International Journal of Research in Advanced Engineering and Technology
ISSN: 2455-0876; Impact Factor: RJIF 5.44
www.engineeringresearchjournal.com
Volume 2; Issue 3; May 2016; Page No. 150-156

The simulation and modeling of active filters

Chiagunye Tochukwu, Okereke Eze Aru, Udeani Henrietta

Computer Engineering Dept. Michael Okpara University of Agriculture, Umudike, Abia State Nigeria

Abstract

Filters are very vital electronic systems. They are basically used to remove unwanted signals. This project showcases the concept of electronic filters, as well as basic configurations of commonly used filters. The designs of these filters, including parameter value definition were presented. The systems are designed and simulated with MATLAB, which subsequently generates a plot of the filtering action of the various filters.

Keywords: Simulation, Modeling, MATLAB, Active Filter, Controller, Signal, Op-Amp

1. Introduction

Applications involving processing of signals from external analog sources; sensors to be precise usually require some kind of digital filtering. For extremely high filter performance, digital signal processors (DSP) are usually employed, however in many cases these are too expensive to deploy. Consequently, 8- or 16-bit Microcontrollers are used [1]. These are inexpensive, efficient, and have all the required Input/Output (I/O) features and communication modules which are absent in the DSPs. The Peripheral Interface Controllers (PIC) microcontrollers are excellent for signal processing applications due to their powerful architecture, strong instruction set and built-in multi-channel 10-bit Analog to Digital Converter (ADC) [1]. The PIC series further have a hardware multiplier, which is important in signal processing applications. Digital Signal Processors (DSPs) take real-life signals like voice, video, temperature, pressure, position, etc in their digital form.At this point, they are mathematically manipulate to generate the desired signals. A DSP is designed for performing mathematical functions like "add", "Subtract", "multiply" and "divide" very quickly. This processing operation is necessary to extract the important information encapsulated in them. Subsequently, the information can be displayed, analyzed, or converted to another type of signal that may be of use. In the real-life situation, analog sensors which are digitized with analog-digital converters detect signals such as sound, light, temperature or pressure and manipulate them.

Digital filtering form a very important function of the DSP. In fact, their extraordinary performance is one of the key reasons that DSP have become so popular [2]. Filters have two uses: signal separation and signal restoration. Signal separation is needed when a signal has been contaminated with interference, noise, or other signals [2]. For example, imagine a device for measuring the electrical activity of a baby's heart while still in the womb, the raw signal will likely be corrupted by the breathing and heartbeat of the mother. A filter might be used to separate these signals so that they can be individually analyzed. Signal restoration is used when a signal has been distorted in some way. For example, an audio recording performed with a poor equipment has to be filtered for a better representation of the sound.

Another example is the deblurring of an image acquired with an improperly focused lens, or a shaky camera. Moreso, noise may occur from wind or rain at an outdoor music presentation. Here, Filtering out sinusoidal components of the signal that occur at frequencies that cannot be produced by the music itself results in recording the music with little wind and rain noise. Sometimes the signal is corrupted not by noise, but by other signal frequencies that are of no interest. If the signal is an electronic measurement of a brain wave obtained by using probes applied externally to the head, other electronic signals are picked up by the probes, but the physician may be interested only in signals occurring at a particular frequency. By using digital filtering, the signals of interest only will be presented to the physician.

These problems can be solved with either analog or digital filters. Analog filters are cheap, fast, and have a large dynamic range in both amplitude and frequency. Digital filters, in comparison, are vastly superior in the level of performance that can be achieved. For example, Some low-pass digital filter has a gain of 1 +/- 0.0002 from DC to 1000 Hz, and a gain of less than 0.0002 for frequency above 1001 Hz. The entire transition occurs within only 1 Hz. Digital filters can achieve thousands of times better performance than analog filters. With analog filters, the emphasis is on handling limitations of the electronics, such as the accuracy and stability of the resistors and capacitors, but in contrast, digital filters are so good that the performance of the filter is frequently ignored. The emphasis shifts to the limitations of the signals, and the theoretical issues regarding their processing [2].

2. Materials and Methods

The system was conceived with the realization of the block diagram shown in fig 1 below. The block diagram of fig 1 below, shows the PIC microcontroller as the major unit of the project, since the Analog to Digital conversion takes place inside the Microcontroller and all the filtering processes also happens in the Microcontroller unit. The microcontroller has an ADC inbuilt, this is one reason that makes PIC Derivatives of the microcontroller excellent for signal processing and analysis. The visual output from the microcontroller is sent to the LCD and the audio output is sent to the speakers. The tone generator unit is made of a 555 IC configured in astable mode. It generates signal of IKHz which is fed to the microcontroller for processing

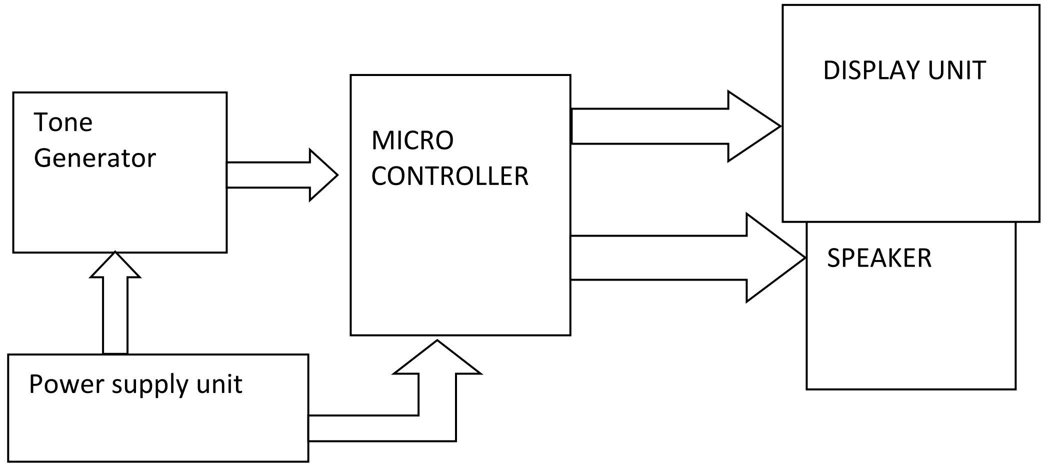

Fig 1: The block diagram of the system hardware unit.

A Tone Generator of 1 Hz was realized with a 555 timer IC configured in Astable mode to generate a 1Hz signal which was sent to the microcontroller for conversion and subsequent filtering. The PIC microcontroller with an inbuilt ADC was used to convert the analog signal from the tone generator into digital signal. The MCU uses the stored assembly program to manipulate this digitized signal, performing filtering, sending same to an inbuilt DAC for digital to analog conversion. Assembly language was used to develop codes for the microcontroller to enable it read the values sent by the Tone generator (555 IC) so as to take appropriate action.
The system realized was modeled and analyzed with Matlab 7.5. The generated Transfer Function of the Op-Amp filter circuit is fed into the Matlab. Here, the values of the parameters; resistance and capacitance were assigned to the model based on the design considerations. The Transfer Function was fed directly into the MATLAB, the op amp _tf command was then used to generate the Transfer Function values. For example, given the Transfer Function of a band pass filter as Eo/Ei = -[$C_1S/R_1C_1C_2S$][R_2/R_2C_2S+1] The following code are applied to MATLAB:

```
>> R1 = 20000; R2 = 500; C1 = 0.1E-6; C2 = 0.1E-6;
>> s = tf('s')
```

Transfer function:

s

*>> opamp_tf = -(C1*s/(R1*C1*s+1))*(R2/(R2*C2*s+1))*

Transfer function:

```
                -5e-005 s
      --------------------------
   1e-007 s^2 + 0.00205 s +
                    1
```

Here the MATLAB BODE command, bode(opamp_tf) is used to plot the Frequency Response of the system. The approach that was adopted in the implementation of the project was the method that allows the representation of the Transfer Function into numerator [num] and denominator [denum] values and input same into the MATLAB.

3. Basic Filter Designs

System Development

Using complex impedance analysis, the time domain parameters of an Op-Amp filters are converted into their equivalent frequency domain equivalence starting from the basic characteristics as governed by the Ohm's Law. Subsequently, using Laplace transforms, the impedance of these linear passive elements as functions of the Laplace variable, *S.* Ohm's Law in the frequency domain is E(*s*) = I(*s*) Z(*s*). Frequently for convenience, the (*s*) is dropped.

Table 1: Basic Voltage-Current Relationships

Element	Time Domain Voltage-Current Relationship	Frequency Domain Voltage-Current Relationship	Complex Impedance Voltage-Current Relationship
Resistor	$e = Ri$	$E(s) = RI(s)$	$Z(s) = R$
Capacitor	$e = \frac{1}{C}\int i\,dt$	$E(s) = \frac{1}{Cs}I(s)$	$Z(s) = \frac{1}{Cs}$
Inductor	$e = L\frac{di}{dt}$	$E(s) = LsI(s)$	$Z(s) = Ls$

A single pole Low-pass Filter with Op-Amp

The fig.2 below show the circuit diagram of the Op-Amp filter which is also referred to as the integrator.

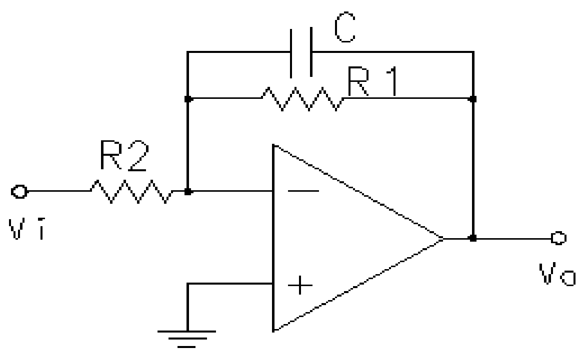

Fig 2: A Single pole low pass filter circuit

Complex impedance analysis is used to derive the Transfer Function of the circuit. Here, all elements in the circuit diagram to their equivalent complex impedances. In this case:

R_1, R_2 and C_1 are replaced with their equivalent complex impendence values, thus:

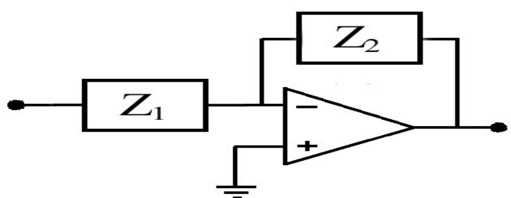

Fig 3: Complex Impedance representation

From the figure above, where Z_2 is the parallel combination of R_1 and C while $Z_1=R_2$

Therefore, with reference to op amp characteristics, the circuit is:

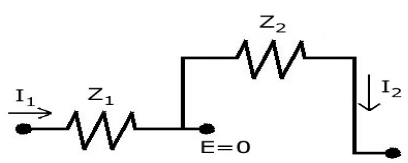

Fig 4: The reduced Op-Amp circuit

From the circuit above,

$$I_1(s) = E_i \frac{1}{Z_1} \quad \text{…1}$$

Applying Ohm's Law to Z_2:

$$I_2(s) = (0 - E_o)\frac{1}{Z_2} = -E_o\frac{1}{Z_2} \quad \text{…2}$$

Since $I_1 = I_2$:

$$E_i\frac{1}{Z_1} = -E_o\frac{1}{Z_2} \quad \text{… 3}$$

Rearranging:

$$\frac{E_o}{E_i} = -\frac{Z_2}{Z_1} \quad \text{… 4}$$

Subsitituting for the values of Z_1 and Z_2 gives the Transfer Function of the referenced FIR Digital filters is therefore computed as:

$Z_2=R_1//C$

$=[R_1*1/sC]/[R_1+1/sC]$ …5

$=[(R_1/sC)] / [(R_1sC+1)/sC]$,

Then dividing Num and Denum by sC gives;

$= R_1/(R_1 sC+1)$

$Z_1 = R_2$ …6

$Eo/Ei =-(Z_2/Z_1)$

Then subsitituting the values of Z_2 and Z_1 thus;

$Eo/Ei = -[R_1/(R_1 sC+1)] / R_2$ …7

Rearranging the equation gives

$Eo/Ei = -[(1/R_2C)] / [s+(1/R_1C)]$ …8

The Single Pole High-pass Filter with Op-Amp

Applying similar top-down design method used for the low-pass filter, the transfer function of a high-pass filter can also be derived by modeling fig 3 using Complex Impedance analysis.

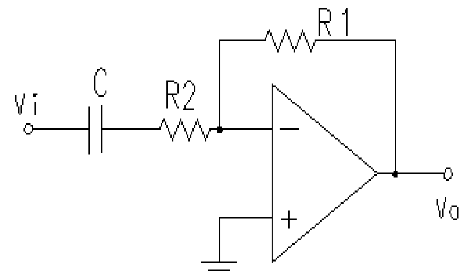

Fig 5: Single pole high-pass filter circuit.

The high-pass filter circuit can be represented in complex impedance notation as shown below.

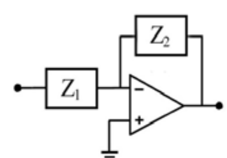

Fig 6: complex impedance representation

The transfer function can be obtained by using the relationship as derived in section 4 above. Thus

$$\frac{E_o}{E_i} = -\frac{Z_2}{Z_1} \quad \text{…9}$$

From fig 14, $Z_1 = R_2 + C$ and $Z_2 = R_1$.The transfer function will now be the substitution of their individual complex impedance values as shown below.

$Eo/Ei = - R_1/[R_2+(1/sC)]$ …10

$= [R_1] /[(R_2sC+1)/sC]$

$= [R_1sC] / [(R_2sC+1)]$

Rearranging the equation gives:

$$Eo/Ei = -[(R_1/R_2)s] / [s+(1/R_2C)] \quad \ldots 11$$

The equation represents the transfer function of a single pole first order high-pass with Op-Amp.

The band-pass filter with Op-Amp

The band-pass filter obtains from the parallel combination of high-pass and low-pass filters as shown fig 7.

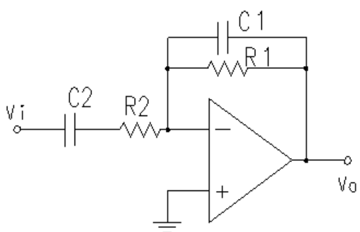

Fig 7: The band-pass circuit

The band-pass circuit can be reduced to as shown Fig 8.

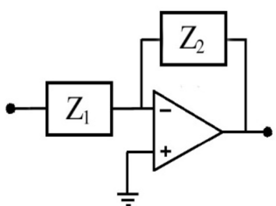

Fig 8: the complex impedance representation of the band-pass filter

Using complex impedance design analysis, the circuit can be modeled based on the Op-Amp assumptions to obtain the relationship shown.

$$\frac{E_o}{E_i} = -\frac{Z_2}{Z_1} \quad \ldots 12$$

From the circuit of fig 8, $Z_2=R_1//C_1$ and $Z_1=R_2 + C_2$.The transfer function of a band-pass filter can be obtained by subsitituting their various complex impedance values to equation above;

$$Eo/Ei = - [(R_1*(1/sC_1)) /(R_1+(1+sC_1))] / [(R_2+(1/sC_2))] \quad \ldots 13$$

$$= [R_1/ (R_1 sC_1+1)] / [(R_2sC_2+1)/sC_2] \quad \ldots 14$$

and rearranging the expression is shown in equation 15.

$$\frac{E_o}{E_i} = -\left(\frac{C_1 s}{R_1C_1 s+1}\right)\left(\frac{R_2}{R_2C_2 s+1}\right) \quad \ldots 15$$

4. System Modeling and implementation

Matlab 7.5 is deployed to help in the analysis and simulation of the modeled FIR filter circuits.The transfer function obtained for the various FIR digital filters are rearranged into vector format of denominator(den) and numerator (num).The resistor and the capacitor values are selected based on the design of the signal generation subsystem. The MATLAB program codes are generated and edited using m. file text editor. Here, the transfer functions obtained in the designs above are rearranged in a vector format to represent the den and num used in the MATLAB programming. The source codes and arrangements are used for the BODE plot which shows the relationship between the magnitude, phase and the frequency of the filter. The BODE plot as obtained after executing the source codes, are represented in the proceeding graphs.

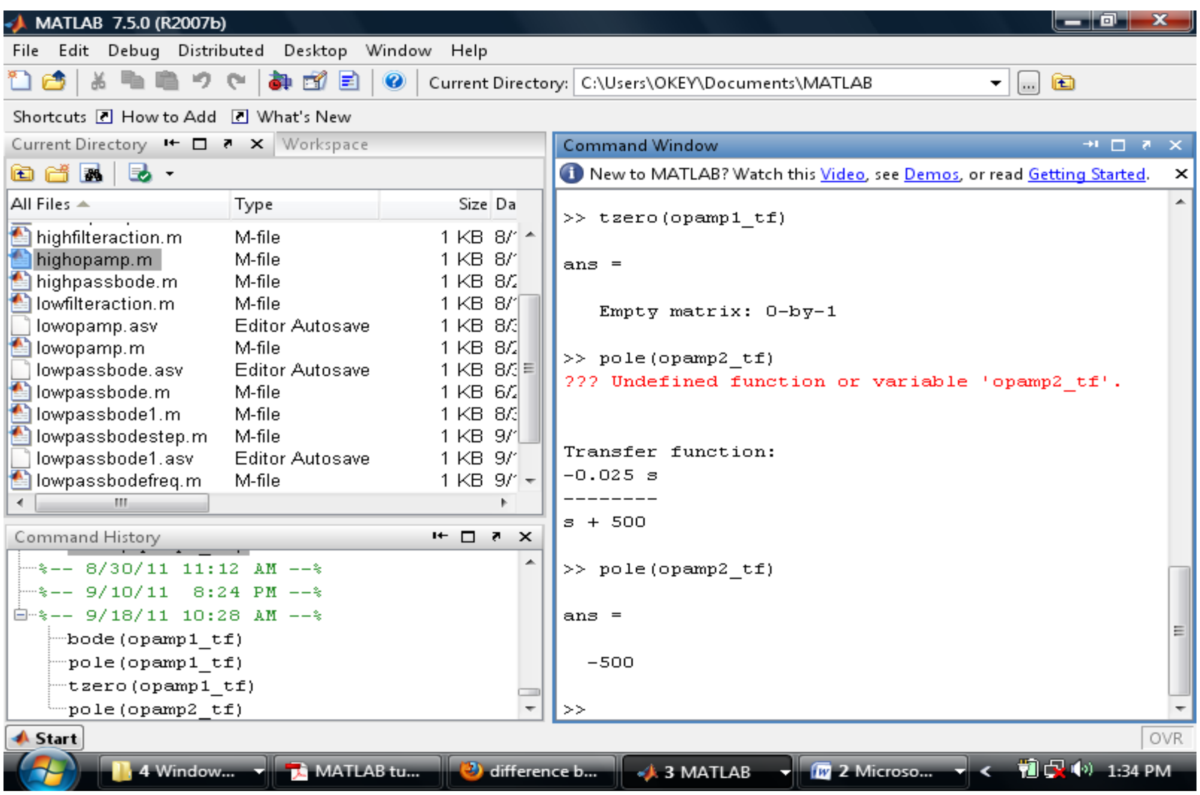

Fig 9: the pole values of the high-pass filter

Filtering action of a low-pass filter

The filter will only allow the signal with low frequency to pass while attenuating the signal with high frequency. The test process is achieved in MATLAB 7.5 using the lsim function. The program code is edited using the m. file editor and saved with. m extension as usual. The executable low-pass filter filtering testing is shown in fig 10.

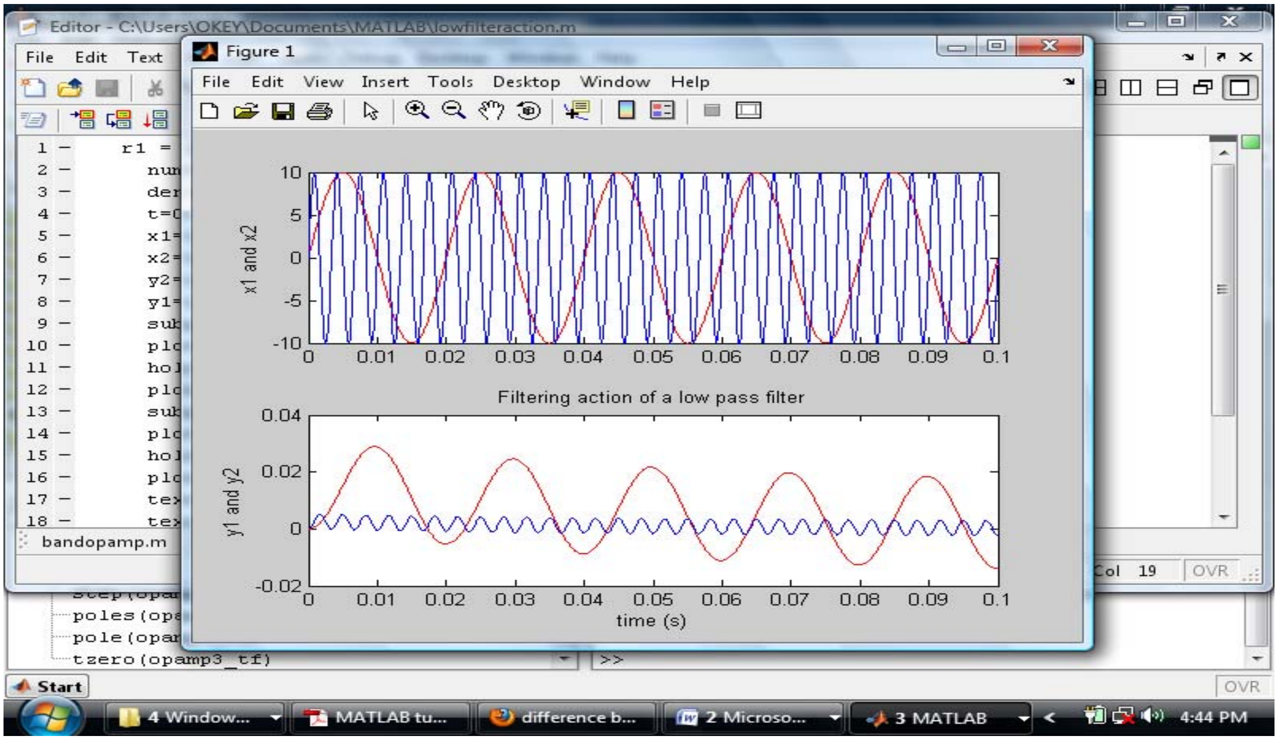

Fig 10: Filtering action low-pass filter

The filtering action of High-pass filter.

When two sinusoidal signals with different frequencies are pased to the designed high-pass filter, the filter will only allow the high frequency signal to pass attenuating the low frequency signal as shown in the plot obtained after executing the program.

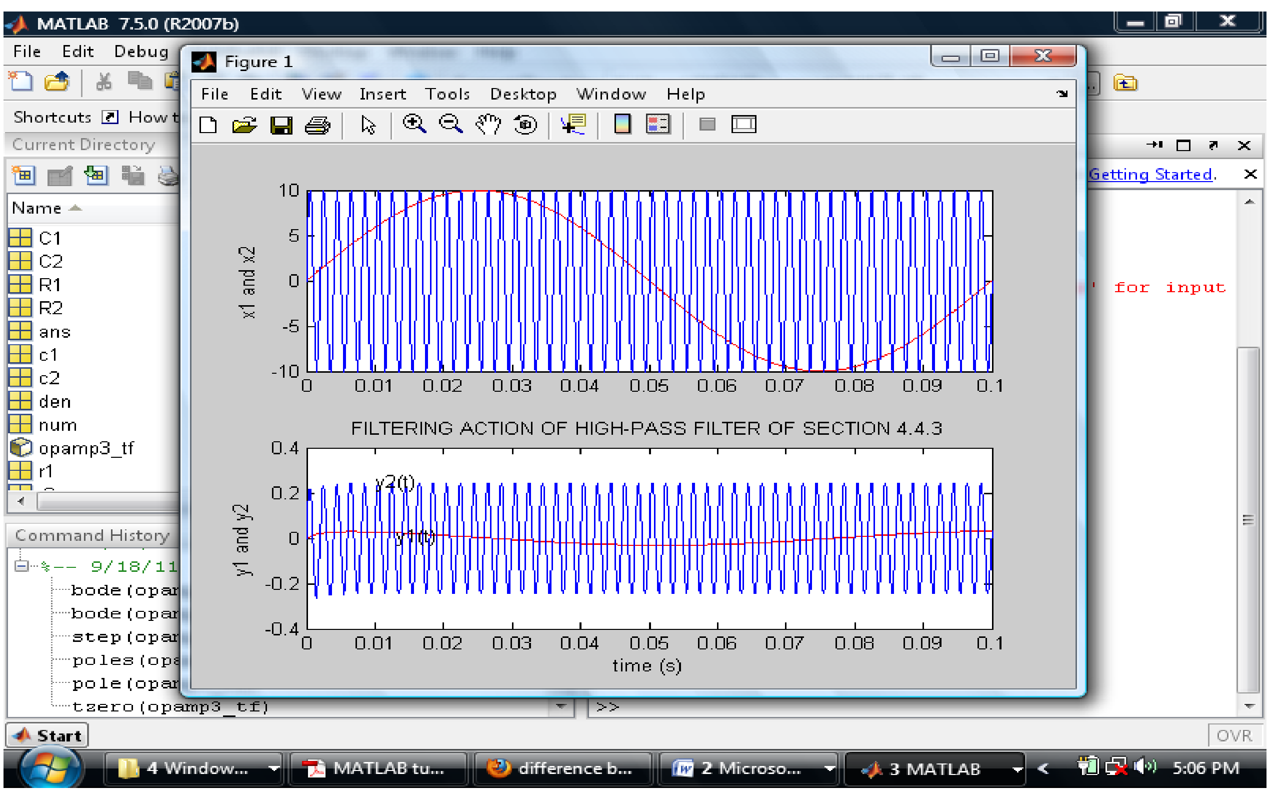

Fig 11: filtering action of a high-pass filter

The filtering action of Band-pass filter

The filtering process of the designed band-pass filter was also tested. The generated m. file program codes were executed and the results obtained were as shown in fig 12.

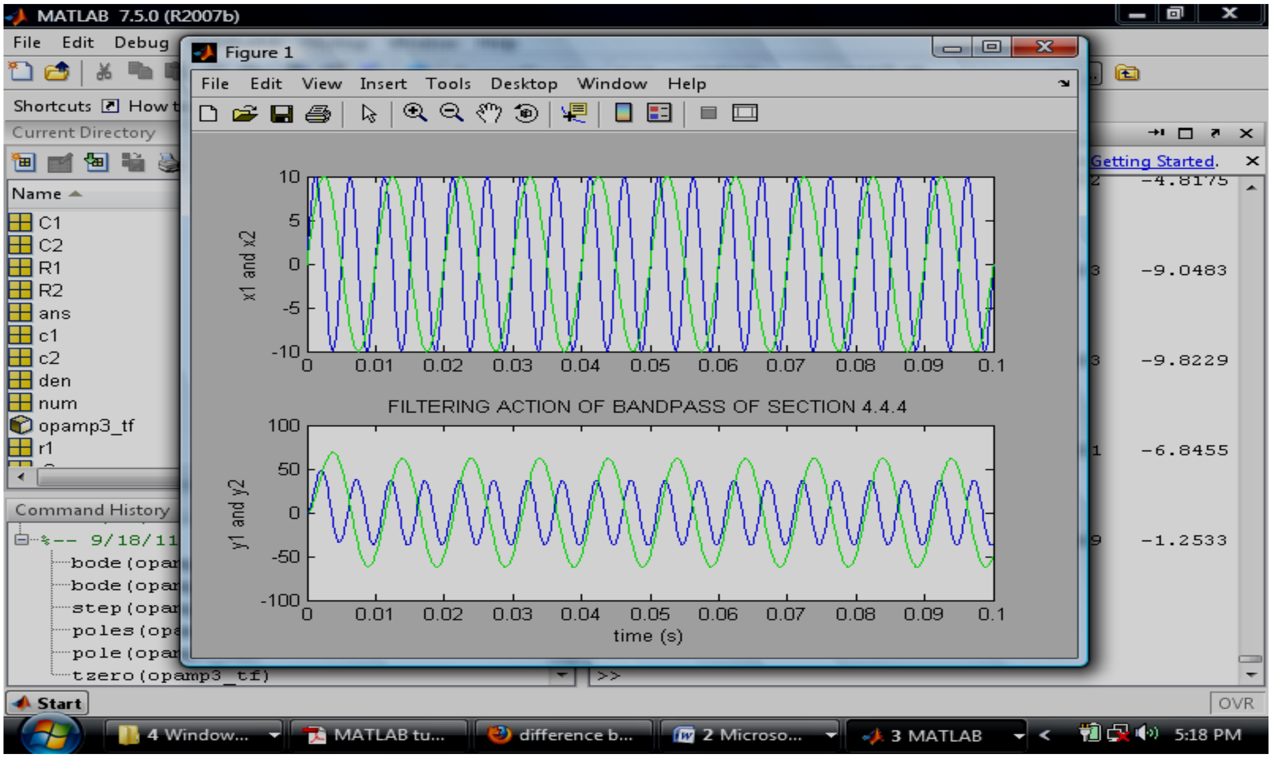

Fig 12: filtering process of Band-pass filter

5. Tests and Results

The filtering action of each of the designed FIR digital filters are also tested in the MATLAB by passing two signals of different frequencies into the MATLAB as input and observing the output signal. A function called Isim helps to separate this signal based on the desired output.

6. Recommendation and Conclusion

The project, Microcontroller-based digital filter was designed and implemented, the program can be deployed in high level languages such as Assembly, JAVA, C, C++ and C# and program modules can be imported as external function parameters in the use of such high level languages in the development of signal processing devices. Giving to the fact that MATLAB codes are readily incorporated into these high level languages, the execution and compilation delay is very minimal.

A signal processing operating system could be developed without great energy in running filters for function-specific signal processors. These devices can then be deployed in engineering labs for the training of staff and students. This program was developed using MATLAB 7.5 the use of a lower version to modify any part of this program could result to errors. Also, the presence of a strong graphic user interface platform possessed by Windows Vista also affects the kernel function application in a way that the processing is delayed for a varied time. This actually, will not affect the final result of the program. If for any reason, there happen to be a bug generated in the program procedure, a self-debugging event can be generated

7. References

1. Steven.W.Smith,2011.<http;//www.dspguide.com/ch14.htm>
2. Howstuffsworks.inc,2012<http;//electronics.howstuffwork.com/microcontroller4.htm>
3. Jullius.O.SmithIII,9th June2012,<https://ccrma.stanford.edu/~jos/filters/Examples_Digital_Filters.html>
4. UBM.Tech,2012<http://www.eetimes.com/discussion/beginner-s-corner/4024479/Introduction-to-Digital-Filters>
5. eWerks.inc,2012<http://www.ewerksinc.com/refdocs/IIR%20and%20FIR%20filters%20with%20PIC18>
6. Nicola.Asuni,2011,<http;//www.technick.net/public/code/cp_dpage.php:?aiocp_dp=guide_edft_008>
7. IVR Tech.inc,2012,<http://www.ivr.com/files/mbch1.pdf>
8. MikroElektronika,2012,<http://www.mikroe.com/eng/chapters/view/71/chapter-1-basic-concepts-of-digital-filtering-and-types-of-digital-filter>
9. Dorf and Bishop "Modern Signal Processing System" 10th Edition Prentice hall 2005
10. eWerks.inc,2012<http://www.ewerksinc.com/refdocs/IIR%20and%20FIR%20filters%20with%20PIC18.pdf>
11. Paul.Gu,2012<http://www.0x00f.com/files/ebooks/%B5Cs%20%20Fundamentals%20and%20Appl%20with%20PIC.pdf
12. MaximIntegrated,2012<http://www.maximic.com/products/microcontrollers/training/basics>
13. The Mathworks.Using Simulink Version 6.The MathWorks Natick,MA.1990-2004
14. Hahn, Essential MATLAB for Scientists and Engineers, John Wiley & Sons, New York, NY, 1997

15. http;//www.asic-soc.blogspot.com/2009/01/digital-filter-implemntation-using-html.
16. Larson RE, Edwards BH. Elementary Linear Algebra, Third edition, D.C. Heath and Company, Lexington, MA, 1996
17. G.Strang.Linear Algebra and Applications,Academic Press FL 2000
18. Chapman SJ. MATLAB Programming for Engineers. Thomson, 2004
19. Dorf, Bishop. Modern Signal processing systems", 10th Edition, Prentice Hall, 2005