CSV to png and workspace

Csv name format - <name1>_<name2>_<name3>.csv no - '({ /...'

CSV to workspace: will add to workspace vectors <type>_<element>_<mesurmant parameter> :type = dB_mag or lin_mag, elemant = <name1>, mesurmant = <name2>

```
% Set the directory containing the CSV files
csv_directory = '/Users/ohadformanair/Documents/Git/AML/LAB_3p4/AML_3&4_measurmants/CS

% Set the directory for saving the plots
save_directory = '/Users/ohadformanair/Documents/Git/AML/LAB_3p4/AML_3&4_measurmants/c

%Set stat result directory
stat_directory = '/Users/ohadformanair/Documents/Git/AML/LAB_3p4/Stat_results';
```

```
% Initialize the output variables
linear_magnitude_data = [];
dB_magnitude_data = [];
phase_data = [];

% Initialize table for magnitude and phase statistics
mag_stats = table();
phase_stats = table();

% Loop over each file in the directory
filelist = dir(fullfile(csv_directory, '*.csv'));
for i = 1:length(filelist)
    % Extract the filename components
    [~, filename, ~] = fileparts(filelist(i).name);
    name_parts = split(filename, '_');
    name1 = name_parts{1};
    name2 = name_parts{2};

    % Load the data from the current file
    data = readmatrix(fullfile(csv_directory, filelist(i).name), 'HeaderLines', 3);

    % Extract the frequency and formatted data columns
    frequency = data(:, 1); % Don't normalize frequency
    formatted_data = data(:, 2);

    % Calculate the magnitude and phase data
    if strcmp(name_parts(end), 'MAG')
        linear_magnitude = formatted_data;
        dB_magnitude = 20*log10(formatted_data);
        linear_phase = [];
        phase = [];

        % Calculate magnitude statistics and add to table
        mag_mean = mean(linear_magnitude);
```

```
        mag_median = median(linear_magnitude);
        mag_std = std(linear_magnitude);
        mag_min = min(linear_magnitude);
        mag_max = max(linear_magnitude);
        mag_stats = [mag_stats; table({name1}, {name2}, mag_mean, mag_median, mag_std,

    elseif strcmp(name_parts(end), 'PHASE')
        linear_phase = formatted_data;
        phase = formatted_data;
        linear_magnitude = [];
        dB_magnitude = [];

        % Calculate phase statistics and add to table
        phase_mean = mean(linear_phase);
        phase_median = median(linear_phase);
        phase_std = std(linear_phase);
        phase_min = min(linear_phase);
        phase_max = max(linear_phase);
        phase_stats = [phase_stats; table({name1}, {name2}, phase_mean, phase_median,

    else
        % Invalid name3 value, skip this file
        continue;
    end

    % Append the data for this file to the output variables
    linear_magnitude_data = [linear_magnitude_data; linear_magnitude];
    dB_magnitude_data = [dB_magnitude_data; dB_magnitude];
    phase_data = [phase_data; phase];

    % Create variables in the workspace for the frequency and magnitude data
    freq_varname = ['freq_' name1 '_' name2];
    assignin('base', freq_varname, frequency);
    if ~isempty(linear_magnitude)
        lin_mag_varname = ['lin_mag_' name1 '_' name2];
        assignin('base', lin_mag_varname, linear_magnitude);
    end
    if ~isempty(dB_magnitude)
        dB_mag_varname = ['dB_mag_' name1 '_' name2];
        assignin('base', dB_mag_varname, dB_magnitude);
    end
    if ~isempty(phase)
        phase_varname = ['phase_' name1 '_' name2];
        assignin('base', phase_varname, phase);
    end

    % Display the progress
    fprintf('Processed file %d of %d: %s\n', i, length(filelist), filelist(i).name);
end
```

```
Processed file 1 of 44: ATTENUATOR10dB_S11_MAG.CSV
Processed file 2 of 44: ATTENUATOR10dB_S11_PHASE.CSV
Processed file 3 of 44: ATTENUATOR10dB_S21_MAG.CSV
Processed file 4 of 44: ATTENUATOR10dB_S21_PHASE.CSV
Processed file 5 of 44: ATTENUATOR10dB_S22_MAG.CSV
Processed file 6 of 44: ATTENUATOR10dB_S22_PHASE.CSV
Processed file 7 of 44: CIRCULATOR_S11_MAG.CSV
Processed file 8 of 44: CIRCULATOR_S11_PHASE.CSV
Processed file 9 of 44: CIRCULATOR_S13_MAG.CSV
Processed file 10 of 44: CIRCULATOR_S13_PHASE.CSV
Processed file 11 of 44: CIRCULATOR_S22_MAG.CSV
Processed file 12 of 44: CIRCULATOR_S22_PHASE.CSV
Processed file 13 of 44: CIRCULATOR_S23_MAG.CSV
Processed file 14 of 44: CIRCULATOR_S23_PHASE.CSV
Processed file 15 of 44: CIRCULATOR_S31_MAG.CSV
Processed file 16 of 44: CIRCULATOR_S31_PHASE.CSV
Processed file 17 of 44: CIRCULATOR_S32_MAG.CSV
Processed file 18 of 44: CIRCULATOR_S32_PHASE.CSV
Processed file 19 of 44: CIRCULATOR_S33_MAG.CSV
Processed file 20 of 44: CIRCULATOR_S33_PHASE.CSV
Processed file 21 of 44: COUPLER_S11_MAG.CSV
Processed file 22 of 44: COUPLER_S11_PHASE.CSV
Processed file 23 of 44: COUPLER_S12_MAG.CSV
Processed file 24 of 44: COUPLER_S12_PHASE.CSV
Processed file 25 of 44: COUPLER_S13_MAG.CSV
Processed file 26 of 44: COUPLER_S13_PHASE.CSV
Processed file 27 of 44: COUPLER_S14_MAG.CSV
Processed file 28 of 44: COUPLER_S14_PHASE.CSV
Processed file 29 of 44: MAGICT_S11_MAG.CSV
Processed file 30 of 44: MAGICT_S11_PHASE.CSV
Processed file 31 of 44: MAGICT_S13_MAG.CSV
Processed file 32 of 44: MAGICT_S13_PHASE.CSV
Processed file 33 of 44: MAGICT_S21_MAG.CSV
Processed file 34 of 44: MAGICT_S21_PHASE.CSV
Processed file 35 of 44: MAGICT_S33_MAG.CSV
Processed file 36 of 44: MAGICT_S33_PHASE.CSV
Processed file 37 of 44: MAGICT_S41_MAG.CSV
Processed file 38 of 44: MAGICT_S41_PHASE.CSV
Processed file 39 of 44: MAGICT_S42_MAG.CSV
Processed file 40 of 44: MAGICT_S42_PHASE.CSV
Processed file 41 of 44: MAGICT_S43_MAG.CSV
Processed file 42 of 44: MAGICT_S43_PHASE.CSV
Processed file 43 of 44: MAGICT_S44_MAG.CSV
Processed file 44 of 44: MAGICT_S44_PHASE.CSV
```

```
% Assign the output variables to the workspace
assignin('base', 'linear_magnitude_data', linear_magnitude_data);
assignin('base', 'dB_magnitude_data', dB_magnitude_data);
assignin('base', 'phase_data', phase_data);

% Display the magnitude statistics table
disp(mag_stats);
```

Var1	Var2	mag_mean	mag_median	mag_std	mag_min	mag_max
{'ATTENUATOR10dB'}	{'S11'}	0.16397	0.17282	0.035995	0.087018	0.22521
{'ATTENUATOR10dB'}	{'S21'}	0.31421	0.31912	0.017231	0.27307	0.34011
{'ATTENUATOR10dB'}	{'S22'}	0.093437	0.088995	0.029913	0.043255	0.16793
{'CIRCULATOR' }	{'S11'}	0.49783	0.51505	0.14108	0.28723	0.68689

{'CIRCULATOR' }	{'S13'}	0.76035	0.75967	0.069075	0.61367	0.90136
{'CIRCULATOR' }	{'S22'}	0.55653	0.59575	0.13933	0.33223	0.73661
{'CIRCULATOR' }	{'S23'}	0.24892	0.24827	0.013325	0.22471	0.27525
{'CIRCULATOR' }	{'S31'}	0.33749	0.33856	0.029507	0.27417	0.39551
{'CIRCULATOR' }	{'S32'}	0.72664	0.74376	0.10954	0.52562	0.89943
{'CIRCULATOR' }	{'S33'}	0.49268	0.48455	0.089414	0.36707	0.63779
{'COUPLER' }	{'S11'}	0.17728	0.17136	0.08728	0.012918	0.36812
{'COUPLER' }	{'S12'}	0.94952	0.95142	0.018375	0.88912	0.9896
{'COUPLER' }	{'S13'}	0.074495	0.075058	0.0064454	0.059652	0.090273
{'COUPLER' }	{'S14'}	0.0058987	0.005521	0.002993	0.00061361	0.013538
{'MAGICT' }	{'S11'}	0.26583	0.22323	0.12873	0.070193	0.51578
{'MAGICT' }	{'S13'}	0.58715	0.58667	0.025565	0.51528	0.64796
{'MAGICT' }	{'S21'}	0.3587	0.36393	0.029809	0.2876	0.41046
{'MAGICT' }	{'S33'}	0.55396	0.55514	0.034313	0.47595	0.67052
{'MAGICT' }	{'S41'}	0.65161	0.65684	0.039543	0.56437	0.72828
{'MAGICT' }	{'S42'}	0.64722	0.65225	0.038255	0.56088	0.72784
{'MAGICT' }	{'S43'}	0.0075694	0.0078122	0.0032121	0.00024736	0.017084
{'MAGICT' }	{'S44'}	0.35495	0.39884	0.11997	0.090516	0.49761

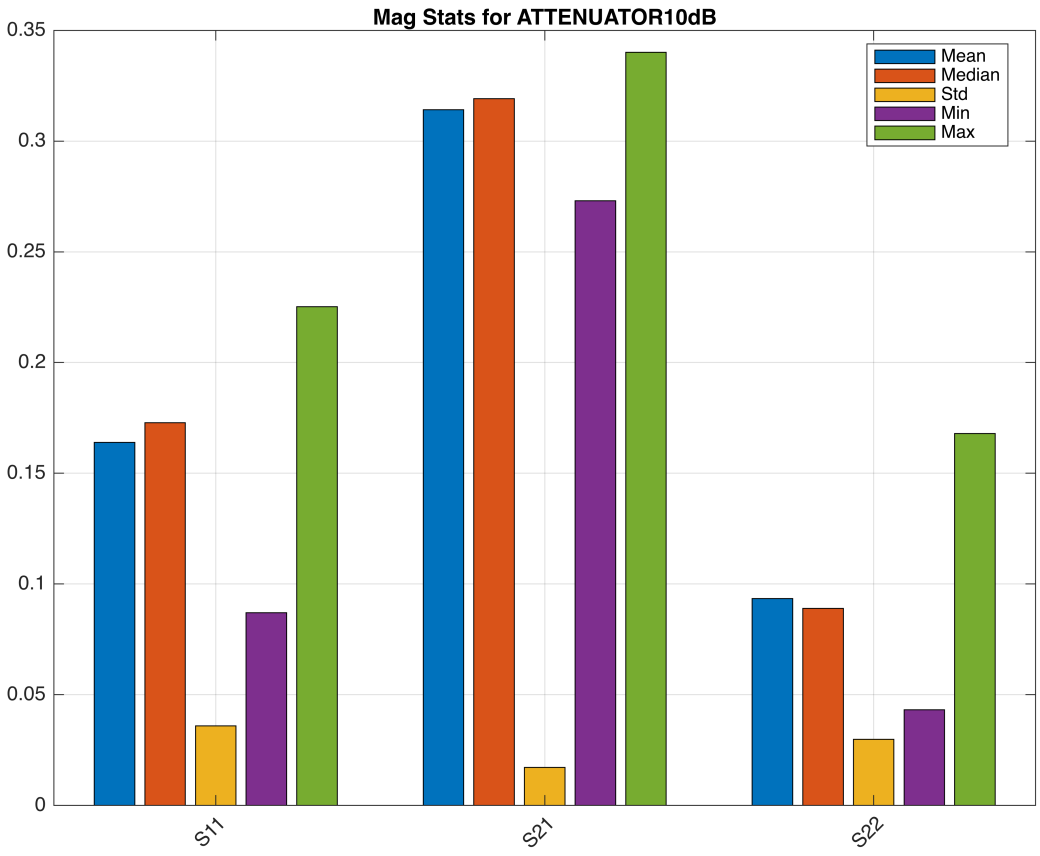
```
% Display the phase statistics table
disp(phase_stats);
```

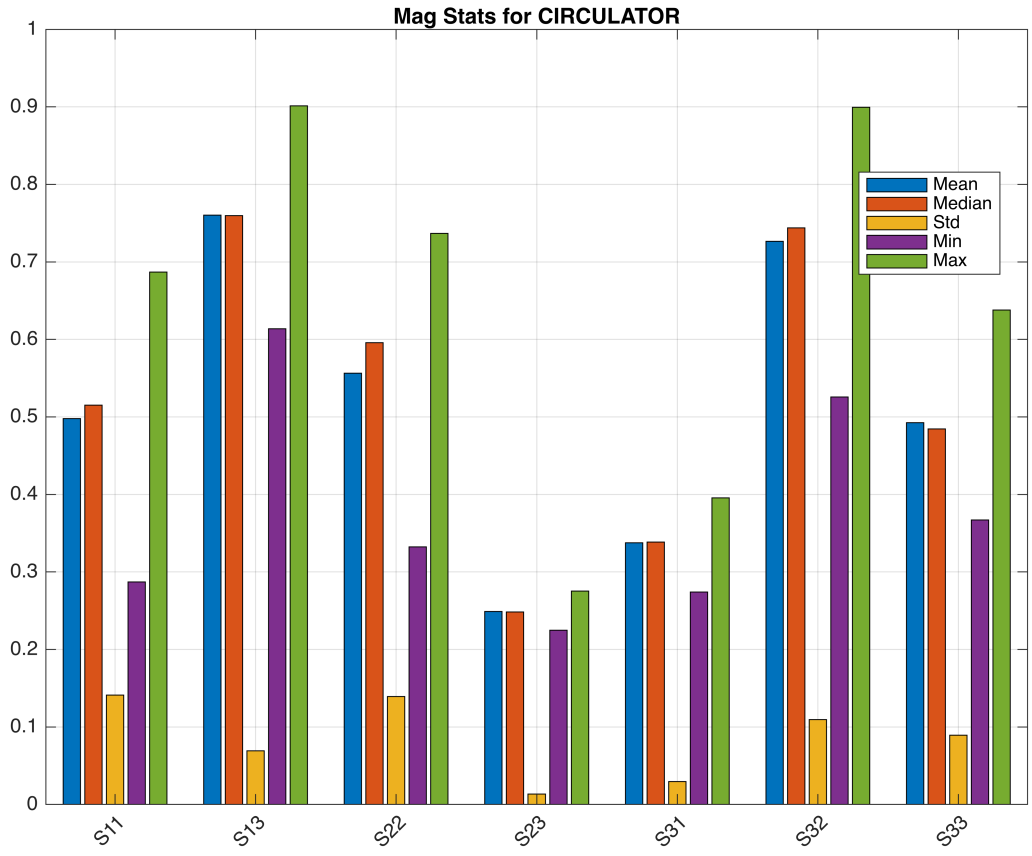
Var1	Var2	phase_mean	phase_median	phase_std	phase_min	phase_max
{'ATTENUATOR10dB'}	{'S11'}	-106.63	-112.83	21.144	-147.07	-65.95
{'ATTENUATOR10dB'}	{'S21'}	4.5724	13.198	104.3	-178.73	179.34
{'ATTENUATOR10dB'}	{'S22'}	-43.578	-93.802	125.89	-179.48	178.21
{'CIRCULATOR' }	{'S11'}	-17.616	-47.636	108.08	-179.88	179.29
{'CIRCULATOR' }	{'S13'}	9.3243	44.271	117.41	-178.7	178.78
{'CIRCULATOR' }	{'S22'}	-17.353	-53.338	111.48	-178.28	178.29
{'CIRCULATOR' }	{'S23'}	-0.27321	-2.5134	73.447	-128.83	131.45
{'CIRCULATOR' }	{'S31'}	-27.124	-27.364	78.233	-164.09	101.42
{'CIRCULATOR' }	{'S32'}	4.8117	44.873	120.88	-179.92	179.99
{'CIRCULATOR' }	{'S33'}	-39.843	-43.619	88.738	-178.68	178.86
{'COUPLER' }	{'S11'}	-103.65	-108.24	24.778	-160.38	-15.819
{'COUPLER' }	{'S12'}	14.222	26.107	105.11	-179.84	178.21
{'COUPLER' }	{'S13'}	2.7499	-17.162	113.27	-179.69	179.68
{'COUPLER' }	{'S14'}	17.013	27.333	94.461	-178.69	177.4
{'MAGICT' }	{'S11'}	-40.4	-56.221	101.67	-175.88	179.62
{'MAGICT' }	{'S13'}	6.3671	26.09	102.6	-179.6	178.91
{'MAGICT' }	{'S21'}	32.364	41.459	99.256	-178.87	177.98
{'MAGICT' }	{'S33'}	13.712	33.198	109.57	-178.85	179.68
{'MAGICT' }	{'S41'}	11.355	25.422	103.16	-177.9	179.89
{'MAGICT' }	{'S42'}	8.712	22.353	102.86	-179.75	177.4
{'MAGICT' }	{'S43'}	-1.9307	0.43641	105.34	-179.7	178.99
{'MAGICT' }	{'S44'}	-52.3	-67.433	81.351	-179.07	176.99

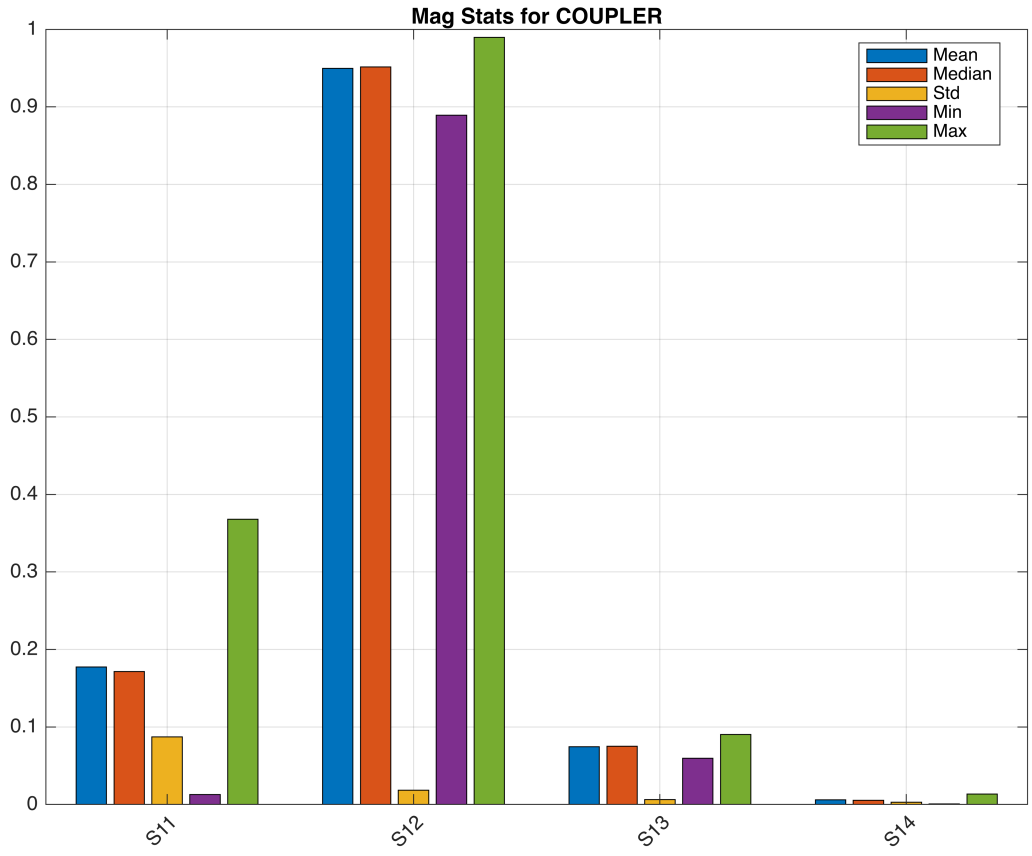
```
% Get unique values of Var1
Var1_vals = unique(mag_stats.Var1);

% Define figure width and height
fig_width = 800;
fig_height = 600;

% Loop over each unique value of Var1
for i = 1:length(Var1_vals)
```

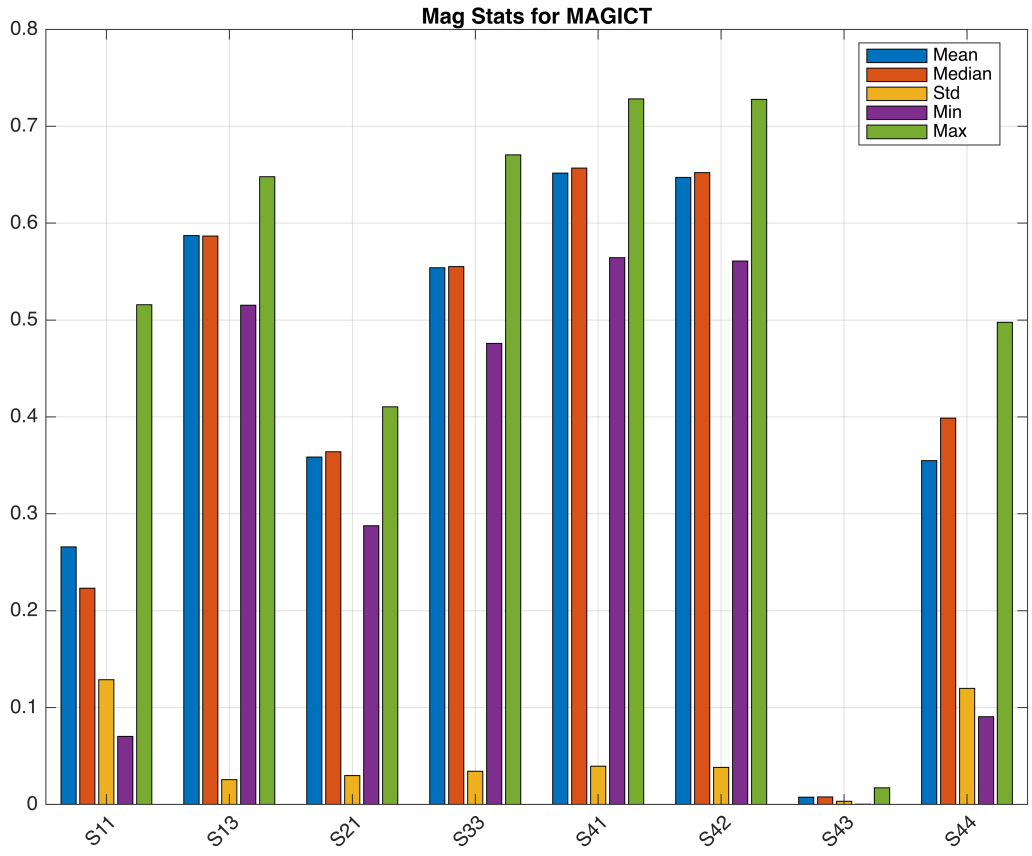
```
    % Subset the table to only include rows with the current Var1 value
    subset = mag_stats(strcmp(mag_stats.Var1, Var1_vals{i}),:);

    % Get unique values of Var2 for this subset
    Var2_vals = unique(subset.Var2);

    % Initialize empty arrays for the statistics
    mag_mean = [];
    mag_median = [];
    mag_std = [];
    mag_min = [];
    mag_max = [];

    % Loop over each unique value of Var2 for this subset
    for j = 1:length(Var2_vals)

        % Subset the table to only include rows with the current Var1 and Var2 values
        subsubset = subset(strcmp(subset.Var2, Var2_vals{j}),:);

        % Check if each statistic is available for this subset
        if ~any(ismissing(subsubset.mag_mean))
            mag_mean(j) = subsubset.mag_mean;
        end
        if ~any(ismissing(subsubset.mag_median))
            mag_median(j) = subsubset.mag_median;
        end
        if ~any(ismissing(subsubset.mag_std))
            mag_std(j) = subsubset.mag_std;
        end
        if ~any(ismissing(subsubset.mag_min))
            mag_min(j) = subsubset.mag_min;
        end
        if ~any(ismissing(subsubset.mag_max))
            mag_max(j) = subsubset.mag_max;
        end
    end

    % Create new figure with specified size
    fig = figure('Position', [0, 0, fig_width, fig_height]);

    % Plot the statistics for this Var1 value
    bar([mag_mean' mag_median' mag_std' mag_min' mag_max'])
    xticklabels(Var2_vals)
    xtickangle(45)
    legend({'Mean','Median','Std','Min','Max'},'Location','Best')
    title(sprintf('Mag Stats for %s', Var1_vals{i}))
    grid on

    % Save the figure as a PNG file
```

```
    filename = fullfile(stat_directory, sprintf('mag_stats_%s.png', Var1_vals{i}));
    saveas(fig, filename);

end
```




Mag Stats for CIRCULATOR
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Mean
Median
Std
Min
Max
S11
S13
S22
S23
S31
S32
S33


Mag Stats for COUPLER
Mean
Median
Std
Min
Max
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
S11
S12
S13
S14




```
% Save the table as a CSV file
filename = fullfile(stat_directory, 'mag_stats.csv');
writetable(mag_stats, filename);
```

the plot title will be: <name1>: <name2> <name3>

the y axis will be: Phase(deg) if <name3> is PHASE, if MAG it will be : Magnitude (unit less)

the x axis is frequency.

set in script:

csv_directory = 'path/to/csv/files'

plot_directory = 'path/to/save/plots'

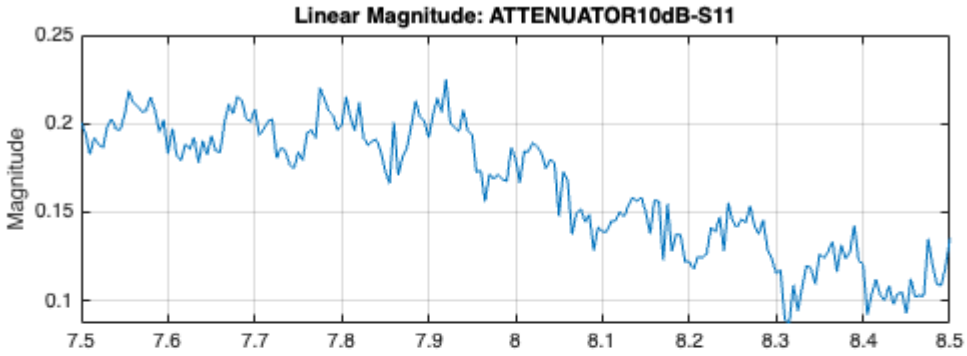
plot_mag_dB = (true for dB, false for linear)

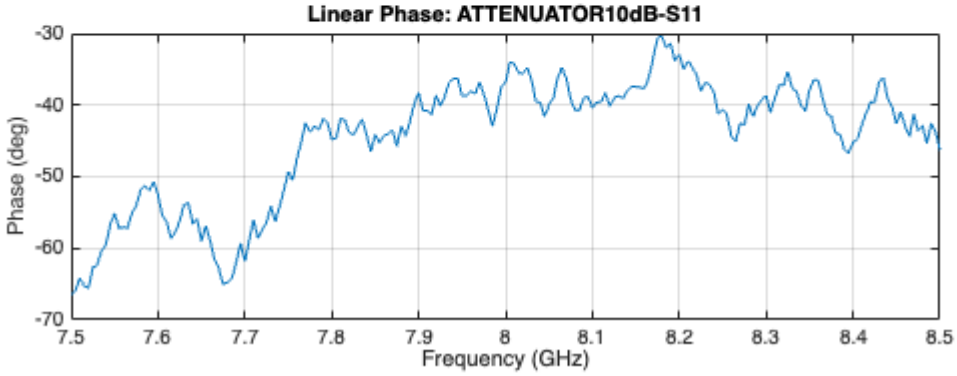
```
% Set smoothing options
```

```
smoothing_enabled = false;  % Set to true to enable smoothing
smoothing_window_size = 1; % Size of the smoothing window (odd number)
% Set plot type options
is_bode_plot = false; % Set to true for Bode plot, false for linear magnitude plot
```

```
% Get a list of all CSV files in the directory
filelist = dir(fullfile(csv_directory, '*.csv'));

% Loop over each file in the directory
for i = 1:length(filelist)
    % Extract the filename components
    [~, filename, ~] = fileparts(filelist(i).name);
    name_parts = strsplit(filename, '_');
    name1 = name_parts{1};
    name2 = name_parts{2};
    name3 = name_parts{3};

    % Load the data from the current file
    data = readmatrix(fullfile(csv_directory, filelist(i).name), 'HeaderLines', 3);

    % Extract the frequency and formatted data columns
    frequency_data = data(:, 1);
    formatted_data = data(:, 2);

    % Smooth the data if enabled
    if smoothing_enabled
        smoothed_data = smooth(formatted_data, smoothing_window_size);
    else
        smoothed_data = formatted_data;
    end

    % Plot the magnitude and phase data in a Bode or linear plot
    if strcmp(name3, 'MAG')
        mag_data = smoothed_data;
        % Find the corresponding phase file
        phase_filename = fullfile(csv_directory, [name1 '_' name2 '_PHASE.csv']);
        phase_data = readmatrix(phase_filename, 'HeaderLines', 3);
        phase_data = phase_data(:, 2);

        % Create the plot
        if is_bode_plot
            % Bode plot
            figure;
            subplot(2,1,1);
            semilogx(frequency_data/1e9, 20*log10(abs(mag_data)));
            ylabel('Magnitude (dB)');
            grid on;
            hold on;
            subplot(2,1,2);
```

```
            semilogx(frequency_data/1e9, unwrap(phase_data));
            ylabel('Phase (deg)');
            xlabel('Frequency (GHz)');
            grid on;
            title(sprintf('Bode: %s-%s', name1, name2));
            filename = fullfile(save_directory, ['Bode-' name1 '_' name2 '.png']);
        else
            % Linear plot with subplots
            figure;
            subplot(2,1,1);
            plot(frequency_data/1e9, mag_data);
            ylabel('Magnitude');
            grid on;
            title(sprintf('Linear Magnitude: %s-%s', name1, name2));

            % Add subplot for phase data
            subplot(2,1,2);
            plot(frequency_data/1e9, unwrap(phase_data));
            ylabel('Phase (deg)');
            xlabel('Frequency (GHz)');
            grid on;
            title(sprintf('Linear Phase: %s-%s', name1, name2));

            % Save the figure with the name1_name2_LINMAG.png
            filename = fullfile(save_directory, ['LinMag-' name1 '_' name2 '.png']);
            saveas(gcf, filename);
        end
        % Save the figure
        saveas(gcf, filename);

        % Close the figure
        close(gcf);
    end
end
```


Linear Magnitude: ATTENUATOR10dB-S11
Magnitude
0.25
0.2
0.15
0.1
7.5
7.6
7.7
7.8
7.9
8
8.1
8.2
8.3
8.4
8.5


Linear Phase: ATTENUATOR10dB-S11
Phase (deg)
-30
-40
-50
-60
-70
7.5
7.6
7.7
7.8
7.9
8
8.1
8.2
8.3
8.4
8.5
Frequency (GHz)