) b_2nd(ii,1) = Deltax*Deltax+q*(Nacc+elec(ii,1))/eps0*0.5; % interface doping level
73         else b_2nd(ii,1) = Deltax*Deltax+q*(Nacc+elec(ii,1))/eps0; % interface doping level
74
75     end
76 end
77 b_2nd(N,1) = - ( work_vac - int_si) - v_bias;
78
79
80 phi_2nd = A#b_2nd;
81
82 elec_2nd = zeros(N,1);
83 for ii = interface1:interface2
84     elec_2nd(ii,1) = ni * exp(q*phi_2nd(ii,1)/(k_B*T)); % only silicon region that is doped is cacluated
85 end
86
87 figure(3)
88 plot(x/1e-9, phi_2nd+1); % Plot with scailing the unit
89 title('The electrostatic potential using the initial result')
90 figure(4)
91 stem(x/1e-9, elec_2nd*1e-6); % Plot with scailing the unit
92 title('The electrostatic density using the initial result')
93
94 %%% Differenece between depletion assumption and consideration of fermi level variation
95 diff_Electropotential(:,1) = phi-phi_2nd;
96 diff_Electrodensity(:,1) = elec-elec_2nd;
97
98 figure(5)
99 plot(x/1e-9, diff_Electropotential);
100 title('The electrostatic potential difference')
101 figure(6)
102 stem(x/1e-9, diff_Electrodensity);
103 title('The electrostatic density difference')

```

B. This code is for question 5th

```

1 - clc; clear all;
2
3 -   *** Set Parameter
4 -   q = 1.602192e-19;      % elementary charge, C
5 -   eps0 = 8.854187817e-12; % Vacuum permittivity, F/m
6 -   k_B = 1.380662e-23;    % Boltzmann constant, J/K
7 -   T = 300.0;             % Temperature, K
8 -   Deltax = 0.1e-9;       % 0.1 nm spacing
9 -   N = 61;                % 6 nm thick
10 -  x = Deltax*transpose([0:N-1]); % real space, m
11 -  interface1 = 6;         % At x=0.5 nm
12 -  interface2 = 56;       % At x=5.5 nm
13 -  eps_si = 11.7;  eps_ox = 3.9; % Relative permittivity
14 -  Nacc = 1e24;           % 1e18 /cm^3
15 -  ni = 1.075e16;         % 1.075e10 /cm^3
16 -  work_vac = 4.3;        % Work function under vacuum
17 -  int_si = 4.63374;      % Work function under silicon
18 -  v_bias = 1;            % v_bias is varied from 0 to 1 with the 0.1 step
19
20 -  A = zeros (N,N);
21 -  b = zeros(N,11);
22 -  elec = zeros(N,11);
23
24 -  b_2nd = zeros(N,11);
25 -  elec_2nd = zeros(N,11);
26
27 -  for j=1:11
28 -      *** Set 'A' Matrix
29
30 -      A(1,1) = 1.0;
31 -      for ii=2:N-1
32 -          if (ii < interface1) A(ii,ii-1) = eps_ox; A(ii,ii) = -2*eps_ox; A(ii,ii+1) = eps_ox;
33 -          elseif (ii == interface1) A(ii,ii-1) = eps_ox; A(ii,ii) = -eps_ox-eps_si; A(ii,ii+1) = eps_si;
34 -          elseif (ii < interface2) A(ii,ii-1) = eps_si; A(ii,ii) = -2*eps_si; A(ii,ii+1) = eps_si;
35 -          elseif (ii == interface2) A(ii,ii-1) = eps_si; A(ii,ii) = -eps_si-eps_ox; A(ii,ii+1) = eps_ox;
36 -          elseif (ii > interface2) A(ii,ii-1) = eps_ox; A(ii,ii) = -2*eps_ox; A(ii,ii+1) = eps_ox;
37 -          end
38 -      end
39 -      A(N,N) = 1.0;
40
41 -      *** Initial poisson equation
42
43 -      b(1,j) = - ( work_vac - int_si) - (j-1)/10;
44 -      for ii = interface1:interface2 % only silicon region is doped
45 -          if (ii==interface1) b(ii,j) = Deltax*Deltax*q*Nacc/eps0*0.5; % interface doping level
46 -          elseif (ii==interface2) b(ii,j) = Deltax*Deltax*q*Nacc/eps0*0.5; % interface doping level
47 -          else b(ii,j) = Deltax*Deltax*q*Nacc/eps0; % interface doping level
48 -          % interface doping level
49 -          % interface doping level
50 -      end

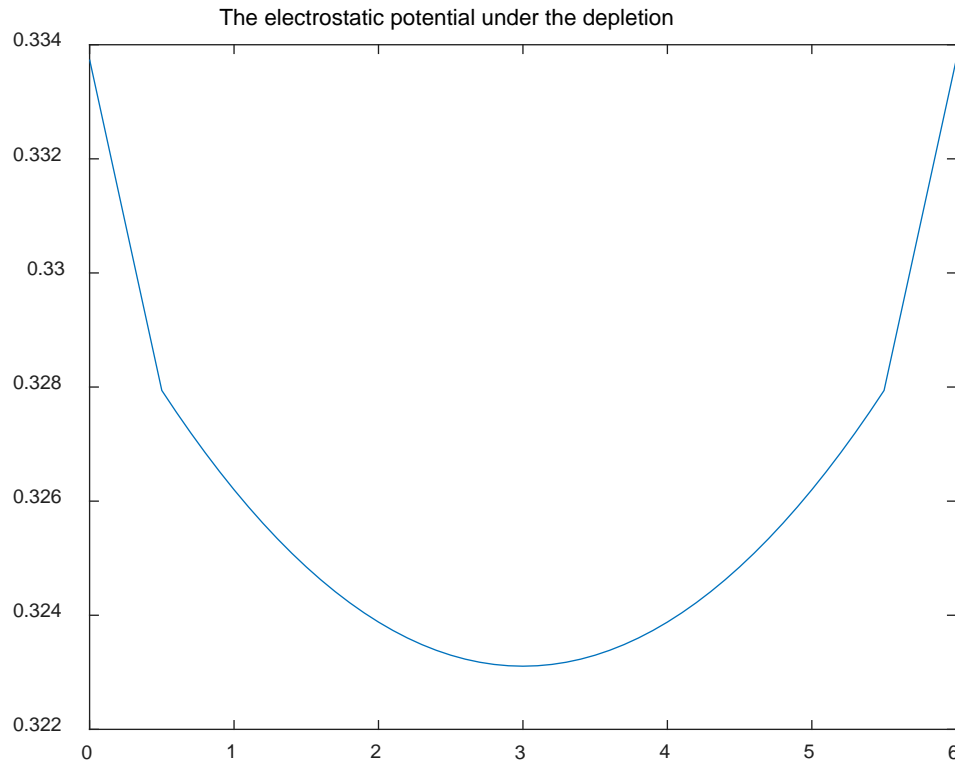
```

```

51 -     end
52 - end
53 - b(N,j) = - ( work_vac - int_si) - (j-1)/10;
54 -
55 - %%% Calculation for electrostatic potential
56 - phi(:,j) = A*b(:,j);
57 -
58 -
59 - for ii = interface1:interface2
60 -     elec(ii,j) = ni * exp(q*phi(ii,j)/(k_B*T)); % only silicon region that is doped is cacluated
61 - end
62 -
63 -
64 -
65 - b_2nd(1,j) = - ( work_vac - int_si) - (j-1)/10;
66 - for ii = interface1:interface2 % only silicon region is doped
67 -
68 -     if (ii==interface1) b_2nd(ii,j) = Deltax*Deltax*q*(Nacc+elec(ii,j))/eps0+0.5; % interface doping level
69 -     elseif (ii==interface2) b_2nd(ii,j) = Deltax*Deltax*q*(Nacc+elec(ii,j))/eps0+0.5; % interface doping level
70 -     else b_2nd(ii,j) = Deltax*Deltax*q*(Nacc+elec(ii,j))/eps0; % interface doping level
71 -
72 -     end
73 - end
74 - b_2nd(N,j) = - ( work_vac - int_si) - (j-1)/10;
75 -
76 -
77 - phi_2nd(:,j) = A*b_2nd(:,j);
78 -
79 -
80 - for ii = interface1:interface2
81 -     elec_2nd(ii,j) = ni * exp(q*phi_2nd(ii,j)/(k_B*T)); % only silicon region that is doped is cacluated
82 - end
83 -
84 - end
85 -
86 -
87 - %%% Differnece between depletion assumption and consideration of fermi level variation
88 - diff_Electropotential(:,j) = phi(:,j)-phi_2nd(:,j);
89 - diff_Electrodensity(:,j) = elec(:,j)-elec_2nd(:,j);
90 -
91 -
92 - figure(1)
93 - for j=1:11
94 -     plot(x/1e-9,elec(:,j))/1e+6;
95 -     hold on;
96 - end
97 - xlabel('distance (nm)');
98 - ylabel('Electron density (cm^3)');
99 -
100 - figure(2)
101 - for j=1:11
102 -     plot(x/1e-9,phi_2nd(:,j));
103 -     hold on;
104 - end
105 - xlabel('distance (nm)');
106 - ylabel('Electrostatic potential (V)');
107 -
108 - figure(3)
109 - for j=1:11
110 -     plot(x/1e-9,diff_Electropotential(:,j));
111 -     hold on;
112 - end
113 - xlabel('distance (nm)');
114 - ylim([-0.5e-5 3.5e-5])
115 - ylabel('Potential difference (V)');

```

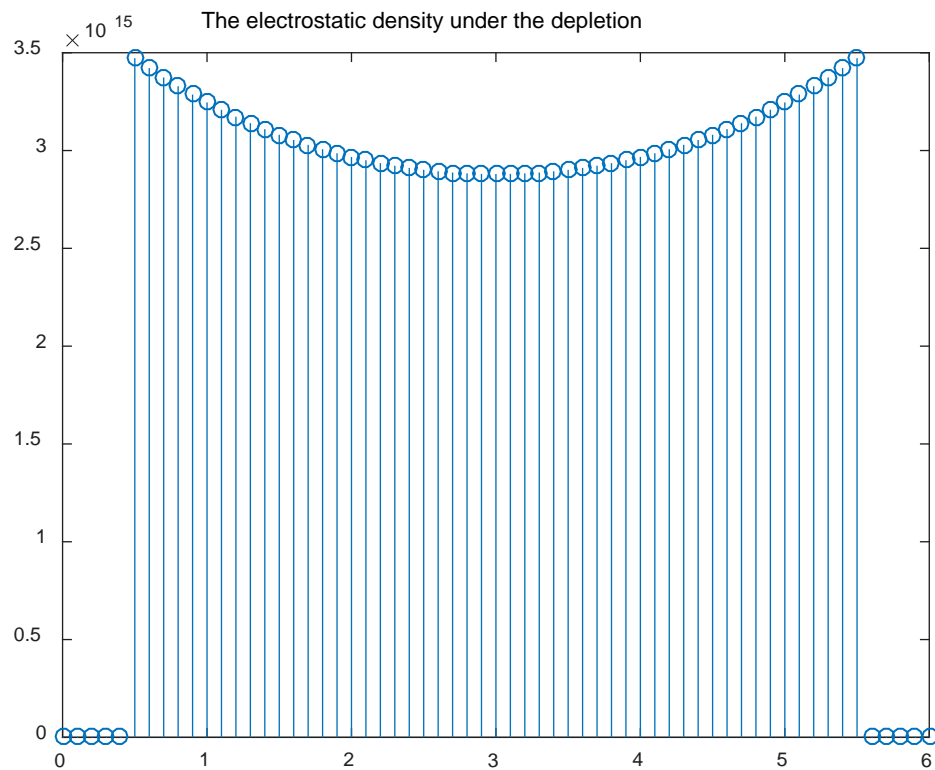
2. The electrostatic potential is obtained under the depletion approximation. (The initial potential)



A. The electrostatic potential is divided by three region

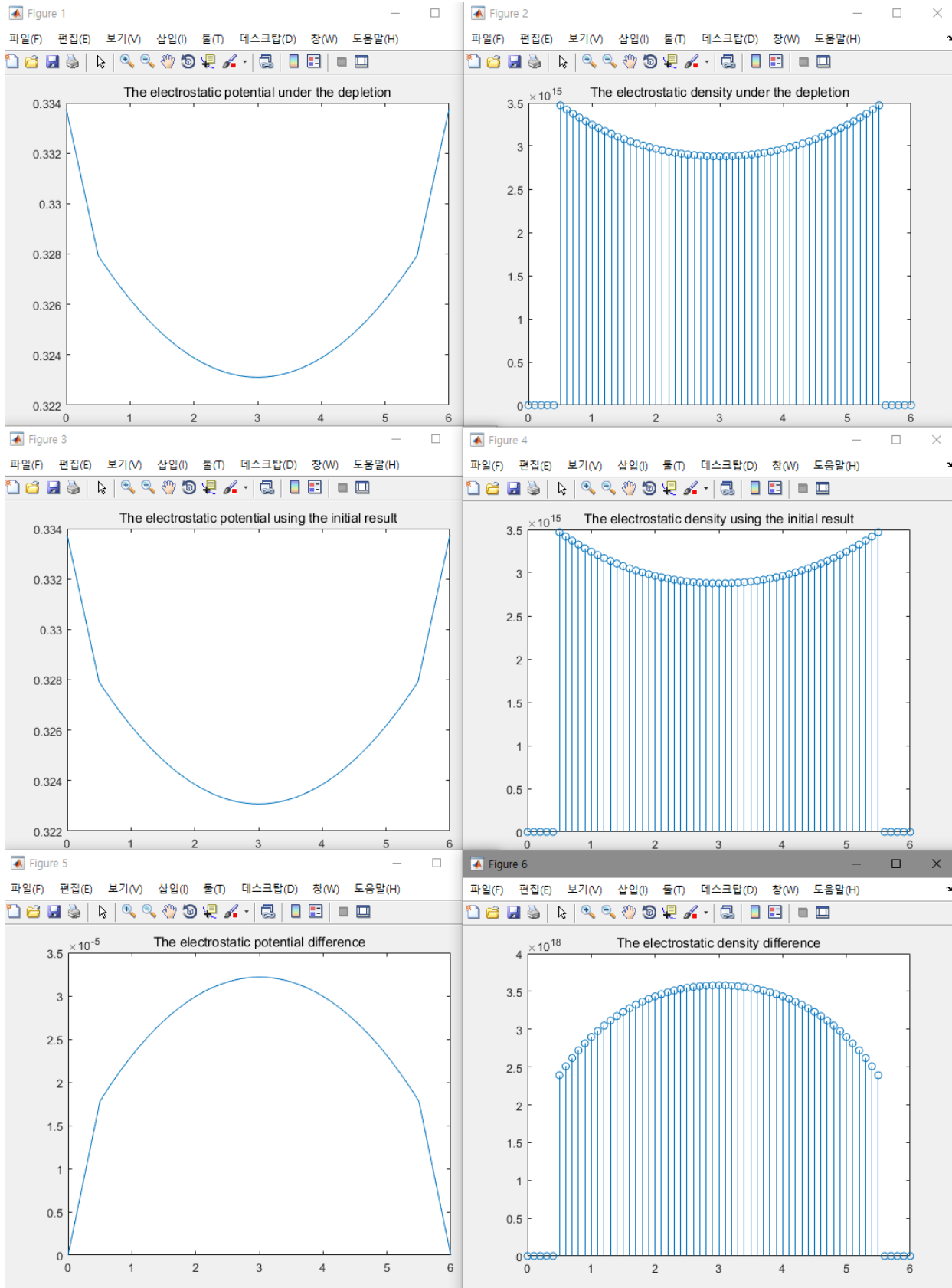
- i. The first region is ' $x_{\text{SiO}_2} = [0, 5\text{nm}]$ '. At this point because of the free-charge space, the slop is linear.
- ii. The second region is ' $x_{\text{Si}} = [5, 55\text{nm}]$ '. At this point because of the existence of the dopant, the slop is parabolic.
- iii. The first region is ' $x_{\text{SiO}_2} = [55, 60\text{nm}]$ '. At this point because of the free-charge space, the slop is linear.

3. The electron density is calculated by assuming $n_i \cdot \exp(q \cdot \phi / k_B / T)$



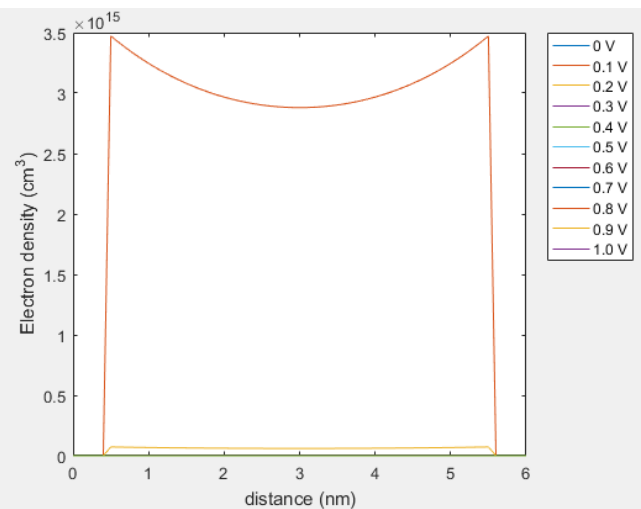
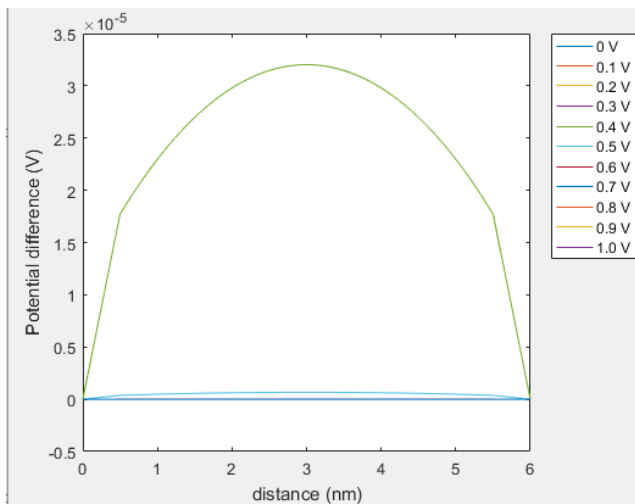
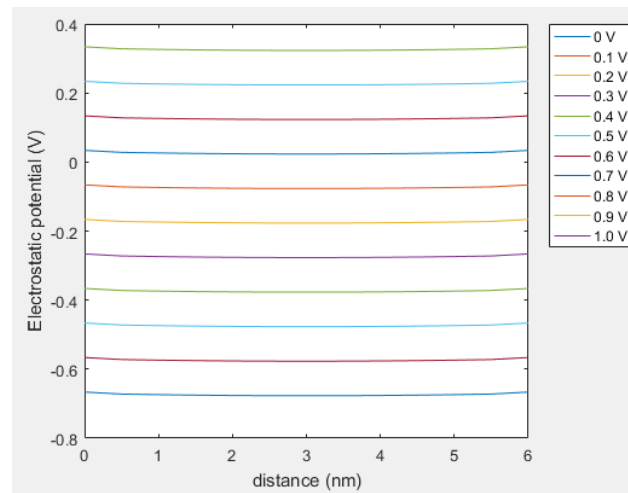
- A. Using the interface doping charge equation, we can find the electrostatic density
- B. Because of the free charge space of the SiO_2 , the ranges of '[0, 5 nm]', '[55 nm, 60 nm]' have zero electron density.

4. Then, using the electron density, re-calculate the potential. (The updated potential)



- A. The difference electrostatic potential and electron density exist
- B. This difference is introduced by electron density that it was assumed by depletion in the Si region (range from 5 nm to 55nm)
- C. This charge which make potential difference can be accurately calculated by this iteration process

5. Check their difference for several gate voltages from 0 V to 1V. (The voltage step is 0.1 V. 11points in total)



- A. When the gate voltage is varied from 0V to 1V, the difference is drastically reduced

- i. The electron density is function of energy of conduction band
- ii. This relation is inverse proportional and exponentially decrease with E_c
- iii. When the larger gate bias is applied, the energy of conduction band is greater than no bias condition
- iv. So, the electron density in the Si region is drastically reduced