

Simulation of EMI Filters Using Matlab

M.Satish Kumar, Dr.A.Jhansirani,

Asst Professor, ECE Department, Sri Vasavi Engineering College, Tadepalligudem, India
Professor, ECE Department, V.R.Siddhartha Engineering College, Vijayawada, India.

Abstract: Filtering is an important mitigation technique for suppressing undesired conducted electromagnetic interference, when a system incorporates shielding, undesired coupling caused by radiated EMI is reduced. Conventional filter analysis and design assumes idealized and simplified conditions. These assumptions are not completely valid in many EMI filter because of unavoidable and severe impedance mismatch. Classical passive filter theory is well developed for communication circuits, where one can operate under impedance-matched conditions. Such filter characteristics are evaluated with 50Ω terminations. Filter evaluated with this procedure may behave differently when used in a circuit, where the impedance presented by the circuit to the filters is not exactly 50Ω . Now a day, digital signals are mostly used to avoid such EMI effects. These are caused by the capacitors, inductors, which are also part of the filtering circuits. Filter design using software, like MATLAB is very useful in avoiding hardware, is highly immune to noise and possesses considerable parameter stability, can be operated over a wide range of frequencies. The frequency response can be changed by changing the filter coefficients and can minimize the Insertion losses (IL).

Key Words: Mitigation, Shielding, Insertion losses

I. Introduction

MATLAB is a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. Typical uses include

1. Math and computation
2. Algorithm development
3. Data acquisition
4. Modeling, simulation, and prototyping
5. Data analysis, exploration, and visualization
6. Scientific and engineering graphics
7. Application development, including graphical user interface building.

The name MATLAB stands for *matrix laboratory*. MATLAB was originally written to provide easy access to matrix software developed by the LINPACK and EISPACK projects. MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. This allows you to solve many technical computing problems, especially those with matrix and vector formulations, in a fraction of the time it would take to write a program in a scalar non-interactive language such as C or FORTRAN.

II. Filter Design Toolbox

It is a collection of tools that provides advanced techniques for designing, simulating, and analyzing digital filters. It extends the capabilities of Signal Processing Toolbox with filter architectures and design methods for complex real-time DSP applications, including adaptive filtering and multirate filtering, as well as filter transformations. Key features of Filter Design Toolbox are

1. FIR filter design, including minimum-order, minimum-phase, constrained-ripple, half band, Nyquist, interpolated FIR, and nonlinear phase.
2. IIR filter design, including arbitrary magnitude and phase, group-delay equalizers, constrained-pole radius, peaking, notching, and comb filters.
3. Multirate filter design, analysis, and implementation, including cascaded integrator-comb (CIC) fixed-point multirate filters and compensators.
4. Farrow filter design.
5. Multirate, multistage filter design.
6. Wave digital filter design.
7. FIR and IIR filter transformations, including low pass to low pass, low pass to high pass, and low pass to multiband.

8. Adaptive filter design, analysis, and implementation, including LMS-based, RLS-based, lattice-based, frequency-domain, fast transversal and affine projection adaptive filters.
9. Round-off noise analysis for filters implemented in single-precision floating point or fixed point.
10. Analysis and implementation of digital filters in single-precision floating-point and fixed-point arithmetic.

How Filter Design Toolbox Works: The unique feature of Filter Design Toolbox is that you do not need to know any specific filter algorithms to design a good working filter. You take a given set of design parameters for the filter, such as a stop-band frequency, a pass-band frequency, and a stop-band attenuation, for example, and using these parameters design a specification object for the filter. Then, using this specification object, you design the filter. There are two distinct objects involved in filter design:

1. Specification Object — captures the required design parameters of a filter.
2. Filter Object — describes the designed filter; includes the array of coefficients and the filter structure.

Basic Filter Design Process: Use the following two steps to design a simple filter.

1. Create a filter specification object.

To create a filter specification object type or evaluate the following code at the MATLAB prompt:

```
>> BandPassSpecObj = fdesign.bandpass BandPassSpecObj =
Response: 'Bandpass'
```

```
Specification: 'Fst1,Fp1,Fp2,Fst2,Ast1,Ap,Ast2' Description: {7x1 cell}
Normalized Frequency: true Fstop1: 0.35
Fpass1: 0.45
```

```
Fpass2: 0.55
```

```
Fstop2: 0.65
```

```
Astop1: 60
```

```
Apass: 1
```

```
Astop2: 60
```

2. Design your filter.

Design the filter by using the design command. Evaluate or type the following at the MATLAB prompt:

```
>> BandPassFilt = design(BandPassSpecObj)
```

To check your work, you can plot the filter magnitude response using the Filter Visualization tool. Verify that all the design parameters are met:

```
>>fvtool(BandPassFilt) %plot the filter magnitude response.
```

3. EMI FILTERS MATLAB PROGRAM

1. BANDPASS FILTER: A Band-pass filter allows a particular band of frequencies to pass through without attenuation and rejects signals outside this band. Here we will use the transfer function of a particular order band-pass digital filter is given by

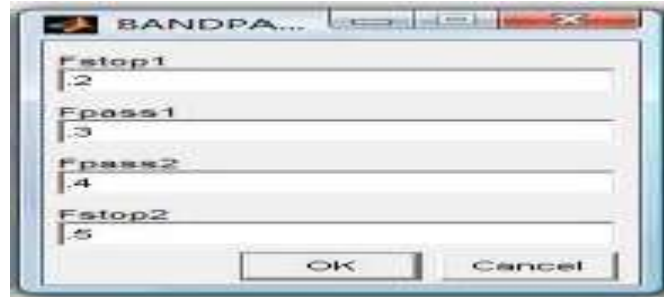
$$H(z) = \frac{1-a}{2} \frac{1+z^{-1}}{1+b(1-a)z^{-1}+az^{-2}}$$

And, the frequency response of the filter is.

$$H(e^{j\omega}) = \frac{1-a}{2} \frac{1+e^{-j\omega}}{1+b(1-a)e^{-j\omega}+ae^{-j2\omega}}$$

By the above transfer function MATLAB has developed a function “fdesign.bandpass” We are going to calculate the filter magnitude response and phase response simply enter the frequency pass band and stop band values. Here we also used the Dialog box function to get the input values and made special function with the help of .m file to operate or design this filter. Simply using the transfer functions we are going to design a filter instead of analog circuit made-up of capacitors resistors and inductors which are source of insertion losses.

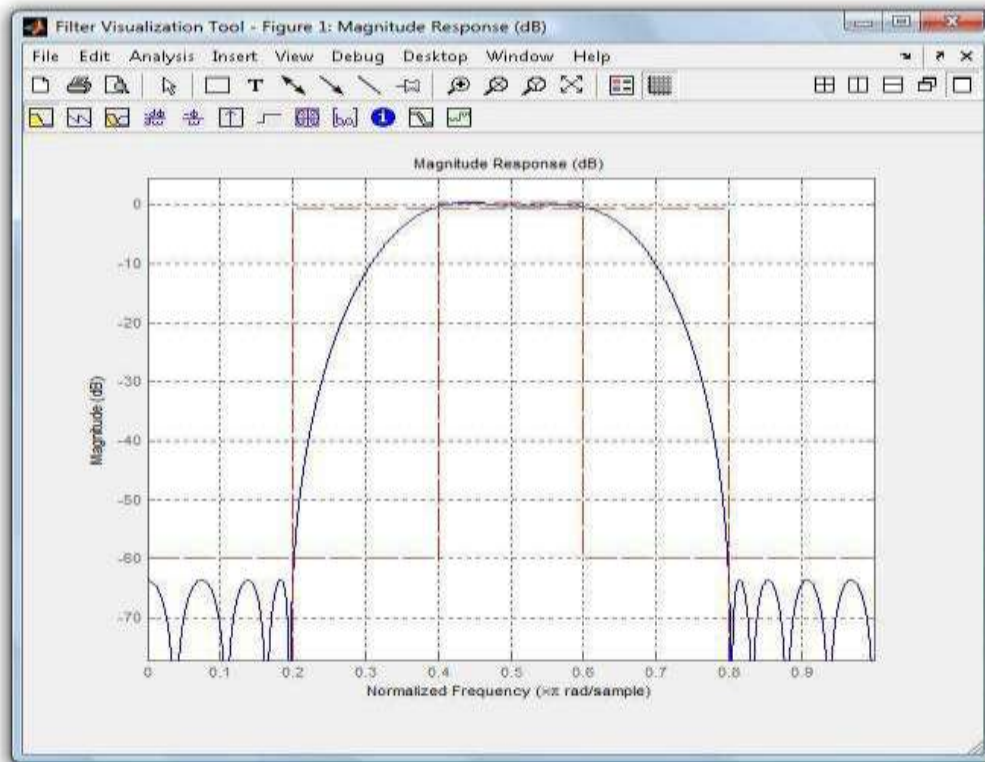
By simply typing the word “EMIBP” in command window of the MATLAB, a dialog box will appear as shown below to get the user desired values to design a band-pass filter.



To entering the values in particular columns, these values are taken as string values as per instruction of MATLAB, so we need to convert these values into number and that values has fed to the filter designed function of “fdesign.bandpass” and immediately our program is going to display the graph of the filter using “fvtool (bandpassfilter)” MATLAB function.

Result

Displayed graph has shown below.



In the above Graph window we have options to watch the

1. Phase response,
2. Phase & magnitude response at a time,
3. Group delay response,
4. Phase delay response,

5. Impulse response,
6. Step response,
7. Pole\zero plot,
8. Filter coefficients data,
9. Filter information and
10. Round off noise power spectrum.

Similarly we have design for BANDSTOP, NOTCH, HIGHPASS and LOWPASS Filters. These can simulate in software typing functions “EMIBS”, “EMINT”, EMIHP” and “EMILP” respectively.

MATLAB PROGRAM

```
function [bandpass]=EMIBP(Fstop1,Fpass1,Fpass2,Fstop2)

    prompt= {'Fstop1','Fpass1','Fpass2','Fstop2'} defans=
    {'.2','.3','.4','.5'}
    fields = {'Fstop1','Fpass1','Fpass2','Fstop2'} info =
    inputdlg(prompt, 'BANDPASS FILTER!', 1,
defans)
    if ~isempty(info) %see if user hit cancel info =
    cell2struct(info,fields);
    Fs1 = str2num(info.Fstop1);%convert string to
number
    Fp1 = str2num(info.Fpass1); Fp2 =
    str2num(info.Fpass2); Fs2 =
    str2num(info.Fstop2);
    bandPassSpecObj = fdesign.bandpass
    set(bandPassSpecObj,
'Fstop1',Fs1,'Fpass1',Fp1,'Fpass2',Fp2,'Fstop2',Fs2)
    BandPassFilt = design(bandPassSpecObj)
    fvtool(BandPassFilt) %plot the filter magnitude
response end
```

NOTE: This is the main body of the program and supporting programs are written as calling functions which has not shown in above program.

III. Conclusion

Filter design using software, like MATLAB is very useful in avoiding hardware, is highly immune to noise and possesses considerable parameter stability, can be operated over a wide range of frequencies.

Future Scope

As the Future work, The frequency response can be changed by simply changing the filter coefficients.

Acknowledgement

I thank to my beloved Mr.P.SIDDIAH, Professor,ANU who gave me such a Opportunity,Initiation Encouragement in the Ph.D. program

References

- [1]. V.P.Kodali, “Engineering EMC Principles, Measurements and Technologies”, IEEE Press, Newyork, 1996.
- [2]. Clayton R.Paul,” Introduction to Electromagnetic Compatibility”, John Wiley Publications, 2008
- [3]. V. Oppenheim and T. A. Baran, 6.341x Discrete-Time Signal Processing, on edX, Spring 2015.
- [4]. MATLAB command prompt: Enter filter Designer.
- [5]. Wikipedia\EMI\Filters Design