Electrical Engineering Department,

Faculty of Engineering,

Alexandria University.

**MATLAB Project**

**For April 2019**

|  |  |
| --- | --- |
| **Names** | **Seat Numbers** |
| **اية سعيد ابراهيم محمد** | **1** |
| **احمد محمد حمدان محمود** | **31** |
| **بسمة ممدوح جابر حنفي** | **57** |
| **بسنت اسلام عبدالحميد السيد** | **58** |
| **علاء عبدالوهاب سعيد** | **106** |
| **عمر احمد علي حجازي** | **111** |
| **محمد مجدي السيد الورداني** | **166** |
| **محمد مصطفى حلمي** | **169** |

# **Part I:**

## ***A)Power Method:***

### MATLAB Code****:****

function [n]=largest\_eigen\_value( A,k )

x = [1;0];

for i = 0 : k-1

p = A \* x;

n = max(p);

x= p/ n;

end

end

### Code Testing with eig() function:

## ***B)Least Squares Method:***

### MATLAB Code:

function [a0,a1]=No\_2\_Least\_Squares\_Method(x,y)

X = [ones(length(x),1) x]; %first column treated as all ones since x\_1=1

Y = y; %column vector for proper dimension during multiplication

a = inv(X'\*X)\*X'\*Y; % Least Squares Estimator - equivalent code X\y

plot ( x , y , '\*'); % original data

a0=a(1);%intersection of the line

a1=a(2);%slope of the line

hold on;

plot( x , a0+a1\*x , 'r-' ); %Fitted line

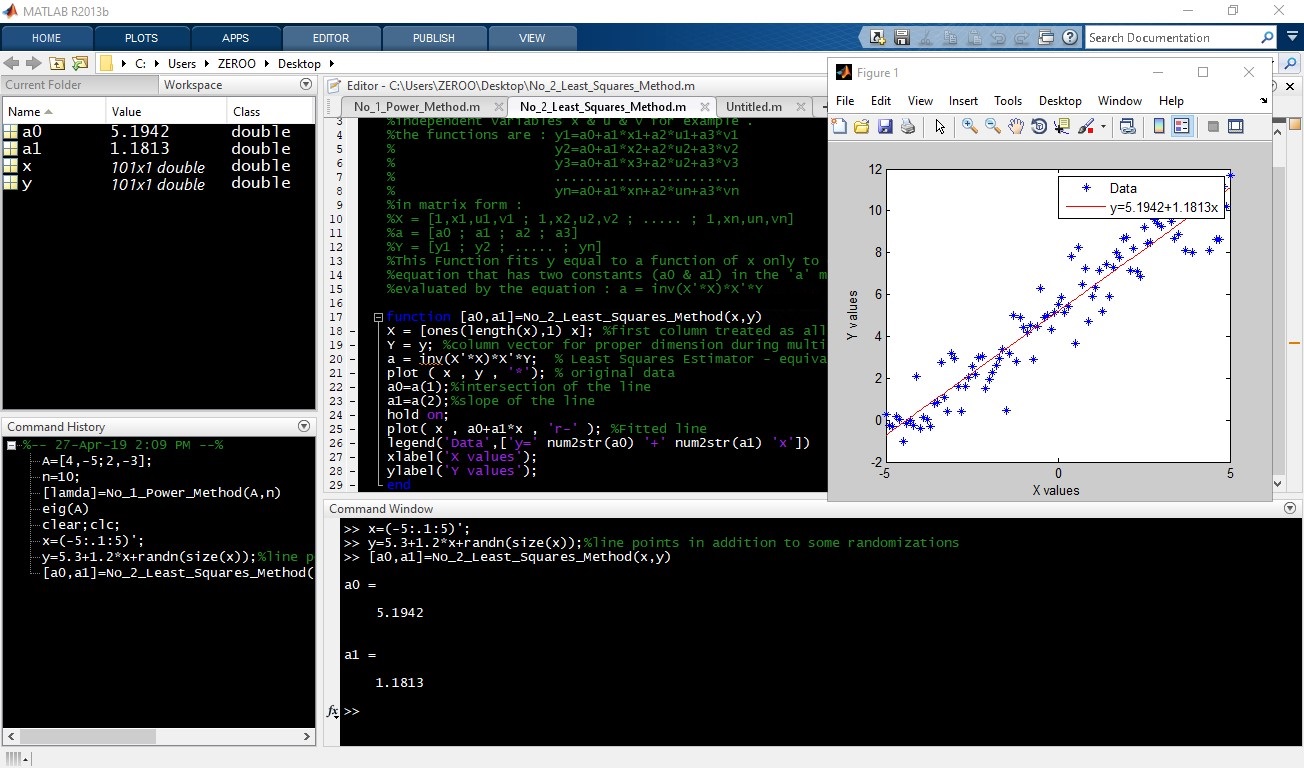
legend('Data',['y=' num2str(a0) '+' num2str(a1) 'x'])

xlabel('X values');

ylabel('Y values');

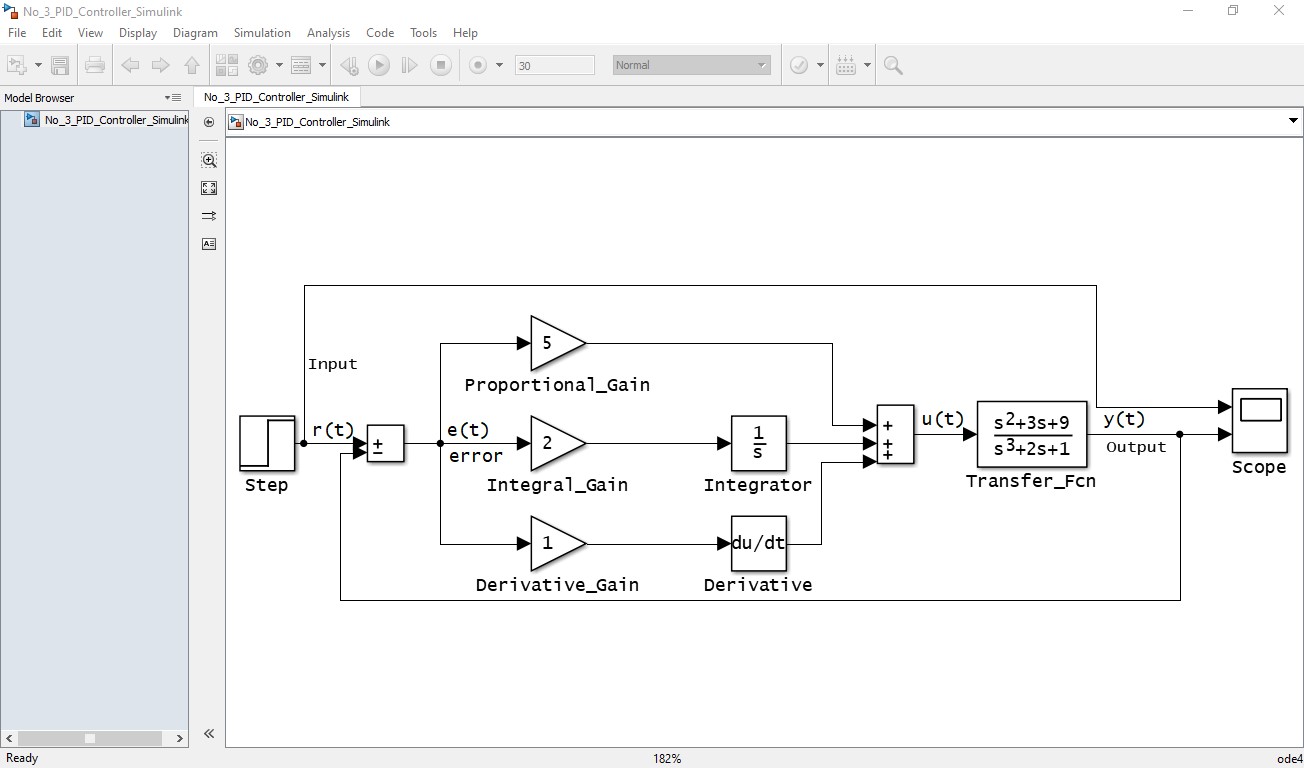
end

### Code Testing With The Plot Diagram:

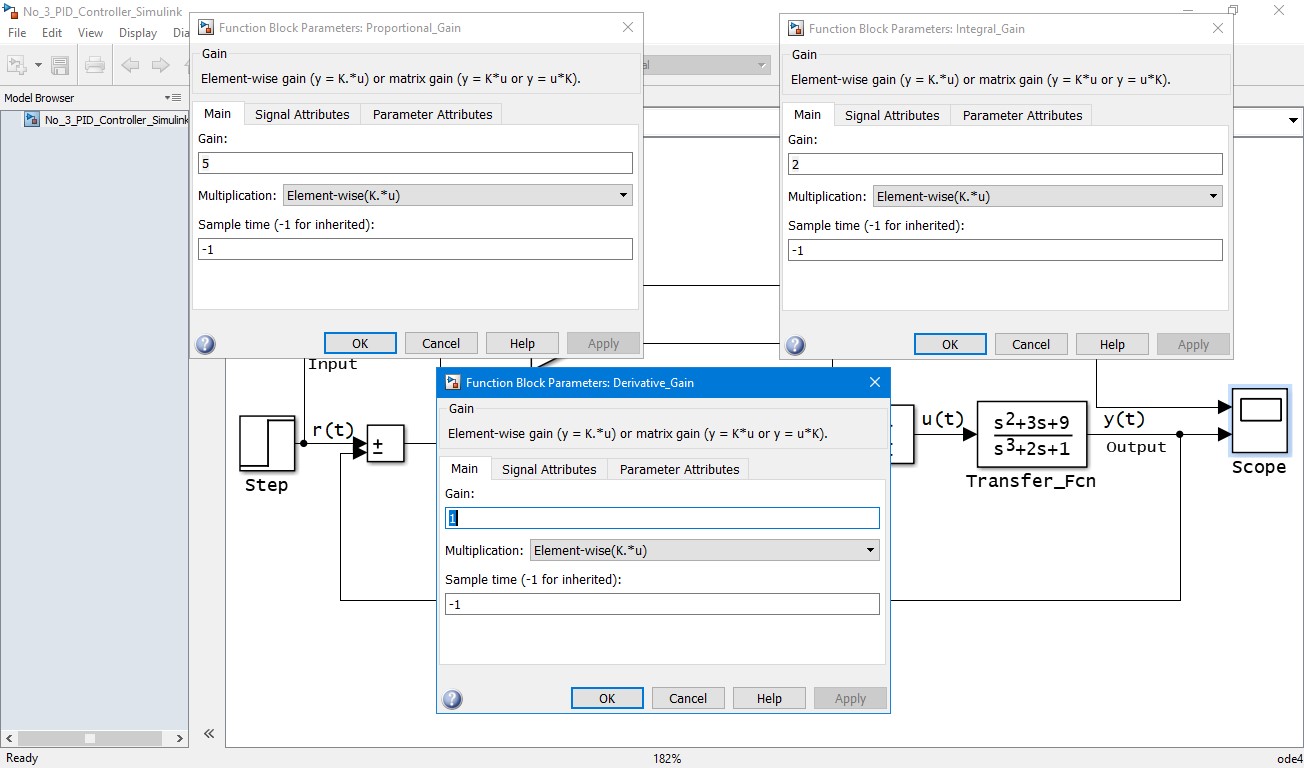


# **Part 2:**

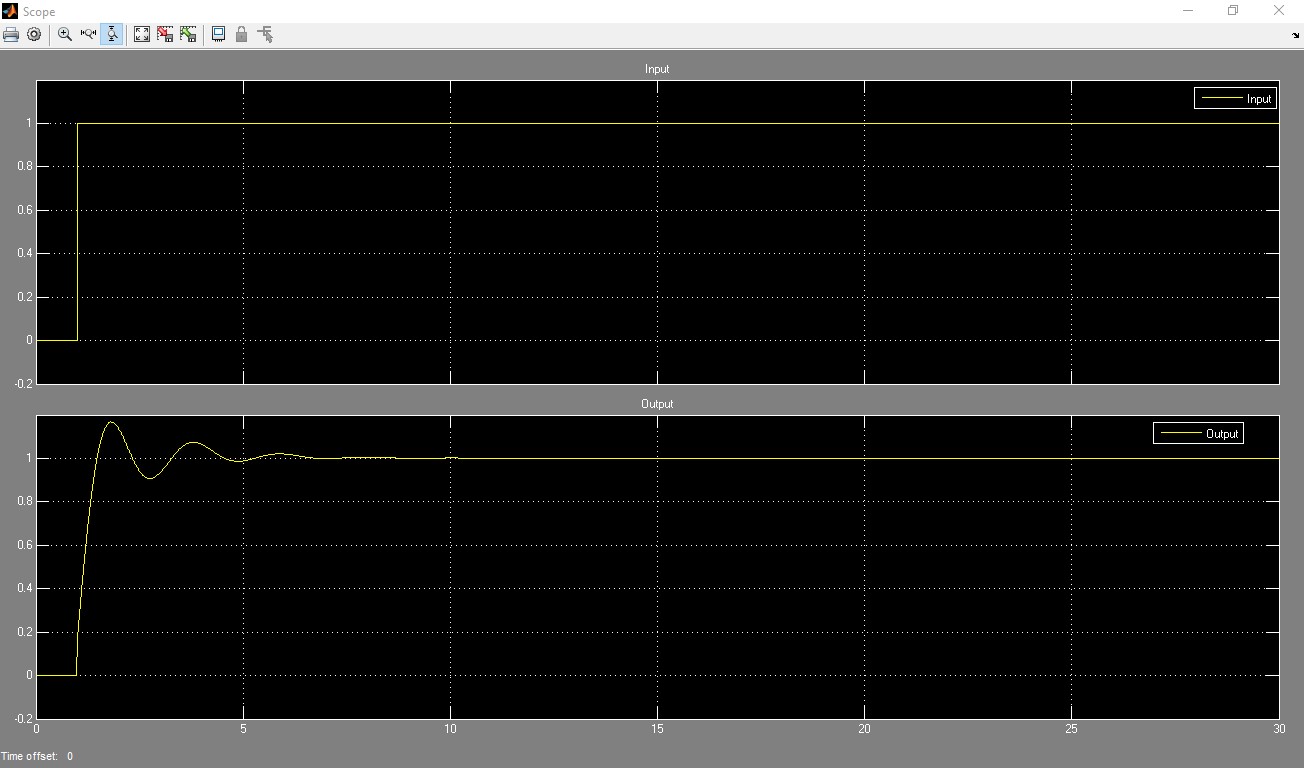
### Simulink Model:



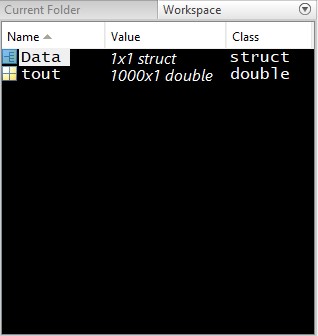
### Modified Simulink Parameter:



**Figure:**

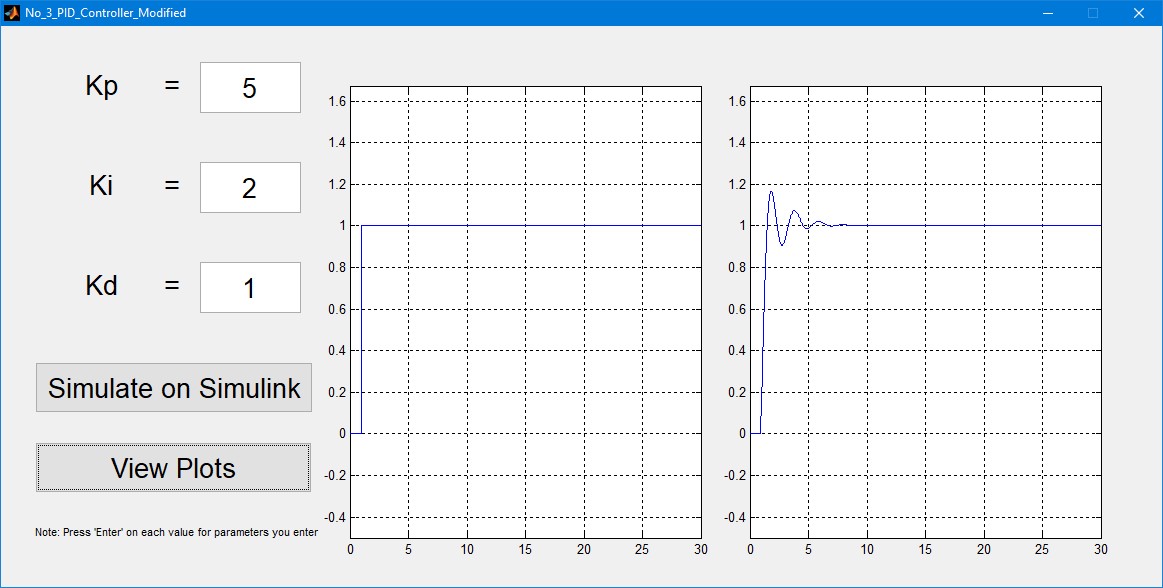


**Workspace:**

****

# ****Bonus:****

**GUI:**



**edited parts of the associated code:**

Function Change No. 1:

function No\_3\_PID\_Controller\_Modified\_OpeningFcn(hObject, eventdata, handles, varargin)

% This function has no output args, see OutputFcn.

% hObject handle to figure

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

% varargin command line arguments to untitled1 (see VARARGIN)

%%% %%%

%%% Start Changes %%%

%%% %%%

model\_open(handles)

% Choose default command line output for final

handles.output = hObject;

% Update handles structure

guidata(hObject, handles);

% Now we can use the figure, as required.

% Set model parameters to match GUI settings

model\_open(handles)

%%% %%%

%%% End Changes %%%

%%% %%%

% Choose default command line output for No\_3\_PID\_Controller\_Modified

handles.output = hObject;

% Update handles structure

guidata(hObject, handles);

% --- Outputs from this function are returned to the command line.

function varargout = No\_3\_PID\_Controller\_Modified\_OutputFcn(hObject, eventdata, handles)

% varargout cell array for returning output args (see VARARGOUT);

% hObject handle to figure

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure

varargout{1} = handles.output;

%end function No\_3\_PID\_Controller\_Modified

%%% %%%

%%% Start Changes %%%

%%% %%%

% Ensure that the Simulink model is open

function model\_open(handles)

% Make sure the diagram is still open

if isempty(find\_system('Name','No\_3\_PID\_Controller\_Simulink')),

% check whether our Simulink model is opened or not

open\_system('No\_3\_PID\_Controller\_Simulink');

end

%endfunction model\_open

%%% %%%

%%% End Changes %%%

%%% %%%

Function Change No 2:

function edit1\_Callback(hObject, eventdata, handles)

% hObject handle to edit1 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

% Ensure model is open

model\_open(handles)

% Get the new value

kp\_NewStrVal = get(hObject,'String');

kp\_NewVal = str2double(kp\_NewStrVal);

% Set the Gain parameter of the Kp Gain Block to the new value

set\_param('No\_3\_PID\_Controller\_Simulink/Proportional\_Gain','Gain',kp\_NewStrVal);

Function Change No 3:

function edit2\_Callback(hObject, eventdata, handles)

% hObject handle to edit2 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

% Ensure model is open

model\_open(handles)

% Get the new value

ki\_NewStrVal = get(hObject,'String');

ki\_NewVal = str2double(ki\_NewStrVal);

% Set the Gain parameter of the Ki Gain Block to the new value

set\_param('No\_3\_PID\_Controller\_Simulink/Integral\_Gain','Gain',ki\_NewStrVal);

Function Change No 4:

function edit3\_Callback(hObject, eventdata, handles)

% hObject handle to edit3 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

% Ensure model is open

model\_open(handles)

% Get the new value

kd\_NewStrVal = get(hObject,'String');

kd\_NewVal = str2double(kd\_NewStrVal);

% Set the Gain parameter of the Kd Gain Block to the new value

set\_param('No\_3\_PID\_Controller\_Simulink/Derivative\_Gain','Gain',kd\_NewStrVal);

Function Change No 4:

function simulatebutton\_Callback(hObject, eventdata, handles)

% hObject handle to simulatebutton (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

myfunc()

function myfunc

a = sim('No\_3\_PID\_Controller\_Simulink','SimulationMode','normal');

b = a.get('Data');

assignin('base','Data',b);

% --- Executes on button press in plotbutton.

function plotbutton\_Callback(hObject, eventdata, handles)

sim('No\_3\_PID\_Controller\_Simulink');

axes(handles.axes1)

x=Data.time;

y1=Data.signals(1).values;

y2=Data.signals(2).values;

plot(x,y1);

grid on;

axis([min(x) max(x) min(y2)-0.5 max(y2)+0.5]);

axes(handles.axes2)

x=Data.time;

y1=Data.signals(1).values;

y2=Data.signals(2).values;

plot(x,y2);

grid on;

axis([min(x) max(x) min(y2)-0.5 max(y2)+0.5]);