APPLICATION CODE WITH GUI COMPONENTS

OMAR IBRAHIM	7442
MAZEN NAGY	7475

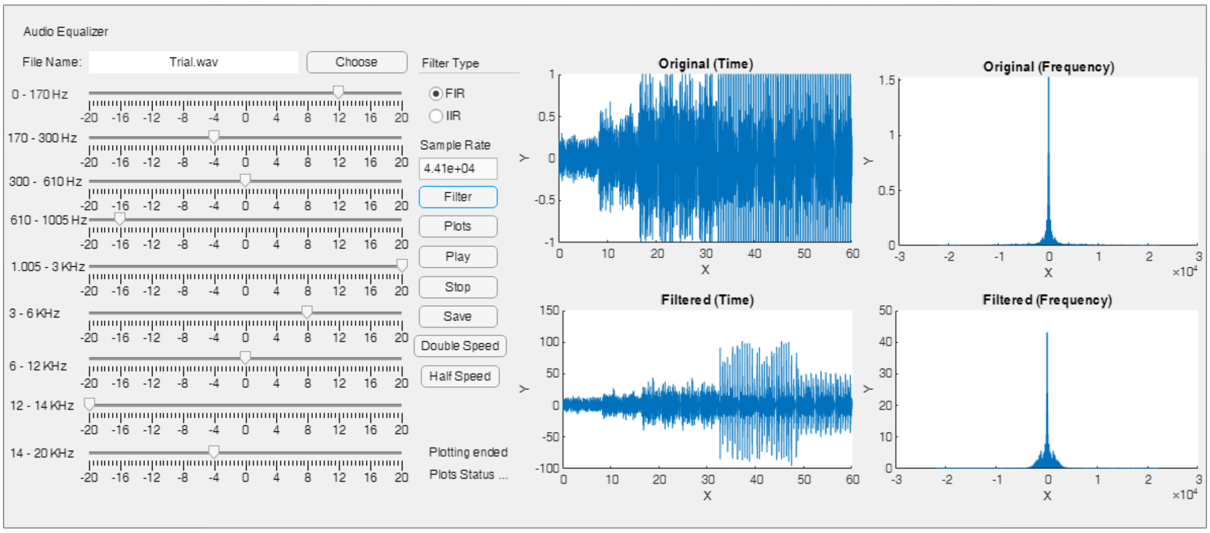
```
classdef app1 < matlab.apps.AppBase
    % Properties that correspond to app components
    properties (Access = public)
        UIFigure                  matlab.ui.Figure
        sampleRateInput           matlab.ui.control.NumericEditField
        SampleRateEditFieldLabel  matlab.ui.control.Label
        halfButton                matlab.ui.control.Button
        doubleButton              matlab.ui.control.Button
        plotsStatus               matlab.ui.control.Label
        filterStatus              matlab.ui.control.Label
        PlotsButton               matlab.ui.control.Button
        SaveButton                matlab.ui.control.Button
        StopButton                matlab.ui.control.Button
        PlayButton                matlab.ui.control.Button
        FilterButton              matlab.ui.control.Button
        FilterTypeButtonGroup     matlab.ui.container.ButtonGroup
        IIR                       matlab.ui.control.RadioButton
        FIR                       matlab.ui.control.RadioButton
        bandSlider9               matlab.ui.control.Slider
        KHzLabel_2                matlab.ui.control.Label
        bandSlider8               matlab.ui.control.Slider
        KHzLabel                  matlab.ui.control.Label
        bandSlider7               matlab.ui.control.Slider
        KHzSlider_3Label          matlab.ui.control.Label
        bandSlider6               matlab.ui.control.Slider
        KHzSlider_2Label          matlab.ui.control.Label
        bandSlider5               matlab.ui.control.Slider
        HzLabel                   matlab.ui.control.Label
        bandSlider4               matlab.ui.control.Slider
        HzLabel_4                 matlab.ui.control.Label
        bandSlider3               matlab.ui.control.Slider
        HzLabel_3                 matlab.ui.control.Label
        bandSlider2               matlab.ui.control.Slider
        HzLabel_2                 matlab.ui.control.Label
        title                     matlab.ui.control.Label
        bandSlider1               matlab.ui.control.Slider
        HzSlider_2Label           matlab.ui.control.Label
        chooseFileButton          matlab.ui.control.Button
        currentFile               matlab.ui.control.Label
        FileNameLabel             matlab.ui.control.Label
        filteredPlot_2            matlab.ui.control.UIAxes
        originalPlot_2            matlab.ui.control.UIAxes
        filteredPlot_1            matlab.ui.control.UIAxes
        originalPlot_1            matlab.ui.control.UIAxes
    end


    properties (Access = public)
        file_name
        file_path
        filter_type = 0;
        bands = ones(1, 9);
        sample_rate = 0;
        audio_data;
        default_audio_data;
        default_sample_rate = 0;
        filtered_audio_data;
    end

    methods (Access = public)

        function y = apply_iir_filters(app,data,fs,gain)
            Frequencies = [170 300 610 1005 3000 6000 12000 14000 20000];
            f = fs/2;
            order_iir = 4;
            [b1 a1] = butter(order_iir, Frequencies(1)/f);
            [b2 a2] = butter(order_iir, [Frequencies(1) Frequencies(2)]/f);
            [b3 a3] = butter(order_iir, [Frequencies(2) Frequencies(3)]/f);
            [b4 a4] = butter(order_iir, [Frequencies(3) Frequencies(4)]/f);
            [b5 a5] = butter(order_iir, [Frequencies(4) Frequencies(5)]/f);
            [b6 a6] = butter(order_iir, [Frequencies(5) Frequencies(6)]/f);
            [b7 a7] = butter(order_iir, [Frequencies(6) Frequencies(7)]/f);
            [b8 a8] = butter(order_iir, [Frequencies(7) Frequencies(8)]/f);
            [b9 a9] = butter(order_iir, [Frequencies(8) Frequencies(9)]/f);

            %low pass filter with cutoff frequency of 170Hz
            dummy_data = filter(b1,a1,data);
            composite_data = dummy_data * gain(1);

            %band pass filter 170 - 300 Hz
            dummy_data = filter(b2,a2,data);
            composite_data = composite_data + dummy_data * gain(2);

            %band pass filter 300 - 610 Hz
            dummy_data = filter(b3,a3,data);
            composite_data = composite_data + dummy_data * gain(3);

            %band pass filter 610 - 1005 Hz
            dummy_data = filter(b4,a4,data);
            composite_data = composite_data + dummy_data * gain(4);

            %band pass filter 1005 - 3k Hz
            dummy_data = filter(b5,a5,data);
            composite_data = composite_data + dummy_data * gain(5);

            %band pass filter 3k - 6k Hz
            dummy_data = filter(b6,a6,data);
            composite_data = composite_data + dummy_data * gain(6);
            %band pass filter 6k - 12k Hz
            dummy_data = filter(b7,a7,data);
            composite_data = composite_data + dummy_data * gain(7);

            %band pass filter 12k - 14k Hz
            dummy_data = filter(b8,a8,data);
            composite_data = composite_data + dummy_data * gain(8);

            %band pass filter 14k - 20k Hz
            dummy_data = filter(b9,a9,data);
            composite_data = composite_data + dummy_data * gain(9);

            y = composite_data;


        end

        function y = apply_fir_filters(app,data,fs,gain)
            Frequencies = [170 300 610 1005 3000 6000 12000 14000 20000];
            f = fs/2;
            order_fir = 50;
            b1 = fir1(order_fir, Frequencies(1)/f);
            b2 = fir1(order_fir, [Frequencies(1) Frequencies(2)]/f);
            b3 = fir1(order_fir, [Frequencies(2) Frequencies(3)]/f);
            b4 = fir1(order_fir, [Frequencies(3) Frequencies(4)]/f);
            b5 = fir1(order_fir, [Frequencies(4) Frequencies(5)]/f);
            b6 = fir1(order_fir, [Frequencies(5) Frequencies(6)]/f);
            b7 = fir1(order_fir, [Frequencies(6) Frequencies(7)]/f);
            b8 = fir1(order_fir, [Frequencies(7) Frequencies(8)]/f);
            b9 = fir1(order_fir, [Frequencies(8) Frequencies(9)]/f);
            %low pass filter with cutoff frequency of 170Hz
            dummy_data = filter(b1,1,data);
            composite_data = dummy_data * gain(1);

            %band pass filter 170 - 300 Hz
            dummy_data = filter(b2,1,data);
            composite_data = composite_data + dummy_data * gain(2);

            %band pass filter 300 - 610 Hz
            dummy_data = filter(b3,1,data);
            composite_data = composite_data + dummy_data * gain(3);

            %band pass filter 610 - 1005 Hz
            dummy_data = filter(b4,1,data);
            composite_data = composite_data + dummy_data * gain(4);

            %band pass filter 1005 Hz - 3k Hz
            dummy_data = filter(b5,1,data);
            composite_data = composite_data + dummy_data * gain(5);

            %band pass filter 3k - 6k Hz
            dummy_data = filter(b6,1,data);
            composite_data = composite_data + dummy_data * gain(6);

            %band pass filter 6k - 12k Hz
            dummy_data = filter(b7,1,data);
            composite_data = composite_data + dummy_data * gain(7);

            %band pass filter 12k - 14k Hz
            dummy_data = filter(b8,1,data);
            composite_data = composite_data + dummy_data * gain(8);

            %band pass filter 14k - 20k Hz
            dummy_data = filter(b9,1,data);
            composite_data = composite_data + dummy_data * gain(9);

            y = composite_data;

        end

    end

    % Callbacks that handle component events
    methods (Access = private)

        % Button pushed function: chooseFileButton
        function chooseFileButtonPushed(app, event)
            [f_name,f_path] = uigetfile({'*.wav'},'File Selector');
            if(f_name ~= 0)
                app.file_name = f_name;
                app.file_path = strcat(f_path, f_name);
                [app.default_audio_data, app.default_sample_rate] =
audioread(app.file_path); % reads the audio data
                app.sampleRateInput.Value = app.default_sample_rate; % shows
the file sample rate in the input box
                app.sample_rate = app.default_sample_rate;
                app.audio_data = app.default_audio_data;
                app.currentFile.Text = app.file_name; % updates current file
            end
        end

        % Selection changed function: FilterTypeButtonGroup
        function FilterTypeButtonGroupSelectionChanged(app, event)
            selectedButton = app.FilterTypeButtonGroup.SelectedObject;
            if(selectedButton == app.FIR)
                app.filter_type = 0;
            else
```

```
                app.filter_type = 1;
            end
        end

        % Value changed function: bandSlider1
        function bandSlider1ValueChanged(app, event)
            value = app.bandSlider1.Value;
            app.bands(1) = 10^(value/10);
        end

        % Value changed function: bandSlider2
        function bandSlider2ValueChanged(app, event)
            value = app.bandSlider2.Value;
            app.bands(2) = 10^(value/10);
        end

        % Value changed function: bandSlider3
        function bandSlider3ValueChanged(app, event)
            value = app.bandSlider3.Value;
            app.bands(3) = 10^(value/10);
        end

        % Value changed function: bandSlider4
        function bandSlider4ValueChanged(app, event)
            value = app.bandSlider4.Value;
            app.bands(4) = 10^(value/10);
        end

        % Value changed function: bandSlider5
        function bandSlider5ValueChanged(app, event)
            value = app.bandSlider5.Value;
            app.bands(5) = 10^(value/10);
        end

        % Value changed function: bandSlider6
        function bandSlider6ValueChanged(app, event)
            value = app.bandSlider6.Value;
            app.bands(6) = 10^(value/10);
        end

        % Value changed function: bandSlider7
        function bandSlider7ValueChanged(app, event)
            value = app.bandSlider7.Value;
            app.bands(7) = 10^(value/10);
        end

        % Value changed function: bandSlider8
        function bandSlider8ValueChanged(app, event)
            value = app.bandSlider8.Value;
            app.bands(8) = 10^(value/10);
        end

        % Value changed function: bandSlider9
        function bandSlider9ValueChanged(app, event)
            value = app.bandSlider9.Value;
            app.bands(9) = 10^(value/10);
        end

        % Button pushed function: FilterButton
        function FilterButtonPushed(app, event)
            app.filterStatus.Text = "Filtering started";
            if(app.filter_type == 0)
                app.filterStatus.Text = "Filtering using FIR";
                app.filtered_audio_data = 
apply_fir_filters(app,app.audio_data, app.sample_rate, app.bands);
            else
                app.filterStatus.Text = "Filtering using IIR";
                app.filtered_audio_data = 
apply_iir_filters(app,app.audio_data, app.sample_rate, app.bands);
            end
            app.filterStatus.Text = "Filtering ended";
            app.filterStatus.Text = "Plotting started";
            
plot(app.originalPlot_1,linspace(0,(length(app.default_audio_data)*(1/app.def
ault_sample_rate))-
(1/app.default_sample_rate),length(app.default_audio_data)),app.audio_data);
            
plot(app.filteredPlot_1,linspace(0,(length(app.filtered_audio_data)*(1/app.de
fault_sample_rate))-
(1/app.default_sample_rate),length(app.filtered_audio_data)),app.filtered_aud
io_data);
            plot(app.originalPlot_2,linspace(-
app.default_sample_rate/2,app.default_sample_rate/2,length(abs(fftshift(fft(a
pp.default_audio_data))))),abs(1/app.default_sample_rate.*fftshift(fft(app.de
fault_audio_data))));
            plot(app.filteredPlot_2,linspace(-
app.default_sample_rate/2,app.default_sample_rate/2,length(abs(fftshift(fft(a
pp.filtered_audio_data))))),abs(1/app.default_sample_rate.*fftshift(fft(app.f
iltered_audio_data))));
            app.filterStatus.Text = "Plotting ended";
        end

        % Button pushed function: PlayButton
        function PlayButtonPushed(app, event)
                sound(app.filtered_audio_data,app.default_sample_rate); % 
plays filtered sound for now
        end
        % Button pushed function: StopButton
        function StopButtonPushed(app, event)
            clear sound; % stops filtered sound for now
        end

        % Button pushed function: PlotsButton
        function PlotsButtonPushed(app, event)
            app.plotsStatus.Text = "generating plots";

plot(app.originalPlot_1,linspace(0,(length(app.default_audio_data)*(1/app.def
ault_sample_rate))-
(1/app.default_sample_rate),length(app.default_audio_data)),app.audio_data);

plot(app.filteredPlot_1,linspace(0,(length(app.filtered_audio_data)*(1/app.de
fault_sample_rate))-
(1/app.default_sample_rate),length(app.filtered_audio_data)),app.filtered_aud
io_data);
            plot(app.originalPlot_2,linspace(-
app.default_sample_rate/2,app.default_sample_rate/2,length(abs(fftshift(fft(a
pp.default_audio_data))))),abs(1/app.default_sample_rate.*fftshift(fft(app.de
fault_audio_data))));
            plot(app.filteredPlot_2,linspace(-
app.default_sample_rate/2,app.default_sample_rate/2,length(abs(fftshift(fft(a
pp.filtered_audio_data))))),abs(1/app.default_sample_rate.*fftshift(fft(app.f
iltered_audio_data))));
            app.plotsStatus.Text = "generation ended";
        end

        % Button pushed function: SaveButton
        function SaveButtonPushed(app, event)
            filename = 'filtered.wav';

audiowrite(filename,app.filtered_audio_data,app.default_sample_rate);
        end

        % Button pushed function: doubleButton
        function doubleButtonPushed(app, event)
            y =
resample(app.default_audio_data,app.default_sample_rate*0.5,app.default_sampl
e_rate);
            sound(y,app.default_sample_rate);

plot(app.originalPlot_1,linspace(0,(length(app.default_audio_data)*(1/app.def
ault_sample_rate))-
(1/app.default_sample_rate),length(app.default_audio_data)),app.audio_data);

plot(app.filteredPlot_1,linspace(0,(length(y)*(1/app.default_sample_rate))-
(1/app.default_sample_rate),length(y)),y);
```

```
            plot(app.originalPlot_2,linspace(-
app.default_sample_rate/2,app.default_sample_rate/2,length(abs(fftshift(fft(a
pp.default_audio_data))))),abs(1/app.default_sample_rate.*fftshift(fft(app.de
fault_audio_data))));
            plot(app.filteredPlot_2,linspace(-
app.default_sample_rate/2,app.default_sample_rate/2,length(abs(fftshift(fft(y
))))),abs(1/app.default_sample_rate.*fftshift(fft(y))));
        end

        % Button pushed function: halfButton
        function halfButtonPushed(app, event)
            y =
resample(app.default_audio_data,app.default_sample_rate*2,app.default_sample_
rate);
            sound(y,app.default_sample_rate);

plot(app.originalPlot_1,linspace(0,(length(app.default_audio_data)*(1/app.def
ault_sample_rate))-
(1/app.default_sample_rate),length(app.default_audio_data)),app.audio_data);

plot(app.filteredPlot_1,linspace(0,(length(y)*(1/app.default_sample_rate))-
(1/app.default_sample_rate),length(y)),y);
            plot(app.originalPlot_2,linspace(-
app.default_sample_rate/2,app.default_sample_rate/2,length(abs(fftshift(fft(a
pp.default_audio_data))))),abs(1/app.default_sample_rate.*fftshift(fft(app.de
fault_audio_data))));
            plot(app.filteredPlot_2,linspace(-
app.default_sample_rate/2,app.default_sample_rate/2,length(abs(fftshift(fft(y
))))),abs(1/app.default_sample_rate.*fftshift(fft(y))));
        end
    end

    % Component initialization
    methods (Access = private)

        % Create UIFigure and components
        function createComponents(app)

            % Create UIFigure and hide until all components are created
            app.UIFigure = uifigure('Visible', 'off');
            app.UIFigure.Color = [0.9412 0.9412 0.9412];
            app.UIFigure.Position = [100 100 1188 515];
            app.UIFigure.Name = 'UI Figure';

            % Create originalPlot_1
            app.originalPlot_1 = uiaxes(app.UIFigure);
            title(app.originalPlot_1, 'Original (Time)')
            xlabel(app.originalPlot_1, 'X')
            ylabel(app.originalPlot_1, 'Y')
            app.originalPlot_1.Position = [507 248 340 220];

            % Create filteredPlot_1
            app.filteredPlot_1 = uiaxes(app.UIFigure);
            title(app.filteredPlot_1, 'Filtered (Time)')
            xlabel(app.filteredPlot_1, 'X')
            ylabel(app.filteredPlot_1, 'Y')
            app.filteredPlot_1.Position = [507 25 340 210];

            % Create originalPlot_2
            app.originalPlot_2 = uiaxes(app.UIFigure);
            title(app.originalPlot_2, 'Original (Frequency)')
            xlabel(app.originalPlot_2, 'X')
            ylabel(app.originalPlot_2, 'Y')
            app.originalPlot_2.Position = [847 245 340 220];

            % Create filteredPlot_2
            app.filteredPlot_2 = uiaxes(app.UIFigure);
            title(app.filteredPlot_2, 'Filtered (Frequency)')
            xlabel(app.filteredPlot_2, 'X')
            ylabel(app.filteredPlot_2, 'Y')
            app.filteredPlot_2.Position = [847 25 340 210];

            % Create FileNameLabel
            app.FileNameLabel = uilabel(app.UIFigure);
            app.FileNameLabel.VerticalAlignment = 'top';
            app.FileNameLabel.Position = [20 453 64 15];
            app.FileNameLabel.Text = 'File Name:';

            % Create currentFile
            app.currentFile = uilabel(app.UIFigure);
            app.currentFile.BackgroundColor = [1 1 1];
            app.currentFile.HorizontalAlignment = 'center';
            app.currentFile.Position = [85 449 207 22];
            app.currentFile.Text = 'Choose a file ...';

            % Create chooseFileButton
            app.chooseFileButton = uibutton(app.UIFigure, 'push');
            app.chooseFileButton.ButtonPushedFcn = createCallbackFcn(app,
@chooseFileButtonPushed, true);
            app.chooseFileButton.Position = [300 449 100 22];
            app.chooseFileButton.Text = 'Choose';

            % Create HzSlider_2Label
            app.HzSlider_2Label = uilabel(app.UIFigure);
            app.HzSlider_2Label.HorizontalAlignment = 'right';
            app.HzSlider_2Label.VerticalAlignment = 'top';
            app.HzSlider_2Label.FontColor = [0.149 0.149 0.149];
            app.HzSlider_2Label.Position = [1 421 63 15];
            app.HzSlider_2Label.Text = '0 - 170 Hz';

            % Create bandSlider1
            app.bandSlider1 = uislider(app.UIFigure);
            app.bandSlider1.Limits = [-20 20];
            app.bandSlider1.MajorTicks = [-20 -16 -12 -8 -4 0 4 8 12 16 20];
            app.bandSlider1.MajorTickLabels = {'-20', '-16', '-12', '-8', '-
4', '0', '4', '8', '12', '16', '20'};
            app.bandSlider1.ValueChangedFcn = createCallbackFcn(app, 
@bandSlider1ValueChanged, true);
            app.bandSlider1.FontColor = [0.149 0.149 0.149];
            app.bandSlider1.Position = [85 428 309 3];

            % Create title
            app.title = uilabel(app.UIFigure);
            app.title.Position = [20 477 126 26];
            app.title.Text = 'Audio Equalizer';

            % Create HzLabel_2
            app.HzLabel_2 = uilabel(app.UIFigure);
            app.HzLabel_2.HorizontalAlignment = 'right';
            app.HzLabel_2.VerticalAlignment = 'top';
            app.HzLabel_2.Position = [1 371 72 22];
            app.HzLabel_2.Text = '170 - 300 Hz';

            % Create bandSlider2
            app.bandSlider2 = uislider(app.UIFigure);
            app.bandSlider2.Limits = [-20 20];
            app.bandSlider2.MajorTicks = [-20 -16 -12 -8 -4 0 4 8 12 16 20];
            app.bandSlider2.ValueChangedFcn = createCallbackFcn(app, 
@bandSlider2ValueChanged, true);
            app.bandSlider2.Position = [85 384 309 3];

            % Create HzLabel_3
            app.HzLabel_3 = uilabel(app.UIFigure);
            app.HzLabel_3.HorizontalAlignment = 'right';
            app.HzLabel_3.VerticalAlignment = 'top';
            app.HzLabel_3.Position = [1 328 77 22];
            app.HzLabel_3.Text = '300 -  610 Hz';

            % Create bandSlider3
            app.bandSlider3 = uislider(app.UIFigure);
            app.bandSlider3.Limits = [-20 20];
            app.bandSlider3.MajorTicks = [-20 -16 -12 -8 -4 0 4 8 12 16 20];
            app.bandSlider3.ValueChangedFcn = createCallbackFcn(app, 
@bandSlider3ValueChanged, true);
            app.bandSlider3.Position = [85 341 309 3];

            % Create HzLabel_4
            app.HzLabel_4 = uilabel(app.UIFigure);
            app.HzLabel_4.HorizontalAlignment = 'right';
            app.HzLabel_4.VerticalAlignment = 'top';
            app.HzLabel_4.Position = [1 290 82 22];
            app.HzLabel_4.Text = '610 - 1005 Hz';

            % Create bandSlider4
            app.bandSlider4 = uislider(app.UIFigure);
            app.bandSlider4.Limits = [-20 20];
            app.bandSlider4.MajorTicks = [-20 -16 -12 -8 -4 0 4 8 12 16 20];
            app.bandSlider4.ValueChangedFcn = createCallbackFcn(app,
@bandSlider4ValueChanged, true);
            app.bandSlider4.Position = [85 303 309 3];

            % Create HzLabel
            app.HzLabel = uilabel(app.UIFigure);
            app.HzLabel.HorizontalAlignment = 'right';
            app.HzLabel.VerticalAlignment = 'top';
            app.HzLabel.Position = [1 244 80 22];
            app.HzLabel.Text = '1.005 - 3 KHz';

            % Create bandSlider5
            app.bandSlider5 = uislider(app.UIFigure);
            app.bandSlider5.Limits = [-20 20];
            app.bandSlider5.MajorTicks = [-20 -16 -12 -8 -4 0 4 8 12 16 20];
            app.bandSlider5.ValueChangedFcn = createCallbackFcn(app,
@bandSlider5ValueChanged, true);
            app.bandSlider5.Position = [85 257 309 3];

            % Create KHzSlider_2Label
            app.KHzSlider_2Label = uilabel(app.UIFigure);
            app.KHzSlider_2Label.HorizontalAlignment = 'right';
            app.KHzSlider_2Label.VerticalAlignment = 'top';
            app.KHzSlider_2Label.Position = [1 205 55 15];
            app.KHzSlider_2Label.Text = '3 - 6 KHz';

            % Create bandSlider6
            app.bandSlider6 = uislider(app.UIFigure);
            app.bandSlider6.Limits = [-20 20];
            app.bandSlider6.MajorTicks = [-20 -16 -12 -8 -4 0 4 8 12 16 20];
            app.bandSlider6.ValueChangedFcn = createCallbackFcn(app,
@bandSlider6ValueChanged, true);
            app.bandSlider6.Position = [85 211 309 3];

            % Create KHzSlider_3Label
            app.KHzSlider_3Label = uilabel(app.UIFigure);
            app.KHzSlider_3Label.HorizontalAlignment = 'right';
            app.KHzSlider_3Label.VerticalAlignment = 'top';
            app.KHzSlider_3Label.Position = [1 153 62 15];
            app.KHzSlider_3Label.Text = '6 - 12 KHz';

            % Create bandSlider7
            app.bandSlider7 = uislider(app.UIFigure);
            app.bandSlider7.Limits = [-20 20];
            app.bandSlider7.MajorTicks = [-20 -16 -12 -8 -4 0 4 8 12 16 20];
            app.bandSlider7.ValueChangedFcn = createCallbackFcn(app,
@bandSlider7ValueChanged, true);
            app.bandSlider7.Position = [85 166 309 3];

            % Create KHzLabel
            app.KHzLabel = uilabel(app.UIFigure);
            app.KHzLabel.HorizontalAlignment = 'right';
            app.KHzLabel.VerticalAlignment = 'top';
            app.KHzLabel.Position = [1 114 69 15];
            app.KHzLabel.Text = '12 - 14 KHz';

            % Create bandSlider8
            app.bandSlider8 = uislider(app.UIFigure);
            app.bandSlider8.Limits = [-20 20];
            app.bandSlider8.MajorTicks = [-20 -16 -12 -8 -4 0 4 8 12 16 20];
            app.bandSlider8.ValueChangedFcn = createCallbackFcn(app,
@bandSlider8ValueChanged, true);
            app.bandSlider8.Position = [85 120 309 3];

            % Create KHzLabel_2
            app.KHzLabel_2 = uilabel(app.UIFigure);
            app.KHzLabel_2.HorizontalAlignment = 'right';
            app.KHzLabel_2.VerticalAlignment = 'top';
            app.KHzLabel_2.Position = [1 60 69 22];
            app.KHzLabel_2.Text = '14 - 20 KHz';

            % Create bandSlider9
            app.bandSlider9 = uislider(app.UIFigure);
            app.bandSlider9.Limits = [-20 20];
            app.bandSlider9.MajorTicks = [-20 -16 -12 -8 -4 0 4 8 12 16 20];
            app.bandSlider9.ValueChangedFcn = createCallbackFcn(app,
@bandSlider9ValueChanged, true);
            app.bandSlider9.Position = [85 73 309 3];

            % Create FilterTypeButtonGroup
            app.FilterTypeButtonGroup = uibuttongroup(app.UIFigure);
            app.FilterTypeButtonGroup.SelectionChangedFcn =
createCallbackFcn(app, @FilterTypeButtonGroupSelectionChanged, true);
            app.FilterTypeButtonGroup.BorderType = 'none';
            app.FilterTypeButtonGroup.Title = 'Filter Type';
            app.FilterTypeButtonGroup.FontName = 'MS Sans Serif';
            app.FilterTypeButtonGroup.Position = [411 392 100 76];
            % Create FIR
            app.FIR = uiradiobutton(app.FilterTypeButtonGroup);
            app.FIR.Text = 'FIR';
            app.FIR.Position = [11 31 41 15];
            app.FIR.Value = true;

            % Create IIR
            app.IIR = uiradiobutton(app.FilterTypeButtonGroup);
            app.IIR.Text = 'IIR';
            app.IIR.Position = [11 9 37 15];

            % Create FilterButton
            app.FilterButton = uibutton(app.UIFigure, 'push');
            app.FilterButton.ButtonPushedFcn = createCallbackFcn(app,
@FilterButtonPushed, true);
            app.FilterButton.Position = [411 316 78 22];
            app.FilterButton.Text = 'Filter';

            % Create PlayButton
            app.PlayButton = uibutton(app.UIFigure, 'push');
            app.PlayButton.ButtonPushedFcn = createCallbackFcn(app,
@PlayButtonPushed, true);
            app.PlayButton.Position = [411 257 78 22];
            app.PlayButton.Text = 'Play';

            % Create StopButton
            app.StopButton = uibutton(app.UIFigure, 'push');
            app.StopButton.ButtonPushedFcn = createCallbackFcn(app,
@StopButtonPushed, true);
            app.StopButton.Position = [411 227 78 22];
            app.StopButton.Text = 'Stop';

            % Create SaveButton
            app.SaveButton = uibutton(app.UIFigure, 'push');
            app.SaveButton.ButtonPushedFcn = createCallbackFcn(app,
@SaveButtonPushed, true);
            app.SaveButton.Position = [411 198 78 22];
            app.SaveButton.Text = 'Save';

            % Create PlotsButton
            app.PlotsButton = uibutton(app.UIFigure, 'push');
            app.PlotsButton.ButtonPushedFcn = createCallbackFcn(app,
@PlotsButtonPushed, true);
            app.PlotsButton.Position = [411 287 78 22];
            app.PlotsButton.Text = 'Plots';

            % Create filterStatus
            app.filterStatus = uilabel(app.UIFigure);
            app.filterStatus.HorizontalAlignment = 'center';
            app.filterStatus.Position = [410 68 100 15];
            app.filterStatus.Text = 'Filtering Status ..';

            % Create plotsStatus
            app.plotsStatus = uilabel(app.UIFigure);
            app.plotsStatus.HorizontalAlignment = 'center';
            app.plotsStatus.Position = [412 46 98 15];
            app.plotsStatus.Text = 'Plots Status ...';

            % Create doubleButton
            app.doubleButton = uibutton(app.UIFigure, 'push');
            app.doubleButton.ButtonPushedFcn = createCallbackFcn(app,
@doubleButtonPushed, true);
            app.doubleButton.Position = [406 169 92 22];
            app.doubleButton.Text = 'Double Speed';

            % Create halfButton
            app.halfButton = uibutton(app.UIFigure, 'push');
            app.halfButton.ButtonPushedFcn = createCallbackFcn(app,
@halfButtonPushed, true);
            app.halfButton.Position = [413 139 78 22];
            app.halfButton.Text = 'Half Speed';

            % Create SampleRateEditFieldLabel
            app.SampleRateEditFieldLabel = uilabel(app.UIFigure);
            app.SampleRateEditFieldLabel.Position = [413 371 76 15];
            app.SampleRateEditFieldLabel.Text = 'Sample Rate';

            % Create sampleRateInput
            app.sampleRateInput = uieditfield(app.UIFigure, 'numeric');
            app.sampleRateInput.RoundFractionalValues = 'on';
            app.sampleRateInput.Editable = 'off';
            app.sampleRateInput.HorizontalAlignment = 'left';
            app.sampleRateInput.Position = [411 344 78 22];

            % Show the figure after all components are created
            app.UIFigure.Visible = 'on';
        end
    end

    % App creation and deletion
    methods (Access = public)

        % Construct app
        function app = app1

            % Create UIFigure and components
            createComponents(app)
            % Register the app with App Designer
            registerApp(app, app.UIFigure)

            if nargout == 0
                clear app
            end
        end

        % Code that executes before app deletion
        function delete(app)

            % Delete UIFigure when app is deleted
            delete(app.UIFigure)
        end
    end
end
```

Sample Runs

- Gains in dB are: [12, -4, 0, -16, 20, 8, 0, -20, 4]
  - ❖ FIR Filters



Audio Equalizer

File Name: Trial.wav

Filter Type

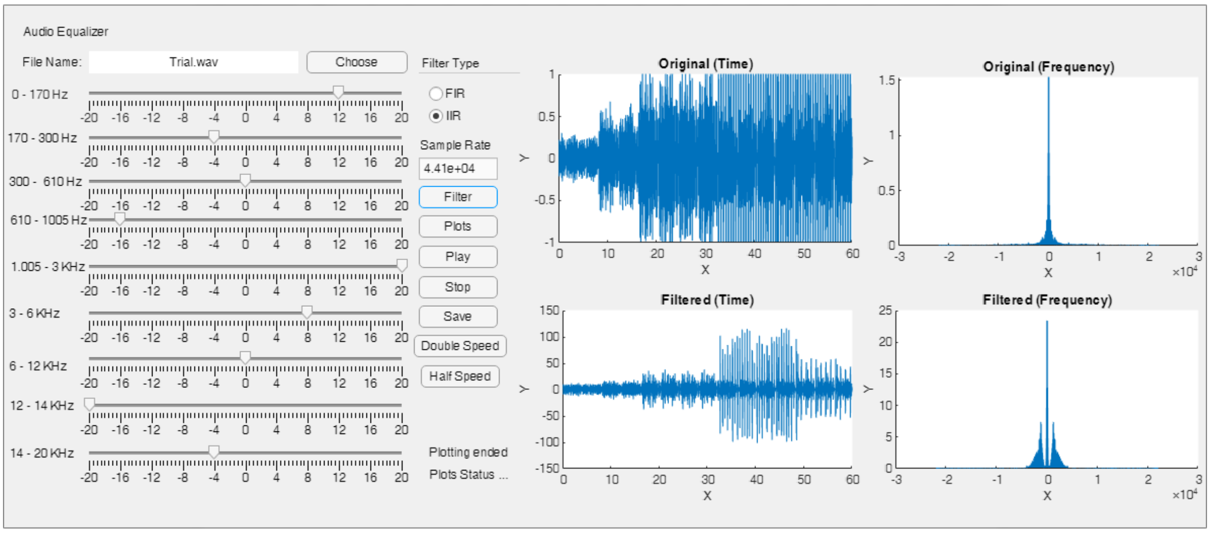
● FIR
○ IIR

Sample Rate 4.41e+04

Plotting ended

Plots Status ...

- ❖ IIR Filters



Audio Equalizer

File Name: Trial.wav

Filter Type

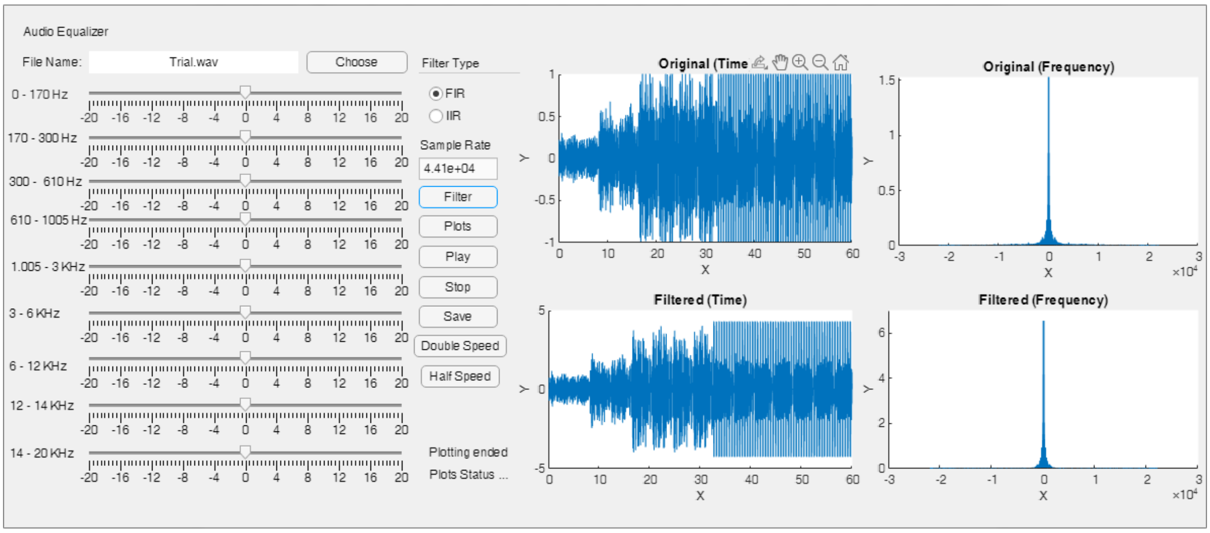
○ FIR
● IIR

Sample Rate 4.41e+04

Plotting ended

Plots Status ...

- Gains in dB are: [0, 0, 0, 0, 0, 0, 0, 0, 0]
    - ❖ FIR Filters



Audio Equalizer

File Name: Trial.wav

Filter Type

● FIR
○ IIR

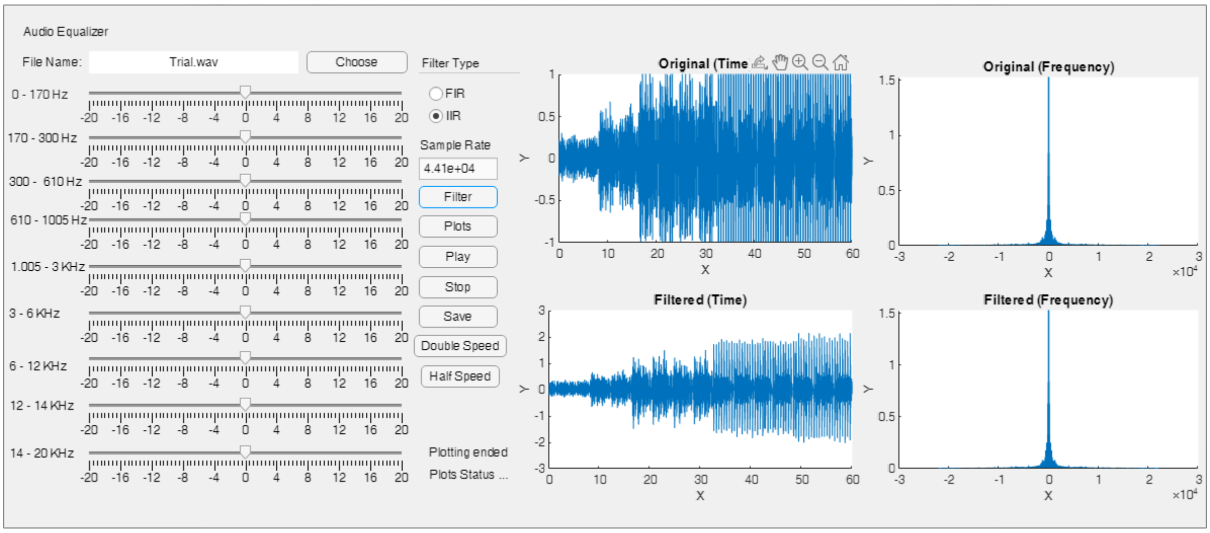
Sample Rate

4.41e+04

Plotting ended

Plots Status ...

- ❖ IIR Filters



Audio Equalizer

File Name: Trial.wav

Filter Type

○ FIR
● IIR

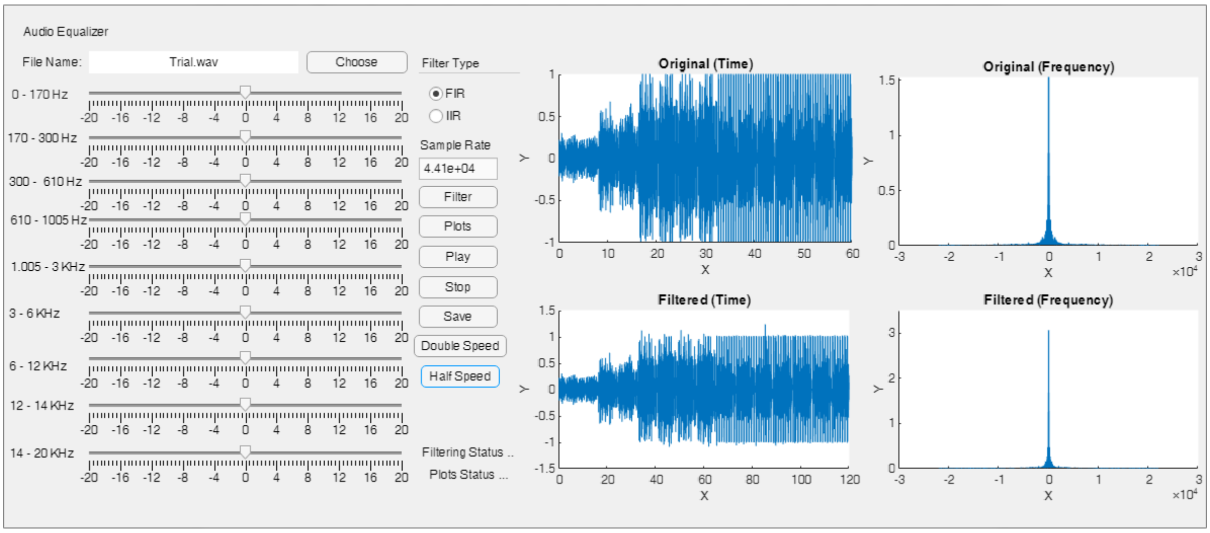
Sample Rate

4.41e+04

Plotting ended

Plots Status ...

- Doubling Frequency “Half Speed”



Audio Equalizer

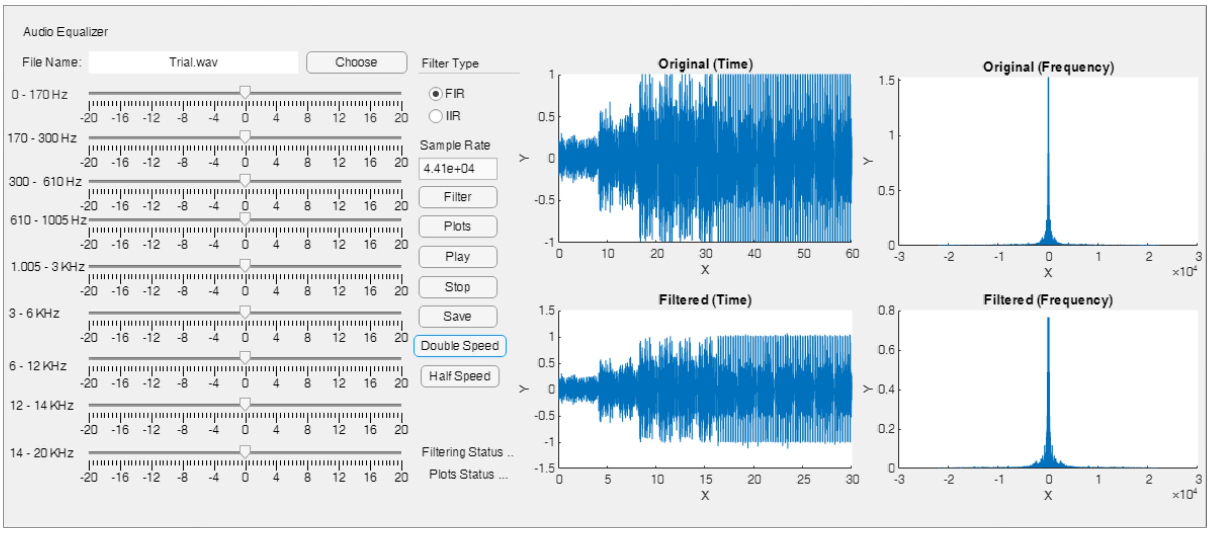
File Name: Trial.wav

Filter Type

● FIR
○ IIR

Sample Rate 4.41e+04

- Halving Frequency “Double Speed”



Audio Equalizer

File Name: Trial.wav

Filter Type

● FIR
○ IIR

Sample Rate 4.41e+04