```
function varargout = eeg_gui(varargin)
% EEG_GUI MATLAB code for eeg_gui.fig
%      EEG_GUI, by itself, creates a new EEG_GUI or raises the existing
%      singleton*.
%
%      H = EEG_GUI returns the handle to a new EEG_GUI or the handle to
%      the existing singleton*.
%
%      EEG_GUI('CALLBACK',hObject,eventData,handles,...) calls the local
%      function named CALLBACK in EEG_GUI.M with the given input arguments.
%
%      EEG_GUI('Property','Value',...) creates a new EEG_GUI or raises the
%      existing singleton*.  Starting from the left, property value pairs are
%      applied to the GUI before eeg_gui_OpeningFcn gets called.  An
%      unrecognized property name or invalid value makes property application
%      stop.  All inputs are passed to eeg_gui_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 eeg_gui

% Last Modified by GUIDE v2.5 17-Feb-2014 23:59:49

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @eeg_gui_OpeningFcn, ...
                   'gui_OutputFcn',  @eeg_gui_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 eeg_gui is made visible.
function eeg_gui_OpeningFcn(hObject, ~, 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 eeg_gui (see VARARGIN)

% Choose default command line output for eeg_gui
```

```
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes eeg_gui wait for user response (see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
function varargout = eeg_gui_OutputFcn(~, ~, 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;
varargout{2} = handles.n_el;
varargout{3} = handles.n_zeit;
varargout{4} = handles.radio;


function n_el_Callback(hObject, ~, handles)
% hObject    handle to n_el (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global n_el
% Hints: get(hObject,'String') returns contents of n_el as text
%        str2double(get(hObject,'String')) returns contents of n_el as a double
n_el = str2num(get(handles.n_el,'String'));
handles.n_el = n_el;
guidata(hObject,handles.n_el);




function n_zeit_Callback(hObject, ~, handles)
% hObject    handle to n_zeit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global n_zeit
% Hints: get(hObject,'String') returns contents of n_zeit as text
%        str2double(get(hObject,'String')) returns contents of n_zeit as a double
n_zeit = str2num(get(handles.n_zeit,'String'));
handles.n_zeit = n_zeit;
guidata(hObject,handles.n_zeit);




% --- Executes on slider movement.
function slider_min_Callback(hObject, eventdata, handles)
% hObject    handle to slider_min (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,'Value') returns position of slider
%        get(hObject,'Min') and get(hObject,'Max') to determine range


% --- Executes during object creation, after setting all properties.
function slider_min_CreateFcn(hObject, ~, handles)
% hObject    handle to slider_min (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end


% --- Executes on slider movement.
function slider_max_Callback(hObject, eventdata, handles)
% hObject    handle to slider_max (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,'Value') returns position of slider
%        get(hObject,'Min') and get(hObject,'Max') to determine range of slider


% --- Executes during object creation, after setting all properties.
function slider_max_CreateFcn(hObject, eventdata, handles)
% hObject    handle to slider_max (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end


% --- Executes on button press in video_speichern_button.
function video_speichern_button_Callback(hObject, eventdata, handles)
% hObject    handle to video_speichern_button (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

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


function axes1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to axes1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called
x = intervallx;
```

```
y = sin(x);
plot(x,y);
% Hint: place code in OpeningFcn to populate axes1


% --- Executes on button press in sin_k.
function sin_k_Callback(hObject, eventdata, handles)
% hObject    handle to sin_k (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of sin_k


% --- Executes on button press in trig_senk.
function trig_senk_Callback(hObject, eventdata, handles)
% hObject    handle to trig_senk (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)


% Hint: get(hObject,'Value') returns toggle state of trig_senk


% --- Executes on button press in trig_zuf.
function trig_zuf_Callback(hObject, eventdata, handles)
% hObject    handle to trig_zuf (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% Hint: get(hObject,'Value') returns toggle state of trig_zuf


% --- Executes when selected object is changed in uipanel2.
function uipanel2_SelectionChangeFcn(hObject, ~, handles)

global typ

if hObject == handles.sin_k;

    typ = '1_sin_kreis';
    handles.radio = typ;
    guidata(hObject,handles.radio);

elseif hObject == handles.trig_senk

    typ = '2_trigamp_senkmod';
    handles.radio = typ;
    guidata(hObject,handles.radio);

elseif hObject == handles.trig_zuf

    typ = '3_trigamp_zufmod';
    handles.radio = typ;
    guidata(hObject,handles.radio);
```

```
end
```