

Design and Verification of FIR Filter Based on Matlab and DSP

Chen-Long hu

The Third Research Institute of Ministry of Pubic Security, Shanghai, China

Email: ahutea@163.com

Abstract—The FIR filter is widely used in digital communication and control systems. This paper presents an analysis in detail of the principle of the FIR filter, and selects the proper window function to cut off impulse response, then presents the designing process with the aided tool in fdatool for obtaining the unit impulse response sequence. It also analyzes the FIR realization on DSP and gives the key program. The FIR filter conducts in AD interrupt service program after the converting of AD sample. Finally, it proposes a simple and rigorous validation scheme for this design with Fourier decomposition. The result shows that the FIR filter is accurate, stable, and enjoys a strong practicability and reliability.

Keywords—FIR digital filter; DSP program; fdatool; TMS320F2812

I. INTRODUCTION

A digital filter is an important component of a digital signal processing system, which is widely used in signal filtration, detection and prediction [1]. Compared with analog filtering, digital filters can avoid inherent voltage unbalance, temperature drift and noise, etc.; it can also meet the strict requirement of a filter to amplitude and phase, and has become a research hotspot in the modern detection system. IIR (limitless shock response) and FIR (limited shock response) are two kinds of commonly-used digital filters in industrial systems. The FIR filter is a recursive type on the structure and can randomly design amplitude characteristics and ensure accurate strict linear phase at the same time, which has been widely applied to digital signal processing field. The TMS320F2812 digital signal processor (DSP) has integrated the best features of a digital signal processor and a micro-controller. It can finish multiplication accumulative operations in one cycle with high-precision calculation and control for the control system of FIR filter. This article fully takes advantages of MATLAB and TMS320F2812 DSP to design FIR digital filters and uses a simple, yet rigorous verification scheme to prove reliability and practicability of design.

II. BASIC PRINCIPLE OF FIR WITH FDATool AIDED DESIGN

A. Basic Principle of FIR

According to the time domain characteristic of the impulse response, a digital filter can be classified into the infinite impulse response filter (IIR) and finite impulse response filter (FIR). The advantage of an FIR is its stability and easy realization of linear phase, so that it is widely used in communication, audio, video, automatic control and other fields.

The basic structure of an FIR filter is a delay line of sections. Summing all output sections with weight can get the output of the filter. $h(n)$ ($n=0, 1, 2, N-1$) is the filter's impulse response, and $x(n)$ is the input sequence, and $y(n)$ is the output sequence of filter; N order FIR filter can be expressed as [1]:

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k) \quad (1)$$

Where: $h(k) = h_d(k)W_N(k)$

$h_d(k)$ is the impulse response coefficient of an ideal filter, which is no-causal and infinite width, but in fact, the impulse response of the FIR filter should be limited, so it needs to add a window function to cut off in order to get the FIR filters with linear phase and causality. $W_N(k)$ is the right window function. Different window functions have different performances in transitional width and minimum stopband weakens, which can be chosen according to the design requirements.

Conduct the z-transform for Formula (1) to get the transfer function of FIR filters.

$$H(z) = \sum_{k=0}^{N-1} h(k)z^{-k} \quad (2)$$

From Formula (2), we can see the common structure drawing of FIR filters, which is shown in Figure 1.

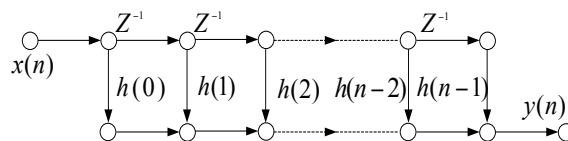


Figure 1. FIR filter structure drawing

We can get that the point of FIR filter design lies to choose a proper window function, which cuts its

impulse response to get a linear phase and causal FIR filter. The design methods mainly are the window function method, frequency sampling method and the best approximating method. In this paper, we utilize the window function method, which is the most commonly used.

B. The Design Steps of the Window Function

1. Given the required frequency response $H_d(e^{j\omega})$;
2. Calculate out the $h_d(k)$ by Fourier transform of $H_d(e^{j\omega})$;
3. According to the transition bandwidth and the minimum attenuation of stop band, it selects the window shapes and estimates the window length N ;
4. Calculate the unit impulse response of filter;
5. Check technical indicators of requirements;

Window function design method should meet the following [2]:

- Main lobe width of window should be as narrow as possible, in order to obtain a steeper transitional bandwidth;
- Minimize the amplitude of window maximum sidelobe, so the energy can gather to main lobe, thereby accelerating the stopband attenuation.

The given filter parameters generally are the cut-off frequency of passband ω_p , cut-off frequency of stopband ω_s , fluctuation of passband R_p , and minimum attenuation of stopband A_s . The empirical formula of window function designing:

$$\Delta\omega = \frac{\omega_s - \omega_p}{2\pi} \quad (3)$$

$$\text{The order of filter } M = \frac{A_s - 7.95}{14.36\Delta\omega}$$

$$\text{When } A_s \geq 50, \beta = 0.1102(A_s - 8.7)$$

$$\text{When } 21 \leq A_s \leq 50,$$

$$\beta = 0.5842(A_s - 21)^{0.4} + 0.07886(A_s - 21);$$

In the engineering, the window function is varied. The performance comparison of all window functions is shown in Table 1 [3].

TABLE 1. THE PERFORMANCE COMPARISON OF ALL WINDOW FUNCTIONS

| Window | Performance index of window | | Performance index of filter | |
|-------------|-----------------------------|-------------------------|----------------------------------|----------------------|
| | Main lobe width | Sidelobe peak value /dB | Minimum stopband attenuation /dB | Transition bandwidth |
| Blackman | $12\pi/N$ | -57 | -74 | $11\pi/N$ |
| Hamming | $8\pi/N$ | -41 | -53 | $6.6\pi/N$ |
| Hann | $8\pi/N$ | -31 | -44 | $6.2\pi/N$ |
| Rectangular | $4\pi/N$ | -13 | -21 | $1.8\pi/N$ |

In contrast, we can see the mainlobe of the rectangular window is narrow, so a steep transition band can be obtained, and its sidelobe peak amplitude is relatively small. The most energy is concentrated in the main lobe. So we use the rectangle window function.

C. Computer-Aided Design of FDATAOOL

FDATAOOL is a special filter designing analytical tool in the MATLAB signal processing toolbox. FDATAOOL can design almost all conventional filters, including various design methods of FIR and IIR and it has a simple and convenient operation [4].

Take the 25-order FIR low-pass filter with cut-off frequency $f_c = 3500\text{HZ}$ and sampling frequency $f_s = 10\text{KHZ}$ as an example to introduce the FDATAOOL aided design process:

- Open the MATLAB, input “fdatool” in the command window to call out the FDATAOOL working interface;
- Choose “Lowpass” in “Response type”;
- Choose “window” under FIR in “Design Method”;
- Input 24 into “Specify order”;
- Choose the Rectangular in “Window” of “Option”;
- Input $F_s = 10\text{KHZ}$, $F_c = 3500\text{HZ}$;
- Click the “Design Filter” and get the design data automatically.

The amplitude/frequency characteristic of the designed filter is shown in Figure 2. Then choose the “Generate C header” option under the menu of “Targets” and export impulse response arrays const real64_TB[25].

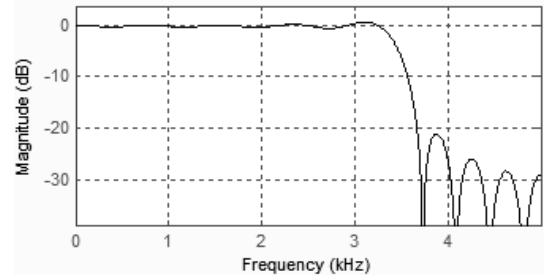


Figure 2. The amplitude/frequency characteristics of the designed filter

III. THE REALIZATION OF FIR DIGITAL FILTER BASED ON DSP

TMS320F2812DSP provides a 16 channel of 12-bit pipeline D/A converter with the highest sampling bandwidth of 12.5 MSPS, which can be used to collect the analog signals in realistic environment as the signal source of filter.

A. Realization Process

1) The filter coefficients and sampling data will be distributed two storage space $x(N)$ and $y(N)$ separately after DSP initialization and impulse response $h(N)$ will be deposited in ROM or RAM with continuous address. The sequence sampled by A/D one by one will be deposited in $x(0)$, $x(1)$, $x(N... - 1)$.

2) FIR filter operations: After clearing the accumulators, the reoccupy conduct two numerical multiplication correspondingly and the result will be accumulated and then deposited into accumulators, the result of which is $y(n)$. Because FIR filter enjoys the linear phase, $h(n)$ is symmetrical, so:

$$y(n) = \sum_{k=0}^{N-1} h(k)x(k) \quad (4)$$

3) Accumulator will be cleared when the new sampled value comes and the original data shift one bit in turn, so the high-bit data is extruded and new data is deposited in $x(0)$ storage location. Then restart accumulative operations as in Step 2 [5].

B. Programming

The program flowchart is shown in Figure 3. It first finishes DSP initialization in the main program, and then the CPU timer starts to count. When the counted number matches the right number given by the CPU timer, the timer produces an interrupt, and then AD transform and digital filter are completed in service subroutines of the interrupt [6], so the AD sampling frequency is decided by the cycle of timer 1.

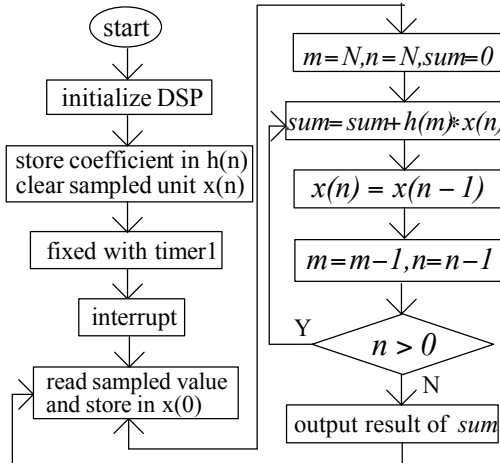


Figure 3. Programming flowchart

The subroutine of the timer interrupt:

```

interrupt void adc_isr(void)
{
  for(i=FIRNUMBER-1;i>0;i--)
  {
    fXn[i]=fXn[i-1];
    fXn[0]=AdcRegs.ADCRESULT0>>4;
    Voltage1[ConCnt]=AdcRegs.ADCRESULT0>>4;
  }
  if(ConCnt ==1024)
  {
    ConCnt =0;
  }
  else ConCnt ++;
  fSum=0;
  for(i=0;i<FIRNUMBER;i++)
  {
    fSum+=(fXn[i]*fHn[i]);
  }
  fOut[n]=fSum;
  if(n==1024)
  {
    n=0;
  }
  else n++;
  AdcRegs.ADCTRL2.bit.RST_SEQ1=1;
  AdcRegs.ADCST.bit.INT_SEQ1_CLR=1;
  PieCtrlRegs.PIEACK.all=PIEACK_GROUP1;
  return;
}
  
```

IV. EXPERIMENT

A. Scheme Selection

To test the design, it must obtain different frequency signal source stacked with each other, and the signal amplitude and frequency should be adjusted, which creates difficulties for verification. However, any periodic function can be separated into dc component, basal wave component and the superposition components of all $n > 1$ harmonics by Fourier decomposition; therefore, Fourier decomposition experiments can verify the design. Given that TMS160C2812DSP has strong peripherals function, it can easily generate a square wave signal, so Fourier decomposition experiment of periodic square wave is the most concise verification. This paper adopts the square wave signal with the frequency of 1KHZ generated by timer 2 in EVA as filtering source which outputted by the pin of T2PWM_T2CMP into the pin of ADCINA0 to sample, but the pin of DSP's own AD transform is very weak and the input voltage can not exceed the amplitude of 3V, so it can string together with two 1 KΩ resistances to distribute the voltage of square wave signal, which will be supplied ADINA0 directly.

The subroutine of producing 1 KHz square wave signal is:

```
void Pulse_generation (void)
{
    EvaRegs.T2CMPR=0x1ad0;
    EvaRegs.T2PR=0x3a98;
    EvaRegs.T2CNT=0x0000;
    EvaRegs.T2CON.all=0x1042;
    EvaRegs.T2CON.bit.TPS=2;
    EvaRegs.GPTCONA.bit.TCMPOE=1;
    EvaRegs.GPTCONA.bit.T2PIN=1;
}
```

B. Verification Results

In order to guarantee the verification effect, this paper carries out 25-order low-pass filter experiment and changes the cut-off frequency f_c in the FDATAOL to control the effect of verification. Because DSP can only process a digital signal, the sampled analog signal should be transformed as in Formula (5)

$$\text{digital value} = \frac{\text{analog value} - \text{ADCLO}}{3} \quad (5)$$

Where: ADCLO has connected to earth in hardware, the value of which is 0. There are digital values from Figure 4 to Figure 6.

Figure 4 is the square wave signal sampled by AD. There are basal wave and DC component after filtering except $n > 1$ harmonic component when $f_c = 1500\text{HZ}$ in Figure 5, and there are others besides above 3 times harmonic when $f_c = 3500\text{HZ}$ in Figure 6. From these figures we can see clearly that the design can well realize digital filter function.

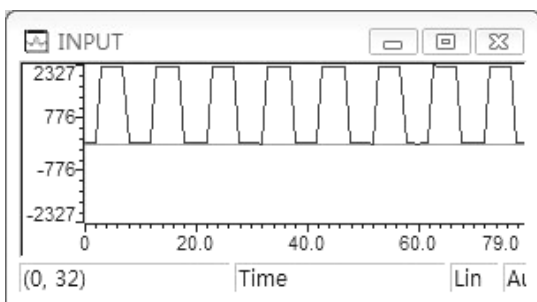


Figure 4. The input sampled signal of square wave

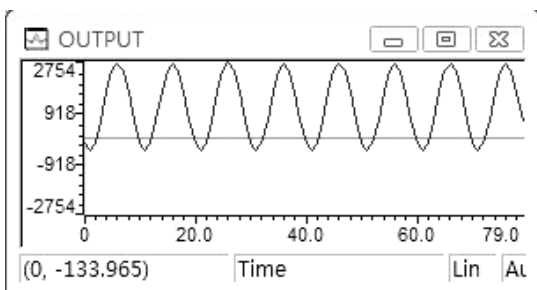


Figure 5. The output wave of filter when $f_c = 1500\text{HZ}$

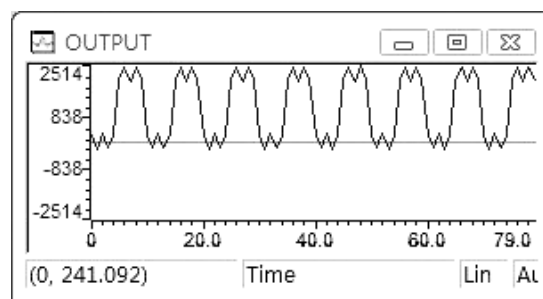


Figure 6. The output wave of filter when $f_c = 3500\text{HZ}$

V. CONCLUSION

Because of its strict linear phase, the FIR filter is widely applied in the digital signal processing system, but the traditional design of FIR is very complicated. In this paper, we can quickly and accurately design parameters of a digital filter using the MATLAB tool. TMS320F2812 DSP enjoys a characteristic of completing a multiply-accumulate operation in single period, especially suitable for high performance FIR filter, and the effect is very good. At the same time, the design of FIR filters can be used as a module added in the application system and as the front processing of digital signal processing.

ACKNOWLEDGEMENT

This paper is supported and financed by the Projects in the National Science & Technology Pillar Program (No. 2012BAH07B01).

REFERENCES

- [1] P. Q. Cheng *Digital signal processing tutorial*. Beijing: Tsinghua University Press, 2002.
- [2] M. Li, J. Wang, Q. Chen, and L. Ke. "The accomplishment of the design of FIR electric filter based on MATLAB program," *Journal of Electric Power*, vol. 23, no. 2, pp. 87-90, 2008.
- [3] Y. Wang. "FDA Tool applies in teaching of FIR filter design with the window function method," *Journal of Zhejiang University of Media and Communications*, vol. 6, pp. 69-71, 2009.
- [4] C. H. Dong. *Matlab signal processing and application*. Beijing: National Defense Industry Press, 2005.
- [5] Y. Liao, P. Li, A. Ruan, W. Li, and W. Li. "Design of area-efficient FIR filter based on ALU architecture," *Microelectronics*, vol. 40, no. 3, pp. 58-61, 2010.
- [6] T. Liu. "DSP realize FIR digital filters," *Journal of Wireless Communication Technology*, vol. 9, no. 2, pp. 53-56, 2000.