Chapter 3: Sampled Signals PhL 6.19.91
last update 3.15.03

In Sections 20-26, we shall used symbols Δt, T_1, t_n, and ω_1 frequently. We use whichever symbol seems most convenient at the moment. (See the footnote on the first page of Chapter 2 for why we use a "1" subscript.) Δt is going to be the time spacing between samples of an analog signal. We define $T_1 = \Delta t$ to make a connection with earlier work where T_1 was the spacing between pulses we superposed to make a pulse train. As before, $\omega_1 = 2\pi/T_1$. Symbol $t_n = n\ \Delta t$ represents the particular times we choose to examine some signal. So:

$T_1 = \Delta t$

$\omega_1 = 2\pi/T_1 = 2\pi/\Delta t$

$t_n = n\ \Delta t = n\ T_1$

20. Sampled signals and their Image Spectra.

As a specific application of our results as boxed in Section 14, we consider the case where our pulse $x_{pulse}(t)$ is a delta function, so we have a pulse train x(t) which is an infinite sequence of these delta functions spaced by amount T_1. As usual, we use $\omega_1 = 2\pi/T_1$.

$x_{pulse}(t) = \delta(t)$

$X_{pulse}(\omega) = 1 \qquad c(\omega) = 1/T_1$

$$x(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT_1) \tag{20.1}$$

$$X(\omega) = (1/T_1) \sum_{m=-\infty}^{\infty} 2\pi\delta(\omega - m\omega_1) \tag{20.2}$$

If we now multiply the delta function sequence x(t) times some reasonable continuous signal y(t), the result is a set of delta spikes which are amplitude modulated by the values that y(t) takes at the spike points $t_n = nT_1$. This product we shall call w(t), it is our "sampled signal", and ω_1 is the (angular) "sampling rate".

$$w(t) = x(t)\ y(t) \tag{20.3}$$

Notice that, whereas y(t) might have some normal dimensions such as "volts", our x(t) has dimensions of 1/time due to the delta functions, so w(t) would then have dimensions "volts/time". This difference in dimensions between y(t) and w(t) causes a factor of T_1 to appear on the left of (20.5) below, because $W(\omega)$ and $Y(\omega)$ inherit the dimensional difference between w(t) and y(t).

According to the "reverse" convolution theorem stated at the end of Section 3, (20.3) implies

$$W(\omega) = (1/2\pi) \int_{-\infty}^{\infty} d\omega' \, Y(\omega - \omega') \, X(\omega') \qquad (20.4)$$

Inserting (20.2) into (20.4) quickly yields a famous result

$$T_1 W(\omega) = \sum_{m=-\infty}^{\infty} Y(\omega - m\omega_1) \qquad (20.5)$$

which we now summarize:

$$w(t) = \sum_{n=-\infty}^{\infty} \delta(t - t_n) \, y(t) = \sum_{n=-\infty}^{\infty} \delta(t - t_n) \, y(t_n) \qquad (20.6)$$

$$T_1 W(\omega) = Y(\omega) + \sum_{m \neq 0} Y(\omega - m\omega_1) \qquad (20.7)$$

The first term on the right side of (20.7) is the good old Fourier Integral spectrum of y(t). The second term is a set of identical copies of $Y(\omega)$ that are shifted by all possible multiples of ω_1. Usually these are called "image spectra", and the term $Y(\omega)$ is called the "main spectrum". Here is a picture:

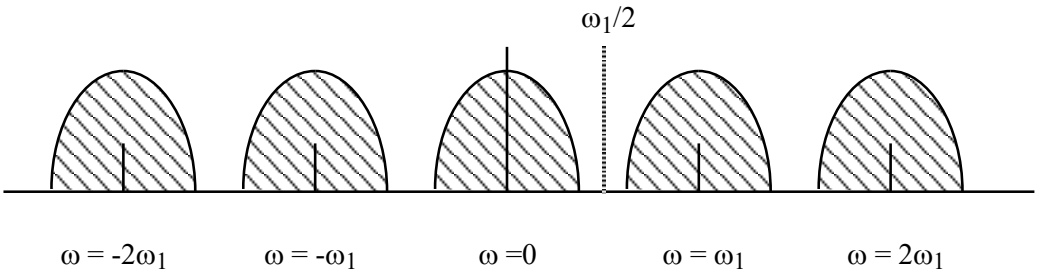


Figure 20.1. Example of a main spectrum with four of the image spectra.

Thus, by sampling signal y(t) with delta functions to create the sampled signal w(t), we have picked up an infinite set of "image spectra" in addition to the main spectrum $Y(\omega)$.

If the spectrum $Y(\omega)$ of the "reasonable" original signal y(t) completely cuts off at or below $\omega_1/2$ (as shown in Figure 20.1), then the image spectra are completely disjoint. If you then run signal w(t) through a low pass filter that removes all these image spectra, you end up with $T_1 W(\omega) = Y(\omega)$. And from $Y(\omega)$, one can presumably reconstruct y(t). Thus, these image spectra can be "dealt with".

Of course if the spectrum of y(t) goes beyond $\omega_1/2$, then the spectra in (20.7) overlap, and it is impossible to recover Y(ω) by itself by using a low pass filter. You would at least pick up something from the m=1 image spectra, and this results in aliasing. Another picture:

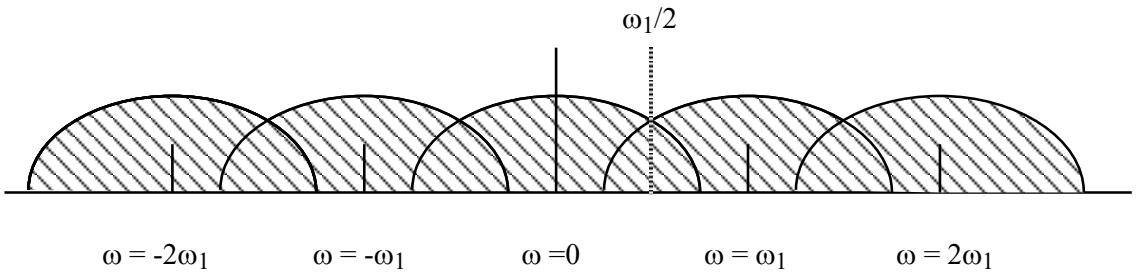


Figure 20.1. Here the image spectra overlap the main one. This is bad news.

People usually associate the Nyquist name with this $\omega_1/2$ boundary, but the whole business really doesn't amount to a hill of beans, as we have just shown.

21. Digital Filters and their Image Spectra.

In Section 3 we derived the "normal" convolution theorem, which we repeat here:

$$a(t) = \int_{-\infty}^{\infty} dt'\, b(t-t')\, c(t') \qquad \text{sometimes written} \qquad a = b * c \tag{21.1}$$

$$A(\omega) = B(\omega)\, C(\omega) \tag{21.2}$$

A *digital* filter can only approximate the continuous integration shown in (21.1). What a digital (FIR) filter really does is this,

$$a(t_n) = \sum_{m=-\infty}^{\infty} \Delta t\; b(t_n - t_m)\, c(t_m) \qquad \text{where} \quad t_n = n\,\Delta t \tag{21.3}$$

But these things are just numbers -- the values the functions a, b and c take at particular times, so you could just as well write this as

$$a_n = \sum_{m=-\infty}^{\infty} \Delta t\; b_{n-m}\, c_m \tag{21.4}$$

Think of the "digital filter" as a set of numbers b_k ; c_k is the input signal to the filter, and a_n is the output of the filter. Typically these numbers are represented by one byte in video, and by two bytes in audio. The set of numbers b_k is in practice finite. For example, a 5 tap filter has only these non-zero values: b_{-2}, b_{-1}, b_0, b_1, b_2 . Therefore the summation in (21.4) in practice is finite. Δt is the time between samples, so perhaps $1/\Delta t$ is 13.5 MHz for 601 video or 44.1 KHz for audio.

Since in practice the convolution integral (21.1) is replaced by the summation (21.3), one is forced to ask oneself: what happens in this case to (21.2)? We have all the tools needed to answer this question.

Recall the Fourier Integral transform pair (1.1) and (1.2) which we repeat here,

$$X(\omega) = \int_{-\infty}^{\infty} dt\, x(t)\, e^{-i\omega t} \qquad \text{projection = transform} \qquad (21.5)$$

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} d\omega\, X(\omega)\, e^{+i\omega t} \qquad \text{expansion = inversion} \qquad (21.6)$$

If we set $t = t_n = n\,\Delta t$, we can rewrite (21.6) as:

$$x(t_n) = \frac{1}{2\pi} \int_{-\infty}^{\infty} d\omega\, X(\omega)\, e^{+i\omega n\,\Delta t} \qquad \text{expansion = inversion} \qquad (21.8)$$

Here now is the set of steps one needs to do:

(a) write a version of (21.8) for each of our functions $a(t_n)$, $b(t_n)$ and $c(t_n)$ in terms of $A(\omega)$, $B(\omega')$, and $C(\omega)$. For example,

$$b(t_n) = \frac{1}{2\pi} \int_{-\infty}^{\infty} d\omega'\, B(\omega')\, e^{+i\omega' n\,\Delta t}$$

(b) jam these three expansions into (21.3)

(c) move the m-summation as far to the right as possible, it comes to rest against an exponential.

(d) do this summation using our exponential addition theorem (13.2), in this form:

$$\sum_{m=-\infty}^{\infty} e^{im\Delta t(\omega - \omega')} = \sum_{m=-\infty}^{\infty} 2\pi\, \delta[\, \Delta t(\omega - \omega') - 2\pi m\,] = (2\pi/\Delta t) \sum_{m=-\infty}^{\infty} \delta(\, \omega - \omega' - \omega_1 m)$$

(e) kill the $d\omega'$ integration against the delta function

(f) identify the dω integrand on both sides. Here is the result:

$$A(\omega) = \sum_{m=-\infty}^{\infty} B(\omega - m\omega_1)\ C(\omega)$$

$$= B(\omega)\, C(\omega) + \sum_{m\neq 0} B(\omega - m\omega_1)\ C(\omega)$$

Thus, we have our new digitalized convolution theorem:

$$a(t_n) = \sum_{m=-\infty}^{\infty} \Delta t\ b(t_n - t_m)\, c(t_m) \qquad t_n = n\,\Delta t \tag{21.9}$$

$$A(\omega) = [\ B(\omega) + \sum_{m\neq 0} B(\omega - m\omega_1)\ ]\ C(\omega) \tag{21.10}$$

This pair of equations should be compared immediately to (21.1) and (21.2) above. We see that there is a *penalty* for working in the imperfect, discrete world of time-sampled signals like $a(t_n)$. The penalty is that there are extra terms in (21.10) that are not present in (21.2).

Recall that $B(\omega)$ is the spectrum of a filter kernel $b(t)$ which we are approximating by a set of coefficients b_k. The extra terms in (21.10) can be interpreted as being due to the "image spectra" of this filter.

A perfect analog filter would of course have no such image spectra, this is what (21.2) is all about. It has no such artifacts because it filters at all times t, not just at particular points t_n. The digital filter is "blind" between sample points, so you can stick it with some high frequency signals which wiggle an arbitrary number of wiggles between the sample points. These high frequency signals are what get passed through the image pass bands of a digital low pass filter. All the above math should not blind the reader to this straightforward understanding of the image spectra.

A common use of a digital filter is to remove the image spectra of digitized signals. These image spectra are sitting staring us in the face as the second term in (20.7). Suppose we use a digital filter with a set of b coefficients to get some kind of low pass filter. Then (21.10) shows that in addition to the low pass band, this filter has an infinite set of equally spaced pass bands at ever increasing frequency, and the spacing is ω_1. We have to be careful that some of the image spectra of our sampled signal w(t) don't slip through these pass bands.

A standard trick ("oversampling") is to make ω_1 be 4X or maybe even 8X larger than it needs to be by the Nyquist requirement. This moves the image spectra of $W(\omega)$ in (20.7) way out away from the central spectrum. This gives lots of "headroom" in the ω domain to implement a digital filter which need not be too fancy because it does not have to drop off too quickly, ie, you can have some reasonable number of coefficients. All digital convolution filters (FIR filters) with symmetric coefficients have a perfect linear

phase characteristic, which means a constant group delay. Thus, the digital filter does a minimal amount of damage to the signal being filtered. The stuff that gets through the image pass bands of the digital filter is now at very high frequency because the oversampling rate ω_1 is so large. This stuff can be easily filtered out with a cheap analog filter -- perhaps a small capacitor.

22. The Digital Fourier Transform X'(ω) .

Looking back at (21.5), we can write down the following *non*-equation:

$$X(\omega) \neq \sum_{n=-\infty}^{\infty} \Delta t\, x(t_n)\, e^{-i\omega n\, \Delta t} \qquad \text{projection = transform} \qquad (22.1)$$

X(ω) is the genuine Fourier Integral transform of x(t). Only in the limit $\Delta t \to 0$ are the two sides equal, and we then reproduce (21.5). So let's define something new called X' that *is* equal for any finite Δt:

$$X'(\omega) \equiv \sum_{n=-\infty}^{\infty} \Delta t\, x(t_n)\, e^{-i\omega n\, \Delta t} \qquad \text{projection = transform} \qquad (22.2)$$

Although X(ω) can have any shape you want, the new spectrum X'(ω) is periodic with period ω_1. This follows directly from the definition (22.2) due to the exponential exp(i2π x integer) = 1. Thus,

$$X'(\omega - m\omega_1) = X'(\omega) \qquad m = \text{any integer} \qquad (22.3)$$

We claim now that the inverse of (22.2) is the following:

$$x(t_m) = \frac{1}{2\pi} \int_{-\omega_1/2}^{\omega_1/2} d\omega\, X'(\omega)\, e^{+i\omega m\, \Delta t} \qquad \text{expansion = inversion} \qquad (22.4)$$

This is identical to (21.8), except the integration endpoints are here finite. Thus, we have a *different* projection formula, and a correspondingly *different* expansion formula. For want of a better name, let us call this new transform the Digital Fourier Transform pair, as opposed to the Fourier Integral transform pair given in (21.5) and (21.6).

To show this really works, plug (22.2) into the right side of (22.4) and do the dω integration. This integration gives

$$\frac{\Delta t}{2\pi} \int_{-\omega_1/2}^{\omega_1/2} d\omega\, e^{+i\omega\Delta t(m\, -\, n)} = \delta_{m,n} \qquad (\delta_{m,n} = 0 \text{ if } m \neq n,\ = 1 \text{ if } m = n) \qquad (22.5)$$

The Kronecker delta $\delta_{m,n}$ then kills off the sum over n, we get $x(t_m) = x(t_m)$, and this proves that we have a valid transform. If we had $-\infty$ to $+\infty$ endpoints in (22.4), the integral in (22.5) would be a delta function [as in (2.1)], and we have no dt integration to make this go away. Then our proof fails. However, notice that as $\Delta t \rightarrow 0$, we do get endpoints $-\infty$ and $+\infty$.

Thus, both sides of our new transform approach the Fourier Integral transform in the limit $\Delta t \rightarrow 0$. For finite Δt, the new transform differs from the Fourier Integral transform.

One should keep in mind that this new transform, the Digital Fourier Transform, is dependent on the constant $\Delta t = T_1$. If you change this constant, you change the transform. The Fourier Integral Transform contains no such constant. In effect, $T_1 = 0$.

Now lets go back to our discrete convolution relation (21.9):

$$a(t_n) = \sum_{m=-\infty}^{\infty} \Delta t \; b(t_n - t_m)\, c(t_m) \qquad t_n = n\,\Delta t \tag{22.6}$$

If we now repeat the analysis described in Section 21, but this time we use expansions of the form (22.4), the finite range of the $d\omega$ integrations causes only a single delta function in (13.2) to be selected out. The result is this:

$$A'(\omega) = B'(\omega)\, C'(\omega) \tag{22.7}$$

Thus, our "new" transform, indicated so far by the primed quantities, yields a simple result with no "extra" terms. Of course we must remain aware that $A'(\omega)$ is not the genuine spectrum of a(t), it is some new thing. Just because we defined a new animal and got (22.7) does not mean that the image spectra go away in (20.7) and (21.10) !

23. Relation between X'(ω) and X(ω).

Obviously, the next problem is to try and find out how the genuine Fourier integral spectrum $X(\omega)$ is related to our "new" spectrum $X'(\omega)$, the Digital Fourier transform. The result is so significant, that we will do all the steps right here. Start with the Fourier integral expansion (21.6) ,

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} d\omega\, X(\omega)\, e^{+i\omega t}$$

Break up the integration into a set of little ranges of width ω_1 :

$$x(t) = \frac{1}{2\pi} \sum_{m=-\infty}^{\infty} \int_{m\omega_1 - \omega_1/2}^{m\omega_1 - \omega_1/2} d\omega\, X(\omega)\, e^{+i\omega t}$$

Change integration variable to $\omega' = \omega + m\omega_1$,

$$x(t) = \frac{1}{2\pi} \int_{-\omega_1/2}^{\omega_1/2} d\omega' \left[\sum_{m=-\infty}^{\infty} X(\omega' - m\omega_1)\, e^{-im\omega_1 t} \right] e^{+i\omega' t}$$

Finally, set $t = t_n = nT_1$. This makes the exponential inside the square bracket equal 1, so

$$x(t_n) = \frac{1}{2\pi} \int_{-\omega_1/2}^{\omega_1/2} d\omega' \left[\sum_{m=-\infty}^{\infty} X(\omega' - m\omega_1) \right] e^{+i\omega' n\Delta t}$$

Now compare this to the Digital Fourier expansion defined in (22.4), which we duplicate here, changing ω to ω', and changing m to n:

$$x(t_n) = \frac{1}{2\pi} \int_{-\omega_1/2}^{\omega_1/2} d\omega'\, X'(\omega')\, e^{+i\omega' n\,\Delta t}$$

Therefore, comparing the $d\omega'$ integrands, the relation between $X'(\omega)$ and $X(\omega)$ is given by:

$$X'(\omega) = \sum_{m=-\infty}^{\infty} X(\omega - m\omega_1) = \left[X(\omega) + \sum_{m\neq 0} X(\omega - m\omega_1) \right] \qquad (23.1)$$

This is that thing that keeps popping up everywhere -- the main spectrum plus all the image spectra. Thus, we have shown that this combination is precisely the Digital Fourier spectrum. We can therefore go back and reexamine some of our earlier results with this new knowledge:

Consider (20.7):

$$T_1 W(\omega) = \left[Y(\omega) + \sum_{m\neq 0} Y(\omega - m\omega_1) \right] = Y'(\omega) \qquad (23.2)$$

This says that the Fourier integral spectrum of a "reasonable" signal y(t) *multiplied* by a sequence of delta functions is exactly the Digital Fourier spectrum of y(t). Of course $Y'(\omega)$ is computed from (22.2) from a knowledge of y(t) only at the sample points t_n.

Next, consider (21.10):

$$A(\omega) = [\ B(\omega) + \sum_{m\neq 0} B(\omega - m\omega_1)\ ]\ C(\omega)$$

$$A(\omega) = B'(\omega)\ C(\omega) \qquad (23.3)$$

This was our low pass digital filter B with input A and output C. The filter with all its image pass bands is now conveniently represented by B'(ω). A(ω) and C(ω) are still the Fourier Integral spectra. However, we already know that (23.3) is true with primes on A and C as well, since we already derived this in (22.7), we rewrite it here:

$$A'(\omega) = B'(\omega)\ C'(\omega) \qquad (23.4)$$

It might seem unusual that both (23.3) and (23.4) can be true at the same time, it has to do with the translation invariance property (22.3). You can derive (23.4) as a superposition of copies of (23.3) making use of (22.3).

23A. The Autocorrelation Function.

At the start of Chapter 6, the autocorrelation function will be studied. Here, we wish only to state the result and show what it looks like in the discrete time domain. From Chapter 6 we have,

$$r_x(t) \equiv \int_{-\infty}^{\infty} dt'\ x^*(t')\ x(t' + t) \qquad (6.1.1)$$

In analogy with (22.6) we shall define a discrete version which approaches the continuous version in the limit that $\Delta t \rightarrow dt$,

$$r'_x(t_i) \equiv \sum_{n=-\infty}^{\infty} \Delta t\ \ x^*(t_n)\ x(t_{n+i}) \qquad t_n = n\ \Delta t\ ,\quad \Delta t = T_1 \qquad (23A.1)$$

We add a prime to indicate that $r' \neq r$ unless we are in the limit just mentioned. Now, if we transform both sides of (23A.1) with the Digital Fourier Transform (22.2), we find that

$$R'_X(\omega) = \Delta t\ |\ X'(\omega)|^2 \qquad (23A.2)$$

which can be compared with the continuous-time result from Chapter 6,

$$R_x(\omega) = |\ X(\omega)\ |^2 \qquad (6.1.4)$$

We now summarize what we know about the Digital Fourier Transform:

1. Let x(t) be any reasonable function.

2. Divide up the time axis into steps $t_n = n\,\Delta t$; let $T_1 \equiv \Delta t$ and $\omega_1 \equiv 2\pi/T_1$.

3. Define the Digital Fourier transform = projection by

$$X'(\omega) \equiv \sum_{n=-\infty}^{\infty} \Delta t\, x(t_n)\, e^{-i\omega t_n}$$

By its definition (and $t_n = n\,\Delta t$), $X'(\omega)$ is periodic in ω with period ω_1:

$$X'(\omega - m\omega_1) = X'(\omega) \qquad m = \text{any integer}$$

4. At the points $t = t_n$, one can expand x(t) by the following expansion = inversion formula:

$$x(t_n) = \frac{1}{2\pi} \int_{-\omega_1/2}^{\omega_1/2} d\omega\, X'(\omega)\, e^{+i\omega t_n} \quad \text{expansion = inversion}$$

Due to the periodicity of $X'(\omega)$, any range of width ω_1 can be used for this integral.

5. The relation between $X'(\omega)$ and the Fourier Integral spectrum $X(\omega)$ is given by:

$$X'(\omega) = \sum_{m=-\infty}^{\infty} X(\omega - m\omega_1) = [\, X(\omega) + \sum_{m\neq 0} X(\omega - m\omega_1) \,]$$

6. The Digital Fourier Transform diagonalizes any convolution sum:

$$a(t_n) = \sum_{m=-\infty}^{\infty} \Delta t\; b(t_n - t_m)\, c(t_m) \qquad t_n = n\,\Delta t$$

$$A'(\omega) = B'(\omega)\, C'(\omega)$$

7. Autocorrelation function:

$$r'_x(t_i) \equiv \sum_{n=-\infty}^{\infty} \Delta t\; x^*(t_n)\, x(t_{n+i}) \qquad t_n = n\,\Delta t\,, \quad \Delta t = T_1$$

$$R'_X(\omega) = \Delta t\, |\, X'(\omega)|^2$$

24. The Z Transform.

The Digital Fourier Transform described in Section 23 is really the Z Transform, apart from an unimportant overall constant. We have sort of concealed this fact up till now because the ω-space version, called X'(ω) in Section 23, allows direct comparison to the Fourier Integral spectrum X(ω). We have already drawn the major conclusions. Here we just change the clothing .

Change variables from ω to

$$z = e^{i\omega\Delta t} = e^{\,i\pi\left[\frac{\omega}{(\omega_1/2)}\right]} \qquad dz = z\, i\, \Delta t\, d\omega \qquad (24.1)$$

Note that as ω wanders over its range $-\omega_1/2$ to $+\omega_1/2$, z wanders from -180° to +180° on a unit circle.

Now define the Z Transform in terms of X'(ω),

$$X''(z) \equiv \frac{1}{\Delta t}\, X'(\omega(z)) \qquad (24.2)$$

With this substitution, and letting

$$x_n = x(t_n) = x(n\Delta t),$$

the above Digital Fourier Transform formulas (22.2) and (22.4) immediately become:

$$X''(z) = \sum_{n=-\infty}^{\infty} x_n\, z^{-n} \qquad \text{projection = transform} \qquad (24.3)$$

$$x_n = \frac{1}{2\pi i} \int_c dz\, X''(z)\, z^{n-1} \qquad \text{expansion = inversion} \qquad (24.4)$$

where C is a contour doing one counterclockwise traversal of the unit circle in the z plane. Note from dz = ziΔtdω that we pick up an i and an extra power of z in the inversion formula.

These two equations are the Z Transform and its Inverse.

So far in these notes we have run into the Fourier Integral Transform, the Laplace Transform, the Fourier Series, the Digital Fourier Transform, and the Z Transform. They are all variations on the same theme, and we have shown how they are all related to each other. They all have an analogous set of "basic results" and "rules", so we pause to observe a few things that I think the reader will find interesting.

(a) Convolution. According to the definition $X''(z) \equiv \frac{1}{\Delta t} X'(\omega(z))$, if we transcribe the convolution result $A'(\omega) = B'(\omega)\, C'(\omega)$, we pick up an extra factor of Δt . Thus (22.6) and (22.7) become:

$$a_n = \sum_{m=-\infty}^{\infty} \Delta t\, b_{n-m}\, c_m \qquad \text{convolution sum}$$

$$A''(z) = \Delta t\, B''(z)\; C''(z) \qquad \text{z-transformed result}$$

If you have a filter where A" and C" represent voltages, B" has the dimensions of inverse time. This same statement applies to a_n, c_n, and b_n. We will see an example of this where $B''(z) = G''(z)$ in our digital RC discussion below.

(b) Unit Impulse. The analog of the unit impulse $\delta(t-a)$ at $t=a$ must be a sequence of numbers x_n which are all zero except the one say at some integer m. You might write this as

$x_n = \delta_m(n) = \delta_{n,m}$ n = all integers, the sequence index

If we stuff this into the Z Transform projection (24.3) , we get

$X''[\, \delta_m(n)\,] = z^{-m}$

This looks a lot like our Fourier Integral result (8.2) which was

$x(t) = \delta(t - t_m)$

$X(\omega) = e^{-i\omega t_m} = e^{-i\omega \Delta t\, m}$

especially when you consider that $z = \exp(i\omega\Delta t)$. The thing on the right is the same, but the thing on the left is different. The Fourier integral transform does to its appropriate "unit impulse" just what the Z Transform does to *its* appropriate "unit impulse". The unit impulses are different.

In a filter with input I and output O we have $O''(z) = \Delta t\, G''(z)\, I''(z)$, where G(z) is some transfer function. If I is taken to be a unit impulse at time t=0, then from the above $I''(z) = z^0 = 1$. Thus, quantity $[\Delta t\, G''(z)]$ *is* the z-plane response of the filter to an impulse at t=0. From (24.4) one can then get the time domain impulse response g_n. We shall do this below for an "RC" filter.

(c) Time translation. Above we show a unit impulse $\delta_m(n)$ at time m , and its Z transform is z^{-m} . A unit impulse one step *later* in time would be $\delta_{m+1}(n)$, and its Z transform would be $z^{-m-1} = z^{-m}\, z^{-1}$. This suggests that if you delay a signal by one step, you alter its Z transform by multiplication by z^{-1}. Advancing a signal one step means multiply by z^{+1}. These facts are true for an arbitrary signal, they follow immediately from (24.4). A very similar things happens in the Fourier Integral world, where time translation generates a multiplicative phase, see last equation in (b) above as an example.

In general, you can delay a digital signal one step in time by running it through a D flip-flop, so this is why such flip-flops are associated with z^{-1} in a digital filter. There is no analogous device to associate with z^{+1} .

(d) Derivative Limit. Based on the preceding subsection, we know that the following difference of two sequences has this transform.

$$\frac{x_n - x_{n-1}}{\Delta t} \quad \leftrightarrow \quad X''(z) \left[\frac{1 - z^{-1}}{\Delta t} \right]$$

This is just a simple superposition. If you take the limit $\Delta t \to 0$, the LHS becomes dx/dt, and the RHS becomes X"(z) iω, since $z^{-1} = \exp(-i\omega\Delta t) \approx 1 - i\omega\Delta t$. Recall that in this limit, Z Transform is the Fourier Integral transform, and the derivative rule of Section 11 says you multiply by iω.

(e) Digital RC filter. Here we are going to write down equations that look like (4.1a) - (4.6) for the analog RC filter back in Section 4. Using the above transcription of d/dt we write the RC filter of equation (4.1a) as:

$$\left[RC \frac{v_o(t_n) - v_o(t_{n-1})}{\Delta t} + v_o(t_n) \right] = v_i(t_n)$$

Using (d) above, we convert this difference equation to Z transform land:

$$\left[(RC/\Delta t) (1 - z^{-1}) + 1 \right] VO''(z) = VI''(z)$$

Now write this in the convolution format shown in (a) above:

$$VO''(z) = \Delta t \, G''(z) \, VI''(z)$$

Thus, we know the transfer function G"(z)

$$G''(z) = \frac{(1/\Delta t)}{\left[(RC/\Delta t) (1 - z^{-1}) + 1 \right]} = (1/\Delta t) \frac{[z/(1+\alpha)]}{z - [\alpha/(1+\alpha)]} \qquad \text{where } \alpha = (RC/\Delta t).$$

If we stick this into (24.4), we note that the integrand converges for $n \geq 0$ inside the unit circle, so we can shrink the contour to the pole in G"(z) and pick up its residue. For $n<0$, the integrand is singular at the origin, but is well behaved and decays away at infinity. In this case we stretch the contour to infinite and get 0 for g_n. Here is the overall result:

$$g_n = (1/RC) \left(\frac{\alpha}{1+\alpha}\right)^{n+1} \theta(n) = (1/RC) \, e^{\left[- (n+1) \ln\{(1+\alpha)/\alpha\} \right]} \theta(n)$$

Thus, g_n decays in an exponential fashion, and is causal -- it does not propagate backward in time due to the θ(n) factor. This result should be compared to (4.6) which we repeat here

$$g(t) = (1/RC)\, e^{-(t/RC)}\, \theta(t)$$

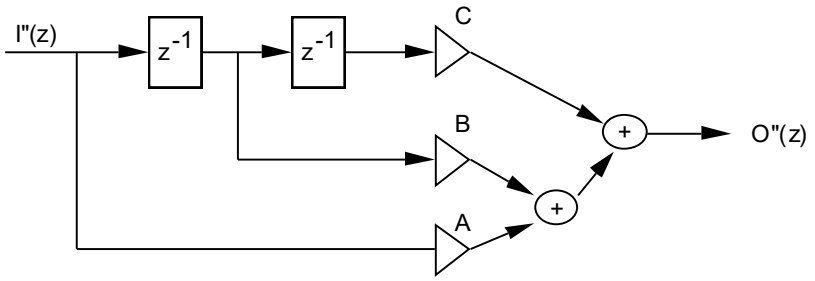
Indeed, if you take the limit $\alpha \to \infty$ of the $g_n = g(t_n)$ result, you arrive at g(t), with an appropriate connection between t_n and t. One might note that our difference equation is a little bit biased toward the past. A fancier rendition of this section might use an average of the forward difference and the backward difference in the original difference equation.

(f) Poles imply feedback. The general form of an implementable transfer function G"(z) is a ratio of polynomials in z. We saw an example in the RC filter above. It is easier to visualize this instead as a ratio of polynomials in z^{-1}, which just means multiplying top and bottom by some power of z. Thus, zeros of G"(z) are associated with powers of z^{-1} in the numerator, and poles of G"(z) are associated with powers of z^{-1} in the denominator.

Suppose there are no poles. Then you get something like $G"(z) = A + Bz^{-1} + Cz^{-2}$. Then if I"(z) and O"(z) are the input and output of your filter, you have:

$$O"(z) = [\ A + Bz^{-1} + Cz^{-2}\ ]\ I"(z)$$

So the output is developed with two flip-flops and three constant multipliers and two adders, and nothing is ever fed back. Here is a picture:



This is an FIR filter since the unit impulse response g_n goes to zero after 3 time periods.

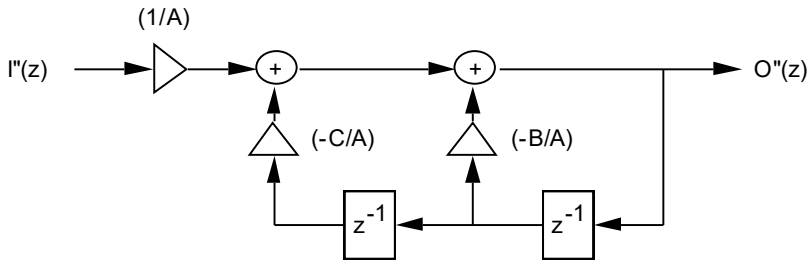
Now take a different example with poles. Suppose G"(z) is the inverse of the one just shown. Then we get I and O swapped, so

$$I"(z) = [\ A + Bz^{-1} + Cz^{-2}\ ]\ O"(z)$$

Solve this for O"(z) to get:

$$O"(z) = (1/A)\ I"(z)\ + (\ -\ B/A)\ z^{-1}\ O"(z) + (\ -\ C/A)\ z^{-2}\ O"(z)$$

This requires three constant multipliers, two adders, and two flip-flops. Here is the filter:



Here you can see the feedback. This is an IIR filter, since the impulse response g_n carries on forever due to the feedback. See g_n in section (e) above: exponential decay, never goes away completely. [Of course it might go away in a real digital circuit with a finite number of quantization bits.]

(g) Scramblers. Typical examples of IIR filters having feedback are serial scramblers and CRC generators. One thinks of the input sequence I"(z) as a huge polynomial in z, where the presence or absence of each power represents a 1 or a 0. That is, think of the incoming stream as a superposition of unit impulses with weights equal to the binary digits of the data stream. The transfer function G"(z) = 1/polynomial, so we write O"(z) = G"(z) I"(z) = I"(z)/polynomial. The output stream is then the quotient of polynomial division. One way to interpret the above discussion is as follows:

no poles $\rightarrow$ polynomial multiplication $\rightarrow$ no feedback $\rightarrow$ FIR

poles $\rightarrow$ polynomial division $\rightarrow$ feedback $\rightarrow$ IIR

We conclude this section with a Z Transform summary:

Z Transform

1. Let x(t) be any reasonable function.

2. Divide up the time axis into steps $t_n = n\,\Delta t$; let $T_1 \equiv \Delta t$ and $\omega_1 \equiv 2\pi/T_1$.

3. As a shorthand, let $x_n = x(t_n)$ = the values x(t) takes at the points t_n.

4. Define the Z transform = projection by

$$X''(z) = \sum_{n=-\infty}^{\infty} x_n z^{-n}$$

5. At the points $t = t_n$, one can expand $x_n = x(t_n)$ by the following expansion = inversion :

$$x_n = \frac{1}{2\pi i}\int_c dz\, X''(z)\, z^{n-1}$$

where contour C goes once counterclockwise around the unit circle in the z-plane.

6. The Z Transform diagonalizes any convolution sum :

$$a_n = \sum_{m=-\infty}^{\infty} \Delta t\; b_{n-m}\, c_m$$

$$A''(z) = \Delta t\; B''(z)\, C''(z)$$

25. Amplitude Modulated Pulse Trains.

In Chapter 2 we studied the spectrum of a pulse train made by superposing pulses $x_{pulse}(t)$ with spacing T_1. We found that the Fourier Integral spectrum $X(\omega)$ of such a pulse train was given by an infinite sequence of delta function spikes with amplitudes determined by an envelope function $c(\omega)$ which is just a multiple of the spectrum of the pulse:

$$x(t) = \sum_{n=-\infty}^{\infty} x_{pulse}(t - t_n) \qquad t_n = nT_1 \tag{25.1}$$

$$X(\omega) = \sum_{m=-\infty}^{\infty} c(\omega)\; 2\pi\,\delta(\omega - m\omega_1) = \sum_{m=-\infty}^{\infty} c_m\; 2\pi\,\delta(\omega - m\omega_1) \tag{25.2}$$

$$\text{where } c(\omega) + (1/T_1)X_{pulse}(\omega) = (1/T_1)\int_{-\infty}^{\infty} dt\, x_{pulse}(t)\, e^{-i\omega t} \qquad (25.3)$$

The numbers $c_m = c(m\omega_1)$ turned out to be exactly the complex Fourier Series coefficients.

We now wish to generalize (25.1) slightly by inserting a set of amplitude coefficients y_n so we now have an amplitude modulated pulse train, which we will call w(t):

$$w(t) = \sum_{n=-\infty}^{\infty} y_n\, x_{pulse}(t - t_n) \qquad (25.4)$$

Insert this w(t) into the Fourier integral projection (1.1) and immediately change the dt integration variable to dt' where $t' = t - t_n$. Then move the summation over n to the right , and recognize this sum as precisely the Digital Fourier spectrum of the y_n (see box at end of Section 23),

$$\sum_{n=-\infty}^{\infty} y_n\, e^{-i\omega t_n} = \frac{1}{T_1}\; Y'(\omega) \qquad (25.5)$$

Recall that $Y'(\omega)$ is the thing with all the image spectra, see (23.1). The remaining terms are just $X_{pulse}(\omega)$, the Fourier integral spectrum of $x_{pulse}(t)$. We combine this with the factor $1/T_1$ to get $c(\omega)$. Thus, we arrive at this very significant result, which deserves its own box:

Amplitude Modulated Pulse train

$$w(t) = \sum_{n=-\infty}^{\infty} y_n\, x_{pulse}(t - t_n)$$

$$W(\omega) = c(\omega)\; Y'(\omega) = c(\omega)\; [\; Y(\omega) + \sum_{m\neq 0} Y(\omega - m\omega_1)\;]$$

$$\text{where } c(\omega) + (1/T_1)X_{pulse}(\omega) = (1/T_1)\int_{-\infty}^{\infty} dt\, x_{pulse}(t)\, e^{-i\omega t}$$

We now recognize our earlier result (20.7) as the special case of the boxed result above where $x_{pulse}(t) = \delta(t)$, so that $c(\omega) = 1/T_1$.

The boxed result above is one of the holy grails of the spectral analysis of digital signals. In the following section, we shall use it to study the problem of aperture correction.

26. A simple application: aperture correction.

The output of a digital system usually involves a D/A converter followed by an analog post-filter. Let us assume that this system is attempting to reproduce some reasonable analog waveform y(t). Assume that each converted value is held as charge on a capacitor for some portion τ of the conversion period T_1, and then the capacitor charge is instantly dumped to ground for the remainder of the period. In this way, we produce a signal w(t) that is a sequence of square pulses modulated by the values $y(t_n)$:

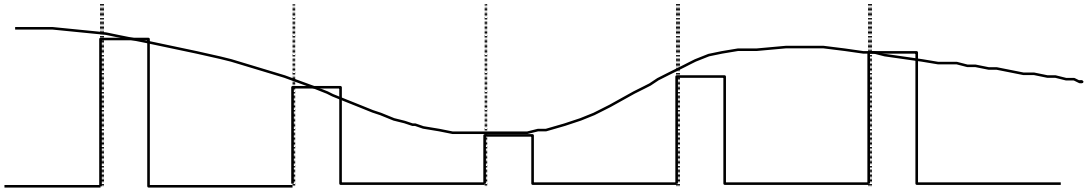


Figure 26.1. The smooth curve is y(t), the pulse train is w(t). Period is $\Delta t = T_1$. Width of each pulse is τ, so duty cycle is τ/T_1.

What is the spectrum of w(t)? It is an amplitude modulated pulse train, so we know the answer at once from the last section:

$$W(\omega) = (1/T_1)X_{pulse}(\omega)\, Y'(\omega) \qquad (26.1)$$

The pulse $x_{pulse}(t)$ now is a square pulse of unit height, width τ, and it has its left edge aligned with t=0. The pulse amplitudes y_n have already been absorbed into the Y'(ω) factor. We know the spectrum of this pulse from (9.2) , but we must add a time shift phase $\exp(-i\omega\tau/2)$ because we are translating our earlier pulse $\tau/2$ units to the right to make the left edge line up at t=0. Thus, from (9.2)

$$X_{pulse}(\omega) = \tau\ \mathrm{sinc}(\omega\tau/2)\ e^{-i\omega\tau/2} \qquad (26.2)$$

so the spectrum of w(t) is

$$W(\omega) = (\tau/T_1)\ \mathrm{sinc}(\omega\tau/2)\ \exp(-i\omega\tau/2)\ Y'(\omega) \qquad (26.3)$$

The magnitude of W(ω) is the product of two functions which we now illustrate :

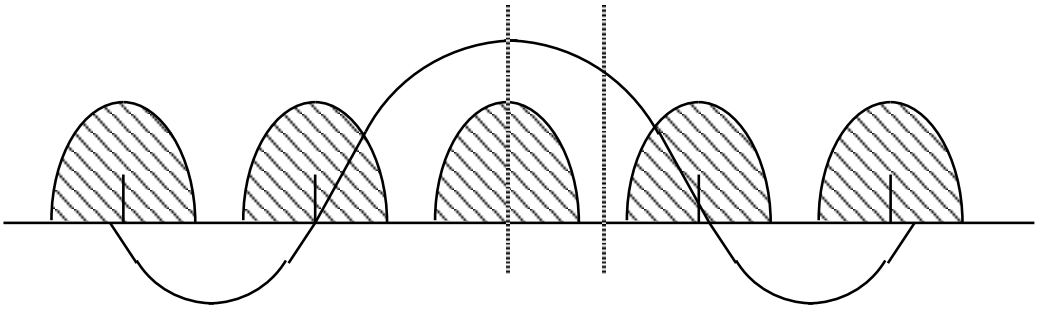


Figure 26.2. The humps are $Y'(\omega)$ and the curve is $(\tau/T_1)\ \mathrm{sinc}(\omega\tau/2)$.

Presumably our low pass analog post-filter is going to select out the portion of the spectrum indicated by the dotted lines in Figure 26.2. In this region, we have

$$W(\omega) = (\tau/T_1)\ \mathrm{sinc}(\omega\tau/2)\ \exp(-i\omega\tau/2)\ Y(\omega), \qquad (26.4)$$

where $Y(\omega)$ is the central spectrum of $Y'(\omega)$. The factor $(\tau/T_1)\mathrm{sinc}(\omega\tau/2)$ represents an undesired magnitude distortion of the spectrum $W(\omega)$ due to the aperture τ. The phase $\phi(\omega) = -i\omega\tau/2$ is a harmless linear phase which just means the whole signal is delayed by time $\tau/2$.

The distortion is at its *worst* when the aperture τ fills the entire period T_1, in which case the signal in Figure 26.1 looks like a traditional stepwise fit to y(t). The distortion is worst because the zeros of $\mathrm{sinc}(\omega\tau/2)$ are at $\omega_m = m\,(2\pi/\tau)$, so they are moved in as close as possible when τ is as large as possible, $\tau = T_1$.

The distortion can be reduced by making τ as small as practicable. In this case, the zeros move out, and the central hump of $\mathrm{sinc}(\omega\tau/2)$ is broad, so its drop-off during $Y(\omega)$ is minimized. Of course the amplitude of $W(\omega)$ also drops off as τ is made small, so there is a tradeoff.

In any event, there is still *some* distortion represented by $\mathrm{sinc}(\omega\tau/2)$ varying in the dotted region in Figure 26.2. Usually one attempts to correct for this aperture distortion by building into the analog post-filter an exactly compensating boost at frequencies near the cutoff region of the filter.

Thus, if the post-filter would normally be some $F(\omega)$ cutting off in the region of the second dotted line in Figure 26.2, a corrected filter would have the spectrum $F(\omega)/\mathrm{sinc}(\omega\tau/2)$. This filter needs to know the aperture time τ in addition to the cutoff frequency. Such a filter is said to have "sine x over x correction".

27. The Discrete Fourier Transform.

This is our last transform, we promise! We shall develop it in exact analogy to what we did in Section 22 for the Digital Fourier Transform, then we shall make some appropriate comments.

Recall from Section 15 the discussion of the Fourier Series transform with complex coefficients c_m. This was summarized in the last box of Section 15, which we duplicate here:

$$c_m = (1/T_1) \int_{-\infty}^{\infty} dt\, x_{pulse}(t)\, e^{-im\omega_1 t} \qquad \text{projection = transform} \qquad (27.1)$$

$$x(t) = \sum_{m=-\infty}^{\infty} c_m\, e^{+im\omega_1 t} \qquad \text{expansion = inversion} \qquad (27.2)$$

Here, x(t) is an infinite pulse train made by superposing pulses $x_{pulse}(t)$ at spacing T_1. Thus, x(t) is a periodic function with period T_1.

We wish now to redefine our concept of interval Δt. In our previous discussion, we set $\Delta t = T_1$. Here we wish instead to break up *each* interval T_1 into N pieces of size Δt, so now we have:

$$\Delta t = T_1 / N \quad T_1 = N\,\Delta t \qquad \omega_1 = (2\pi/T_1) = (2\pi/N)\,\Delta t \qquad (27.3)$$

Now let $t_n = n\Delta t$ represent a sequence of sample points for x(t). There are N such points per period. If we rewrite our second equation above evaluated at these points, we get

$$x(t_n) = \sum_{m=-\infty}^{\infty} c_m\, e^{+imn(2\pi/N)} \qquad (27.4)$$

So far we haven't really done anything except examine x(t) at some sample points.

Now consider, as we did at the start of Section 22, the following *non*-equation:

$$c_m \neq (1/T_1) \sum_{n=-\infty}^{\infty} \Delta t\, x_{pulse}(t_n)\, e^{-im\omega_1 t_n} \qquad (27.5)$$

Only in the limit that $\Delta t \to 0$ does (27.5) become an equality, since it then reproduces (27.1). So let's define something *new* called c'_m that *is* equal for any finite Δt:

$$c'_m \equiv (1/T_1) \sum_{n=-\infty}^{\infty} \Delta t\, x_{pulse}(t_n)\, e^{-im\omega_1 t_n}$$

Using (27.3) , this becomes

$$c'_m \equiv (1/N) \sum_{n=-\infty}^{\infty} x_{pulse}(t_n)\, e^{-imn(2\pi/N)} \qquad (27.6)$$

In the Fourier Series world, you can have any number of unique c_m coefficients that you like. For the c'_m in (27.6) this is no longer true. There are in fact only N unique values of c'_m because they keep repeating. This is because (27.6) implies that

$$c'_{m+kN} = c'_m \qquad \text{for any integer k} \qquad (27.7)$$

In other words, c'_m is a periodic digital sequence of period N.

We claim now that the correct expansion of pulse train x(t) to accompany (27.6) is the following:

$$x(t_n) = \sum_{m=0}^{N-1} c'_m\, e^{+imn(2\pi/N)} \qquad (27.8)$$

This is identical to (27.4) except the summation is finite, having only N terms. Thus, we have a *different* projection formula, and a correspondingly *different* expansion formula. This new transform pair is known as the Discrete Fourier Transform, as opposed to the Fourier Series transform pair given in (27.1) and (27.2).

To show this transform really works, first replace the dummy summation variable n in (27.6) by k, then plug (27.6) for c_m' into the right side of (27.8) and do the m summation. This summation gives

$$\sum_{m=0}^{N-1} e^{+im(2\pi/N)(n-k)} = N \sum_{m=0}^{N-1} \delta_{k,n-mN} \qquad (27.9)$$

If k = n-mN for some integer m, (27.9) is quite obvious, since the summand is 1. For other integer k, the fact that the sum is zero is less obvious. In this case, it is the sum of a set of complex numbers equally spaced around the unit circle, so you do in fact get zero. (It might help to draw this for the cases N = 3 and N = 4.) The above exponential sum is the finite-sum version of (13.2).

Continuing, the Kronecker delta $\delta_{k,n-mN}$ then kills off the sum over k, and we end up with this result,

$$x(t_n) = \sum_{m=-\infty}^{\infty} x_{pulse}(t_{n-mN}) = \sum_{m=-\infty}^{\infty} x_{pulse}(t_n - mT_1)$$

which confirms that (27.8) is the correct expansion to go with projection (27.6).

Note that if we had an infinite sum in (27.8), the sum in (27.9) would diverge, and our proof would fail.

However, notice also that as $\Delta t \to 0$, we do get an infinite sum in (27.8), and in this limit, our Discrete Fourier Transform becomes the Fourier Series Transform. This is a little more clear if you rearrange the elements in the sum (27.8) so that m=0 is in the center of the range. For example, for N even

$$x(t_n) = \sum_{m=-N/2}^{N/2-1} c'_m e^{+imn(2\pi/N)} \qquad N \text{ even}$$

This really is the same as (27.8) . To prove it, take the negative half of the sum and define for it a new summation index $m' = m + N$, make use of periodicity (27.7) , and then this part of the sum becomes the upper half of the sum in (27.8). For N odd, the corresponding formula is this

$$x(t_n) = \sum_{m=-(N-1)/2}^{(N-1)/2} c'_m e^{+imn(2\pi/N)} \quad N \text{ odd}$$

and the proof is the same. Thus, as you take the limit $N \to \infty$ ($\Delta t \to 0$), both these formulas approach (27.4). Again, in this limit, the Discrete Fourier Transform equals the Fourier Series transform.

The fact that you have to distinguish N even and odd in these last two equations is an excellent reason for writing the sum as shown in (27.8), which works for N even or odd.

At this point, we make a box to summarize what we know about the Discrete Fourier Transform:

Discrete Fourier Transform

1. Let $x_{pulse}(t)$ be any reasonable pulse. Construct a pulse train x(t) with spacing T_1:

$$x(t) = \sum_{n=-\infty}^{\infty} x_{pulse}(t - nT_1) \qquad \Rightarrow \qquad x(t + nT_1) = x(t) \qquad n = \text{any integer}$$

By its construction, x(t) is periodic with period T_1. If x(t) is a *known* periodic function of period T_1, a candidate for $x_{pulse}(t)$ is x(t) over any one period.

2. Break up each T_1 interval into N steps of width $\Delta t = T_1/N$. Let $t_n = n\Delta t$ (n = integer).

3. Define the Discrete Fourier coefficients c'_m by this projection = transform:

$$c'_m = (1/N) \sum_{n=-\infty}^{\infty} x_{pulse}(t_n)\, e^{-imn(2\pi/N)} \qquad m = \text{integer}$$

Only N of these are unique because c'_m is periodic in index m with period N:

$$c'_{[\,m + nN\,]} = c'_m \qquad n = \text{any integer}$$

4. The pulse train at sample points t_n is then given by this expansion = inversion:

$$x(t_n) = \sum_{m=0}^{N-1} c'_m\, e^{+imn(2\pi/N)}$$

We are now ready for the "comments" promised at the start of this section. One might wonder about the purpose of the Discrete Fourier Transform, and its relation to earlier transforms. This can be made astoundingly clear by a simple set of pictures.

First, go back to the Section 2 analysis of a making a pulse train x(t) by superposing pulses $x_{pulse}(t)$. Imagine, as in the discussion at the end of Section 14, that $x_{pulse}(t)$ is a Gaussian, and assume moreover that it extends beyond the domain of one period T_1. Here is a picture of this pulse:

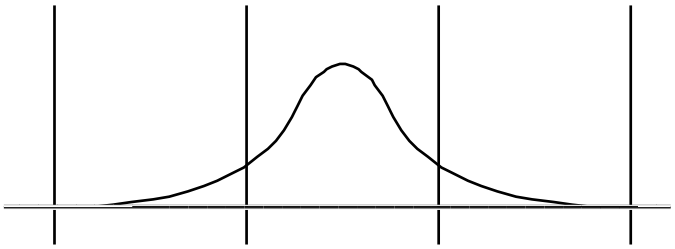


Figure 27.1. The pulse $x_{pulse}(t)$. Bars are distance T_1 apart.

If we now superpose these pulses, we get the following pulse train x(t),

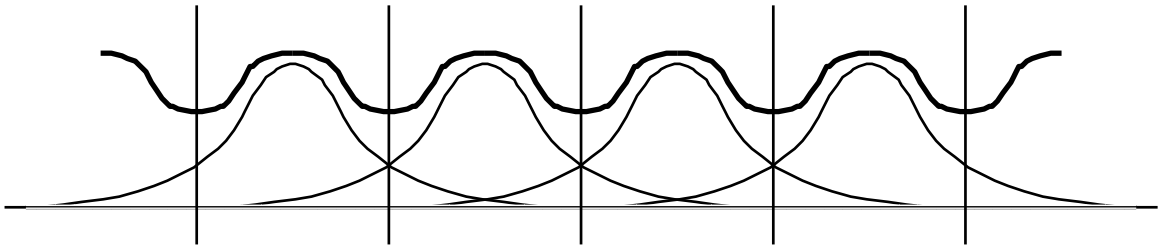


Figure 27.2. The function x(t) is the heavy curve. It is the sum of the gaussians.

This picture gives us a chance to repeat a point made earlier, namely that $x_{pulse}(t)$ is not unique. One could use instead a portion of the heavy curve between any adjacent pair of bars.

The heavy curve is our pulse train, and we have chosen the most complicated case, that where the pulses overlap. Typically they do not overlap.

Now, the heavy curve is a periodic continuous function of time x(t), and it has in principle an infinite set of Fourier Series coefficients c_m. These coefficients are really determined from the underlying pulse $x_{pulse}(t)$. In general, it takes an infinite number of time points to represent the smooth function $x_{pulse}(t)$, so there are an infinite number of coefficients c_m in the "transformed space", which is where these coefficients live.

We know that the transform of the Gaussian $x_{pulse}(t)$ is a Gaussian $X_{pulse}(\omega)$, and we know that the Fourier coefficients are given by

$$c_m = (1/T_1)\, X_{pulse}(m\omega_1) \qquad \text{where } \omega_1 = 2\pi/T_1 \ .$$

You can imagine the infinite set of the c_m as tracing the envelope of this Gaussian $X_{pulse}(\omega)$. Of course in general it may happen that many of the c_m vanish if $x_{pulse}(t)$ has simple harmonic content. The point is that there *could* be an infinite number of them.

This infinitude matches the infinitude of points along the pulse $x_{pulse}(t)$. If we consider the regular Fourier integral spectrum $X_{pulse}(\omega)$ in its own right, you again get an infinitude of complex numbers needed to describe the pulse in the transformed space.

Having said all this, we are now ready to move from analog to digital. We now consider the same pulse $x_{pulse}(t)$ evaluated only at the discrete points t_n, so we have the pulse now represented by this sequence of numbers:

$$x_n(\text{pulse}) = x_{pulse}(t_n). \qquad t_n = n\,\Delta t$$

and we assume that there are N sample points in each period T_1. In our figures below, N = 8.

Here then is a picture of the set of numbers x_n(pulse) which describe our pulse:

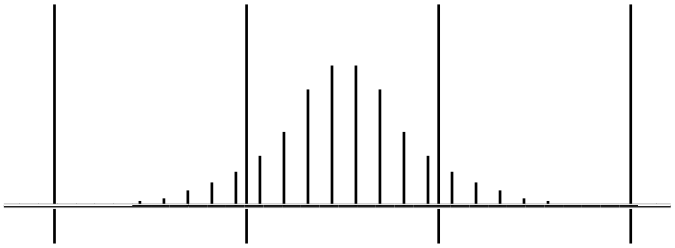


Figure 27.3. Pulse is now a set of 18 numbers x_n(pulse) . N = 8

Now as before, build a digital pulse train by superposing pulses. Here is the result:

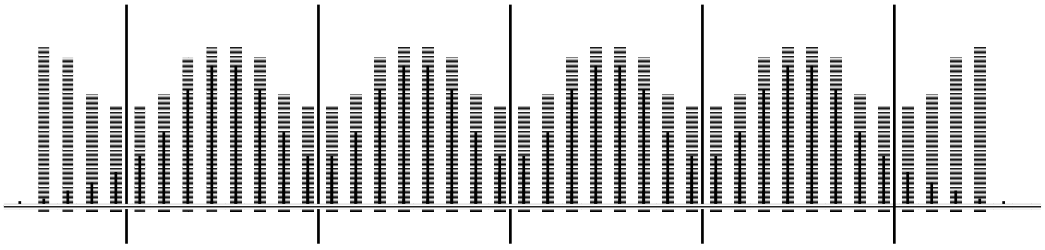


Figure 27.4. Digital pulse train represented by the fat hatched bars.

In this figure, the thin dark bars are the numbers which describe the pulse. The fat hatched bars represent the sum of the thin bars -- remember that we have overlap here.

Note that the resulting sequence -- the fat bars -- form a periodic sequence, just as we had a periodic function given by the heavy curve in Figure 27.2. Note also that again we could have used an "equivalent pulse sequence" here consisting of just the set of 8 fat bars in one interval.

Thus, although our original pulse contained 18 numbers, the minimal pulse contains only 8 numbers. If we now compute the Discrete Fourier Transform coefficients c_m' according to the formula

$$c'_m \equiv (1/N) \sum_{n=-\infty}^{\infty} x_{pulse}(t_n)\, e^{-imn(2\pi/N)}$$

we find that only 8 of them are unique because of the translation rule shown back in the DFT summary box. Select those with m = 0,1,2,3,4,5,6,7. We should be happy to find that it takes only 8 numbers in the transform space (where the c_m' live) to represent the 8 numbers in the time domain which represented our pulse in its minimal representation -- the 8 fat bars in period T_1. For this minimal pulse, there are only 8 non-vanishing terms in the above sum. Thus, the c_m' are related to the eight fat bar heights by a set of numbers which form an 8x8 matrix, namely

$$M_{mn} = (1/N)\ e^{-imn(2\pi/N)}$$

Of the 64 matrix elements, only 8 are unique. They lie equally spaced on a unit circle of radius (1/N).

As a possible application of the Discrete Fourier Transform (DFT), suppose you had some sort of digital circuit that puts out a set of numbers that repeat after every N numbers. Perhaps this is what a scrambler does with a constant input. In this case, you can think of the set of N numbers as forming the envelope of the pulse $x_{pulse}(t)$. The only appropriate "frequency domain" transform of this repeating sequence of numbers is the DFT. In the frequency domain you get a finite set of N numbers c_m' as the transform.

Just as with regular Fourier series coefficients, the DFT coefficients c_m' are a measure of the frequency content of the signal. Recall that the pulse train x(t) is the mapped out by:

$$x(t_n) = \sum_{m=0}^{N-1} c'_m\, e^{+imn(2\pi/N)} = \sum_{m=0}^{N-1} c'_m\, e^{+im\omega_1 t_n} \qquad \omega_1 = (2\pi/T_1)$$

One should think of n taking lots of values and tracing out the envelope of the function x(t). Clearly, coefficient $c_{m'}$ is the weight of frequency component $m\omega_1$. So c'_0 measures the DC component, and c'_1 measures the amount of frequency component ω_1 and so on.

There is a limit on how high a frequency component you can have. If you had a sine wave with period Δt = the sample spacing, it would have the same value at every sample point, and would thus show up in the DC component. The frequency corresponding to period Δt is $\omega_N = N\omega_1$. This is why $c'_N = c'_0$. In a similar fashion, potential frequencies ω_n with n>N are also "aliased" down into lower frequencies according to the translation rule for the c'_m. Thus, the highest frequency we can really have is this one: $(N-1)\omega_1$.

So this gives a reasonable "Fourier explanation" of why there are a finite number of c'_m coefficients involved in the spectral expansion above for $x(t_n)$.

We repeat one more time an important fact stressed earlier: as N, the number of sample points per T_1 interval, increases, the number of DFT coefficients c'_m increases as well, and these c'_m becomes closer and closer to the Fourier Series coefficients c_m. In the limit $N \to \infty$, $c'_m = c_m$, and the DFT and the Fourier Series exactly align.

For finite N, the c'_m differ from the c_m in exactly the same way that the area under a stepwise approximated curve differs from the area under the smooth curve. This fact follows directly from the definitions of the c'_m and c_m.

28. Alternate form of the DFT.

Suppose we arbitrarily define new DFT coefficients as follows:

$$c''_m = N\, c_m' \tag{28.1}$$

With these coefficients the DFT looks like so:

$$\begin{aligned} c''_m &= \sum_{n=-\infty}^{\infty} x_{pulse}(t_n)\, e^{-imn(2\pi/N)} \\ x(t_n) &= (1/N) \sum_{m=0}^{N-1} c''_m\, e^{+imn(2\pi/N)} \end{aligned} \tag{28.2}$$

This change is like moving the 2π around in the Fourier Transform, nothing has changed except the scale of the coefficients. I think (28.2) is the more common normalization used in the literature, and we shall use it from now on. The normal application is to think of $x_{pulse}(t)$ as the signal x(t) over one period, rather than thinking in terms of our Gaussian pulse choice of pulse, so let's rewrite (28.2) in the more conventional form,

$$\begin{aligned} c''_m &= \sum_{n=0}^{N-1} x(t_n)\, e^{-imn(2\pi/N)} \\ x(t_n) &= (1/N) \sum_{m=0}^{N-1} c''_m\, e^{+imn(2\pi/N)} \end{aligned} \tag{28.3}$$

29. Symmetry relation for the c''_m for real x(t).

Just using the first equation above, it is complete trivial to show that

$$c''_{N-m} = c''_m{}^* \tag{29.1}$$

which implies that

$$| c''_{N-m} |^2 = | c''_m |^2 \tag{29.2}$$

This relation holds for either the c''_m or the c'_m normalizations.

Let's look explicitly what this means in a few cases:

N=2	$\| c''_0 \|^2, \| c''_1 \|^2$	0,1 unique
N=3	$\| c''_2 \|^2 = \| c''_0 \|^2 , \| c''_1 \|^2$	0,1 unique
N=4	$\| c''_3 \|^2 = \| c''_1 \|^2, \| c''_2 \|^2, \| c''_0 \|^2$	0,1,2 unique
N=5	$\| c''_4 \|^2 = \| c''_1 \|^2, \| c''_3 \|^2 = \| c''_2 \|^2, \| c''_0 \|^2$	0,1,2 are unique

In general, then, there are N/2+1 unique coefficients squared, integer division implied. This means that when you plot a spectrum in the DFT world, you need only plot coefficients square from m = 0 up through m = N/2+1. The spectrum is **mirror-symmetric** about the midpoint.

30. Parseval's Theorem for the DCT:

For the full Fourier Transform, Parseval's Theorem states that

$$\int_{-\infty}^{\infty} dt\, |x(t)|^2 = \int_{-\infty}^{\infty} df\, |X(f)|^2 \tag{30.1}$$

which we showed in Chapter 1 equation (10.5), but here we use the variable f instead of ω. This says that the energy in a pulse is the same no matter which domain you compute it in.

In the DFT world, we have a similar result, which is:

$$\sum_{n=0}^{N-1} |x(t_n)|^2 = N \sum_{n=0}^{N-1} | c'_n |^2 = (1/N) \sum_{n=0}^{N-1} |c''_n|^2 \tag{30.2}$$

where we show the right side for both normalizations. To prove this result, start with the LHS and insert the DFT expansion twice, using indices m and m' for the time expansions. Then make use of this result,

$$\sum_{n=0}^{N-1} e^{+imn2\pi/N}\, e^{-im'n2\pi/N} = N\,\delta_{m,m'} \qquad (30.3)$$

This is obvious for m = m'. For other cases, it is the usual situation of adding equally spaced vectors around a circle, and you show the result in this case using $1 + x + x^2 + ... + x^{N-1} = (1- x^N) / (1-x)$.

31. Sample DFT

Consider x(t) = sin(2πkt/T) where T is our period which we have divided up into N sample times, and k is the number of full sine waves we jam into our period. We will then have $x(t_n) = \sin(2\pi k t_n/T)$. But we know that $t_n = (n/N)T$ so $x(t_n) = \sin(2\pi kn/N)$. Let's now compute the coefficients:

$$c''_m = \sum_{n=0}^{N-1} x(t_n)\, e^{-imn(2\pi/N)} = \sum_{n=0}^{N-1} \sin(2\pi kn/N)\, e^{-imn(2\pi/N)}$$

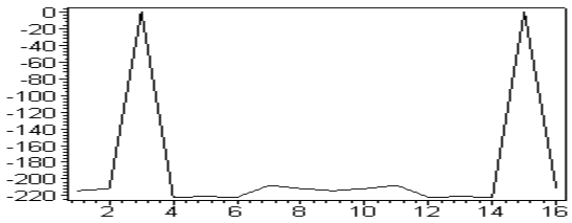
$$= (1/2i)\, N\, [\ \delta_{k,m} - \delta_{N-k,m}\ ]$$

which tells us that, for the sine wave with k periods in T, we have $c''_k = (N/2i)$ and $c''_{N-k} = -(N/2i)$ and all other coefficients are 0. Notice that this satisfies the symmetry relation (29.1). Also, $|c''_k| = N/2$.

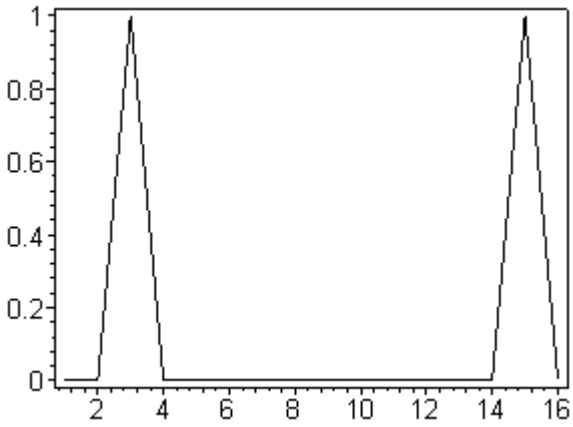
Here is a Maple program used to compute the DFT of this test function, the program is saved somewhere, I put comments into italics here:

```
> restart;
> M := 8:    Exponent of 2 which sets size of the FFTN := 2^M:  Size of the FFT
> k := 2.0:  Parameter in our test sine function below
> x := array(1..N): y := array(1..N):c := array(1..N):  Define empty vectors  for use below
> for n to N do x[n] := evalf(sin(2*Pi*k*(n-1)/N)) od:  Install test function into the x vector
If you don't put evalf above, it puts out messy functional expressions for the x[n]. We just want numbers!
> # for n to N do x[n] od;   View the components of the x vector if you want.
> for n to N do y[n] := 0 od:   Install 0's into the imaginary parts of the test vector
> readlib(FFT): FFT(M,x,y):    Run the FFT, first arg is that power of 2 that makes N, it returns N.
> # for n to N do x[n] od:  View output of the FFT if you want
> # for n to N do y[n] od:
> for n to N do c[n] := ((sqrt(x[n]^2 + y[n]^2))) od:   Compute |c| of the coefficients
Want to normalize coefficients so maximum one is 1.000, so do a scan for cmax
> cmax := 0: for n to N do if c[n] > cmax then cmax := c[n] fi od: cmax:
Rescale the coefficients into dB, not the max will be at 0 dB
for n to N do c[n] := 10*log(c[n]/cmax) od:
> # for n to N do c[n] od; View them if you want
> with(plots): listplot(c,axes=boxed);
```

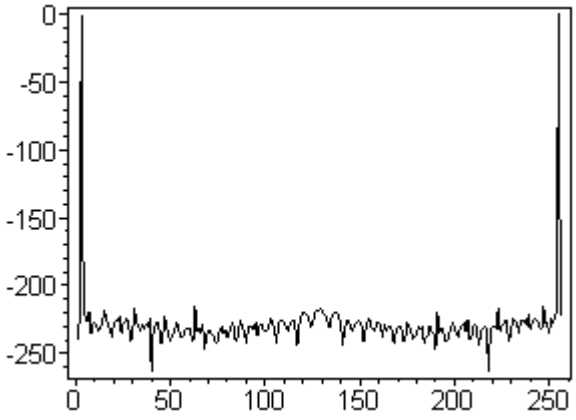
And here are a few plots from this program, notice that we plot 10*log |c| here,



This plot with N = 16 shows our points made above, that only two coefficients are non-zero and that things are mirrored about the center. Due to the log plot, even the smallest numerical calculation error shows up way down low. If you plot this same thing without the log it looks like this, with the same normalization to max peak of unity,



and now the bottom appears to be at the zero level. Here is one more log plot with N = 256, since this is what you might typically see in the literature,



The noise at the bottom is all meaningless, just comes from the slight computational error.

32. The FFT

The Fast Fourier Transform is simply an efficient way to compute a DFT in the special case that N is a power of 2. The method was discovered in 1965 by Cooley and Tukey, and exposes a large amount of redundancy in the calculation for such values of N. The DFT requires N^2 complex multiply/accumulate operations, while the FFT reduces this number to $N \log_2 N$. For N = 1024, the computation is therefore sped up by a factor of about 100. This provides strong motivation for people to do DFT calculations with binary power values of N. The Maple program shown in Section 31 above in fact uses an FFT algorithm, and we used it with N = 256.

33. Lee & Messerschmitt Notation

First, we need to review the various notations *we* have used up to this point and where the corresponding "data" is located:

(a) $X(\omega)$ = true **Fourier** Transform, defined in (1.1), Chapter 1, page 1

(b) $X_{Laplace}(s) = X(s/j)$ at least for $x(t<0) = 0$ functions, (6.3), Chapter 1, page 7

$X_{Laplace}(j\omega) = X(\omega)$ **Laplace** connection (33.1)

(c) $X'(\omega)$ = **Digital Fourier Transform**, Chapter 3 page 9 box

$X'(\omega) = \Sigma_m X(\omega - m\,\omega_1)$ Chapter 3, page 7, (23.1) (33.2)

This shows how my DFT spectrum is related to my FT spectrum.

(d) $X''(z) = X'(\omega)/\Delta t$ Chapter 3, page 10, (24.2) (33.3)

This relates the **Z-Transform** X" spectrum to the DFT X' spectrum.

Now, if we look at Lee and Messerschmitt page 14 equation (2.9), we must conclude that

Me	L&M	
$X_{Laplace}(j\omega)=X(\omega)$	$X(j\omega)$	(33.4)

so they have chosen to use the Laplace form at their fundamental functional form X. This, any time I see something of the form $X(j\omega)$, I will recognize it as my $X(\omega)$.

Next, look at L&M page 14 equation (2.10). They call this their DTFT. If I compare this to my DFT, I can make these connections:

Me	L&M	
T_1	T	(33.5)
ω	ω	

$$X''(z=e^{i\omega T_1}) \qquad X(e^{j\omega T}) \tag{33.6}$$

Therefore, in *these* two equations, L&M are identifying their X with my Z-transform object. Thus, any time I see L&M refer to something as $X(e^{j\omega T})$, I will recognize it as my $X''(e^{j\omega T})$.

Now, can I make a connection between my Z-transform X" object and my regular Fourier X object?

$$X''(z=e^{i\omega T_1}) = X'(\omega)/\Delta t = (1/\Delta t)\,\Sigma_m\, X(\omega - m\,\omega_1) \tag{33.7}$$

If we convert this to L&M notation, by the way, we get this,

$$X(z=e^{i\omega T}) = (1/T)\,\Sigma_m\, X(j[\omega - m\,\omega_1]) \tag{33.8}$$

which appears in L&M as the "fundamental sampling theorem" (2.17) on page 16. Notice that there is **not** a 1-to-1 simple connection here between X" and X. Also, there is a scale factor sitting there! So in no way could you claim that $X''(e^{j\omega T}) = X(\omega)$. I would never claim this, but in their notation they have two objects $X(e^{j\omega T})$ and $X(j\omega)$ which are as different as $f(e^{j\omega T})$ and $g(j\omega)$, but which they make you think might actually be the same function evaluated at a different loci in a complex plane. Perhaps it is because I am used to seeing analytic continuation that this bothers me.

Therefore, to say it again, L&M are using *two different meanings* of function X in equation (2.9) compared with (2.10). The meaning of the first X is Laplace, and the meaning of the second X is Z-Transform. They use the same letter X without any way to distinguish which transform they are talking about, I do not like this. You really need to examine the form of their argument to know which transform they are using!

Next, look at their equation (2.13). This starts with equation (2.2) which is the usual PAM equation which I wrote earlier as w(t) = x(t)y(t) in my Chapter 3 (see (20.3) and (20.6)). I have to switch the roles of my y and x to match L&M, so we have,

$$w(t) = \Sigma_m\, x(t_m)\ \delta(t - mT) \qquad // = \hat{x}(t) \text{ in L\&M (2.2)} \tag{33.9}$$

My Fourier Transform of this equation gave my (20.5),

$$T_1 W(\omega) = \Sigma_m\, X(\omega - m\omega_1) = \text{main spectrum plus the images} \tag{33.10}$$

What I call $W(\omega)$, L&M are calling $\hat{X}(j\omega)$. This is reasonable, because they refer to my w(t) as their $\hat{x}(t)$. So they mean by $\hat{X}(j\omega)$ the normal FT in *their* notation, which is what I mean by $W(\omega)$.

Now, compare the last two equations above to obtain this result:

$$W(\omega) = X''(z=e^{i\omega T_1}) \tag{33.11}$$

If I translate this result into L&M notation, I should get:

$$\hat{X}(j\omega) = X(z= e^{i\omega T}) \tag{33.12}$$

and this agrees with L&M (2.13). Now, what does this (33.10) or (33.11) say? The RHS is the Z-Transform of some reasonable sampled function x(t), evaluated at $z = e^{i\omega T}$. The LHS is the Fourier Transform of the delta pulse train that you make from the samples of x(t). We use the appropriate tool on the left side (the FT) to deal with the continuous time function w(t), and we use the appropriate tool on the right side (the ZT) to deal with a sampled-time function $x(t_n)$. We then find the above connection between the two transforms evaluated at the points shown.

34. The Nyquist Sampling Theorem

It seems a bit strange that we are mentioning this idea formally at such a late point in our discussion, we have been more or less assuming it earlier. Here we make the official statement of this theorem.

Suppose we take our delta-function PAM pulse train w(t) as shown in (33.9) above and run it through a box filter that cuts off at $\omega_1/2 = \pi/T$, and we examine the output o(t). We then have the following convolution from (3.1),

$$o(t) = w(t) * box(t) = \int_{-\infty}^{\infty} dt' \, box(t-t') \, w(t') \qquad (34.1)$$

Inserting (33.8) for w(t), we get

$$o(t) = \Sigma_m \, x(t_m) \, box(t-t_m) = \Sigma_m \, x(t_m) \, (1/T_1) \, sinc(\pi/T_1 \, [\, t - t_m]) \qquad (34.2)$$

where I used (9.2) for box(t) by setting $\tau = \omega_1 = 2\pi/T_1$ and $t \leftrightarrow \omega$, and also adding an extra factor $1/2\pi$ because our Fourier Transform is not symmetrical! Maybe it is good to write all this out:

$$Box(\omega) = [\, \theta(\omega + \omega_1/2) - \theta(\omega - \omega_1/2) \,]. \qquad (34.3)$$

Apply (1.2) to get the time domain version of this spectrum,

$$box(t) = (1/2\pi) \int_{-\omega_1/2}^{\omega_1/2} d\omega \, e^{+i\omega t}$$
$$= (1/2\pi) \, (\omega_1) \, sinc(t \, \omega_1/2) = (1/T_1) \, sinc(\pi t/T_1) \qquad (34.4)$$

So, what is this o(t) ? Look in the frequency domain and repeat the above process,

$$O(\omega) = Box(\omega) \, W(\omega) \qquad (34.5)$$

But from (33.10), *if* we have non-overlapping image spectra, then we can write

$$O(\omega) = Box(\omega) \, W(\omega)$$
$$= Box(\omega) \, (1/T_1) \, \Sigma_m \, X(\omega - m\omega_1) = (1/T_1) \, Box(\omega) X(\omega) \qquad (34.6)$$

because the box filter blocks all the image spectra. But, since we assume that the entire main spectrum of $X(\omega)$ fits under the box, we end up with

$$O(\omega) = (1/T_1)\, X(\omega) \tag{34.7}$$

from which we conclude that

$$o(t) = (1/T_1)x(t) \tag{34.8}$$

which gives us our final result, namely,

$$\begin{aligned} x(t) = T_1\, o(t) &= T_1 \Sigma_m\, x(t_m)\, (1/T_1)\, \mathrm{sinc}(\pi/T_1\, [\, t - t_m]) \\ &= \Sigma_m\, x(t_m)\, \mathrm{sinc}(\pi/T_1\, [\, t - t_m]) \end{aligned} \tag{34.9}$$

and this shows how $x(t)$ can be *reconstructed* from the samples $x(t_m)$, assuming you obey the Nyquist Limit. This last result is the **Nyquist Sampling Theorem**, and it appears in L&M as (2.19). In the reconstruction, you just weight each sample with a sinc function centered at that sample's time.