McCauley DSP FIR Filters

Mark DeArman

Introduction

As DSPs become available with increasing horsepower, Finite Impulse Response (FIR) filters are becoming more and more common. They were generally thought to be computationally inefficient in comparison to IIR design techniques. Their primary advantage is that an arbitrary magnitude response can be designed with exactly linear phase. It is important to understand a few facts about FIR filter design and implementation to use them effectively in the field since there are many varieties of realizations.

Mathematical Background

Let $H(e^{j\omega})$ be the frequency response of a arbitrary FIR digital filter, equation (1.2) should be its impulse response.

$$H(e^{j\omega}) = \sum_{-\infty}^{+\infty} h[n]e^{-j\omega n} \tag{1.1}$$

$$h[n] = \frac{1}{2\pi}\int_0^{2\pi} H(e^{j\omega})e^{j\omega n}d\omega \tag{1.2}$$

Without giving too much attention to the mathematics, besides noting that (1.1) and (1.2) are a Fourier pair, there is one thing which stands out right away. Since the bounds of the sum in equation (1.1) run from $-\infty$ to $+\infty$ there are two problems with it being a realizable FIR filter. First, it is not casual, since $h[n]$ is not generally zero for $n < 0$. Second, and most obvious, its duration is infinite.

If we are to realize this filter in a DSP then $h[n]$ should also be the coefficients we store in memory.

$$y[n] = \sum_{k=0}^{m-1} h[k] * x[n-k] \tag{1.3}$$

Equation (1.3) shows the code which executes inside the DSP to perform the filtering operation. To design a $h[n]$ which is of finite length m there are two common methods: Windowing and Optimization. The optimization method is computationally intensive, at least on a low clock rate DSP, but can produce highly accurate arbitrary magnitude response filters. Common implementations use a host computer to design a filter and then transfer the coefficients to the DSP to perform the filtering. The Window method if very efficient and can run in real time on a low clock rate DSP producing amazing results. It is especially useful when arbitrary magnitude design is not necessary.

Window Method

The M-Series DSP uses the windowing method to design filter coefficients on the fly based on the user's selection of cutoff frequency and the number of taps.

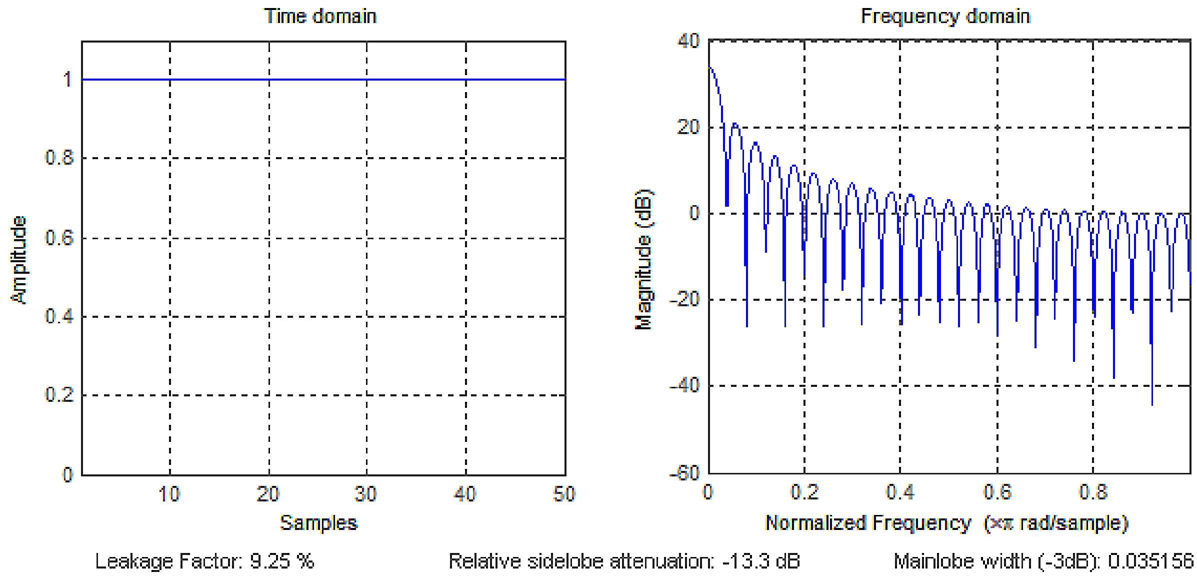

Figure 1

The simplest window which could be applied to $h[n]$ would be the rectangular window, or simply truncating it after m coefficients. Figure 1 shows an analysis of the simple truncation.

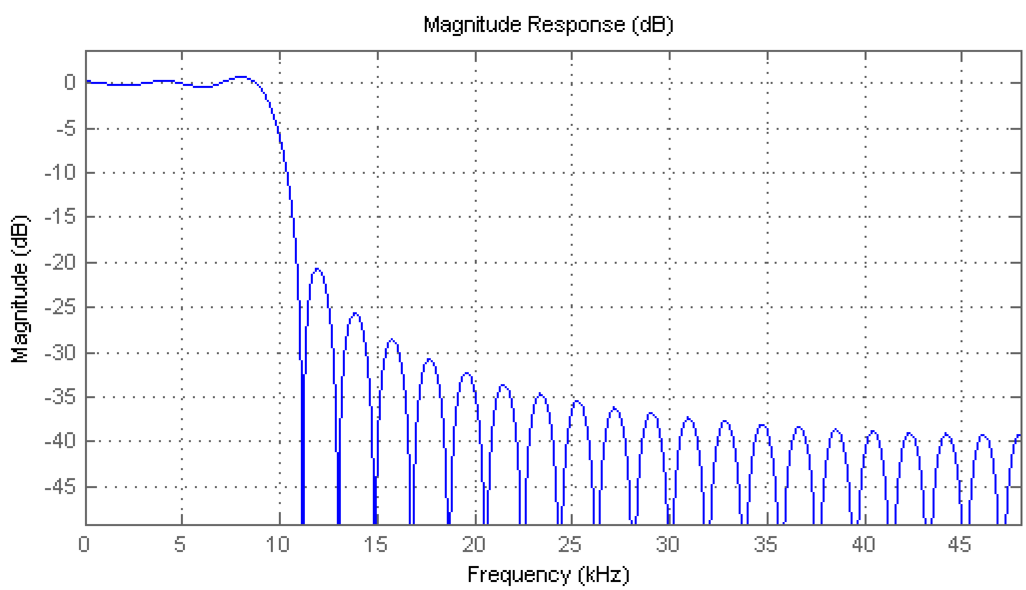

Figure 2

The filter produced using truncation shows how the large leakage produces pass band ripple and over/under shoot around the cutoff frequency. Figure 2 shows a low-pass 50 tap FIR with a cutoff frequency of 10 kHz. Figure 3 shows a low-pass filter with the same cutoff frequency and 200 taps. Increasing the number of taps does nothing to reduce the amplitude of the ripple; it simply confines the overshoot to a smaller region around the cutoff frequency [1].

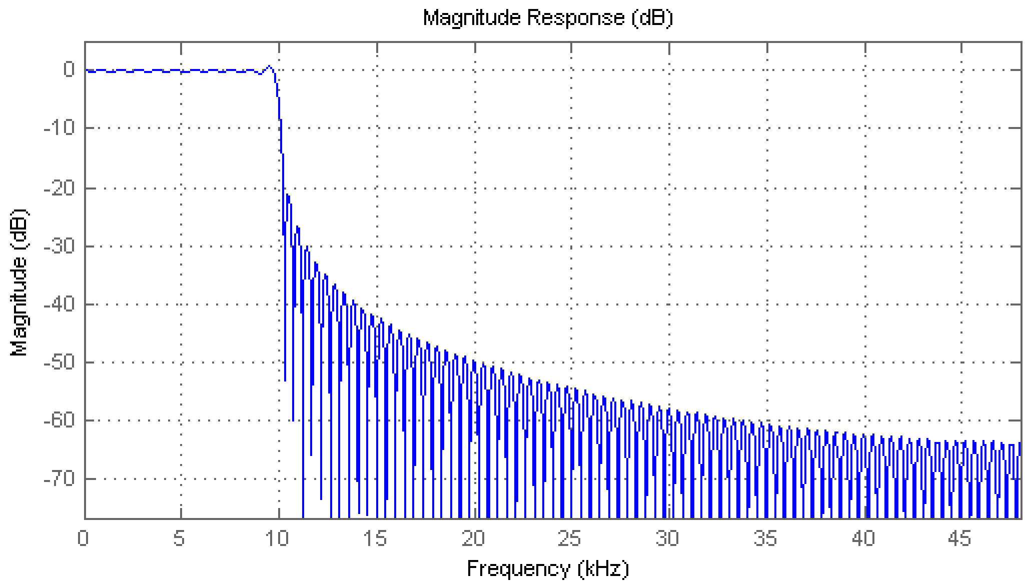

Figure 3

The author in [1] identified two criterions which are important to choosing a filter:

1. Small width of main lobe of frequency response containing as much of the total energy as possible.
2. Side lobes of the frequency response that decrease in energy rapidly as ω tends to π.

Stated more generally, the problem of finding a good windowing function, is finding a function which has time support and which is the best approximation of a function with frequency support. Of course, no function exists which can satisfy both qualities. Kaiser solved the problem in an optimal sense for continuous function and proposed a discrete time approximation known now as the Kaiser window.

$$w_K(n, \beta) \tag{1.4}$$

Without going into the mathematical intricacies, we can think of the definition of the Kaiser window as equation (1.4), where n is the length and β is a parameter which controls main lobe peak height versus side lobe ripples.

β	D	Passband Ripple (dB)	Stopband Ripple (dB)
2.120	1.50	±0.27	-30
3.384	2.23	±0.0864	-40
4.538	2.93	±0.0274	-50
5.658	3.62	±0.00868	-60
6.764	4.32	±0.00275	-70
7.865	5.0	±0.000868	-80
8.960	5.7	±0.000275	-90
10.056	6.4	±0.000087	-100

Table 1 - J. Kaiser, Bell Laboratories

Table 1 shows how β effects the FIR filter design. The M-Series DSP uses a value of $\beta = 12.265$, the time and frequency domain is plotted below in Figure 4. Figure 5 shows a chart of cutoff frequency versus number of taps N, such that the transition bandwidth, $D * N$ from Table 1, is at a minimum. Table 1 shows that as β is increased the transition bandwidth will widen.

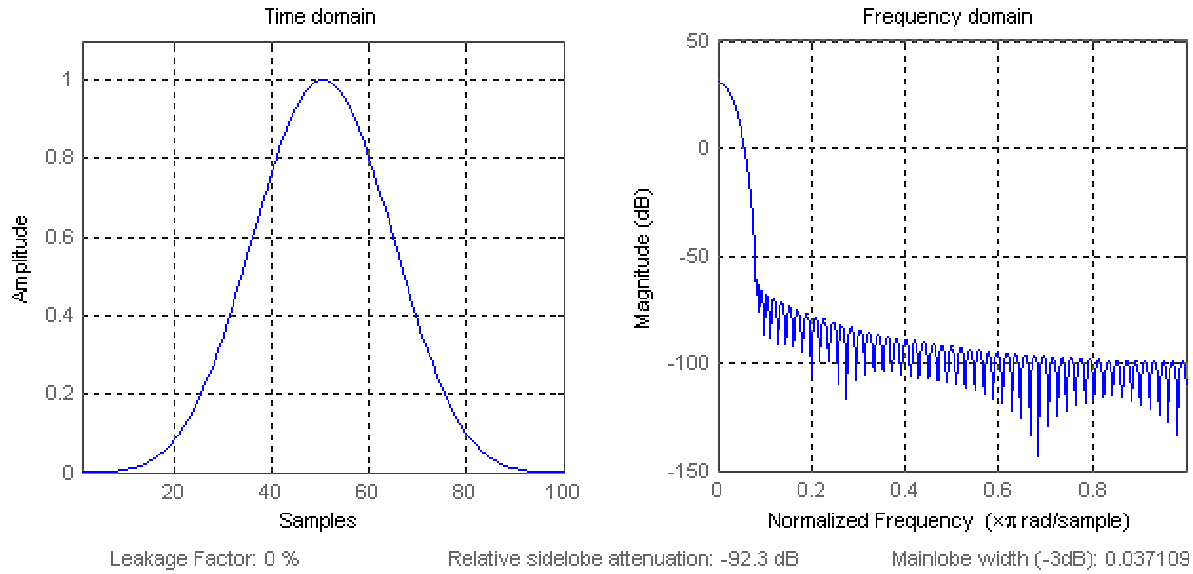

Figure 4

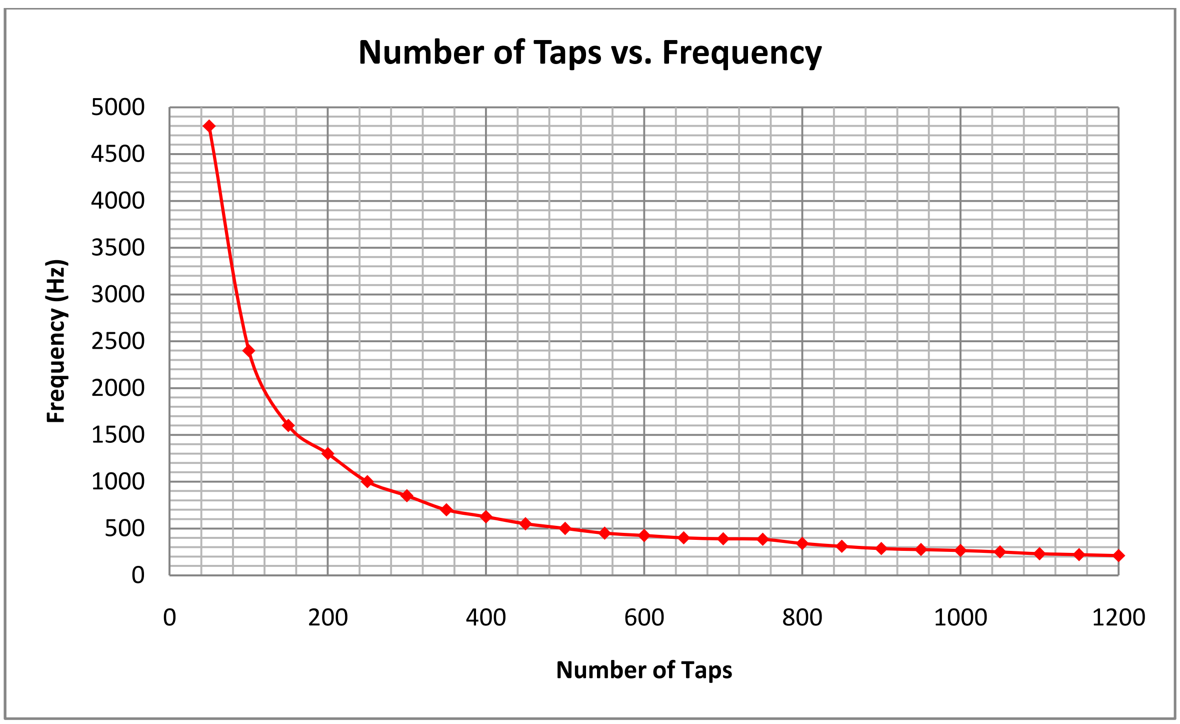

Figure 5

Figure 6 and 7 below repeat Figure 2 and 3, the blue dotted line, and display the Kaiser Window version overlaid in green solid. As can be seen, the Kaiser Window version has no visible pass band ripple and falls of quite far due to the increased frequency support. This is of course at the expense of cut off frequency sharpness. Increasing N, the number of taps, will lead to a steeper cut off region but increase transition bandwidth. With properly chosen parameters this effect is negligible.

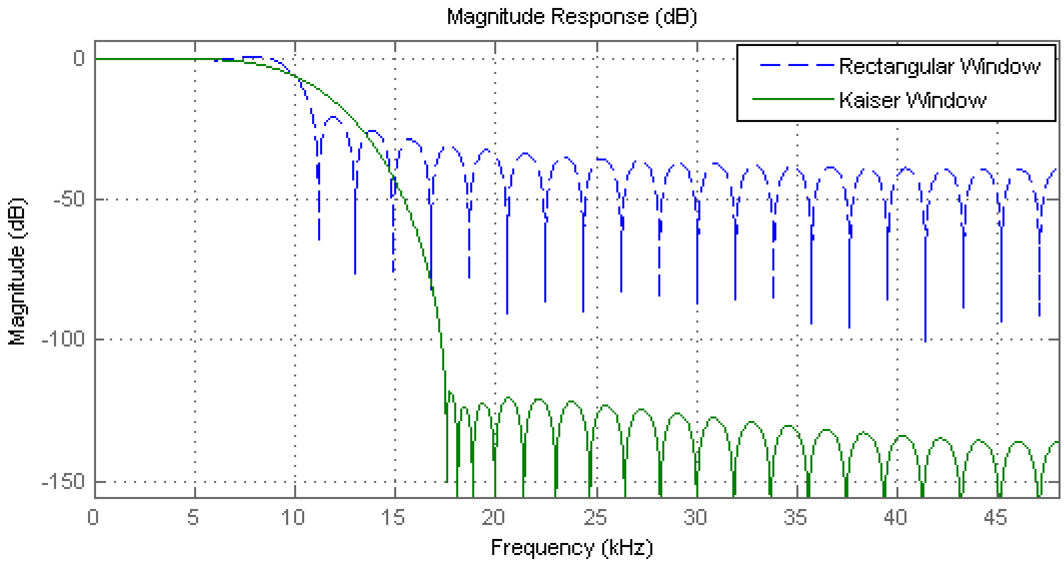

Figure 6

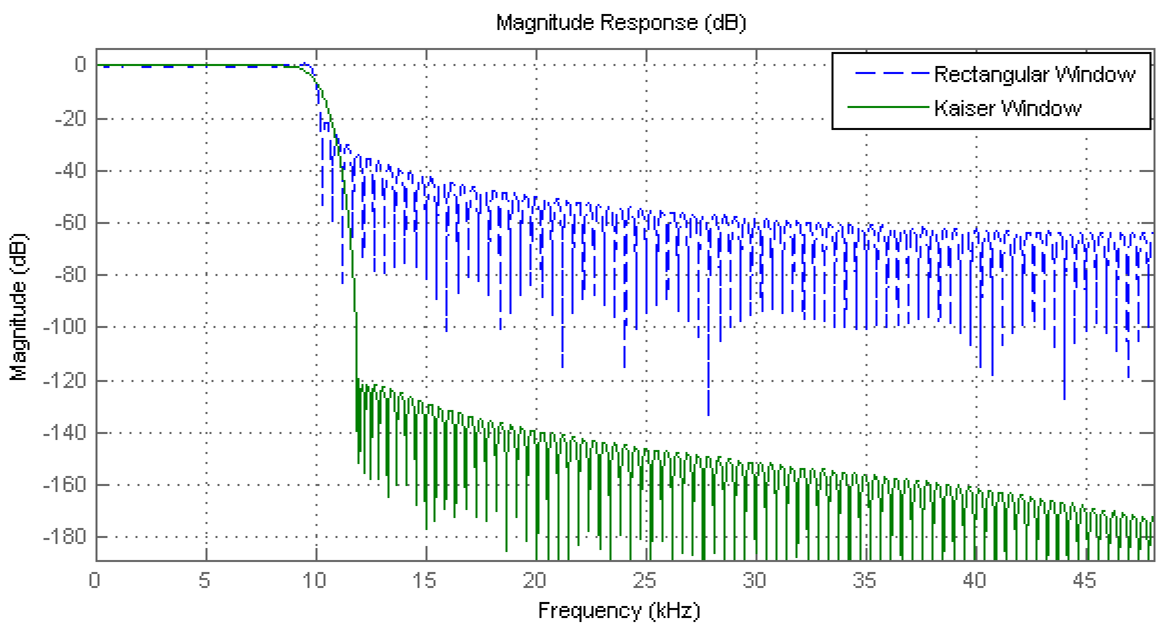

Figure 7

Since the M-Series DSP uses only 40-bit floating point representations of numbers, creating a filter with little frequency support can be problematic. For example, a high-pass filter with a cutoff frequency 10 Hz or a low-pass filter with a cutoff close to Fs/2 (48000 Hz). Figure 8 shows the impulse response of a 50 tap low-pass filter with a cutoff of 47980 Hz. Notice that the response is almost a perfect delta function with very small variations around zero symmetric from the impulse. Since the time support is so narrow this implies the frequency support must be broad. The filter will not behave like the tables and figure above predict. Also, the coefficients when convolved with the input signal to the DSP will have increased precision errors due to their disproportionally small size. The filter will effectively behave as a simple delay element equal to its number of taps.

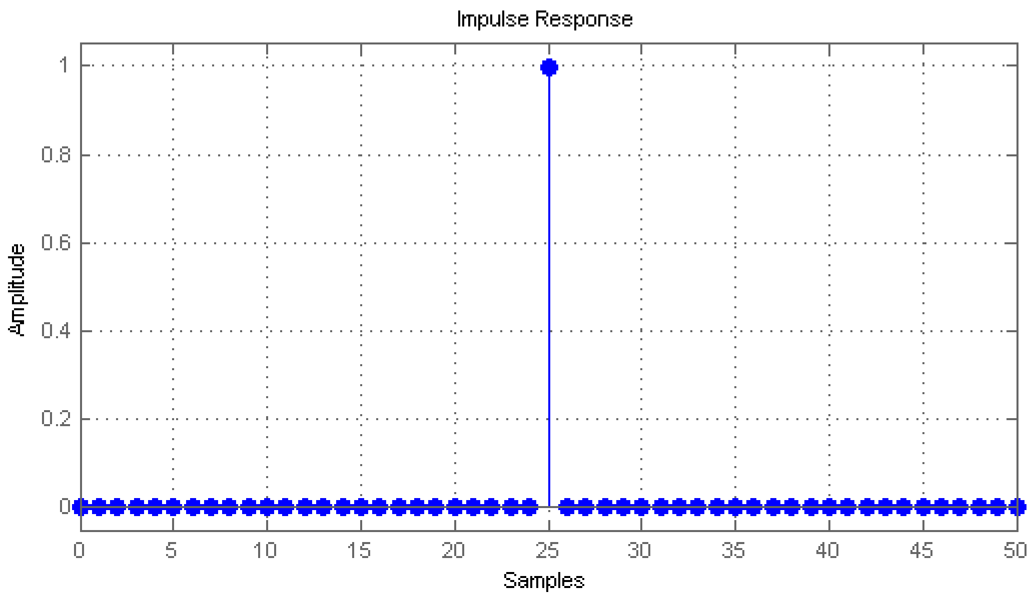

Figure 8

Conclusion

This paper presents a brief introduction to FIR filtering as implemented in the M-Series DSP. It is by no means exhaustive of this vast subject. Although the information presented here gives some rough guidelines for filter parameter choices, the best way to use the correct filter is still test and measurement in a specific situation.

References

[4] L. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing,* Englewood Cliffs, N.J. : Prentice-Hall, c1975.