MATLAB Code Appendix (in order of the problems).

```
% Mohamed Ghonim - ECE 515 Fundamentals of Semiconductor Devices
% Project 2 MOSFET Characterization
% Dr. Malgorzata Chrzanowska-Jeske

clc
clear % This clears all variables
close all % This closes all figures

% Reading the data from CSV
filename = 'Id_Vd.csv'; % update with your filename
data = xlsread(filename);

% Separating the data into vectors
V_DS = data(:,1); % Drain-Source Voltage
I_D = data(:,2); % Drain Current
V_GS = data(:,3); % Gate-Source Voltage

% Get unique V_GS values
uniqueV_GS = unique(V_GS);

% Create a figure
figure;
hold on; % This will allow multiple plots on the same figure

% Loop over unique V_GS values
for i = 1:length(uniqueV_GS)
    % Get the indices for the current V_GS
    indices = V_GS == uniqueV_GS(i);
    % Plot I_D vs V_DS for this V_GS
    plot(V_DS(indices), I_D(indices), 'LineWidth', 1.5, 'DisplayName', ['V_{GS} = ', 
num2str(uniqueV_GS(i)), 'V']);
end

% Adding labels and title
xlabel('V_{DS} (V)');
ylabel('I_D (A)');
title('Output Characteristics of the MOSFET');
legend('show'); % Show legend

% Holding off the figure
hold off;
```

```matlab
% Mohamed Ghonim - ECE 515 Fundamentals of Semiconductor Devices
% Project 2 MOSFET Characterization
% Dr. Malgorzata Chrzanowska-Jeske

clc
clear % This clears all variables
close all % This closes all figures

% Reading the data from Excel
filename = 'Id_Vg.csv'; % update with your filename
data = xlsread(filename);

% Separating the data into vectors
V_GS = data(:,1); % Gate-Source Voltage
I_D = data(:,2); % Drain Current
V_DS = data(:,4); % Drain-Source Voltage

% Get unique V_DS values
uniqueV_DS = unique(V_DS);

% Create a figure
figure;
hold on; % This will allow multiple plots on the same figure

% Loop over unique V_DS values
for i = 1:length(uniqueV_DS)
    % Get the indices for the current V_DS
    indices = V_DS == uniqueV_DS(i);
    % Plot I_D vs V_GS for this V_DS
    plot(V_GS(indices), I_D(indices), 'LineWidth', 1.5, 'DisplayName', ['V_{DS} = ',
num2str(uniqueV_DS(i)), 'V']);
end

% Adding labels and title
xlabel('V_{GS} (V)');
ylabel('I_D (A)');
title('Transfer Characteristics of the MOSFET');
legend('show','Location','northwest'); % Show legend

% Holding off the figure
hold off;
```

```
% Mohamed Ghonim - ECE 515 Fundamentals of Semiconductor Devices
% Project 2 MOSFET Characterization
% Dr. Malgorzata Chrzanowska-Jeske

clc
clear % This clears all variables
close all % This closes all figures

% Reading the data from Excel
filename = 'Id_Vg.csv'; % update with your filename
data = xlsread(filename);

% Separating the data into vectors
V_GS = data(:,1); % Gate-Source Voltage
I_D = data(:,2); % Drain Current
V_DS = data(:,4); % Drain-Source Voltage

% Specify the two V_DS values of interest
V_DS1 = min(unique(0.25)); % e.g., minimum V_DS
V_DS2 = max(unique(2.5)); % e.g., maximum V_DS

% Create a figure
figure;
hold on; % This will allow multiple plots on the same figure

% Loop over the two selected V_DS values
for V_DS_val = [V_DS1, V_DS2]
    % Get the indices for the current V_DS
    indices = V_DS == V_DS_val;
    % Plot I_D vs V_GS for this V_DS
    plot(V_GS(indices), I_D(indices), 'LineWidth', 1.5, 'DisplayName', ['V_{DS} = ',
num2str(V_DS_val), 'V']);
end

% Adding labels and title
xlabel('V_{GS} (V)');
ylabel('I_D (A)');
title('Transfer Characteristics of the MOSFET for Selected V_{DS} Values');
legend('show'); % Show legend

% Holding off the figure
hold off;
```

```matlab
% Mohamed Ghonim - ECE 515 Fundamentals of Semiconductor Devices
% Project 2 MOSFET Characterization
% Dr. Malgorzata Chrzanowska-Jeske

clc
clear % This clears all variables
close all % This closes all figures

% Reading the data from Excel
filename = 'Id_Vg.csv'; % update with your filename
data = xlsread(filename);

% Separating the data into vectors
V_GS = data(:,1); % Gate-Source Voltage
I_D = data(:,2); % Drain Current
V_DS = data(:,4); % Drain-Source Voltage

% Specify the V_DS value of interest
V_DS_val = 0.25; % e.g., V_DS = 0.25V

% Get the indices for the specified V_DS
indices = V_DS == V_DS_val;

% Create a figure
figure;

% Plot I_D vs V_GS for this V_DS
plot(V_GS(indices), I_D(indices), 'LineWidth', 1.5);
hold on; % To allow multiple plots on the same figure

% Define the points for fitting the line
V_GS_line_points = [2.4, 2.8];
ID_line_points = [I_D(V_GS == 2.4 & indices), I_D(V_GS == 2.8 & indices)];

% Fit a line through the specified points
P = polyfit(V_GS_line_points, ID_line_points, 1);

% Generate points to draw the fitted line
V_GS_line = linspace(min(V_GS(indices)), max(V_GS(indices)), 100);
ID_line = polyval(P, V_GS_line);

% Plot the line
plot(V_GS_line, ID_line, 'r', 'LineWidth', 1.5);

% Calculate the x-axis intersection
V_T_estimate = -P(2) / P(1);
disp(['The estimated threshold voltage (x-axis intersection of the line) is: ',
num2str(V_T_estimate), ' V']);

% Add the equation of the line to the plot
str = sprintf('y = %.3f x + %.3f', P(1), P(2));
text(mean(V_GS_line), mean(ID_line), str, 'Color', 'red');
```

```
% Adding labels and title
xlabel('V_{GS} (V)');
ylabel('I_D (A)');
title(['Drain Current vs. Gate-Source Voltage for V_{DS} = ', num2str(V_DS_val),
'V']);

% Holding off the figure
hold off;
```

```
% Mohamed Ghonim - ECE 515 Fundamentals of Semiconductor Devices
% Project 2 MOSFET Characterization
% Dr. Malgorzata Chrzanowska-Jeske


clc
clear % This clears all variables
close all % This closes all figures

% Reading the data from Excel
filename = 'Id_Vg.csv'; % update with your filename
data = xlsread(filename);

% Separating the data into vectors
V_GS = data(:,1); % Gate-Source Voltage
I_D = data(:,2); % Drain Current
V_DS = data(:,4); % Drain-Source Voltage

% Specify the V_DS value of interest
V_DS_val = 2.5; % e.g., V_DS = 2.5V

% Get the indices for the specified V_DS
indices = V_DS == V_DS_val;

% Create a figure
figure;
% Plot sqrt(I_D) vs V_GS for this V_DS
plot(V_GS(indices), sqrt(I_D(indices)), 'LineWidth', 1.5);
% Adding labels and title
xlabel('V_{GS} (V)');
ylabel('sqrt(I_D) (A^{1/2})');
title(['Square Root of Drain Current vs. Gate-Source Voltage for V_{DS} = ',
num2str(V_DS_val), 'V']);
```

```
% Mohamed Ghonim - ECE 515 Fundamentals of Semiconductor Devices
% Project 2 MOSFET Characterization
% Dr. Malgorzata Chrzanowska-Jeske

clc
clear % This clears all variables
close all % This closes all figures

% Reading the data from Excel
filename = 'Id_Vg.csv'; % update with your filename
data = xlsread(filename);

% Separating the data into vectors
V_GS = data(:,1); % Gate-Source Voltage
I_D = data(:,2); % Drain Current
V_DS = data(:,4); % Drain-Source Voltage

% Specify the V_DS value of interest
V_DS_val = 2.5; % e.g., V_DS = 2.5V

% Get the indices for the specified V_DS
indices = V_DS == V_DS_val;

% Create a figure
figure;

% Plot sqrt(I_D) vs V_GS for this V_DS
plot(V_GS(indices), sqrt(I_D(indices)), 'LineWidth', 1.5);
hold on; % To allow multiple plots on the same figure

% Define the points for fitting the line (asymptote)
V_GS_asymptote_points = [2.5, 2.6];
sqrt_ID_asymptote_points = [sqrt(I_D(V_GS == 2.5 & indices)), sqrt(I_D(V_GS == 2.6 &
indices))];

% Fit a line through the specified points
P = polyfit(V_GS_asymptote_points, sqrt_ID_asymptote_points, 1);

% Generate points to draw the fitted line (asymptote)
V_GS_line = linspace(min(V_GS(indices)), max(V_GS(indices)), 100);
sqrt_ID_line = polyval(P, V_GS_line);

% Plot the asymptote
plot(V_GS_line, sqrt_ID_line, 'r', 'LineWidth', 1.5);

% Calculate the x-axis intersection
V_T_estimate = -P(2) / P(1);
disp(['The estimated threshold voltage (x-axis intersection of the asymptote) is: ',
num2str(V_T_estimate), ' V']);

% Add the equation of the line to the plot
str = sprintf('y = %.3f x + %.3f', P(1), P(2));
text(mean(V_GS_line), mean(sqrt_ID_line), str, 'Color', 'red');
```

```
% Adding labels and title
xlabel('V_{GS} (V)');
ylabel('sqrt(I_D) (A^{1/2})');
title(['Square Root of Drain Current vs. Gate-Source Voltage for V_{DS} = ',
num2str(V_DS_val), 'V']);

% Holding off the figure
hold off;
```

Q3 linear region

```
% Mohamed Ghonim - ECE 515 Fundamentals of Semiconductor Devices
% Project 2 MOSFET Characterization
% Dr. Malgorzata Chrzanowska-Jeske

clc
clear % This clears all variables
close all % This closes all figures

% Reading the data from Excel
filename = 'Id_Vg.csv'; % update with your filename
data = xlsread(filename);

% Separating the data into vectors
V_GS = data(:,1); % Gate-Source Voltage
I_D = data(:,2); % Drain Current
V_DS = data(:,4); % Drain-Source Voltage

% Specify the V_DS value of interest
V_DS_val = 0.25; % e.g., V_DS = 0.25V

% Get the indices for the specified V_DS
indices = V_DS == V_DS_val;

% Define the points for fitting the line
line_indices = indices & V_GS >= 2.4 & V_GS <= 2.8;
V_GS_line_points = V_GS(line_indices);
ID_line_points = I_D(line_indices);

% Fit a line through the specified points
P = polyfit(V_GS_line_points, ID_line_points, 1);

% Generate points to draw the fitted line
V_GS_line = linspace(min(V_GS_line_points), max(V_GS_line_points), 100);
ID_line = polyval(P, V_GS_line);

% Calculate the x-axis intersection (threshold voltage, V_T)
V_T_estimate = -P(2) / P(1);

% Define the W/L ratio
WL_ratio = 31; % W/L = 31, from your provided info
```

```
% Calculate the µnCox product from the slope
u_n_C_ox_estimate = P(1) / (V_DS_val * WL_ratio); % dividing by V_DS and WL_ratio
because the slope of ID vs VGS in the linear region is µnCox * V_DS * WL_ratio

% Print the estimated parameters
disp(['The estimated threshold voltage (x-axis intersection of the line) is: ',
num2str(V_T_estimate), ' V']);
disp(['The estimated µnCox product (from the slope of the line) is: ',
num2str(u_n_C_ox_estimate), ' A/V^2']);

% Create a figure
figure;

% Plot I_D vs V_GS for this V_DS
plot(V_GS(indices), I_D(indices), 'LineWidth', 1.5);
hold on; % To allow multiple plots on the same figure

% Plot the line
plot(V_GS_line, ID_line, 'r', 'LineWidth', 1.5);

% Add the equation of the line to the plot
str = sprintf('y = %.3f x + %.3f', P(1), P(2));
text(mean(V_GS_line), mean(ID_line), str, 'Color', 'red');

% Adding labels and title
xlabel('V_{GS} (V)');
ylabel('I_D (A)');
title(['Drain Current vs. Gate-Source Voltage for V_{DS} = ', num2str(V_DS_val),
'V']);

% Holding off the figure
hold off;
```

Q3 Saturation Region

```
% Mohamed Ghonim - ECE 515 Fundamentals of Semiconductor Devices
% Project 2 MOSFET Characterization
% Dr. Malgorzata Chrzanowska-Jeske

clc
clear % This clears all variables
close all % This closes all figures

% Reading the data from Excel
filename = 'Id_Vg.csv'; % update with your filename
data = xlsread(filename);

% Separating the data into vectors
V_GS = data(:,1); % Gate-Source Voltage
I_D = data(:,2); % Drain Current
V_DS = data(:,4); % Drain-Source Voltage
```

```matlab
% Specify the V_DS value of interest
V_DS_val = 2.5; % e.g., V_DS = 2.5V

% Get the indices for the specified V_DS
indices = V_DS == V_DS_val;

% Define the points for fitting the line
line_indices = indices & V_GS >= 2.4 & V_GS <= 2.8;
V_GS_line_points = V_GS(line_indices);
sqrt_ID_line_points = sqrt(I_D(line_indices));

% Fit a line through the specified points
P = polyfit(V_GS_line_points, sqrt_ID_line_points, 1);

% Generate points to draw the fitted line
V_GS_line = linspace(min(V_GS_line_points), max(V_GS_line_points), 100);
sqrt_ID_line = polyval(P, V_GS_line);

% Calculate the x-axis intersection (threshold voltage, V_T)
V_T_estimate = -P(2) / P(1);

% Define the W/L ratio
WL_ratio = 31; % W/L = 31, from your provided info

% Calculate the μnCox product from the slope
u_n_C_ox_estimate = (P(1) / sqrt(0.5 * WL_ratio))^2; % dividing by sqrt(0.5 *
WL_ratio) because the slope of sqrt(ID) vs VGS in the saturation region is sqrt(0.5 *
μnCox * WL_ratio)

% Print the estimated parameters
disp(['The estimated threshold voltage (x-axis intersection of the line) is: ',
num2str(V_T_estimate), ' V']);
disp(['The estimated μnCox product (from the slope of the line) is: ',
num2str(u_n_C_ox_estimate), ' A/V^2']);

% Create a figure
figure;

% Plot sqrt(I_D) vs V_GS for this V_DS
plot(V_GS(indices), sqrt(I_D(indices)), 'LineWidth', 1.5);
hold on; % To allow multiple plots on the same figure

% Plot the line
plot(V_GS_line, sqrt_ID_line, 'r', 'LineWidth', 1.5);

% Add the equation of the line to the plot
str = sprintf('y = %.3f x + %.3f', P(1), P(2));
text(mean(V_GS_line), mean(sqrt_ID_line), str, 'Color', 'red');

% Adding labels and title
xlabel('V_{GS} (V)');
ylabel('sqrt(I_D) (sqrt(A))');
title(['Square Root of Drain Current vs. Gate-Source Voltage for V_{DS} = ',
num2str(V_DS_val), 'V']);
```

```
% Holding off the figure
hold off;
```

Q4

```
% Mohamed Ghonim - ECE 515 Fundamentals of Semiconductor Devices
% Project 2 MOSFET Characterization
% Dr. Malgorzata Chrzanowska-Jeske
clc
clear % This clears all variables
close all % This closes all figures

% Reading the data from Excel
filename = 'Id_Vd.csv'; % update with your filename
data = xlsread(filename);

% Separating the data into vectors
V_DS = data(:,1); % Drain-Source Voltage
I_D = data(:,2); % Drain Current
V_GS = data(:,3); % Gate-Source Voltage

% Extracted parameters from the saturation region
mu_nCox = 0.0075427;  % (A/V^2)
VT = 2.082; % Threshold voltage (V)
W_L = 31; % Aspect ratio W/L

% V_DS values for the model
V_DS_values = 0:0.01:3.5;

% V_GS values
V_GS_values = [2.15, 2.3, 2.45, 2.6, 2.9, 3.05, 3.2];

figure;

for i = 1:length(V_GS_values)
    % Select the data for the current V_GS value
    indices = abs(V_GS - V_GS_values(i)) < 1e-3;
    V_DS_data = V_DS(indices);
    I_D_data = I_D(indices);

    % Plot the experimental data
    scatter(V_DS_data, I_D_data, 'DisplayName', ['V_GS = ', num2str(V_GS_values(i)),
'V (Measured)']);
    hold on;

    % MOSFET I-V model for both regions
    V_DSAT = max(0, V_GS_values(i) - VT); % calculate V_DSAT
    I_D_model = mu_nCox * W_L * ((V_GS_values(i) - VT).*V_DS_values -
0.5.*V_DS_values.^2);
    I_D_model_saturation = mu_nCox * W_L * ((V_GS_values(i) - VT).*V_DSAT -
0.5.*V_DSAT.^2);

    % Limiting I_D_model to saturation region when V_DS >= V_DSAT
```

```
    I_D_model(V_DS_values >= V_DSAT) = I_D_model_saturation;

    % Plot the modeled data
    plot(V_DS_values, I_D_model, 'DisplayName', ['V_GS = ', num2str(V_GS_values(i)),
'V (Model)'], 'LineWidth', 1.5);
end

title('I_D vs V_DS for different V_GS values (Measured and Modeled)');
xlabel('V_DS (V)');
ylabel('I_D (A)');
legend('Location', 'northwest');
grid on;
```

```
% Mohamed Ghonim - ECE 515 Fundamentals of Semiconductor Devices
% Project 2 MOSFET Characterization
% Dr. Malgorzata Chrzanowska-Jeske
clc
clear % This clears all variables
close all % This closes all figures

% Reading the data from Excel
filename = 'Id_Vd.csv'; % update with your filename
data = xlsread(filename);

% Separating the data into vectors
V_DS = data(:,1); % Drain-Source Voltage
I_D = data(:,2); % Drain Current
V_GS = data(:,3); % Gate-Source Voltage

% Extracted parameters from the saturation region
mu_nCox = 0.0075427;  % (A/V^2)
VT = 2.082; % Threshold voltage (V)
W_L = 31; % Aspect ratio W/L

% V_DS values for the model
V_DS_values = 0:0.01:3.5;

% Unique V_GS values
V_GS_values = unique(V_GS);

figure;

for i = 1:length(V_GS_values)
    % Select the data for the current V_GS value
    indices = abs(V_GS - V_GS_values(i)) < 1e-3;
    V_DS_data = V_DS(indices);
    I_D_data = I_D(indices);

    % Plot the experimental data
    scatter(V_DS_data, I_D_data, 'DisplayName', ['V_GS = ', num2str(V_GS_values(i)),
'V (Measured)']);
    hold on;
```

```
    % MOSFET I-V model for both regions
    V_DSAT = max(0, V_GS_values(i) - VT); % calculate V_DSAT
    I_D_model = mu_nCox * W_L * ((V_GS_values(i) - VT).*V_DS_values -
0.5.*V_DS_values.^2);
    I_D_model_saturation = mu_nCox * W_L * ((V_GS_values(i) - VT).*V_DSAT -
0.5.*V_DSAT.^2);

    % Limiting I_D_model to saturation region when V_DS >= V_DSAT
    I_D_model(V_DS_values >= V_DSAT) = I_D_model_saturation;

    % Plot the modeled data
    plot(V_DS_values, I_D_model, 'DisplayName', ['V_GS = ', num2str(V_GS_values(i)),
'V (Model)'], 'LineWidth', 1.5);
end

title('I_D vs V_DS for different V_GS values (Measured and Modeled)');
xlabel('V_DS (V)');
ylabel('I_D (A)');
legend('Location', 'northwest');
grid on;
```

```
% Mohamed Ghonim - ECE 515 Fundamentals of Semiconductor Devices
% Project 2 MOSFET Characterization
% Dr. Malgorzata Chrzanowska-Jeske
clc
clear % This clears all variables
close all % This closes all figures

% Reading the data from Excel
filename = 'Id_Vg.csv'; % update with your filename
data = xlsread(filename);

% Separating the data into vectors
V_GS = data(:,1); % Gate-Source Voltage
I_D = data(:,2); % Drain Current
V_DS = data(:,4); % Drain-Source Voltage

% Extracted parameters from the saturation region
mu_nCox = 0.0075427;  % (A/V^2)
VT = 2.082; % Threshold voltage (V)
W_L = 31; % Aspect ratio W/L

% V_GS values for the model
V_GS_values = 0:0.01:3.5;

% Selected V_DS values
V_DS_values = [1, 1.75, 2,2.25, 2.5];

figure;

for i = 1:length(V_DS_values)
    % Select the data for the current V_DS value
    indices = abs(V_DS - V_DS_values(i)) < 1e-3;
```

```
    V_GS_data = V_GS(indices);
    I_D_data = sqrt(abs(I_D(indices)));

    % Plot the experimental data
    scatter(V_GS_data, I_D_data, 'DisplayName', ['V_DS = ', num2str(V_DS_values(i)),
'V (Measured)']);
    hold on;

    % MOSFET I-V model for both regions
    V_DSAT = max(0, V_GS_values - VT); % calculate V_DSAT
    I_D_model = mu_nCox * W_L * ((V_GS_values - VT).*V_DS_values(i) -
0.5.*V_DS_values(i).^2);
    I_D_model_saturation = mu_nCox * W_L * ((V_GS_values - VT).*V_DSAT -
0.5.*V_DSAT.^2);

    % Limiting I_D_model to saturation region when V_DS >= V_DSAT
    I_D_model(V_DS_values(i) >= V_DSAT) = I_D_model_saturation(V_DS_values(i) >=
V_DSAT);

    % Plot the modeled data
    plot(V_GS_values, sqrt(abs(I_D_model)), 'DisplayName', ['V_DS = ',
num2str(V_DS_values(i)), 'V (Model)'], 'LineWidth', 1.5);
end

title('sqrt(I_D) vs V_GS for different V_DS values (Measured and Modeled)');
xlabel('V_GS (V)');
ylabel('sqrt(I_D) (sqrt(A))');
legend('Location', 'northwest');
grid on;
```

```
% Mohamed Ghonim - ECE 515 Fundamentals of Semiconductor Devices
% Project 2 MOSFET Characterization
% Dr. Malgorzata Chrzanowska-Jeske
clc
clear % This clears all variables
close all % This closes all figures

% Reading the data from Excel
filename = 'Id_Vg.csv'; % update with your filename
data = xlsread(filename);

% Separating the data into vectors
V_GS = data(:,1); % Gate-Source Voltage
I_D = data(:,2); % Drain Current
V_DS = data(:,4); % Drain-Source Voltage

% Choosing V_DS = 2V and near threshold voltage
V_DS_selected = 2;
VT = 2.082; % Threshold voltage (V)
VT_window = 0.1; % Range around VT to consider
```

```
% Filtering data
mask = (V_DS == V_DS_selected) & (V_GS > VT - VT_window) & (V_GS < VT + VT_window);
V_GS_subthreshold = V_GS(mask);
I_D_subthreshold = I_D(mask);

% Plotting
figure
plot(V_GS_subthreshold, I_D_subthreshold, 'b', 'LineWidth', 1.5) % Line plot
hold on
scatter(V_GS_subthreshold, I_D_subthreshold, 'MarkerEdgeColor', 'r') % Scatter plot
hold off
title('Subthreshold characteristics (I_D vs V_GS for V_DS = 2V)')
xlabel('V_GS (V)')
ylabel('I_D (A)')
legend('Data trend', 'Measured data')
grid on
```

```
% Mohamed Ghonim - ECE 515 Fundamentals of Semiconductor Devices
% Project 2 MOSFET Characterization
% Dr. Malgorzata Chrzanowska-Jeske
clc
clear % This clears all variables
close all % This closes all figures

% Reading the data from Excel
filename = 'Id_Vg.csv'; % update with your filename
data = xlsread(filename);

% Separating the data into vectors
V_GS = data(:,1); % Gate-Source Voltage
I_D = data(:,2); % Drain Current
V_DS = data(:,4); % Drain-Source Voltage

% Choosing V_DS = 2V and near threshold voltage
V_DS_selected = 2;
VT = 2.082; % Threshold voltage (V)
VT_window = 0.1; % Range around VT to consider

% Filtering data
mask = (V_DS == V_DS_selected) & (V_GS > VT - VT_window) & (V_GS < VT + VT_window);
V_GS_subthreshold = V_GS(mask);
I_D_subthreshold = I_D(mask);

% Plotting
figure
semilogy(V_GS_subthreshold, I_D_subthreshold, 'b', 'LineWidth', 1.5) % Line plot
hold on
scatter(V_GS_subthreshold, log10(I_D_subthreshold), 'MarkerEdgeColor', 'r') % Scatter
plot
hold off
title('Logarithmic subthreshold characteristics (log(I_D) vs V_GS for V_DS = 2V)')
xlabel('V_GS (V)')
ylabel('log_{10}(I_D) (A)')
```

```
legend('Data trend')
grid on
```

```
% Mohamed Ghonim - ECE 515 Fundamentals of Semiconductor Devices
% Project 2 MOSFET Characterization
% Dr. Malgorzata Chrzanowska-Jeske
clc
clear % This clears all variables
close all % This closes all figures

% Reading the data from Excel
filename = 'Id_Vg.csv'; % update with your filename
data = xlsread(filename);

% Separating the data into vectors
V_GS = data(:,1); % Gate-Source Voltage
I_D = data(:,2); % Drain Current
V_DS = data(:,4); % Drain-Source Voltage

% Choosing V_DS = 2V
V_DS_selected = 2;

% Filtering data
mask = (V_DS == V_DS_selected);
V_GS_all = V_GS(mask);
I_D_all = I_D(mask);

% Plotting
figure
semilogy(V_GS_all, I_D_all, 'b', 'LineWidth', 1.5) % Line plot
hold on
scatter(V_GS_all, log10(I_D_all), 'MarkerEdgeColor', 'r') % Scatter plot
hold off
title('Logarithmic I_D vs V_GS for V_DS = 2V')
xlabel('V_GS (V)')
ylabel('log_{10}(I_D) (A)')
legend('Data trend')
grid on
```

```
% Mohamed Ghonim - ECE 515 Fundamentals of Semiconductor Devices
% Project 2 MOSFET Characterization
% Dr. Malgorzata Chrzanowska-Jeske
clc
clear % This clears all variables
close all % This closes all figures

% Reading the data from Excel
filename = 'Id_Vg.csv'; % update with your filename
data = xlsread(filename);

% Separating the data into vectors
```

```
V_GS = data(:,1); % Gate-Source Voltage
I_D = data(:,2); % Drain Current
V_DS = data(:,4); % Drain-Source Voltage

% Select data for V_DS=2V
V_DS_target = 2;
indices = find(V_DS == V_DS_target);
V_GS_target = V_GS(indices);
I_D_target = I_D(indices);

% Threshold voltage
VT = 2.082;

% Margin around V_T
margin = 0.2;

% Select data around V_T
indices_VT = find((V_GS_target >= VT - margin) & (V_GS_target <= VT + margin));
V_GS_target_VT = V_GS_target(indices_VT);
I_D_target_VT = I_D_target(indices_VT);

% Plot log10(I_D) vs V_GS
figure
scatter(V_GS_target_VT, log10(I_D_target_VT), 'filled');
hold on;

% Fitting line
P = polyfit(V_GS_target_VT, log10(I_D_target_VT), 1);
fit_line = polyval(P, V_GS_target_VT);
plot(V_GS_target_VT, fit_line, 'LineWidth', 1.5);

% Compute Subthreshold Slope
S = P(1);

% Boltzmann constant in J/K
k = 1.38e-23;

% Charge of an electron in C
q = 1.6e-19;

% Room temperature in K
T = 300;

% Calculate c_r
c_r = S * q / (2.3 * k * T);

fprintf('Subthreshold slope, S = %.4f V/decade\n', S);
fprintf('The parameter c_r = %.4e\n', c_r);

% Equation of the fitted line
m = S;
c = P(2);
str = sprintf('y = %.2fx + %.2f', m, c);

% Add equation to the plot
```

```
text(VT, min(log10(I_D_target_VT)), str, 'FontSize', 12);
hold off;

xlabel('V_G_S (V)');
ylabel('log_{10}(I_D) ');
title('log_{10}(I_D) vs V_G_S for V_D_S=2V around V_T');
legend('Measured data', 'Fitted line', 'Location', 'northwest');
```

```
% Mohamed Ghonim - ECE 515 Fundamentals of Semiconductor Devices
% Project 2 MOSFET Characterization
% Dr. Malgorzata Chrzanowska-Jeske
clc
clear % This clears all variables
close all % This closes all figures

% Reading the data from Excel
filename = 'Id_Vg.csv'; % update with your filename
data = xlsread(filename);

% Separating the data into vectors
V_GS = data(:,1); % Gate-Source Voltage
I_D = data(:,2); % Drain Current
V_DS = data(:,4); % Drain-Source Voltage

% Select data for V_DS=2V
V_DS_target = 2;
indices = find(V_DS == V_DS_target);
V_GS_target = V_GS(indices);
I_D_target = I_D(indices);

% Threshold voltage
VT = 0.8;

% Margin around V_T
margin = 0.2;

% Select data around V_T
indices_VT = find((V_GS_target >= VT - margin) & (V_GS_target <= VT + margin));
V_GS_target_VT = V_GS_target(indices_VT);
I_D_target_VT = I_D_target(indices_VT);

% Plot log10(I_D) vs V_GS
figure
scatter(V_GS_target_VT, log10(I_D_target_VT), 'filled');
hold on;

% Fitting line
P = polyfit(V_GS_target_VT, log10(I_D_target_VT), 1);
fit_line = polyval(P, V_GS_target_VT);
plot(V_GS_target_VT, fit_line, 'LineWidth', 1.5);

% Compute Subthreshold Slope
S = 1 / P(1); % Corrected computation of S
```

```matlab
% Boltzmann constant in J/K
k = 1.38e-23;

% Charge of an electron in C
q = 1.6e-19;

% Room temperature in K
T = 300;

% Calculate c_r
c_r = S * q / (2.3 * k * T);

fprintf('Subthreshold slope, S = %.4f V/decade\n', S);
fprintf('The parameter c_r = %.4e\n', c_r);

% Equation of the fitted line
m = P(1); % m here is the slope of the line, not S.
c = P(2);
str = sprintf('y = %.2fx + %.2f', m, c);

% Add equation to the plot
text(VT, min(log10(I_D_target_VT)), str, 'FontSize', 12);
hold off;

xlabel('V_G_S (V)');
ylabel('log_{10}(I_D) ');
title('log_{10}(I_D) vs V_G_S for V_D_S=2V around V_T');
legend('Measured data', 'Fitted line', 'Location', 'northwest');
```