LECTURE 16

DIGITAL FILTERS - III

We are now ready to consider the systematic design of non-recursive filters. The design method is based on the Figure 16-1, which has 6 parts. On the upper left is a sketch of the ideal filter you wish to have. It can be a low pass, a high pass, a band pass, a band stop, a notch filter, or even a differentiator. For other than differentiator filters you usually want either 0 or 1 as the height in the various intervals, while for the differentiator you want $i\omega$ since the derivative of the eigenfunction is

$$d/dt[\exp\{i\omega t\}] = i\omega \exp\{i\omega t\}$$

hence the desired eigenvalues are the coefficient $i\omega$. For a differentiator there is likely to be a cutoff at some frequency because, as you can see, differentiation magnifies, multiplies by ω, and is larger at the high frequencies, which is where the noise usually is, Figure 16-2. See also Figure 15-2.

The coefficients of the corresponding formal Fourier series are easily computed since the integrands of their expressions are straightforward, (using integration by parts when you have a derivative). Suppose we represent the series in the form of the complex exponentials. Then the coefficients of the filter are just the Fourier coefficients of the corresponding exponential terms. On the upper right of Figure 16-1 we have a sketch of the coefficients, symbolically, (they are, of course, complex numbers).

Next, we must truncate the infinite Fourier series to 2N + 1 terms, (meaning use a rectangular window), shown just below in Figure 16-1, with the corresponding Fourier representation on the left showing the Gibbs' effect.

Third, we then choose a window to remove the worst of this Gibbs' effect. The windowed coefficients are shown on the lower right, with the corresponding final digital filter on the lower left. In practice, you should round off the filter coefficients before evaluating the transfer function so that their effect will be seen.

In the method as sketched above, you must choose both the N, the number of terms to be kept, and the particular window shape, and if what you get does not suit you then you must make new choices. It is a "trial and error" design method.

J. F. Kaiser has given a design method that finds both the N and the member of a family of windows to do the job. You have to specify two things beyond the shape: the vertical distance you are willing to tolerate missing the ideal, labeled δ, and the transition width between the pass and stop bands, labeled ΔF, Figure 16-3.

For a band pass filter, with f_p as the band pass and f_s as the band stop frequencies, the sequence of design formulas is:

$$A = -20 \log_{10}\delta$$

$$N \geq (A - 7.95)/28.72\Delta F \qquad (N = \text{an integer})$$

If N is too big you stop and reconsider your design. Otherwise you go ahead and compute in turn:

$$\alpha = \begin{cases} 0.1102(A - 8.7) & 50 < A \\ 0.5842(A - 21)^{0.4} + 0.07886(A - 21) & 21 \leq A \leq 50 \\ 0 & A < 21 \end{cases}$$

(this is plotted in Figure 16-4). The original Fourier coefficients for a band pass filter are given by:

$$c_0 = 2(f_s - f_p)$$

$$c_k = (1/\pi k)[\sin 2\pi k f_s - \sin 2\pi f_p] \qquad (k = 1, 2, \ldots, N)$$

These coefficients are to be multiplied by the corresponding weights w_k of the window

$$w_k = I_0[\alpha\sqrt{\{1 - (k/N)^2\}}/I_0(\alpha) \qquad (|k| \leq N), \text{ and } 0 \text{ else}$$

where

$$I_0(x) = 1 + \text{SUM}[n=1,\infty; [(x/2)^n/n!]^2$$

$I_0(x)$ is the pure imaginary Bessel function of order 0. For computing it you will need comparatively few terms as there is an n! squared in the denominator and hence the series converges rapidly. $I_0(x)$ is best computed recursively; for a given x the successive terms of the series are given by

$$u_n = [(x/2)/n])^2 u_{n-1}$$

where $u_0 = 1$.

For a low pass or a high pass one of the two frequencies f_p or p_s has the limit possible for it. For a band stop filter there are slight changes in the formulas for the coefficients c_k.

Let us examine Kaiser's window coefficients, the w_k.

At $k = 0$ $\quad w_0 = I_0(\alpha)/I_0(\alpha) = 1$

At $k = N$ $\quad w_N = I_0(0)/I_0(\alpha) = 1/I_0(\alpha)$

As we examine these numbers we see that they have, for $\alpha > 0$, something like the shape of a raised cosine

$$a + b \cos x$$

and resemble the von Hann and Hamming windows. There is a "platform" when A > 21. For A < 21 then $\alpha = 0$, all the $w_k = 1$ and it is a Lanczos' type window. As A increases the platform gradually appears. Thus the Kaiser window has properties like many of the more popular ones, and the particular window you use is determined from your specifications via his window rather than by guess or prejudice.

How did Kaiser find the formulas? To some extent by trial and error. He first assumed that he had a single discontinuity and he ran a large number of cases on a computer to see both the rise time ΔF and the ripple height δ. With a fair amount of thinking, plus a touch of genius, and noting that as a function of A, as A increases we pass from a Lanczos' window, (A < 21), to a platform of increasing height, $1/I_0(\alpha)$. Ideally he wanted a prolate spheroidal function but he noted that they are accurately approximated, for his values, by the $I_0(x)$. He plotted the results and approximated the functions. I asked him how he got the exponent 0.4. He replied he tried 0.5 and it was too large, and 0.4, being the next natural choice, seemed to fit very well. It is a good example of using what one knows plus the computer as an experimental tool, even in theoretical research, to get very useful results.

Kaiser's method will fail you once in a while because there will be more than one edge (indeed, there is the symmetric image of an edge on the negative part of the frequency line) and the ripples from different edges may by chance combine and make the filter ripples go beyond the designated amount. In this case, which seldom arises, you simply repeat the design with a smaller tolerance. The whole program is easily accommodated on a primitive hand held programmable computer like the TI-59, let alone on a modern PC.

We next turn to the finite Fourier series. It is a remarkable fact that the Fourier functions are orthogonal, not only over a line segment, but for any discrete set of equally spaced points. Hence the theory will go much the same, except that there can be only as many coefficients determined in the Fourier series as there are points. In the case of 2N points, the common case, there is one term of the highest frequency only, the cosine term, (the sine term would be identically zero at the sample points). The coefficients are determined as sums of the data points multiplied by the appropriate Fourier functions. The resulting representation will, within roundoff, reproduce the original data.

To compute an expansion it would look like 2N terms each with 2N multiplications and additions, hence something like $(2N)^2$ operations of multiplication and addition. But using both: (1)

the addition and subtraction of terms with the same multiplier before doing the multiplications, and (2) producing higher frequencies by multiplying lower ones, the Fast Fourier Transform (FFT) has emerged requiring about N log N operations. This reduction in computing effort has greatly transformed whole areas of science and engineering - what was once impossible in both time and cost is now routinely done.

Now for another story from life. You have all heard about the Fast Fourier Transform, and the Tukey-Cooley paper. It is sometimes called the Tukey-Cooley transform, or algorithm. Tukey had suggested to me, sort of, the basic ideas of the FFT. I had at that time an IBM Card Programmed Calculator (CPC), and the "butterfly" operation meant that it was completely impracticable to do with the equipment I had. Some years later I had an internally progammed IBM 650 and he remarked on it again. All I remembered was that it was one of Tukey's few bad ideas; I completely forgot why it was bad - namely because of the equipment I had at that time. So I did not do the FFT, though a book I had already published (1961) shows that I knew all the facts necessary, and could have done it easily!

Moral: when you know that something cannot be done, also remember the essential reason why, so that later, when the circumstances have changed, you will not say, "It can't be done." Think of my error! How much more stupid can anyone be? Fortunately for my ego, it is a common mistake, (and I have done it more than once), but due to my goof on the FFT I am very sensitive to it now. I also note when others do it - which is all too often! Please remember the story of how stupid I was and what I missed, and not make that mistake yourself. When you decide that something is not possible, don't say at a later date that it is still impossible without first reviewing all the details of why you originally were right in saying it couldn't be done.

I must now turn to the delicate topic of power spectra, which is the sum of the squares of the two coefficients of a given frequency in the real domain, or the square of the absolute value in the complex notation. An examination of it will convince you that this quantity does not depend on the origin of the time, but only on the signal itself, contrary to the dependence of the coefficients on the location of the origin. The spectrum has played a very important role in the history of science and engineering. It was the spectral lines that opened the black box of the atom and allowed Bohr to see inside. The newer Quantum Mechanics, starting around 1925, modified things slightly to be sure, but the spectrum was still the key. We also regularly analyse black boxes by examining the spectrum of the input and the spectrum of the output, along with correlations, to get an understanding of the insides - not that there is always a unique insides, but generally we get enough clues to formulate a new theory.

Let us analyse carefully what we do and its implications, because what we do to a great extent controls what we can see. There is, usually, in our imaginations at least, a continuous signal. This is usually endless, and we take a sample in time of length 2L. This is the same as multiplying the signal by a Lanczos' window, a box car if you prefer. This means that the original signal is convolved with the corresponding function of the form (sin x)/x function, Figure 16-5, - the longer the signal the narrower the (sin x)/x loops are. Each pure spectral line is smeared out into its (sin x)/x shape.

Next we sample at equal spaces in time, and all the higher frequencies are aliased into lower frequencies. It is an obvious that interchanging these two operations, and sampling and then limiting the range, will give the same results - and as I earlier said I once carefully worked out all the algebraic details to convince myself that what I thought had to be true from theory was indeed true in practice.

Then we use the FFT, which is only a cute, accurate, way of getting the coefficients of a finite Fourier series. But when we assume the finite Fourier series representation we are making the function periodic - and the period is exactly the sampling interval size times the number of samples we take! This period has generally nothing to do with the periods in the original signal. We force all nonharmonic frequencies into harmonic ones - we force a continuous spectrum to be a line spectrum! This forcing is not a local effect, but as you can easily compute, a nonharmonic frequency goes into all the other frequencies, most strongly into the adjacent ones of course, but nontrivially into more remote frequencies.

I have glossed over the standard statistical trick of removing the mean, either for convenience, or because of calibration reasons. This reduces the amount of the zero frequency in the spectrum to 0, and produces a significant discontinuity in the spectrum. If you later use a window, you merely smear this around to adjacent frequencies. In processing data for Tukey I regularly removed linear trend lines and even trend parabolas from some data on the flight of an airplane or a missile, and then analyzed the remainder. But the spectrum of a sum of two signals is not the sum of the spectra - not by a long shot! When you add two functions the individual frequencies are added algebraically, and they may happen to reinforce or cancel each other, and hence give entirely false results! No one I know has any reasonable reply to my objections here - we still do it partly because we do not know what else to do - but the trend line has a big discontinuity at the end (remember we are assuming that the functions are all periodic) and hence its coefficients fall off like 1/k, which is not rapid at all!

Let us turn to theory. Every spectrum of real noise falls off reasonably rapidly as you go to infinite frequencies, or else it would have infinite energy. Figure 16-6. But the sampling process aliases the higher frequencies into lower ones, and the folding as indicated, tends to produce a flat spectrum - remember that the frequencies when aliased are algebraically added. Hence we tend to see a flat spectrum for noise, and if it is flat then we call it white noise. The signal, usually, is mainly in the lower frequencies. This is true for several reasons, including the reason that "over sampling" (sampling more often than is required from the Nyquist theorem), means that we can get some averaging to reduce the instrumental errors. Thus the typical spectrum will look as shown in the Figure 16-6. Hence the prevalence of low pass filters to remove the noise. No linear method can separate the signal from the noise at the same frequencies, but those beyond the signal can be so removed by a low pass filter. Therefore, when we "over sample" we have a chance to remove more of the noise by a low pass filter.

Remember, there is the implicit understanding that we are processing a linear system. The old stock market Fourier analysis that revealed that there was only white noise was interpreted to mean that there was no way of predicting the future prices of the stocks - and this is correct only if you intend to use simple linear predictors. It says nothing about the practical use of nonlinear predictors, however. Once again a wide spread misinterpretation of a result because of a lack of understanding of the basics behind the mathematical tool, and only knowing the tool itself. A little knowledge is a dangerous thing - especially if you lack the fundamentals!

I carefully said in the opening talk on digital filters that I thought at that time I knew nothing about them. What I did not know was that, because I was then ignorant of recursive digital filter design, I had effectively created it when I examined closely the theory of predictor-corrector methods of numerically solving ordinary differential equations. The corrector is practically a recursive digital filter!

While doing the study on how to integrate a system of ordinary differential equation numerically I was unhampered by any preconceived ideas about digital filters, and I soon realized that a bounded input, in the words of the filter experts, could produce, if you were integrating, an unbounded output - which they said was unstable, but clearly it is just what you must have if you are to integrate; even a constant will produce a linear growth in the output. Indeed, when later I faced integrating trajectories down to the surface of the moon where there is no air, hence no drag, hence no first derivatives explicitly in the equations, and wanted to take advantage of this by using a suitable formula for numerical integration, I found that I had to have a quadratic error growth; a small roundoff error in the computation of the acceleration would not be corrected and would lead to a quadratic error in position: an error in the acceleration produces a quadratic growth in position. That is the nature of the problem, unlike on earth where the air drag provides some feedback correction to the wrong value of the acceleration and hence some correction to the error in the position. Thus I have to this day the attitude that stability in digital filters means "not exponential growth" from bounded inputs, but allows polynomial growth, and this is not the standard stability criterion derived from classical analog filters, where if it were not bounded you would melt things down - and anyway they had never really thought hard about integration as a filter process.

We will take up this important topic of recursive filters, which are necessary for integration, in the next Lecture.

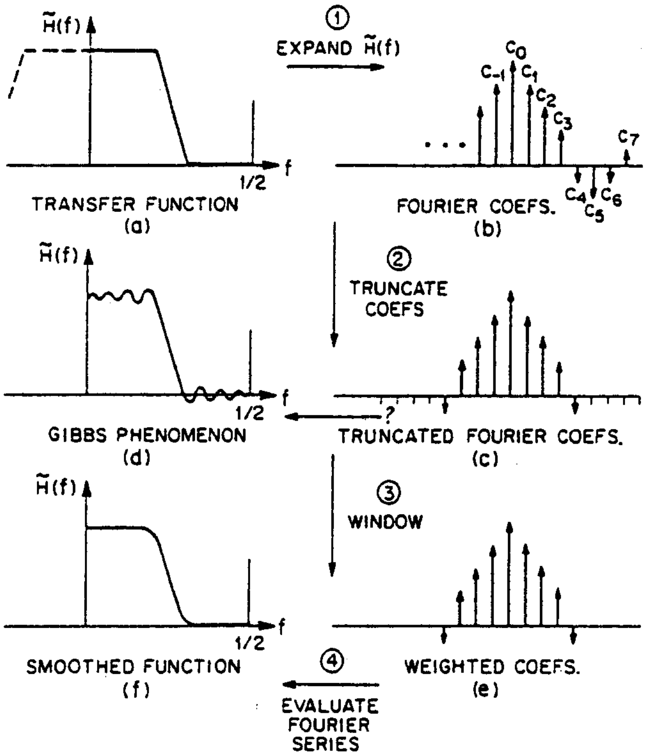
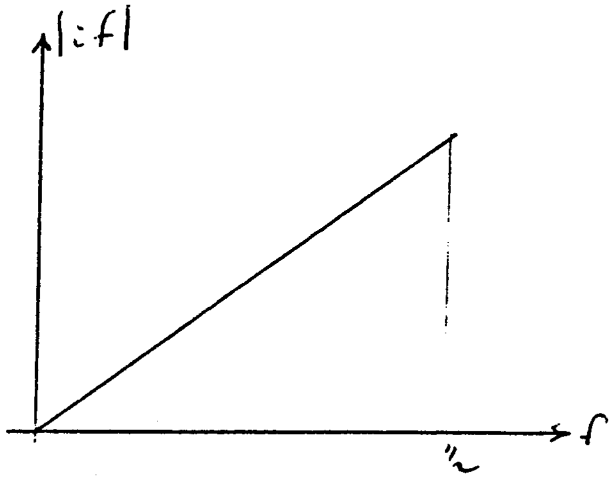
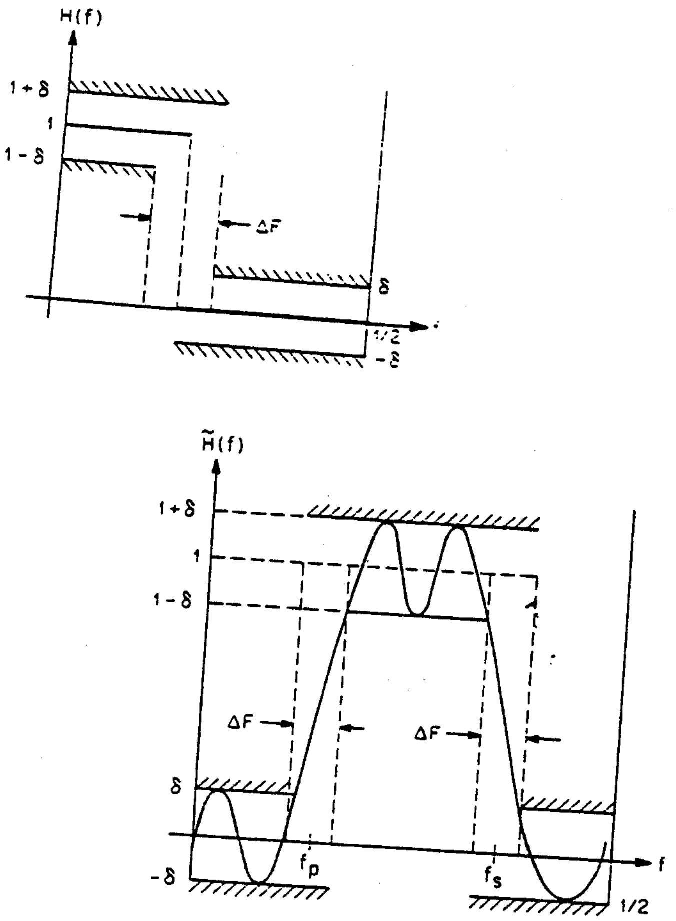


Figure K-1



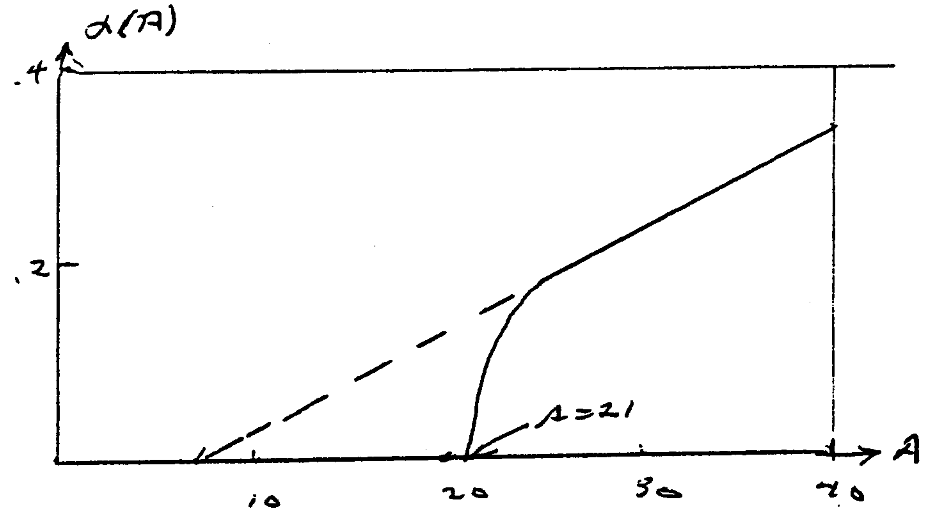
Amplification for differentiation

Fig K-2



BAND PASS FILTER

Figure 16-3



α(A)

Figure 16-~~B~~4

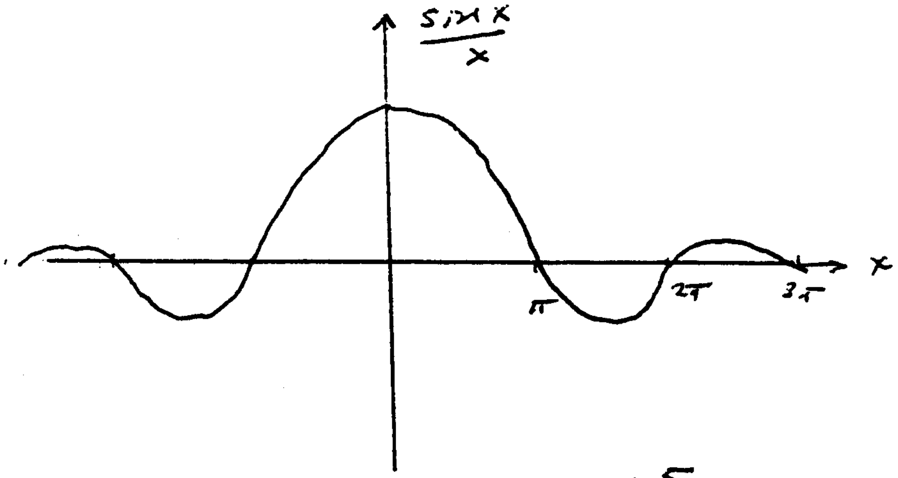


Figure 16-~~X~~5

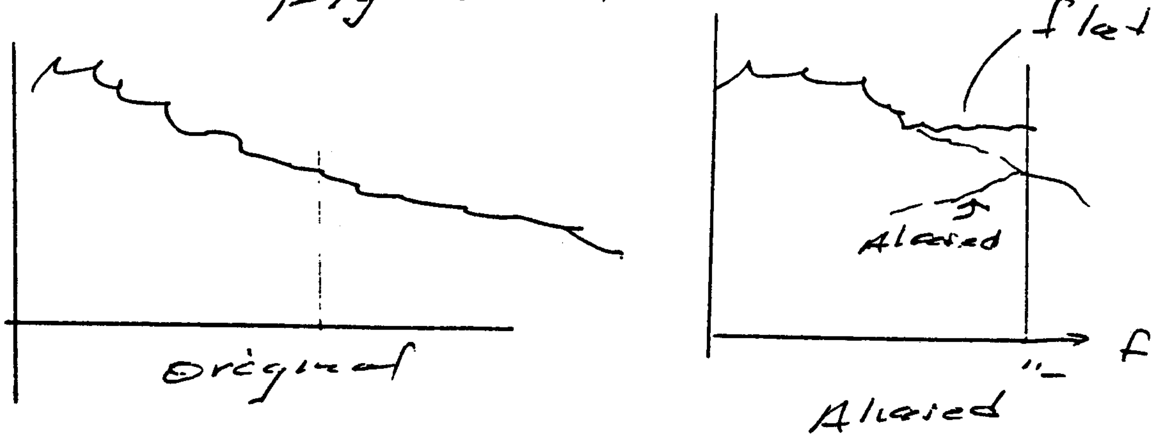


Figure 16-~~F~~6