**SECTION 1**

THE CODE USED IN THE PROJECT IS AS FOLLOWS:

f=input(‘ enter frequency’);

fs=input(‘ enter sample frequency’);

t=0:0.1:100;

ts=0:10:1000;

x=sin(2.\*3.14.\*f.\*t);

xsq=x.^2;

Energy\_c=trapz(t,abs(xsq))

subplot(3,1,1)

plot(t,x);

xlabel(‘time’);

ylabel(‘sin(X)’);

title(‘graph of continuous signal’)

y=sin(2.\*3.14.\*f.\*ts./fs);

ysq=y.^2;

Energy\_d=sum(ysq)

subplot(3,1,2);

stem(ts,y)

xlabel(‘time’);

ylabel(‘sin(Y)’);

title(‘graph of sampled discrete signal’)

[x, y] = meshgrid(-1.5:0.1:1.5 , -1.5:0.1:1.5);

z = x.^2 – y.^2;

subplot(3,1,3)

plot3(x,y,z),grid on

**SECTION 2**

GUI CODE USED:

function varargout = test(varargin)

% TEST M-file for test.fig

% TEST, by itself, creates a new TEST or raises the existing

% singleton\*.

%

% H = TEST returns the handle to a new TEST or the handle to

% the existing singleton\*.

%

% TEST(‘CALLBACK’,hObject,eventData,handles,…) calls the local

% function named CALLBACK in TEST.M with the given input arguments.

%

% TEST(‘Property’,’Value’,…) creates a new TEST or raises the

% existing singleton\*. Starting from the left, property value pairs are

% applied to the GUI before test\_OpeningFcn gets called. An

% unrecognized property name or invalid value makes property application

% stop. All inputs are passed to test\_OpeningFcn via varargin.

%

% \*See GUI Options on GUIDE’s Tools menu. Choose “GUI allows only one

% instance to run (singleton)”.

%

% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help test

% Last Modified by GUIDE v2.5 22-May-2018 00:12:24

% Begin initialization code – DO NOT EDIT

gui\_Singleton = 1;

gui\_State = struct(‘gui\_Name’, mfilename, …

‘gui\_Singleton’, gui\_Singleton, …

‘gui\_OpeningFcn’, @test\_OpeningFcn, …

‘gui\_OutputFcn’, @test\_OutputFcn, …

‘gui\_LayoutFcn’, [] , …

‘gui\_Callback’, []);

if nargin && ischar(varargin{1})

gui\_State.gui\_Callback = str2func(varargin{1});

end

if nargout

[varargout{1:nargout}] = gui\_mainfcn(gui\_State, varargin{:});

else

gui\_mainfcn(gui\_State, varargin{:});

end

% End initialization code – DO NOT EDIT

% --- Executes just before test is made visible.

Function test\_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 test (see VARARGIN)

% Choose default command line output for test

handles.output = hObject;

% Update handles structure

guidata(hObject, handles);

% UIWAIT makes test wait for user response (see UIRESUME)

% uiwait(handles.figure1);

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

Function varargout = test\_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;

function frequency\_Callback(hObject, eventdata, handles)

% hObject handle to frequency (see GCBO)

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

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

% Hints: get(hObject,’String’) returns contents of frequency as text

% str2double(get(hObject,’String’)) returns contents of frequency as a double

% --- Executes during object creation, after setting all properties.

Function frequency\_CreateFcn(hObject, eventdata, handles)

% hObject handle to frequency (see GCBO)

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

% handles empty – handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.

% See ISPC and COMPUTER.

If ispc && isequal(get(hObject,’BackgroundColor’), get(0,’defaultUicontrolBackgroundColor’))

set(hObject,’BackgroundColor’,’white’);

end

function sample\_freq\_Callback(hObject, eventdata, handles)

% hObject handle to sample\_freq (see GCBO)

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

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

% Hints: get(hObject,’String’) returns contents of sample\_freq as text

% str2double(get(hObject,’String’)) returns contents of sample\_freq as a double

% --- Executes during object creation, after setting all properties.

Function sample\_freq\_CreateFcn(hObject, eventdata, handles)

% hObject handle to sample\_freq (see GCBO)

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

% handles empty – handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.

% See ISPC and COMPUTER.

If ispc && isequal(get(hObject,’BackgroundColor’), get(0,’defaultUicontrolBackgroundColor’))

set(hObject,’BackgroundColor’,’white’);

end

% --- Executes on button press in continuous.

Function continuous\_Callback(hObject, eventdata, handles)

% hObject handle to continuous (see GCBO)

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

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

t=0:0.1:100;

f = str2double(get(handles.frequency,’string’));

X=sin(2.\*3.14.\*f.\*t);

set(handles.continuous,’string’,plot(t,X)),grid on

Xsq=X.^2;

Energy\_c=trapz(t,abs(Xsq));

set(handles.E\_c,’string’,Energy\_c)

xlabel(‘time’);

ylabel(‘sin(X)’);

title(‘graph of continuous signal’)

% --- Executes on button press in discrete.

Function discrete\_Callback(hObject, eventdata, handles)

% hObject handle to discrete (see GCBO)

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

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

ts=0:10:1000;

f = str2double(get(handles.frequency,’string’));

fs = str2double(get(handles.sample\_freq,’string’));

Y=sin(2.\*3.14.\*f.\*ts./fs);

set(handles.discrete,’string’,stem(ts,Y)), grid on

Ysq=Y.^2;

Energy\_d=sum(Ysq);

set(handles.E\_d,’string’,Energy\_d)

xlabel(‘time’);

ylabel(‘sin(Y)’);

title(‘graph of sampled discrete signal’)

% --- Executes on button press in discrete\_3d.

function discrete\_3d\_Callback(hObject, eventdata, handles)

% hObject handle to discrete\_3d (see GCBO)

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

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

% ts=0:10:1000;

% f = str2double(get(handles.frequency,’string’));

% fs = str2double(get(handles.sample\_freq,’string’));

[x,y]=meshgrid(-1.5:0.1:1.5, -1.5:0.1:1.5);

z=x.^2-y.^2;

set(handles.discrete\_3d,’string’,stem3(x,y,z)), grid on

title(‘graph of sampled discrete signal’)

% --- Executes on button press in continuous\_3d.

function continuous\_3d\_Callback(hObject, eventdata, handles)

% hObject handle to continuous\_3d (see GCBO)

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

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

% t=0:0.1:100;

% f = str2double(get(handles.frequency,’string’));

[x,y]=meshgrid(-1.5:0.1:1.5, -1.5:0.1:1.5);

z=x.^2-y.^2;

set(handles.continuous\_3d,’string’,plot3(x,y,z)), grid on

title(‘graph of continuous signal’)