

Matlab/DSK GUI for FIR Filter Design Using a TMS320C3X DSK

Michael W. Thompson
Michael_W_Thompson@baylor.edu
Department of Engineering
Baylor University
Waco, Texas

Joan M. Burleigh
Burleigh@engr.colostate.edu
Director, Center for Central Auditory Research
Department of Electrical and Computer Engineering
Colorado State University
Fort Collins, Colorado

Abstract

This paper takes a tutorial approach toward constructing a Matlab/DSK Graphical User Interface (GUI) for implementing Finite Impulse Response (FIR) filter designs on a TMS320C3X DSK. We have found that Matlab GUIs that interface with TI DSKs are useful for DSP applications in both the classroom and for our research program.

Introduction and Motivation

One of the challenges for instructors teaching DSP classes is to achieve a balance in teaching important theoretical concepts and providing practical exposure to DSP hardware. This paper discusses the use of Matlab/DSK GUIs that allow students to quickly explore DSP concepts in Matlab and then download their Matlab based designs to run on a TMS320C3X DSK. This approach provides students with exposure to DSP hardware without requiring DSP programming. For instructors who wish to include DSP programming in their courses, having students develop their own Matlab/DSK GUI application can serve as a “fun” motivational tool.

At the Center for Central Auditory Research we have also made use of Matlab/DSK GUIs for research applications. For several of our research studies we have the need to generate or alter the characteristics of auditory signals in real-time using TI DSKs. Matlab/DSK GUIs make it easy for audiologists at the Center (as shown in Figure 1) to quickly change the acoustic parameters in order to generate the desired test conditions.



Figure 1: Dr. Joan Burleigh conducts an audiological test using a Matlab/DSK GUI.

FIR GUI Example

In this example we will be building a Matlab GUI for implementing “window based” FIR designs on a TMS320C3X DSK. For improved clarity, the GUI developed here is relatively simple with only a few filter design options and no “user input” error checking features. Our task is to develop a Matlab/DSK GUI that will allow a user to choose FIR filter design parameters such as the filter order, the filter type and the choice of window function. After choosing the desired parameters, the user then can “click on a pushbutton” that will load and run the filter design on the DSK.

Step 0. To begin you must have Matlab with the signal processing toolbox and also the software that comes with the 3X DSK (which can also be downloaded from TI’s web site).

Step 1. Find the assembly language file fir.asm from the software that comes with the DSK and copy it to another file called myfir.asm . Open myfir.asm with a text editor (such as notepad) and look for the section of code shown below: Note that one could also use other DSK FIR filter implementation code such as that which can be found in [1].

```
;- - - - -
; FIR filter coefficients
;- - - - -
FIR_coef      .float      1.0          ; FIR filter coefficients
               .float      0.0          ;
               .float      0.0          ;
               .float      0.0          ;
               .float      0.0          ; These coefficients will create a
               .float      0.0          ; comb filter at Hz. By
inspection
               .float      0.0          ; all taps are zero except for the
0th
               .float      0.0          ; and 15th that are set to 1 and -1
               .float      0.0          ; respectively. Therefor a
cancellation
               .float      0.0          ; will occur for any signal which
is
               .float      0.0          ; 180' out of phase when delayed by
               .float      0.0          ; 15 samples. The Z transform
is...
               .float      0.0          ;
               .float      0.0          ;
               .float      0.0          ; -15
               .float     -1.0          ; H(z)=1-z
END_coef      ;
```

Delete this entire section of code and replace it with the following:

```
;- - - - -
; FIR filter coefficients
;- - - - -
               .include "coeffs.asm";
;-----
```

The idea here is that we will be generating the “coeffs.asm” file using Matlab to determine the filter coefficients.

Step 2. Generate a rough version of the Matlab/DSK GUI by first generating an x-y plot, (use the Matlab command `>plot(0,0)` to generate an x-y plot) and then using “guide” (Matlab command `>>guide`) to place the text boxes, list boxes, edit boxes and pushbuttons as shown in the figure below. Change the “String” field of each GUI element to match that of Figure 2. If you are unfamiliar with this process then we recommend that you consult the Matlab provided documentation entitled “Building GUIs with Matlab” [2].

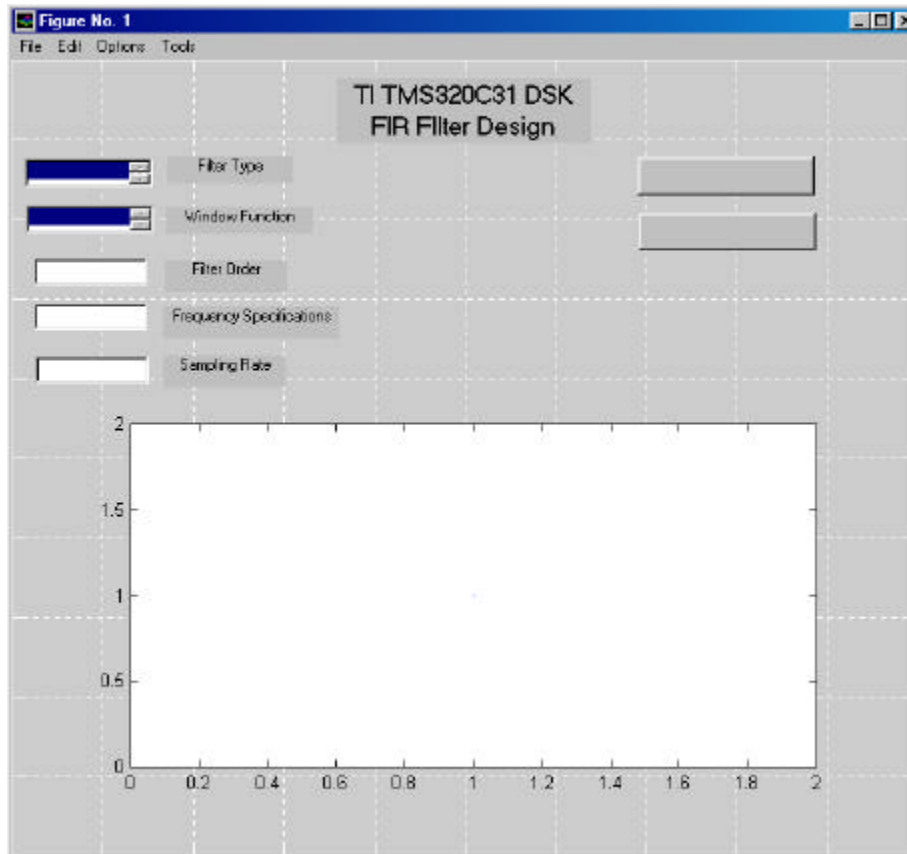


Figure 2: Rough Layout of Matlab/DSK GUI

Step 3: Edit the “Tag” field that correspond to the Filter Type and Window Function list boxes by changing the Tag to “Type” and “Window”, respectively. This step allows you to easily identify the two list boxes when the time comes to read the information that the user selects. Repeat this process for the “Filter Order”, “Frequency Specification”, and “Sampling Rate” edit boxes using the “Tag” names “Order”, “Specs”, and “Sampling,” respectively.

Step 4: In the Matlab command window, generate multi-line strings by entering `s1={'Low-Pass';'High-Pass';'Band-Pass';'Band-Stop'}`, and

```
s2={'Rectangular';'Bartlett';'Hamming';'Hanning';'Blackman'
    }
```

Set the “String” field in the Filter Type and Window Function list boxes to s1 and s2, respectively. This process generates the Filter type and Window function choices for the Matlab/DSK GUI.

Step 5: Edit the String field of the Pushbutton boxes to match that of the final version of the Matlab/DSK GUI shown in Figure 3. Also edit the Callback field for the appropriate pushbutton box to read “fir_gui_action plot” and “fir_gui_action program”. The idea is that upon pushing the “Plot Magnitude Response” pushbutton or the “Program DSP” button, the Matlab m-file called fir_gui_action will read the appropriate FIR filter design choices and then plot the magnitude response and/or initiate the programming of the DSK.

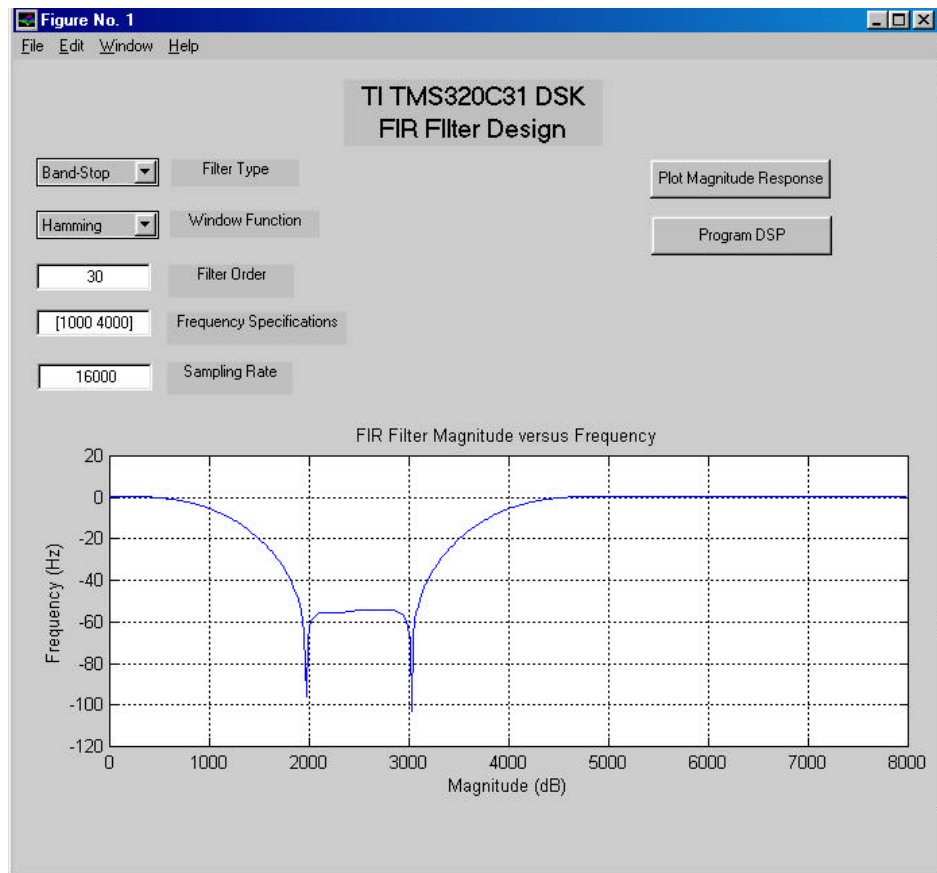


Figure 3: Completed FIR Graphical User Interface

Step 6. In this step Matlab Code is used to “drive the GUI”. Comment lines in the Matlab code indicate the steps that are needed to read the FIR filter parameters, plot the appropriate frequency response and to load the FIR filter coefficients and DSK code to the DSP board.

```
function firgui_action(invalue)
```

```

%Get the filter order
EditHandle=findobj(gcf,'Tag','Order');
N=get(EditHandle,'String');
N=str2num(N);

%Get the Sampling Rate
EditHandle=findobj(gcf,'Tag','Sampling');
Fs=get(EditHandle,'String');
Fs=str2num(Fs);

%Get the frequency spec
EditHandle=findobj(gcf,'Tag','Specs');
Fspec=get(EditHandle,'String');
Fspec=eval(Fspec)/(Fs/2);

% Get the filter type
EditHandle=findobj(gcf,'Tag','Type');
Val=get(EditHandle,'Value');
switch Val
case 1
    type=[]; %Matlab Defaults to Low Pass or Band Pass
case 2
    type='high';
case 3
    type=[];
case 4
    type='stop';
end

b=fir1(N,Fspec,type,boxcar(N+1)); % Let Matlab design the filter

% Window the impulse response.
EditHandle=findobj(gcf,'Tag','Window');
Val=get(EditHandle,'Value');
switch Val
case 1

case 2
    b=b.*bartlett(N+1)';
case 3
    b=b.*hamming(N+1)';
case 4
    b=b.*hanning(N+1)';
case 5
    b=b.*blackman(N+1)';
end

[H w]=freqz(b,1,500);
w=w/pi*Fs/2;
plot(w,20*log10(abs(H)));
title('FIR Filter Magnitude versus Frequency');
xlabel('Magnitude (dB)');
ylabel('Frequency (Hz)');
grid on

```

```

switch(lower(invalue))

case 'plot'

    %This case is here so that you can plot the magnitude without
    %programming the DSP - When you Program the DSP, the magnitude plot
    %will also be updated.

case 'program'

    % Make the FIR coefficients file
    fid=fopen('coeffs.asm','w');
    fprintf(fid,'FIR_coef      .float %12.8f\n',b(1));
    for k=2:N
        fprintf(fid,'      .float %12.8f\n',b(k));
    end
    fprintf(fid,'END_coef\n');
    fclose(fid);

    %Assemble (using DSK tools) and load/run the FIR filter on the DSK.

    dos('dsk3a myfir');
    dos('dsk3load myfir');

end

```

Summary and Conclusions

This paper provides a brief tutorial for building a Matlab/DSK GUI. The Matlab and DSK code needed to run this application can be downloaded from my web site at www.baylor.edu/~Michael_W_Thompson/my_stuff.htm.

References

Rulph Chassaing, *Digital Signal Processing: Laboratory Experiments Using C and the TMS320C31 DSK*, Wiley-Interscience, 1999.

Mathworks, Building Graphical User Interfaces (GUIs) with Matlab, available as of 7/2000 at www.mathworks.com/support/graphics/doc.shtml.