(Notes supplied by Dr Frank Stootman, UWS, 2001)

The Fourier Transform

1.0 Basic Ideas

- Any general periodic signal has the automatic property $f(t) = f(\frac{2\pi t}{T})$ where T is the period of the signal. The 2π is "snuck" in because we know that trigonometric functions are good examples of repetition. The complexity of $f(t)$ is irrelevant as long as it repeats itself faithfully. Please keep in mind that 't' for radioastronomy is usually time, but in fact it is an arbitrary variable and so what follows below is applicable provided the variable has functional repetition in some way with a repeat T. Thus spatial repetition is another important variable to which we may apply the theory.

- Fourier discovered that such a complex signal could be decomposed into an infinite series made up of cosine and sine terms and a whole bunch of coefficients which can (surprisingly) be readily determined.

$$f(t) = \frac{1}{2}a_0 + \sum_{n=1}^{\infty} a_n \cos(\frac{2\pi nt}{T}) + \sum_{n=1}^{\infty} b_n \sin(\frac{2\pi nt}{T})$$

- If you like, we have decomposed the original function $f(t)$ into a series of basis states. For those of you who like to be creative this immediately begs the question of: is this the only decomposition possible? The answer is no.

- The coefficients are "readily" determined by integration.

$$a_n = \frac{2}{T}\int_{-T/2}^{T/2} f(t)\cos(\frac{2\pi nt}{T})dt$$
$$b_n = \frac{2}{T}\int_{-T/2}^{T/2} f(t)\sin(\frac{2\pi nt}{T})dt$$

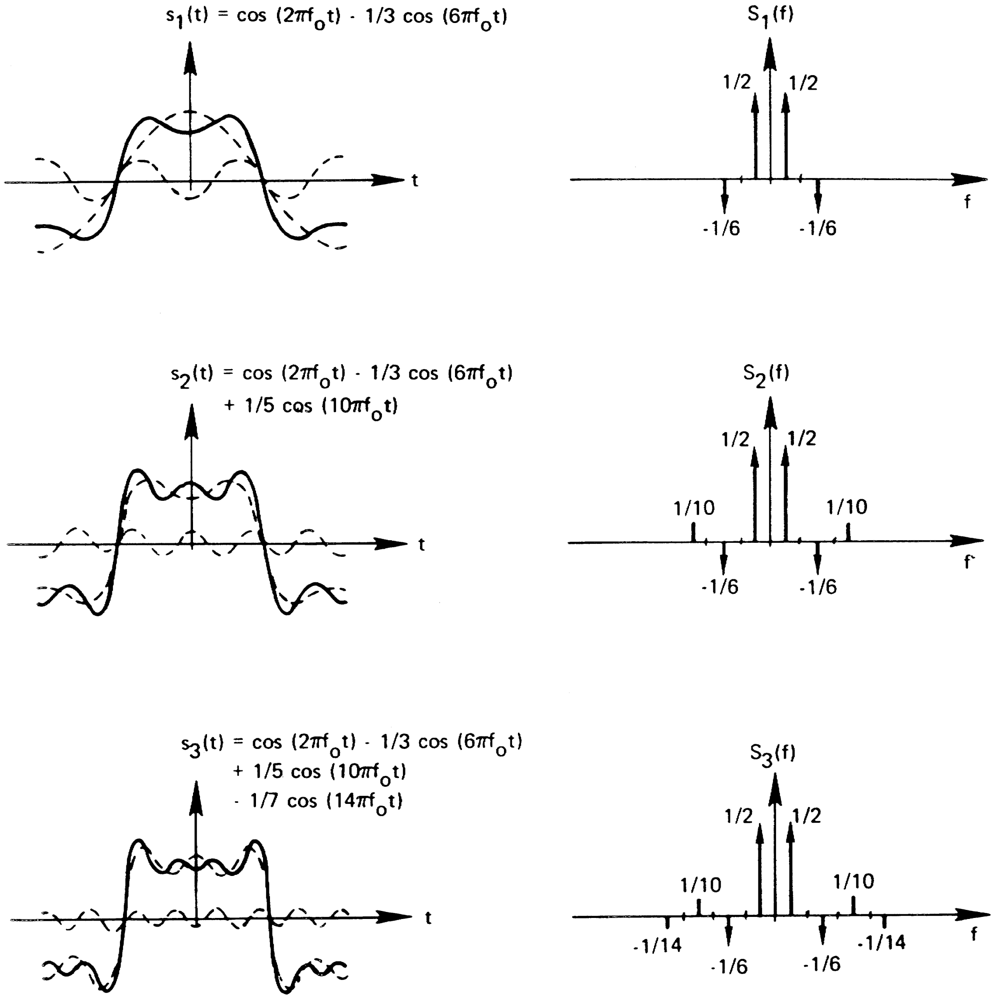
- Introducing complex notation we can simplify all of the above to what you often see in textbooks.

$$f(t) = \sum_{n=-\infty}^{\infty} c_n \exp(-i\frac{2\pi nt}{T})$$

$$c_n = \frac{1}{T}\int_{-T/2}^{T/2} f(t)\exp(i\frac{2\pi nt}{T})dt$$

- Here $c_0 = \frac{1}{2}a_0$, $c_n = \frac{1}{2}(a_n + ib_n)$, and $c_{-n} = \frac{1}{2}(a_n - ib_n)$.

- The graphical example below indicates how addition of cosine time function terms are Fourier transformed into coefficients. In this case only $c_n = a_n / 2$. Take care the centre line with the big arrow is to mark the axis only – it is **not** part of the coefficient display. Notice also that **two** coefficient lines appear for every frequency. The latter is related to the Nyquist sampling theorem (see below) and is also why the coefficient magnitudes are halved. Notice also the **spacing** of the coefficients to be an integral multiple of $f_0 = 1/T$ with the sign consistent with the input waveform.



s1(t) = cos (2πf0t) - 1/3 cos (6πf0t)
t
S1(f)
1/2
1/2
-1/6
-1/6
f
s2(t) = cos (2πf0t) - 1/3 cos (6πf0t) + 1/5 cos (10πf0t)
t
S2(f)
1/2
1/2
1/10
1/10
-1/6
-1/6
f
s3(t) = cos (2πf0t) - 1/3 cos (6πf0t) + 1/5 cos (10πf0t) - 1/7 cos (14πf0t)
t
S3(f)
1/2
1/2
1/10
1/10
-1/14
-1/6
-1/6
-1/14
f

(b)

(after 'The Fast Fourier Transform', E.O. Brigham, Prentice Hall, 1974)

- It is important to stress that it is an ***intrinsic*** property that the c_n are discrete. This is sometimes very confusing in text books because they draw them as continuous functions. It ***is*** possible to make them a continuous function by doing a simple trick and imagining that T is enormously large, or better still it tends to infinity. Thus the repeat period is infinite.

- Thus with a little pure mathematics and the substitutions, $s = \frac{2\pi n}{T}$, leading smoothly to $ds = \frac{2\pi}{T} dn$ as $T \to \infty$, and also introducing the continuous function $F(s)$ to replace the discrete c_n, we get a lovely functional symmetry:

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} F(s) \exp(-ist) ds$$

$$F(s) = \int_{-\infty}^{+\infty} f(t) \exp(ist) dt$$

(Reminder: t and s are arbitrary variables!)

- For a number of functions of time $f(t)$ the corresponding continuous transform shapes $F(s)$ are given in the accompanying diagrams below. It is immediately obvious that ‘s’ has units of frequency and so we talk of transforming the repetitive time function into the frequency domain. We have analysed time behaviour into its corresponding frequency components.

2.0 Digital Fourier Transform

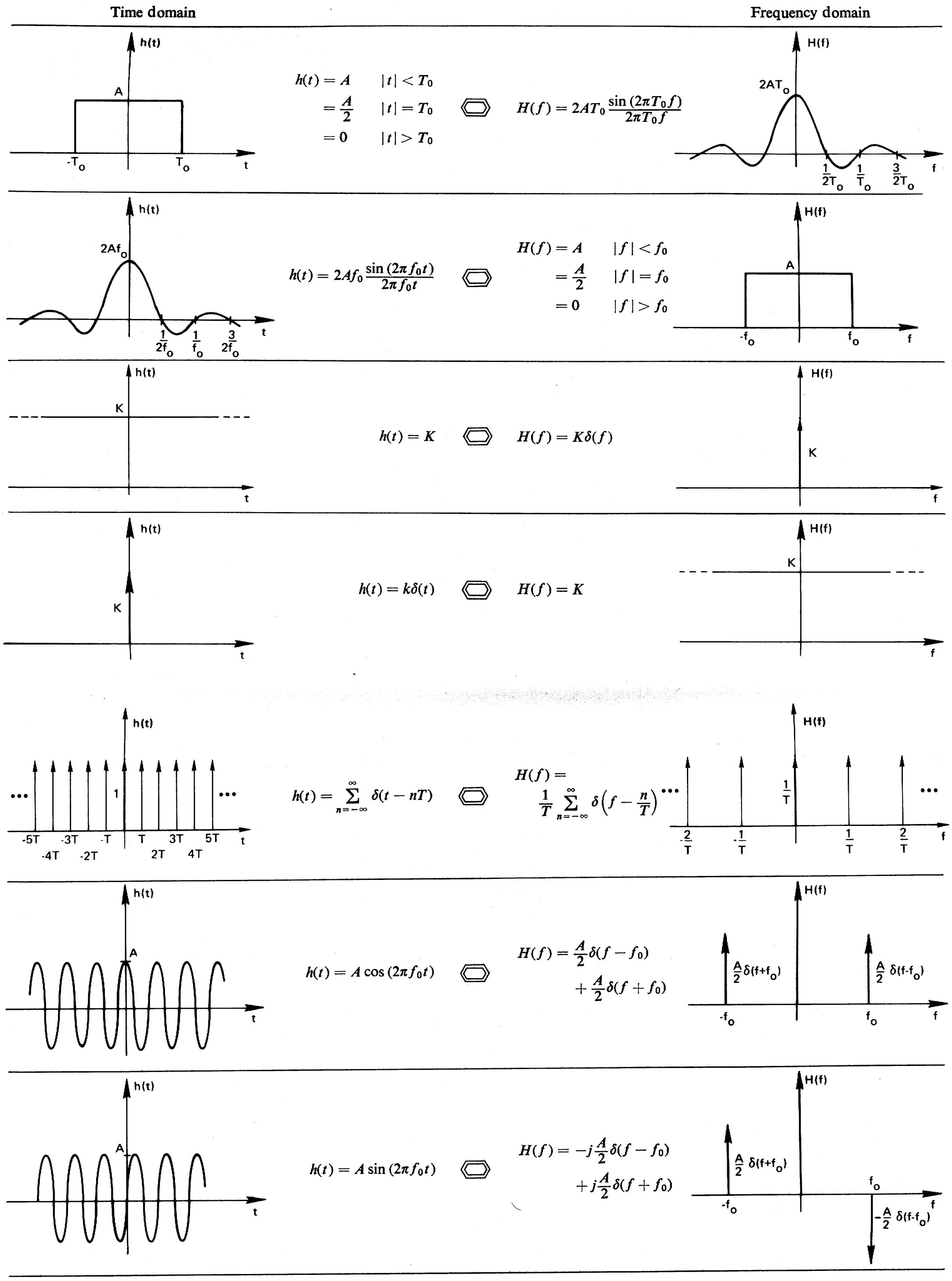
In the real world of experimental physics we do not have the luxury of infinite time nor can we necessarily describe analytically our function $f(t)$. It is normally derived from the experimental equipment attached to something (eg., receiver voltage from an antenna). Thus we need something practical to do an FT.

The above has three simple consequences:

1) We need to choose a total sample time T recognising 1/T will then be the frequency spacing or resolution we can get out of the transform. Thus if T=1.5s we obtain our frequency coefficients at a spacing of 0.67 Hz.

2) During T, we need to sample our waveform N times to produce a sampling vector representing our continuous time domain. Thus

$$f(t) \leftrightarrow \{x_n\} \text{ with } 0 < n < N - 1$$



$$h(t) = A \quad |t| < T_0; \qquad = \frac{A}{2} \quad |t| = T_0; \qquad = 0 \quad |t| > T_0 \qquad \Longleftrightarrow \qquad H(f) = 2AT_0 \frac{\sin(2\pi T_0 f)}{2\pi T_0 f}$$

$$h(t) = 2Af_0 \frac{\sin(2\pi f_0 t)}{2\pi f_0 t} \qquad \Longleftrightarrow \qquad H(f) = A \quad |f| < f_0; \qquad = \frac{A}{2} \quad |f| = f_0; \qquad = 0 \quad |f| > f_0$$

$$h(t) = K \qquad \Longleftrightarrow \qquad H(f) = K\delta(f)$$

$$h(t) = k\delta(t) \qquad \Longleftrightarrow \qquad H(f) = K$$

$$h(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT) \qquad \Longleftrightarrow \qquad H(f) = \frac{1}{T} \sum_{n=-\infty}^{\infty} \delta\left(f - \frac{n}{T}\right)$$

$$h(t) = A\cos(2\pi f_0 t) \qquad \Longleftrightarrow \qquad H(f) = \frac{A}{2}\delta(f - f_0) + \frac{A}{2}\delta(f + f_0)$$

$$h(t) = A\sin(2\pi f_0 t) \qquad \Longleftrightarrow \qquad H(f) = -j\frac{A}{2}\delta(f - f_0) + j\frac{A}{2}\delta(f + f_0)$$

Figure 2-11. Fourier transform pairs.

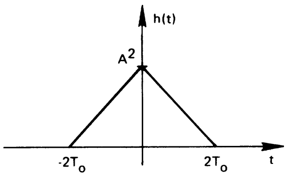
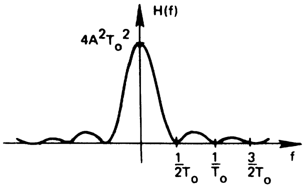
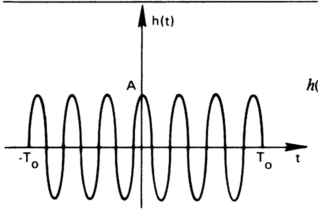
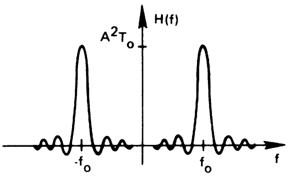
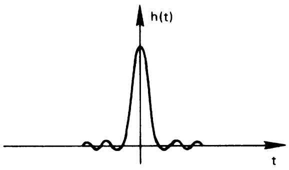
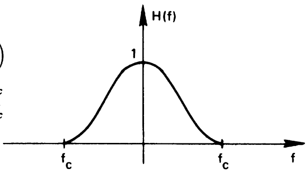
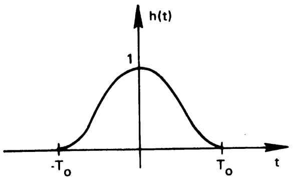
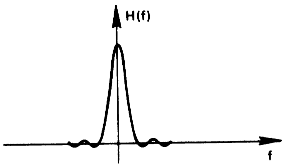
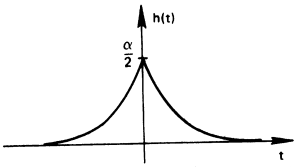
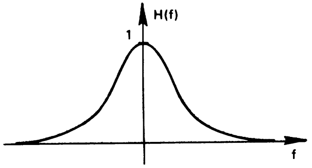
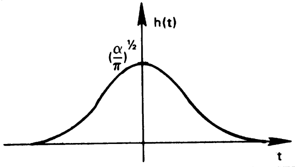
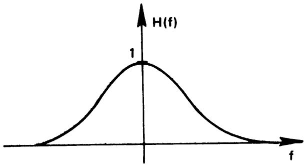
Time domain				Frequency domain
	$h(t) = -\frac{A^2}{2T_0}t + A^2 \quad \lvert t\rvert < 2T_0$ $= 0 \quad \lvert t\rvert > 2T_0$	⇔	$H(f) = A^2\frac{\sin^2(2\pi T_0 f)}{(\pi f)^2}$	
	$h(t) = A\cos(2\pi f_0 t) \quad \lvert t\rvert < T_0$ $= 0 \quad \lvert t\rvert > T_0$	⇔	$H(f) = A^2T_0[Q(f+f_0) + Q(f-f_0)]$ $Q(f) = \frac{\sin(2\pi T_0 f)}{2\pi T_0 f}$	
	$h(t) = \frac{1}{2}q(t) + \frac{1}{4}q\left(t+\frac{1}{2f_c}\right) + \frac{1}{4}q\left(t-\frac{1}{2f_c}\right)$ $q(t) = \frac{\sin(2\pi f_c t)}{\pi t}$	⇔	$H(f) = \frac{1}{2} + \frac{1}{2}\cos\left(\frac{\pi f}{f_c}\right) \quad \lvert f\rvert \leq f_c$ $= 0 \quad \lvert f\rvert > f_c$	
	$h(t) = \frac{1}{2} + \frac{1}{2}\cos\left(\frac{\pi t}{T_0}\right) \quad \lvert t\rvert \leq T_0$ $= 0 \quad \lvert t\rvert > T_0$	⇔	$H(f) = \frac{1}{2}Q(f) + \frac{1}{4}\left[Q\left(f+\frac{1}{2T_0}\right) + Q\left(f-\frac{1}{2T_0}\right)\right]$ $Q(f) = \frac{\sin(2\pi T_0 f)}{\pi f}$	
	$h(t) = \frac{1}{2}\alpha\exp(-\alpha\lvert t\rvert)$	⇔	$H(f) = \frac{\alpha^2}{\alpha^2 + 4\pi^2 f^2}$	
	$h(t) = \left(\frac{\alpha}{\pi}\right)^{1/2}\exp(-\alpha t^2)$	⇔	$H(f) = \exp\left(\frac{-\pi^2 f^2}{\alpha}\right)$	

Figure 2-11 (continued).

(after 'The Fast Fourier Transform', E.O. Brigham, Prentice Hall, 1974)

3) We need to modify our Fourier transform to cope with a discrete input vector. Following our previous work we intuitively make integrals go to sums, T go to N, $f(t)$ go to x_n, and c_n go to X_n (though this is just a notation change since c_n are already a discrete set!). We are left with the practical Digital Fourier Transform. or DFT pair.

$$X_n = \sum_{k=0}^{N-1} x_k \exp(i\frac{2\pi nk}{N})$$

$$x_k = \frac{1}{N}\sum_{n=0}^{N-1} X_n \exp(-i\frac{2\pi nk}{N})$$

Notes:

1) $i = \sqrt{-1}$ and is easily confused with the integer counters n and k.
2) The placement of 1/N factor is somewhat arbitrary, but convention places it as shown. It is important for normalisation – don't lose it!

- The above is readily programmed and the N input samples in time are converted to N frequency samples spaced at 1/T. Essentially the transform is a coefficient matrix multiplication from which you can see the output frequency vector is a weighted sum of input time samples.

$$\begin{pmatrix} X_0 \\ .. \\ X_n \end{pmatrix} = \begin{pmatrix} W_{00} & .. & W_{0n} \\ .. & .. & .. \\ W_{n0} & .. & W_{nn} \end{pmatrix} \begin{pmatrix} x_0 \\ .. \\ x_n \end{pmatrix} \text{with } W_{nk} = \exp\left(i\frac{2\pi nk}{N}\right)$$

From this it can be seen, for instance, that X_0 is the DC average of x_n since $W_{0n} = 1$.

- Practically for large N the above straight forward calculation becomes a massive computational burden of order N^2. Happily, a number of people have discovered patterns in W_{nk} which can be exploited. The best has a computational burden of $N \log N$ and is called a Fast Fourier Transform or FFT. Please explore "Numerical Recipes in C ", 2nd Ed, Cambridge, 1992) . It is important to note the FFT is only a computational device and nothing more!

3.0 Power Spectra

- The complex representation is compact, but needs to be converted to something useful otherwise you end up with two axes of coefficients. Practically we form the power coefficient spectrum as a function of frequency as follows

$$P_n = X_n \times X_n^*$$

It is this which is displayed on spectrum analysers and how most spectra are drawn.

- Fourier transforms have a conservation law which is ultimately related, physically, to the conservation of energy. Thus:

$$\sum_N x_k x_k^* = \frac{1}{N}\sum_N X_n X_n^* = \sum_N P_n$$

Effectively, we are redistributing the input vector into a new basis system but retaining coefficient conservation which is always nice in physics. It is also called Parseval's theorem.

4.0 Imperfections, Limitations and Trade-Offs

- Consider the frequency coefficient X_{N-r}.

$$X_{N-r} = \sum_{k=0}^{N-1} x_k \exp(i\frac{2\pi(N-r)k}{N}) = \sum_{k=0}^{N-1} x_k \exp(i2\pi k)\exp(-i\frac{2\pi rk}{N}) = \sum_{k=0}^{N-1} x_k \exp(-i\frac{2\pi rk}{N})$$

If the input vector x_k is real (and it usually is since it is a time varying voltage) then $X_{N-r} = X_r^*$ from which it follows that $P_r = P_{N-r}$.

- This means that in any Fourier derived power spectrum there is a *duplication* of frequency coefficients. Thus only N/2 points are unique and so by reverse implication if you have your heart set on a particular BW analysed with a resolution of M points you will require 2M samples of the time spectra. This is called the Nyquist sampling theorem.

- The finite sampling time T creates an artificial effect in which the value of the frequency coefficients "leaks" into adjacent coefficient positions. This means you get a reduced value of the wanted coefficient and an adulteration of adjacent coefficients. It is customary therefore to pre-multiply your input time coefficients x_k with a windowing or weighting function over all the time points to reduce this coefficient leakage in X_n. An old friend is the Hanning window which applies a weighting w_k to each x_k given by,

$$w_k = \frac{1}{2}\left(1+\cos\left(\frac{2\pi k}{N}\right)\right)$$

The above is sometimes also called Hanning smoothing. There are other functions. See "Numerical Recipes in C" referred to above for these.

- The maths for the above revolve around convolution and the Fourier transform of convolution. Here is the sketch.

  1. Remember the convolution product, which is a function of time, is defined as $f * g = \int_{-\infty}^{t} f(\tau)g(t-\tau)d\tau$. Good physical examples of this process are "flywheel" action seen in analogue filters, the flywheels in cars and door dampeners. The past affects the present!

  2. The Fourier transform of this is $R(s) = \int_{-\infty}^{+\infty}\left(\int f(\tau)g(t-\tau)d\tau\right)\exp(ist)dt$, which on making the simple substitution $u = t - \tau$, leads after a small fiddle, to $R(s) = F(s)G(s)$. This is an important result which states that convolution in time space is multiplication in frequency space and *vice versa*.

  3. The window problem occurs because you are multiplying the input time samples with a square sampling window. This leads to the convolution of these two in frequency space and thus to practical problems. Provided you pick it well, pre-smoothing with a suitable function minimises this convolution effect.

- Finally, the you must pick a sample time which ensures that all the frequencies in your time signal are resolved in the resulting bandwidth of the frequency coefficients. How do you do this since you don't know what is there beforehand? In practical systems you **filter** out frequencies you are not interested in **before** the FFT process. Failure to do this leads to signal aliasing or the unwanted higher frequencies folding back into your spectrum in unwanted places.

4.0 Thinking in terms of the Fourier Transform

- Digital filtering on a static input time sample can be done taking a FFT of the vector and then applying the desired filter shape to the resultant coefficients. Now apply an inverse FFT and the result is a filtered time set. Clean up your old records this way by converting the sound to digital sample files and process them on a PC.

- Continuous digital filtering on a continuous time sample can be thought of as deliberately convolving the incoming signal with a function which is the inverse FT of the filter shape. This is how many digital filters called FIR filters work. They do the job on the fly.

There are a number of techniques to do FFT's quickly and so get power spectra. Correlators use the following simple idea:

1. The autocorrelation of a signal $f(t)$ with a time shifted version of itself $f(t+\tau)$ is given by $A(\tau)=\int f(t)f(t+\tau)d\tau$.

2. If we take the FT of this in τ space we get, after a fiddle with variable substitution, $\int A(\tau)\exp(i2\pi s\tau)d\tau = F(s)F^{*}(s)$, provided we assume $f(t)$ is a real function (which of course it will be in our case). But this is exactly the desired power spectrum since $P(s)=F(s)F^{*}(s)$.

3. Producing a fast autocorrelator using a shift register and a bit of electronics allows $A(\tau)$ to be produced efficiently by continually multiplying a sampled signal with previous samples of itself. The final vector can then be converted into a power spectrum with a single FFT.