2

The Wireless Communications Engineer's Toolokit

"There is no distance on Earth that radio communications cannot conquer."
—Guglielmo Marconi, 1901, Comment to his own successful first trans-Atlantic radio communication experiment

Communications is information transfer, and information transfer is signal transmission. The conclusion is that the foundations of communications lie in the treatment of signal. Radio signals, in our case. This chapter will review, with the mere intent of performing a survey of supposedly known concepts, the issues of signal analysis

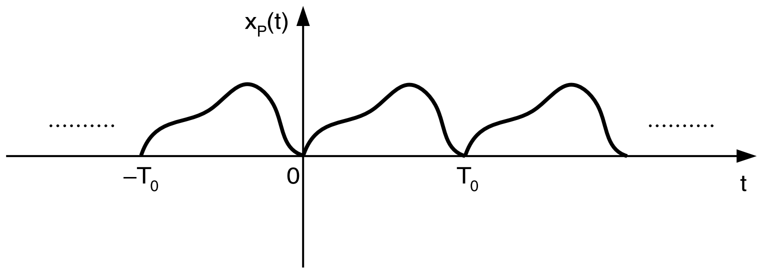


Fig. 2.1 Example of a periodic signal

and filtering, data modulation, synchronization and detection, with a few hints on multiple access. This will be the background expertise on which the remaining three chapters will capitalize.

2.1 BASICS OF FOURIER ANALYSIS OF ANALOG SIGNALS

The "Swiss Knife" of every communications engineer dealing with wireless systems design is Fourier analysis. We do not pretend to perform a comprehensive review of such a huge and fundamental topic. We just want to re-state here the main results and settle a notation concerning the analysis of time-continuous and time-discrete signals that will be the foundation of many, many concepts and tools we will extensively use in the next Chapters.

2.1.1 Periodic Signals and the Fourier Series

We've been taught back in primary schools that white light is a combination of all colors. This is a very first example of Fourier analysis. To be a little bit more specific, we know that every periodic signal $x_p(t)$ (i.e., such that $x_p(t) = x_p(t+T_0)$ for some $T_0 > 0$ that is called *repetition period* as in Fig. 2.1 can be decomposed into a sum of simpler periodic signals, namely, sinusoids as follows:

$$x_p(t) = A_0 + A_1\cos(2\pi f_0 t + \theta_1) + A_2\cos(2\pi 2 f_0 t + \theta_2) + ...$$
$$+ A_k\cos(2\pi k f_0 t + \theta_k) + ... \qquad (2.1)$$

Apart from the constant, DC value, A_0, it is seen that the sinusoids oscillate at frequencies kf_0, the so-called *harmonic frequencies*, that are integer multiples of the *fundamental* or *repetition* frequency $f_0 = 1/T_0$. The k-th component of the expansion (2.1) bears an amplitude A_k, and a phase θ_k. Whilst the value of the oscillation frequencies are always the same for any T_0-periodic signal, the specific values of A_k and θ_k do depend on the shape of the actual $x_p(t)$ under analysis. The sequence of coefficients A_k is called the *amplitude spectrum* of $x_p(t)$, and the sequence of the phases θ_k is the *phase spectrum*. Knowledge of the amplitude and phase spectra is

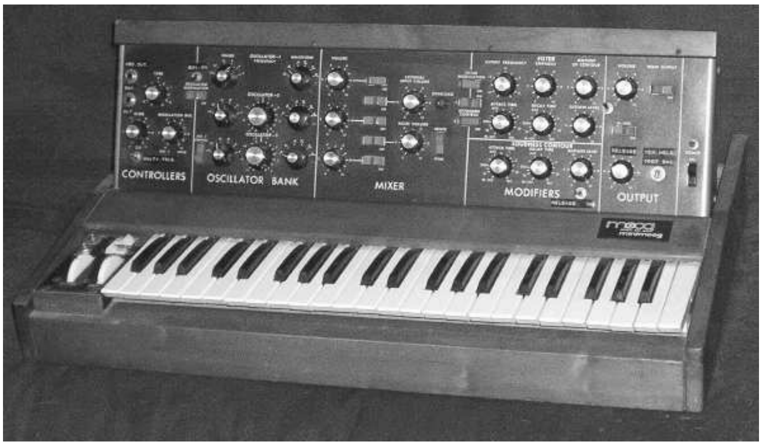


Fig. 2.2 The MiniMoog

equivalent to knowledge of the signal itself, since based on that knowledge we can from (2.1) *synthesize* back the signal in the time domain. This is why (2.1) is called the *synthesis* equation. And this is why the popular musical instrument MiniMoog of the glorious 70's was called a *synthesizer*: it simply implemented (2.1) by heroic low-cost analog hardware to generate (arbitrary) periodic waveforms to be used in musical compositions as a replacement of naturally-generated sounds.

Equation (2.1) is the simplest form of a *Fourier series* for a periodic signal. In a sense, the pitfall of such representation is that exact synthesis of the periodic signal theoretically requires an *infinite* number of components. Nonetheless, such a representation can be used in the practice by truncating the series to a (small) number of significant components only, just as the MiniMoog used to do. Figure 2.3 shows how a periodic rectangular pulse train can be synthesized by the superposition of a finite number of elementary sinusoidal components, according to (2.1). Of course, given a certain waveform that we intend to synthesize, the problem is: what are the correct values of A_k and θ_k to be used in our synthesis equation? Giving a response to this question means *analyzing* signal $x_p(t)$ by means of a proper *analysis equation* that we are to find. This is most easily done by resorting to a complex-number representation of the Fourier series. The key to such representation is Euler's formula for the cosine function:

$$A_k \cos(2\pi k f_0 t + \theta_k) =$$

$$\frac{1}{2}\left[A_k \exp(\jmath 2\pi k f_0 t) \cdot \exp(\jmath\theta) \, + \, A_k \exp(-\jmath 2\pi k f_0 t) \cdot \exp(-\jmath\theta)\right] \qquad (2.2)$$

The real-valued oscillating function is decomposed as the sum of two *rotating vectors* on the complex plane. The first one, $\exp(\jmath 2\pi k f_0 t)$ rotates counterclockwise with a frequency f_0 cycles/s (Hz), and the second one, $\exp(-\jmath 2\pi k f_0 t)$, rotates (counterclockwise) at the *negative frequency*$-f_0$ Hz. Teh sun of the two complex rotating vectors gives just the real-valued cosinusoidal oscillation we started from. This com-

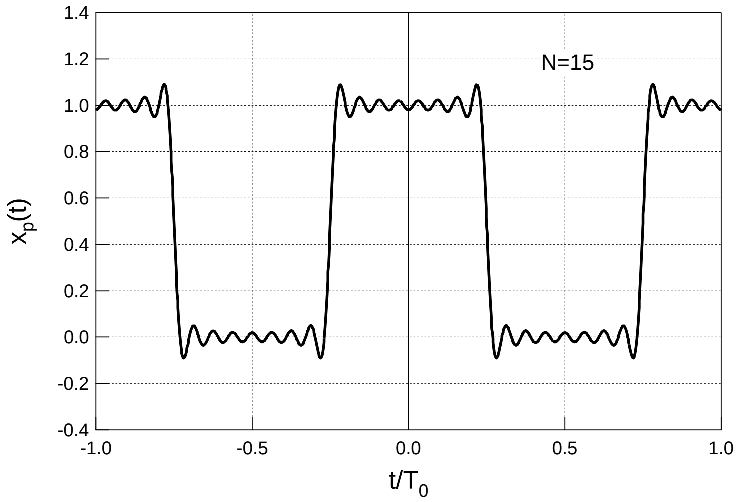


Fig. 2.3 Synthesis of a rectangular pulse train with a finite number N=15 of sinusoidal components

plex decomposition entails the introduction of (complex) signal components with negative frequencies. The amplitude and phase spectra of the sinusoid are collapsed into a single complex-valued coefficient $X_k = A_k \exp[\jmath\theta_k]$ (k positive) that is called the *Fourier coefficient* of $x_p(t)$. Elaborating (2.1) with (2.2), we get the following expression of the Fourier series containing the complex Fourier coefficients X_k:

$$x_p(t) = \sum_{k=-\infty}^{\infty} X_k \exp(\jmath 2\pi k f_0 t) \tag{2.3}$$

that is exactly equivalent to the real-valued form (2.1). Finding the amplitude and phase spectra A_k and θ_k is tantamount to finding the k-th Fourier coefficient X_k. With some effort, it is found that the *analysis equation* we were looking for is relatively simple:

$$X_k = \frac{1}{T_0}\int_{-T_0/2}^{T_0/2} x_p(t)\exp(-\jmath 2\pi k f_0 t)dt \tag{2.4}$$

Example 2.1

Let us analyze the *pulse train* $x_p(t)$ we tried to synthesize in Fig. 2.3. Its Fourier coefficient is given by

$$X_k = \frac{1}{T_0}\int_{-T_0/2}^{T_0/2} x_p(t)\exp(-\jmath 2\pi k f_0 t)dt = \frac{1}{T_0}\int_{-T_0/2}^{T_0/2} x_p(t)\cos(2\pi k f_0 t)dt \tag{2.5}$$

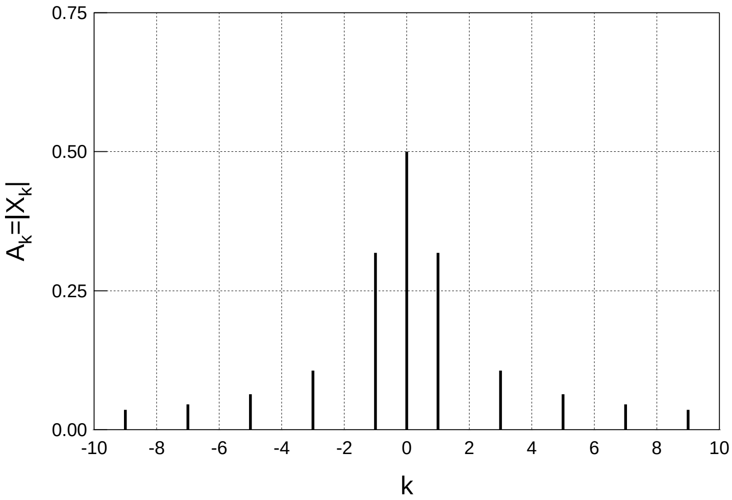


Fig. 2.4 Amplitude spectrum of a rectangular pulse train

where we have exploited the even-symmetry of our waveform. Now, considering that $x_p(t)$ is piecewise-constant, we also have

$$X_k = \frac{1}{T_0}\int_{-T_0/4}^{T_0/4} \cos(2\pi k f_0 t)dt = \frac{1}{T_0}\frac{\sin(2\pi k f_0 t)|_{-T_0/4}^{T_0/4}}{2\pi k f_0} =$$

$$= \begin{cases} 1/2 & k = 0 \\ 0 & k = 2m, m \neq 0 \\ \frac{(-1)^m}{\pi k} & k = 2m+1 \end{cases} \qquad (2.6)$$

The Fourier coefficients X_k turns out to be real-valued (due to the even symmetry of $x_p(t)$); the resulting amplitude line spectrum of $x_p(t)$ is shown in Fig. 2.4.

2.1.2 Non-periodic Signals and the Fourier Transform

What was said until now was only applicable to *periodic* signals. What happens with impulsive signals $x(t)$ that are not periodic? Figure 2.5 shows a *rectangular pulse* that we define as follows:

$$\text{rect}(t/T) \triangleq \begin{cases} 1 & |t| \leq T/2 \\ 1/2 & |t| = T/2 \\ 0 & \text{elsewhere} \end{cases} \qquad (2.7)$$

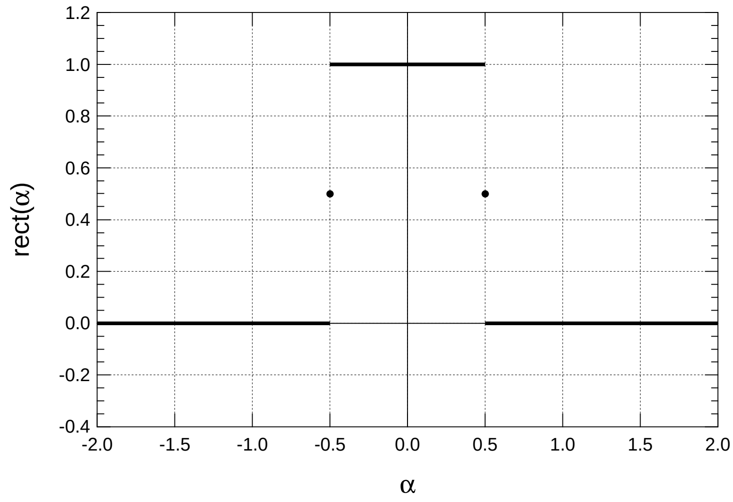


Fig. 2.5 rect signal

The question is: is this kind of signal amenable to Fourier analysis/synthesis? The answer is (of course) positive, and how to do it can be found quite easily if we think of a non-periodic signal as a periodic signal with *infinitely-long* repetition period. The fundamental frequency thus becomes vanishingly small (infinitesimal), and so two components in the frequency spectrum of the signal that were previously separated by $\Delta f = (k+1)f_0 - kf_0 = f_0$ now become infinitesimally close to each other. The frequency spectrum of the signal that used to be *discrete* (the line spectrum of Fig. 2.4 with a frequency "quantum" given by the fundamental frequency f_0) now becomes *continuous*. The relevant analysis equation turns out to be now

$$X(f) = \int_{t=-\infty}^{\infty} x(t) \exp(-\jmath 2\pi f t) dt \tag{2.8}$$

where the frequency f that appears as the argument of this complex-valued quantity $X(f)$ takes all *real* values with continuity.

This counterpart of the former Fourier coefficient is called the *Fourier Transform* (FT) of $x(t)$ and bears the same meaning as X_k. It has an amplitude $|X(f)|$ and a phase $\angle X(f)$, so that we still speak of (continuous) amplitude and phase spectra, respectively. Since the spectrum is now continuous, the synthesis equation cannot be a series any more, rather it is expressed in the form of a *Fourier Integral*:

$$x(t) = \int_{f=-\infty}^{\infty} X(f) \exp(\jmath 2\pi f t) dt \tag{2.9}$$

This relation is also called the *Inverse Fourier Transform* (IFT) of $X(f)$. The physical meaning that we can attach to the pair of relations (2.8)-(2.9) is the same as with the Fourier coefficient-series pair: the IFT is a synthesis equation that tells us how to build our own signal starting from a set of simpler components (the complex-valued sinusoids), and the FT tells us the specific values of the amplitude and phase of each sinusoid that has to be used in our synthesis procedure to build a specific signal. We can show easily that the FT $X(f)$ of a real-valued $x(t)$ (that we will denote at times $\mathcal{F}[x(t)]$) has a particular kind of symmetry that is called *Hermitian*: $X(-f) = X^*(f)$, i,e, $|X(-f)| = |X(f)|$, and $\angle X(-f) = -\angle X(f)$.

We will not waste any precious space in describing the many features of the FT as a tool for signal design and analysis in communications engineering. We just want here to recall some elementary results about FT theory that will be used in many places in the Chapters to follow. For instance, it is an easy exercise for the reader to show that the FT of the time-shifted version $x(t - t_0)$ of the signal $x(t)$ is

$$\mathcal{F}[x(t - t_o)] = X(f) \exp(-\jmath 2\pi f t_0) \tag{2.10}$$

so that the amplitude spectrum of the signal is left unchanged, and the phase spectrum is modified by a term proportional to the frequency of each component. Similarly, it is easy to show what happens if we perform an operation of radio-frequency *modulation* on the signal $x(t)$ as follows:

$$x_{RF}(t) = x(t)\cos(2\pi f_0 t) = \frac{x(t)\exp(\jmath 2\pi f_0 t) + x(t)\exp(-\jmath 2\pi f_0 t)}{2} \tag{2.11}$$

where f_0 is the *carrier frequency*. The corresponding modification of the FT is

$$X_{RF}(f) = \frac{X(f + f_0) + X(f - f_0)}{2} \tag{2.12}$$

that is, a frequency-shift of each of the frequency components the modulating signal is made of.

Example 2.2

Let us find the FT of the rect function $x(t)$ in (2.7):

$$X(f) = \int_{-\infty}^{+\infty} \mathrm{rect}(t)\exp(-\jmath 2\pi f t)dt = \int_{-T/2}^{+T/2} \exp(-\jmath 2\pi f t)dt$$

$$= \frac{\sin(\pi f T)}{\pi f} = T\frac{\sin(\pi f T)}{\pi f T} = T\mathrm{sinc}(fT) \tag{2.13}$$

We have introduced here a new identifier for a special waveform that we will extensively use in the following: the so-called sinc function (represented in Fig.2.6) that we define as $\mathrm{sinc}(\alpha) = \sin(\pi\alpha)/(\pi\alpha)$.

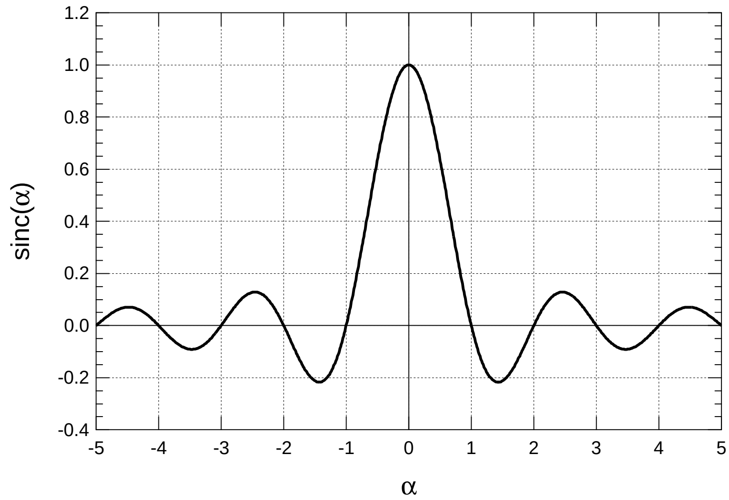


Fig. 2.6 sinc function

2.1.3 Bandlimited Signals

We will say that a signal is *bandlimited* when it has an amplitude spectrum $|X(f)|$ that is confined into a limited frequency band $[-B, B]$. Such limitation may hold exactly (*strictly* bandlimited signal) or to a good approximation, in the sense that out of the interval $[-B, B]$ the signal components, though not exactly null, are so small as to be considered negligible. An example of a (strictly) bandlimited signal is the popular *Frequency Raised Cosine* (FRC)pulse (or *Nyquist's* pulse) given by

$$g_N(t) = \text{sinc}(t/T)\frac{\cos(\beta\pi t/T)}{1-(2\beta t/T)^2}$$

$$G_N(f) = \begin{cases} T & |f| < (1-\beta)/2T \\ \frac{T}{2}\left\{1+\cos\left[\frac{\pi T}{\beta}\left(|f|-\frac{1-\beta}{2T}\right)\right]\right\} & (1-\beta)/2T \le |f| \le (1+\beta)/2T \\ 0 & \text{elsewhere} \end{cases} \tag{2.14}$$

and whose spectrum/waveform are represented in Fig. 2.7. Here, the bandwidth is $B = (1+\beta)/2T$, and $\beta, 0 \le \beta \le 1$ is a parameter that regulates the signal bandwidth and that is called *roll-off factor*. The value $1/2T$ is the so-called *Nyquist frequency*, corresponding to the minimum pulse bandwidth when $\beta = 0$.

2.1.4 Dirac's delta function

A peculiar signal that we will often use in the following chapters is Dirac's *delta function* $\delta(t)$. Its name is a little bit defying, since $\delta(t)$ is not actually a signal in the classical sense. We may speak of a *generalized signal* whose definition and existence

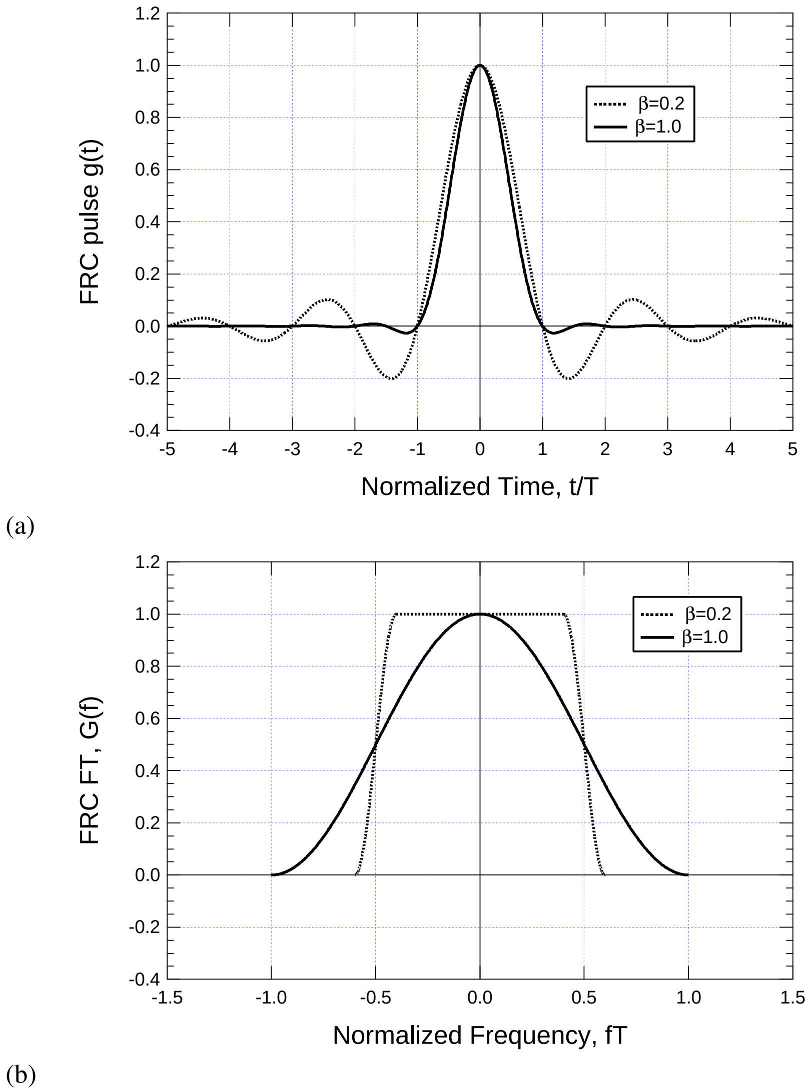


Fig. 2.7 Waveform (a) and Fourier Transform (b) of the FRC pulse

is only justified through an integral property. Dirac's delta is in fact defined through the so-called *sampling* or *sifting* property:

$$\delta(t) \quad : \quad \int_{-\infty}^{-\infty} x(t)\delta(t)dt = x(0) \tag{2.15}$$

where $x(t)$ is any ordinary signal with no discontinuity at $t = 0$. From (2.15) we immediately have as a particular case

$$\int_{-\infty}^{-\infty} \delta(t)dt = 1 \tag{2.16}$$

so we may say that the Dirac's function has "unit area". It is easily argued that no ordinary function with property (2.15) exists, but this new mathematical entity proves useful in system theory and linear filtering, as we'll see in a while.

The standard heuristic representation of the delta function (also called *unit impulse*), whose rigorous treatment is found within the so-called *distributions theory*, can be obtained with the aid of a sequence of functions. Assume we have a rect function with duration $T = 2\varepsilon$ and amplitude $A = 1/2\varepsilon$ as represented in Fig. 2.8(a). The "area" of this signal is 1, irrespective of ε. Assume now that this pulse is made shorter and shorter (and consequently, taller and taller) keeping its unit area but becoming thinner and thinner, as suggested in Fig. 2.8(a). The limit of this pulse is a heuristic representation of $\delta(t)$: something whose time width is null, but whose amplitude is infinite, so that its area is unitary. This is what is symbolically depicted in Fig. 2.8(b) as the standard representation of a delta "function". Of course, such a signal does not exist in the ordinary sense.

The definition of $\delta(t)$ that follows our heuristic representation is

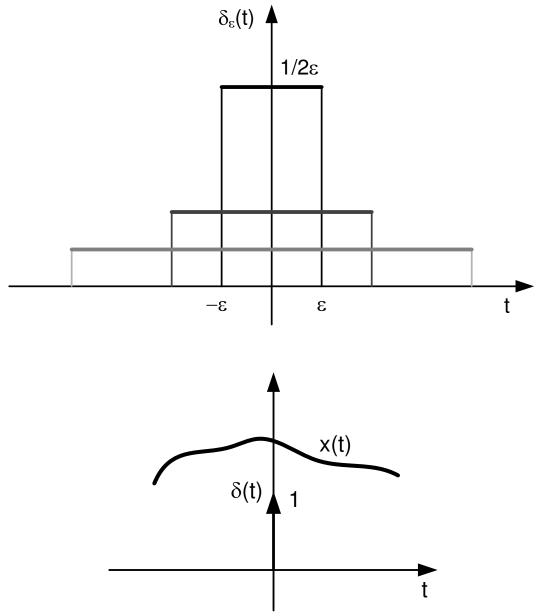
$$\delta(t) \stackrel{\triangle}{=} \lim_{\varepsilon \to 0} \frac{1}{2\varepsilon} \mathrm{rect}\left(\frac{t}{2\varepsilon}\right) \tag{2.17}$$

This relation not only gives an idea about how the delta function "looks like", but can also be used in the practice, provided that i) $\delta(t)$ appears under an integral operator (as in its definition (2.15)), and ii) the limit in (2.17) is moved *outside* the integral operator, i.e., it is computed *subsequently* to the computation of the integral. The reader may verify (2.16) using this new definition. It is also easy to show that the definite integral of $\delta(t)$ on finite intervals of the kind $\int_b^a \delta(t)dt$ gives a value equal to 1 when the instant $t = 0$ lies within (a, b), otherwise it gives 0.

Dirac's delta is also peculiar as far as its FT $\Delta(f)$ is concerned. First, the problem of finding the FT of $\delta(t)$ is well-posed since the FT (2.8) is an integral operator. Second, its computation is trivial, according to (2.15):

$$\Delta(f) = \int_{t=-\infty}^{\infty} \delta(t) \exp(-\jmath 2\pi f t)dt = \exp(-\jmath 2\pi f t)|_{t=0} = 1 \tag{2.18}$$

The FT of the unit impulse is thus *constant* on all frequencies, with no bandlimitation whatever.



(a)

(b)

Fig. 2.8 Definition of Dirac's function (a), and its symbolic representation (b)

2.2 LINEAR FILTERING

In the following Chapters, we will familiarize with a number of signal processing functions that are implemented in a digital data receiver for wireless communications.

2.2.1 Systems and Signals

The simplest and most fundamental of such operations is perhaps *filtering*. In its simplest realization, filtering means designing a device, an electronic circuit, a piece of software or, in a word, a *system* that changes an *input* signal $x(t)$ into an *output* signal $y(t)$ according to some processing criteria. Our notation to indicate this will be

$$y(t) = \mathcal{T}\left[x(\alpha); t\right] \tag{2.19}$$

where $\mathcal{T}$ is an operator representing the signal processing function performed by the system, and where we indicate that the processing depends in general on the *whole* input waveform $x(\alpha)$ and on time as well. The simplest family of system are the *linear filters* that obey the *superposition rule*. Assume that we know that

$$y_1(t) = \mathcal{T}\left[x_1(\alpha); t\right] \quad , \quad y_2(t) = \mathcal{T}\left[x_2(\alpha); t\right] \tag{2.20}$$

and that we build up a signal $x(t)$ as a weighted superposition (i.e., a linear combination) of x_1 and x_2 as $x(t) = \alpha_1 x_1(t) + \alpha_2 x_2(t)$. The system is a *linear* filter iff

$$y(t) = \mathcal{T}\left[x(\alpha); t\right] = \mathcal{T}\left[\alpha_1 x_1(\alpha) + \alpha_2 x_2(\alpha); t\right] = \alpha_1 y_1(t) + \alpha_2 y_2(t) \qquad (2.21)$$

This means that the output $y(t)$ can be obtained as a linear combination with the same coefficients α_1 and α_2 of the "single" outputs of the system to the single inputs x_1 and x_2.

In addition to the property of linearity many filters used in the practice are also *time-invariant*. This means that their behavior does not change with time. Specifically, if we know that $y(t) = \mathcal{T}\left[x(\alpha); t\right]$, and we later submit to the system a time-shifted version of the same signal, namely, $x_{TS}(t) \stackrel{\triangle}{=} x(t - t_0)$, we expect that the filter output $y_{TS}(t)$ be just the same waveform that we had earlier, modified only by the same time shift we introduced on the input:

$$y_{TS}(t) = \mathcal{T}\left[x_{TS}(\alpha); t\right] = \mathcal{T}\left[x(\alpha - t_0); t\right] = y(t - t_0) \qquad (2.22)$$

Linear, Time-Invariant (LTI) systems are particularly simple to design and analyze, nonetheless they prove useful in many instances of signal processing, and especially for the detection of known signals embedded into noise.

2.2.2 Time- and Frequency-Characterization of LTI Systems

The properties of any LTI filter are completely characterized by the knowledge of a particular signal that is associated to the system, namely, its *impulse response*. The impulse response of an LTI system is just the system output in response to a $\delta(t)$ input:

$$h(t) \stackrel{\triangle}{=} \mathcal{T}\left[\delta(\alpha); t\right] \qquad (2.23)$$

In particular, it can be easily shown that the response (output) of an LTI to a generic input $x(t)$ can be found as

$$y(t) = \int_{-\infty}^{\infty} x(\alpha) h(t - \alpha) d\alpha = x(t) \otimes h(t) \qquad (2.24)$$

This kind of "mixing" of the two signals x and h to give y deserves a specific name and notation: it is called the *aperiodic convolution* between the two signals, and is denoted by the symbol $\otimes$ as in (2.24). The convolution is a symmetric, associative, and distributive operator, as can be easily proved. The transformation carried out by an LTI system on its input signal $x(t)$ to give the output signal $y(t)$ is often symbolized in a graphical form as in Fig. 2.9 where the impulse response of the system is explicitly indicated.

Fourier analysis of a signal reveals a fundamental tool also in the characterization of the properties of an LTI system. Everything revolves around a cardinal property

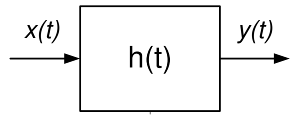


Fig. 2.9 Graphical representation of the transformation effected by an LTI system

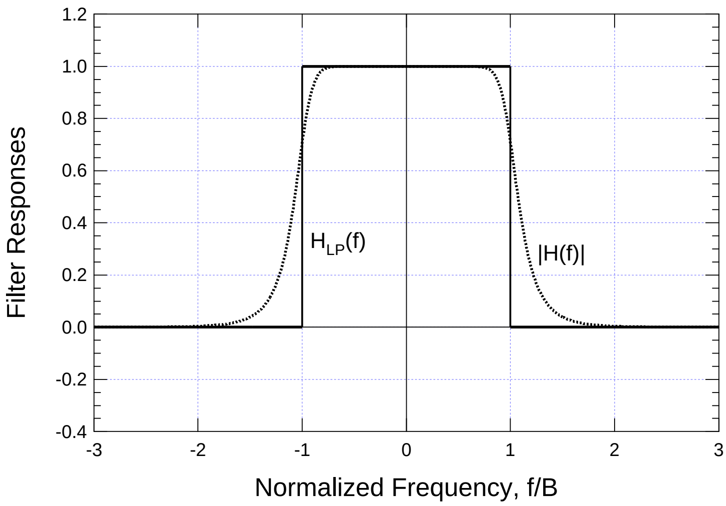


Fig. 2.10 Amplitude response of lowpass filters

of convolution. Starting back from the constituent relation (2.24) of an LTI system, and taking the FT of both sides of the equation, we find:

$$Y(f) = X(f) \cdot H(f) \tag{2.25}$$

where $H(f)$ is the FT of the impulse response $h(t)$. It is seen that a (complicated) operation of convolution in the time domain has changed into a (simple) operation of product between two transforms in the frequency domain. This suggest that often the analysis and design of an LTI system is much simpler in the frequency domain than in the time domain due to the simpler operation that the system implicitly carries out on the FT of the input signal. The quantity $H(f)$ is called the *frequency response* of the filter, and is an alternative means to provide a full characterization of the behavior of the system. In particular, it can be shown that the response of the system to a purely sinusoidal input $x(t) = \cos(2\pi f_0 t)$ is just another sinusoidal signal at the same frequency in the form

$$x(t) = \cos(2\pi f_0 t) \quad \Rightarrow \quad y(t) = |H(f_0)| \cdot \cos(2\pi f_0 t + \angle H(f_0)) \tag{2.26}$$

The effect of the filter on the sinusoidal signal is an amplitude change by a factor equal to $|H(f_0)|$ (the *amplitude response* of the system at the frequency f_0), and a

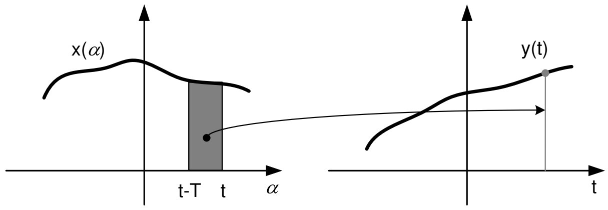


Fig. 2.11 Sliding-window Integrator

phase shift by $\angle H(f_0)$ (the *phase response*). The complex-valued version of (2.73) is

$$x(t) = \exp(\jmath 2\pi f_0 t) \quad \Rightarrow \quad y(t) = H(f_0) \cdot \exp(\jmath 2\pi f_0 t) \tag{2.27}$$

If we change the frequency of the sinusoidal signal, the amplitude and phase responses change, and so our system responds to different components at different frequencies in a different way. This is why $H(f_0)$ is called the frequency response, and also suggests that in general the system bears a *selective* (i.e., unequal) behavior with respect to frequency. Different components in the spectrum of a signals are treated differently. Some may pass substantially unaltered, others may be blocked altogether. An example of such behavior is given by the *low-pass* ideal filter, whose frequency response $H_{LP}(f)$ is represented in Fig. 2.10 (solid line). The ”low” frequency components are passed unaltered (they are multiplied by $H_{LP}(f) = 1$), whilst higher-frequency components, in particular those outside the passband B are blocked. The same is substantially true for the system whose amplitude response $|H(f)|$ is also represented in Fig. 2.10 (dashed line). We can not call this an ”ideal” filter, but it is nonetheless a good approximation of the ideal filter we have just mentioned, and it is easily realizable.

Example 2.3

We intend to find the impulse and the frequency response of a *sliding window* integrator. This LTI system produces an output whose value at a specific instant is the value of the integral of the input signal on a T-wide integration window immediately leading that instant, as i shown in Fig. 2.11:

$$h(t) = \frac{1}{T} \int_{t-T}^{t} x(\alpha) d\alpha \tag{2.28}$$

We leave to the reader the proof that (2.31) defines indeed an LTI. Assuming this, the impulse response is by definition the output of the system when the input is $\delta(t)$ and is given by

$$y(t) = \frac{1}{T} \int_{t-T}^{t} \delta(\alpha) d\alpha \tag{2.29}$$

As we know by the properties of Dirac's δ function, the integral above is either equal to 1 if the instant $\alpha = 0$ (where $\delta(\alpha)$ is applied) is within the integration interval, otherwise it is equal to 0. This means that the output value is going to be 1 whenever $t - T \leq 0 < t$, that is to say, $0 < t \leq T$, and is going to be

0 outside this interval. The impulse resonse we seek for is a causal rectangular pulse that we can cast into the form

$$h(t) = \frac{1}{T}\text{rect}\left(\frac{t - T/2}{T}\right) \tag{2.30}$$

The frequency response is trivially the Fourier transform of $h(t)$, namely,

$$H(f) = \text{sinc}(fT)\exp(-\jmath\pi fT) \tag{2.31}$$

2.3 FILTERING OF RANDOM SIGNALS

Random signals or, in the parlance of probability theory, *random processes* play a central role in communications theory. Not only a random process is the standard mathematical representation of all kinds of noise, interference, and disturbance of any sort that may afflict a signal to be detected. A random process is also the representation for the information-bearing signal itself: were the signal known in advance to the receiver, no need of sending it out by the transmitter would arise. The randomness of such signal is just related to the quantity of information that it contains: the more random it looks to the receiver *before* being received, the larger quantity of information it conveys *after* it is actually received.

2.3.1 Basics of random signals

A random signal (process) is a function of time whose shape is not (exactly) known in advance. The value at time t of a deterministic signal $w(t)$ like those encountered until now in the book is exactly known for every possible t, either because we have a mathematical representation of such signal, or because we have recorded it and stored it in a computer file. On the contrary, we can say that the value at a certain time t of a random signal is a *random variable*. The random process is thus the collection of all random variables at all times t, that we denote with the same notation as an ordinary deterministic signal, $w(t)$.

So we do not know in advance the value of the signal for each value of t. Rather, we only know *statistical properties* of such values. When we observe a random signal, what we get is a *realization* of all such random variables time after time, and, after our observation, we are left with a deterministic signal that could not be predicted in advance. The statistical description that we need to completely characterize the properties of such signal may appear to be the probability density function (pdf) $f_w(a;t)$ of the random variable $w(t)$ at time t, that allows to compute probabilities of any kind on $w(t)$. That is not the case indeed. For such signal it may be needed to know, just to make an example, the probability that $w(t) \leq 0$ and, together with this, that after a certain time τ, the probability that $w(t+\tau) \geq 0$. This *joint* probability cannot be computed from $f_w(a;t)$. What we need here is a *second-order* joint pdf $f_w(a_1, a_2; t, t+\tau)$. But then, why stopping just at the second order? The conclusion is that the complete characterization of a random process requires the knowledge of the *class* of joint pdf's of order K of the form

$$f_w(a_1, a_2, \ldots, a_K; t, t+\tau_1, \ldots, t+\tau_{K-1}) \tag{2.32}$$

for *each* (arbitrary large) K. This piece of knowledge is clearly very difficult to get, apart from simple cases. One such exception is that of *Gaussian processes*, for which the pdf's (2.32) are (jointly) Gaussian for any K.

2.3.2 Expectation, Autocorrelation function, and Power Spectral Density

In the practice, a very partial knowledge of the statistical properties of a random process may be sufficient to solve many problems in communications engineering. The simplest statistical property of a random process is its *expectation* function or simply *mean value*:

$$\eta_w(t) \triangleq \int_{-\infty}^{-\infty} a f_w(a;t) da \tag{2.33}$$

This function represents, time instant by time instant, the most likely value that the process is going to assume before it is actually observed. In particular, the relation (2.33) is often summarized into the following simplified notation:

$$\eta_w(t) = E\{w(t)\} \tag{2.34}$$

where we introduced the linear operator $E\{\cdot\}$ called *Expectation* whose definition is self-evident. In many examples of random signals that are dealt with in communications systems (be them information-bearing or just noise), we will see that $\eta_w(t)$ is a constant, and very very often $\eta_w(t) \equiv \eta = 0$.

Example 2.4

We are given the random process

$$w(t) = A\cos(2\pi f_0 t + \Theta) \tag{2.35}$$

where A and f_0 are known, while Θ is a uniform random variable in the interval $[-\pi/2, \pi/2)$. Let us find first the average value $\eta_w(t)$. According to the expectation theorem, instead of using $f_w(a;t)$ to perform the expectation we need, we can use instead the statistics of the parameter Θ the process depends on:

$$\eta_w(t) = E\{w(t)\} = E\{A\cos(2\pi f_0 t + \Theta)\}$$

$$= \int_{-\infty}^{-\infty} A\cos(2\pi f_0 t + \theta) f_\Theta(\theta) d\theta = A \int_{-\pi/2}^{\pi/2} \cos(2\pi f_0 t + \theta) \frac{1}{\pi} d\theta$$

$$= -\frac{2A}{\pi}\sin(2\pi f_0 t) \tag{2.36}$$

The average value $\eta_w(t)$ of $w(t)$ is a *first-order* statistical quantity, or first-order statistics. It is computed with a first-order pdf, and gives information of the statistical behavior of our random signal when observed at a *single* time instant. Another example of first-order statistics if the average instantaneous power of the process

$$P_w(t) \stackrel{\triangle}{=} E\{w^2(t)\} = \int_{-\infty}^{-\infty} a^2 f_w(a;t)da \tag{2.37}$$

The main example of a *second* order statistics is the *autocorrelation function*

$$\begin{aligned} R_w(t,\tau) &\stackrel{\triangle}{=} E\{w(t)w(t-\tau)\} \\ &= \int_{a_1=\infty}^{-\infty}\int_{a_2=\infty}^{-\infty} a_1 \cdot a_2 f_w(a_1,a_2;t,t-\tau)da_1 da_2 \end{aligned} \tag{2.38}$$

It is computed as the *statistical correlation* between the two random variables $w(t)$ and $w(t-\tau)$ that the random process takes at the two instants t and $t-\tau$, respectively. The autocorrelation function plays a fundamental role in the spectral analysis of a random signal, just as it does for deterministic signals. For deterministic signals, we know that it is a function of the delay τ only, whereas, according to (2.38), $R(t,\tau)$ is a function of both t and τ. We introduce therefore the notion of a wide-sense *stationary* (WSS) process. Assume that

$$R_w(t,\tau) = R_w(\tau) \ \ , \ \eta_w(t) \equiv \eta_w \tag{2.39}$$

We may say that the properties of the random process are always the same, irrespective of the time instant that we choose as our time origin: the average value does not depend on time, and the autocorrelation function depend only on the *time shift* between the two time instant $t_1 = t$ and $t_2 = t - \tau$ that we consider on our signal. The instantaneous power of a WSS process is constant in time and has the following relation with the autocorrelation function:

$$P_w(t) \equiv P_w = R_w(0) \tag{2.40}$$

For WSS processes, we may define a *power spectral density* (psd) function as the FT of the autocorrelation function:

$$S_w(f) \stackrel{\triangle}{=} \int_{-\infty}^{+\infty} R_w(\tau)\exp(-\jmath 2\pi f\tau)d\tau = 2\int_0^{+\infty} R_w(\tau)\cos(2\pi f\tau)d\tau \tag{2.41}$$

where the second equality comes from the fact (easy to show) that $R_w(\tau) = R_w(-\tau)$. The psd function indicates how the power associated to a certain signal is distributed on the different component of the frequency spectrum, and the total signal power can be found as

$$P_w = \int_{-\infty}^{+\infty} S_w(f)df \tag{2.42}$$

As with any random variable, the power of a WSS process can be evaluated as $P_w = \sigma_w^2 + \eta_w^2$.

Example 2.5

Take back into consideration the process we defined in Example 4, but assume that Θ is uniform in $[-\pi, \pi)$. The average value is now easily found to be 0, irrespective of time. We may wonder wether the process is WSS. To possibly verify this, we compute the autocorrelation function $R_w(t,\tau) = E\{w(t)w(t-\tau)\}$ by virtue of the expectation theorem:

$$R_w(t,\tau) = E\{\cos(2\pi f_0 t + \Theta)\cos(2\pi f_0(t-\tau) + \Theta)\}$$

$$\int_{-\infty}^{-\infty} A\cos(2\pi f_0 t + \theta) A\cos(2\pi f_0(t-\tau) + \theta) f_\Theta(\theta) d\theta$$

$$= \frac{A^2}{2\pi}\int_{-\pi}^{\pi} \cos(2\pi f_0 t + \theta)\cos(2\pi f_0(t-\tau) + \theta) d\theta \qquad (2.43)$$

Using trigonometric formulas, it is easy to show that

$$R_w(t,\tau) = \frac{A^2}{2}\cos(2\pi f_0 \tau) = R_w(\tau) \qquad (2.44)$$

that does *not* depend on t. Our suspicion about the wide-sense stationarity of the process was well founded indeed!

A special example of WSS random signal is the *white noise* process. White light is the one made of all of the colors, we've already mentioned. Color means wavelength, and wavelength means frequency. A white noise process is zero-mean, and such that its psd is *constant throughout the whole frequency range*:

$$S_w(f) = N_0/2 \ \ \forall f \qquad (2.45)$$

This is of course an idealization, since no such signal may exist: its associated power is clearly infinite (recall (2.53)). But it is a good model for a random signal that has a much wider bandwidth than that of another "reference" signal or of a system under consideration. This is often the case for the electronic noise (thermal noise, Johnson noise) of active and passive devices, for the "ambient" noise detected by an antennas and so on. In such examples, the noise is generated by the superposition of a multitude of independent elementary components. A well-known result in probability (the central limit theorem) states that the outcome of such superposition is a Gaussian process: any set of random variables $w(t_1), w(t_2),...,w(t_N)$ obtained after observation of the signal at the time instants t_1, t_2,...,t_N has a multivariate Gaussian pdf.

Filtering is a fundamental operation for random signals, just as it is for deterministic waveforms. What happens in particular when a process $w(t)$ is filtered with an LTI system to obtain (the random process) $n(t)$? It is easy to understand how to derive the mean value function. From (2.33) we see that the computation of such function is obtained through *linear* operations: the "Expectation" *operator* is linear. We can invoke thus a "commutative" property of linear operators and say that

$$\eta_n(t) = E\{n(t)\} = E\{\mathcal{T}[w(\alpha);t]\} = \mathcal{T}[E\{w(\alpha)\};t]$$
$$= \mathcal{T}[\eta_w(\alpha);t] = \eta_w(t) \otimes h(t) \tag{2.46}$$

If $w(t)$ is zero-mean, $n(t)$ will always be zero-mean as well, irrespective of the particular LTI filter we may consider.

Example 2.6

Take back into consideration the parametric process of Example 4

$$w(t) = A\cos(2\pi f_0 t + \Theta) \tag{2.47}$$

where Θ is a uniform random variable in the interval $[-\pi/2, \pi/2)$. Its average value was found to be

$$\eta_w(t) = -\frac{2A}{\pi}\sin(2\pi f_0 t) \tag{2.48}$$

What if $w(t)$ is filtered by the sliding-window integrator of Example 3? Let us call $n(t)$ the result of such filtering. From (2.46) we know that $\eta_n(t) = \eta_w(t) \otimes h(t)$. But $\eta_w(t)$ is *sinusoidal* with time, so that the result of filtering can be easily evaluated through the notion of frequency response of the filter, as in (2.73):

$$\eta_n(t) = -\frac{2A}{\pi}|H(f_0)|\sin(2\pi f_0 t + \angle H(f_0))$$
$$= -\frac{2A\text{sinc}(f_0 T)}{\pi}\sin(2\pi f_0 t + \pi f_0 T) \tag{2.49}$$

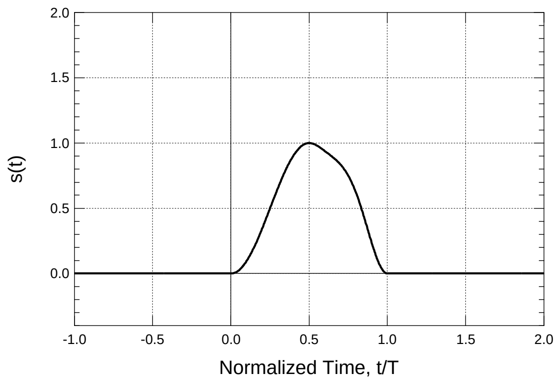
Filtering a WSS random process $w(t)$ is a special case that is easily characterized if we stick to a simple characterization of the output process $n(t)$. We already know from (2.46) how to compute the output average value. The output psd function is also easy to compute since, as happens with deterministic signals,

$$S_n(f) = |H(f)|^2 \cdot S_w(f) \tag{2.50}$$

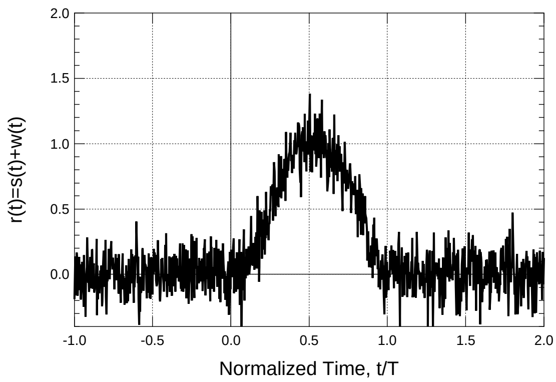
where $|H(f)|^2$ is the *power response* of the LTI filter.

Example 2.7

A certain known signal $s(t)$ is sent out on a wireless radio channel. The receiver collects such signal corrupted by Additive White Gaussian Noise (AWGN) $w(t)$ with psd $N_0/2$ (a sketch of a possible noisy signal is in Fig. 2.12 (b)). The received signal processes $r(t) = s(t) + w(t)$ with an LTI system (filter) to get a filtered signal $y(t) = x(t) + n(t)$, where x and n are the filtered signal and



(a)



(b)

Fig. 2.12 Time-limited signal $s(t)$ (a) and its noisy version $r(t)$ (b)

noise components, respectively. Assume also that $s(t)$ is time-limited within the interval $[0, T)$ as in Fig. 2.12 (a), and that the reception filter has impulse response

$$h(t) = \frac{1}{E_s} s(T - t) \tag{2.51}$$

i.e., a reversed and time-delayed (to be causal) version of the transmitted pulse, scaled by the factor $E_s = \int s^2(t)dt$ that is, by the energy of the time-limited signal s. The output of the receive filter at time T is:

$$y(T) = x(T) + n(T) = \frac{1}{E_s} \int_{-\infty}^{+\infty} s(\alpha) s((T - (T - \alpha)))d\alpha + N = 1 + N \tag{2.52}$$

where N is a zero-mean Gaussian random variable with variance

$$\sigma_N^2 = \int_{-\infty}^{+\infty} S_n(f)df = \int_{-\infty}^{+\infty} \frac{N_0}{2} |H(f)|^2 df$$

$$= \frac{N_0}{2}\frac{1}{E_s^2}\int_{-\infty}^{+\infty} |S(f)|^2 df = \frac{1}{2E_s/N_0} \tag{2.53}$$

We can compute now the *signal to noise ratio* (SNR) as the ratio between the squared signal component (signal power) and the variance of the noise component (noise power):

$$\text{SNR} \triangleq \frac{x^2(T)}{\sigma_N^2} = \frac{1}{(2E_s/N_0)^{-1}} = \frac{2E_s}{N_0} \tag{2.54}$$

It can be shown that (2.54) is the *best* SNR that can be attained upon filtering of $r(t)$ with *any* LTI system. The shape of $h(t)$ as in 2.51 is the "best match" to the received signal, and so this special filter is called the *matched filter*

2.4 BANDPASS SIGNALS AND SYSTEMS

2.4.1 Baseband equivalent of a bandpass signal

The general form of a sinusoidal bandpass signal at frequency f_0 is

$$x_{BP}(t) = A\cos(2\pi f_0 t + \vartheta) \tag{2.55}$$

where A is the amplitude of the signal and ϑ its phase. An alternative formulation of (2.55) is

$$\begin{aligned} x_{BP}(t) &= A\cos(\vartheta)\cos(2\pi f_0 t) - A\sin(\vartheta)\sin(2\pi f_0 t) \\ &= x_I\cos(2\pi f_0 t) - x_Q\sin(2\pi f_0 t) \\ &= \Re\{(x_I + \jmath x_Q)\exp(\jmath 2\pi f_0 t)\} \\ &= \Re\{x_{BB}\exp(\jmath 2\pi f_0 t)\} \end{aligned} \tag{2.56}$$

where we have introduced other quantities than the amplitude and phase of the sinusoid, that will be used extensively in the following. First, we defined the *In-phase/Quadrature components* $x_I = A\cos(\vartheta)$ and $x_Q = A\sin(\vartheta)$ as the "projections" of the sinusoid along the two main quadrature carriers at frequency f_0, namely, $\cos(2\pi f_0 t)$ and $-\sin(2\pi f_0 t)$, respectively. Also, we introduced the complex-valued notation of the *baseband equivalent* of our sinusoidal signal $x_{BB} = x_I + \jmath x_Q$. The amplitude of x_{BB} is the amplitude of our sinusoid, and the phase of x_{BB} is its initial phase, $x_{BB} = A\exp(j\vartheta)$. We summarize all this in the easy-to-remember visual representation given in Fig. 2.13

The spectrum of a sinusoid is monochromatic, i.e., it contains only one component at the frequency f_0 (and its twin at $-f_0$ if we use complex-valued FTs). What happens if by virtue of some kind of *modulation* the amplitude and/or phase of $x_{BP}(t)$ are

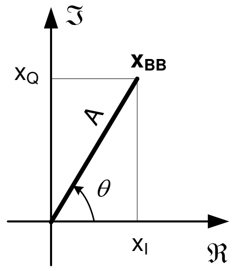


Fig. 2.13 Visual representation of I/Q components and amplitude/phase of a baseband signal

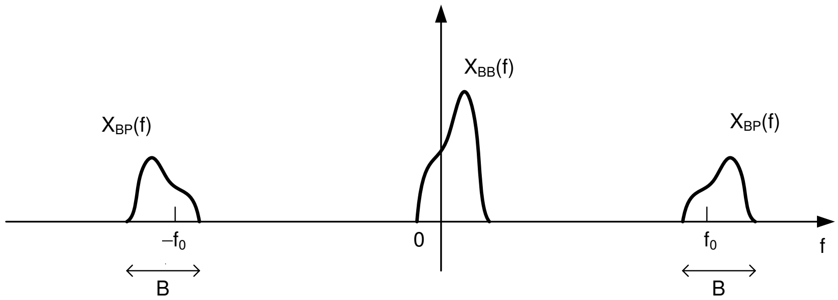


Fig. 2.14 Samples of Bandpass and Baseband-equivalent spectra

made slowly varying in time? We can write

$$\begin{aligned} x_{BP}(t) &= A(t)\cos(2\pi f_0 t + \vartheta(t)) \\ &= A(t)\cos(\vartheta(t))\cos(2\pi f_0 t) - A(t)\sin(\vartheta(t))\sin(2\pi f_0 t) \\ &= x_I(t)\cos(2\pi f_0 t) - x_Q(t)\sin(2\pi f_0 t) \\ &= \Re\{(x_I(t) + \jmath x_Q(t))\exp(\jmath 2\pi f_0 t)\} \\ &= \Re\{x_{BB}(t)\exp(\jmath 2\pi f_0 t)\} \end{aligned} \tag{2.57}$$

The quantities (I/Q components, baseband equivalent) that we mentioned above are now (slowly) varying in time, where "slowly" is to be intended "on a time scale much larger than $1/f_0$". It turns out that the spectrum of $x_{BP}(t)$ as in (2.57) is no longer monochromatic, but it is concentrated around the *carrier frequency* f_0. The *passband* of such spectrum is $B \ll f_0$, and so the signal is *quasi-monochromatic* or *bandpass*. On the contrary, $x_{BB}(t)$ has a spectrum that is confined to baseband, with a bandwdith B much smaller than f_0. Since the signal is complex valued, the spectrum of $x_{BB}(t)$ will not bear any Hermitian symmetry around 0. Figure 2.14 shows fictional examples of spectra of a bandpass, modulated, quasi-monochromatic signal, as well as its (non-Hermitian-symmetric) baseband equivalent.

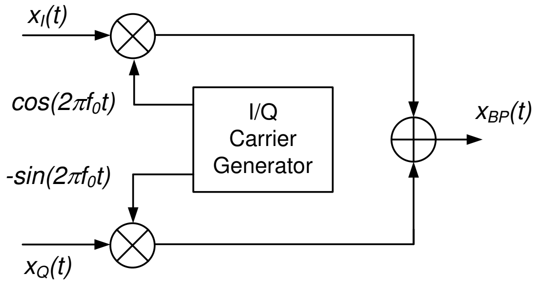


Fig. 2.15 General architecture of an I-Q modulator

2.4.2 The I-Q modulator

So the bandpass signal, once the carrier frequency is known, is completely specified by its baseband equivalent (also called *complex envelope*) $x_{BB}(t)$. Equation (2.57) tells us that $x_{BP}(t) = x_I(t)\cos(2\pi f_0 t) - x_Q(t)\sin(2\pi f_0 t)$. This is not only a mathematical representation, but it also corresponds to the architecture of the so-called *I-Q modulator* that is used in the practice to generate an arbitrary bandpass modulated signal from its I-Q components as in Fig. 2.15. In the Chapters to follow, we will systematically adopt the complex-valued notation of baseband equivalent signals. To make notation shorter, we will drop the subscripts $_BP$ or $_BB$ to denote bandpass or baseband signals, respectively, but we will implicitly assume, unless otherwise stated, that all signals are complex envelopes.

2.4.3 The I-Q demodulator

The usual way of modulating a bandpass signal with a digital data stream is to encode the digital information into either $x_I(t)$, or $x_Q(t)$, or both. Once we have done so, and we send the bandpass modulated signal on a physical medium (radio, copper, fiber), the receiver needs to reconstruct either $x_I(t)$, or $x_Q(t)$, or both, to recover (*demodulate*) the digital data. The simplest way to do this starts from the general expression of the bandpass signal:

$$\begin{aligned} x_{BP}(t) &= \Re\{x_{BB}(t)\exp(\jmath 2\pi f_0 t)\} \\ &= \frac{x_{BB}(t)\exp(\jmath 2\pi f_0 t) + x_{BB}^*(t)\exp(-\jmath 2\pi f_0 t)}{2} \end{aligned} \tag{2.58}$$

From this we have,

$$x_{BP}(t) \cdot 2\exp(-\jmath 2\pi f_0 t) = x_{BB}(t) + x_{BB}^*(t)\exp(-\jmath 2\pi \cdot 2 f_0 t) \tag{2.59}$$

and we see that such (complex-valued) signal contains two components: the first one is just the one we intend to get, and the second is something unwanted and centered at the frequency $-2f_0$. What we have to do to get rid of the latter and keep the former is processing this signal with a *lowpass* filter whose bandwidth is just that of $x_{BB}(t)$

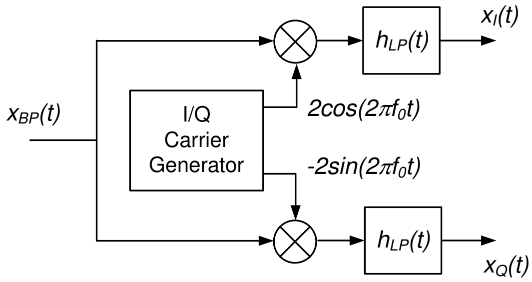


Fig. 2.16 General architecture of an I-Q demodulator

to remove the double-frequency components at $2f_0$. The result of this reasoning is simple:

$$x_{BB}(t) = x_I(t) + \jmath x_Q(t) = \{x_{BP}(t) \cdot 2\exp(-\jmath 2\pi f_0 t)\} \otimes h_{LP}(t) \tag{2.60}$$

where $h_{LP}(t)$ is the impulse response of the lowpass filter. Again, (2.60) is not just mathematics, but it is the outline of the so-called *I-Q demodulator* represented in Fig. 2.16 that is implemented in the vast majority of modern radio receivers. It is seen that the product of the received bandpass signal $x_{BP}(t)$ with the complex oscillation $2\exp(-\jmath 2\pi f_0 t)$ is implemented as a pair of real products between the former and the real-imaginary components of the latter, and the lowpass filter is applied to both components as well.

2.5 FOURIER ANALYSIS OF DIGITAL SIGNALS

2.5.1 Analog and Digital signals

In the previous section we have reviewed the main concepts and results in Fourier analysis of time-continuous (*analog*) signals. The same results can be extended to *digital* signals, or, properly speaking, time-discrete ones. The most immediate way of generating a digital signal is using an analog-to-digital converter (ADC), or, in the parlance of signal analysis, performing the operation of *sampling*. Sampling an analog signal $x(t)$ with a certain sampling rate f_s samples/s (or simply Hz) or equivalently a sampling period $T_s = 1/f_s$ means extracting from $x(t)$ a sequence of *samples* $x[n]$ such as

$$x[n] = x(nT_s) \tag{2.61}$$

The square brackets indicates the the time index n they enclose is discrete, as opposed to continuous time t that is usually enclosed into round brackets. The value at time n of $x[n]$ is real-valued, and so its representation theoretically requires an infinite number of digits. In the practice, the ADC represents each value as an integer on a fixed (finite) number of binary digits. This introduces a (small) representation error: what we get out of the ADC is actually $x_q[n] = x[n] + q[n]$, where x_q is the *quantized* version of $x[n]$, and $q[n]$ is the *quantization noise*. When the number of bits in the digital representation of $x_q[n]$ is large (say, larger than 16), the quantization noise can be safely disregarded.

2.5.2 FT of Digital signals

Sampling and the ADC are the foundation of Digital Signal Processing (DSP). DSP techniques are again heavily based on Fourier analysis, so that we have to review the basics of Fourier transforms for time-discrete signals.

Generalizing the notions already introduced for analog signals, it can be easily shown that a non-periodic sequence $x[n]$ can be Fourier-decomposed as

$$x[n] = \frac{1}{f_s} \int_{-f_s/2}^{+f_s/2} \overline{X}(f) \exp(\jmath 2\pi n f/f_s) df \tag{2.62}$$

where $\overline{X}(f)$ is the FT of the sequence $x[n]$. Equation (2.62) is apparently a *synthesis* equation much similar to (2.8) for analog signals. The corresponding *analysis* equation that gives the FT $\overline{X}(f)$ is

$$\overline{X}(f) = \sum_{n=-\infty}^{+\infty} x[n] \exp(-\jmath 2\pi n f/f_s) \tag{2.63}$$

A fundamental difference exists between the FT of analog and of digital signals. It is seen from (2.62) that the digital signal can be synthesized from a *finite* interval of continuous frequency components, whilst the analog signal requires component at *all* frequencies on the real axis. The reason for this is that the FT $\overline{X}(f)$ of a sequence is a *periodic* function of frequency, with a period equal to the sampling frequency f_s. Thus, the only independent components of $x[n]$ actually lie on an f_s-wide frequency interval, and no more components than those are required in the synthesis. This has also something to do with the property of sampled sinusoids. The sequence extracted by sampling a sinusoidal signal $x_1(t)$ at frequency f_1 is

$$x_1[n] = x_1(n/f_s) = \cos(2\pi n f_1/f_s) \tag{2.64}$$

The ratio f_1/f_s is sometimes indicated with F_1 and is called the *normalized* frequency. Assume now we have a second sinusoidal signal $x_2(t)$ at the frequency $f_1 + f_s$. $x_2(t)$ is clearly (much) different from $x_1(t)$. After sampling $x_2(t)$ we get

$$x_2[n] = \cos(2\pi n f_2/f_s) = \cos(2\pi n f_1/f_s + 2\pi n) = \cos(2\pi n f_1/f_s) = x_1[n] \tag{2.65}$$

After sampling, the two previously different sinusoidal signals look exactly the same! This means that in the digital domain, there can be no more independent sinusoidal components to synthesize a signal than those into a f_s-wide "base" interval: a component outside that interval is actually the "image" of another one that lies into the base interval at a distance equal to an integer multiple of f_s.

Example 2.8

Assume that we sample a time-continuous exponential signal $x(t) = \exp(-t/\alpha)\mathrm{u}(t)$ with a sampling interval $T - s$. What we get is

$$x[n] = x(nT_s) = \exp(-nT/\alpha) = a^n \mathrm{u}[n] \tag{2.66}$$

where $a \triangleq \exp(-T/\alpha) < 1$ is a real constant depending on the time constant of the signal and on the sampling frequency. The FT of the resulting sequence is

$$\overline{X}(f) = \sum x[n] \exp(-\jmath 2\pi n f T_s) = \sum_{n=0}^{\infty} a^n \exp(-\jmath 2\pi n f T_s)$$

$$= \sum_{n=0}^{\infty} [a \exp(-\jmath 2\pi f T_s)]^n = \frac{1}{1 - a\exp(-\jmath 2\pi f T_s)} \tag{2.67}$$

where certainty about the convergence of the series comes form the condition $|a \exp(-\jmath 2\pi f T_s)| = |a| < 1$. The amplitude and phase spectra of $x[n]$ as resulting from (2.67) are shown in Fig. 2.17. Note that they are shown across a frequency span equal to $\pm f_s/2$ since the two functions are *periodic* with period f_s.

Many of the properties that we already mentioned for the FTs of analog signals hold true for FT of sequences as well, whit minor modifications. We refer in particular to Hermitian symmetry, delay and modulation theorems, and so on.

Example 2.9

In the domain of analog signals, special attention was devoted to the definition and properties of Dirac's delta function (Sect. 2.1.4). Is there something similar in the digital domain? The answer is simpler than expected. While $\delta(t)$ was a very special signal, its time-discrete counterpart is just the ordinary sequence

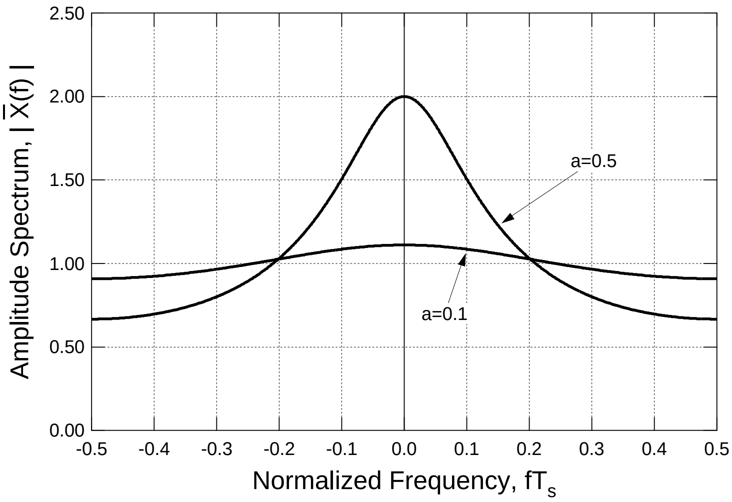
$$\delta[n] \triangleq \begin{cases} 1 & n = 0 \\ 0 & \text{elsewhere} \end{cases} \tag{2.68}$$

Its FT is clearly

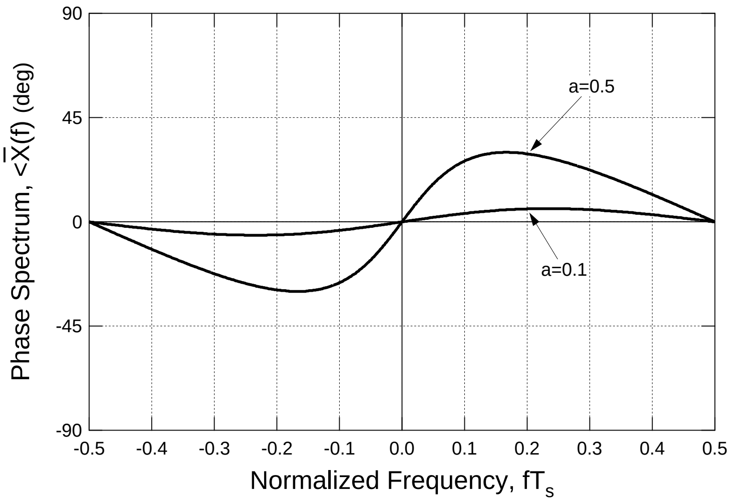
$$\overline{\Delta}(f) = \sum_{n=-\infty}^{+\infty} \delta[n] \exp(-\jmath 2\pi n f/f_s) = 1 \tag{2.69}$$

just like the FT of $\delta(t)$.

If our sequence $x[n]$ comes from sampling of an analog signal $x(t)$ with FT $X(f)$, a



(a)



(b)

Fig. 2.17 Amplitude (a) and phase (b) spectrum of the exponential sequence fundamental question is: what is the relation between the FT $X(f)$ of the signal we start from, and the FT of the resulting sequence $x[n]$? The answer is called *Poisson's relation* and reads

$$\overline{X}(f) = f_s \sum_{k=-\infty}^{+\infty} X(f - kf_s) \tag{2.70}$$

This equation tells us that the FT of $x[n]$ is the superposition of an infinite series of repetitions of the original FT of the analog signal, with a repetition period that is equal to the sampling frequency f_s. This of course gives a periodic FT $\overline{X}(f)$ as that of any digital signal. Notice that there might a sort of "interference" between adjacent repetitions of the original spectrum that is called *aliasing*. An example of aliasing due to sampling is shown in Fig. 2.18 (b) that show the FT $\overline{X}(f)$ of the sequence $x[n]$ obtained after sampling thh analog signal $x(t)$ whose FT $X(f)$ is hown in Fig. 2.18 (a). Such superposition of adjacent spectra does not occur only if the original signal $x(t)$ is bandlimited into the band B, and the sampling frequency is larger than $2B$. The condition $f_s > 2B$ is called the *Nyquist's* condition. When it is verified, there's no aliasing in the sampled signal spectrum. Compare in this respect Fig. 2.18 (a) showing again $\overline{X}(f)$ resulting form the sampling of the signal in Fig. 2.18 (a), this time meeting Nyquist's condition. In particular, the main replica with $k = 0$ in Poisson's relation (2.70) that lies in the main interval $[-f_s/2, f_s/2)$ of the FT is a perfect replica (apart from an immaterial scale factor) of the original spectrum $X(f)$.

2.5.3 Filtering and Interpolation of digital signal

Needless to say, the notion of an LTI system translates nicely and neatly into the digital domain as well. Using a notation similar to the one we introduced for analog systems, a digital LTI filter is identified by an impulse response

$$h[n] \triangleq \mathcal{T}\left[\delta[n]\right] \tag{2.71}$$

and its operation is described by the time-discrete aperiodic convolution between the input signal $x[n]$ and such impulse response $h[n]$:

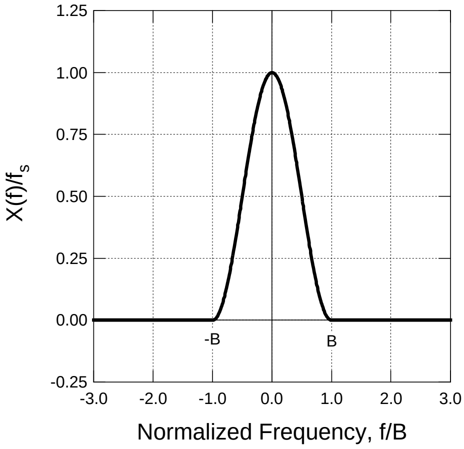
$$y[n] = x[n] \otimes h[n] \triangleq \sum_{m=-\infty}^{+\infty} x[m] \cdot h[n-m] = \sum_{m=-\infty}^{+\infty} h[m] \cdot x[n-m] \tag{2.72}$$

The system may also have a frequency response

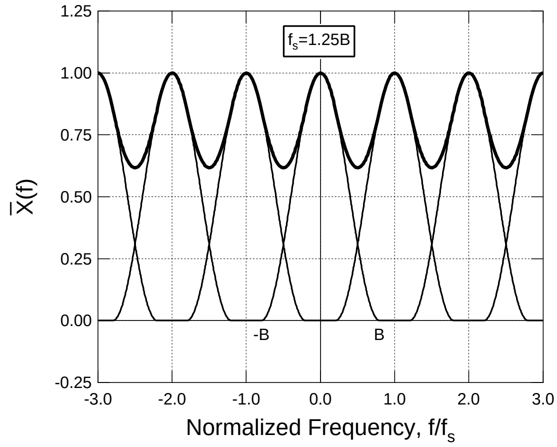
$$\overline{H}(f) \triangleq \sum_{n=-\infty}^{+\infty} h[n] \exp(-\jmath 2\pi n f/f_s) \tag{2.73}$$

so that the frequency-domain input-output relationship still is

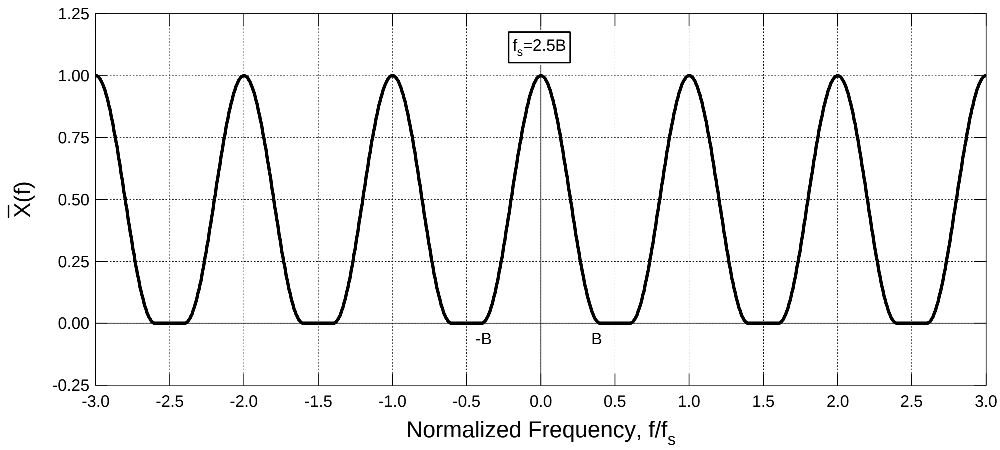
$$\overline{Y}(f) = \overline{X}(f)\overline{H}(f) \tag{2.74}$$



(a)



(b)



(c)

Fig. 2.18 Spectrum of an analog signal (a), of the sampled digital signal with aliasing (b), and of the sampled digital signal without aliasing (c)

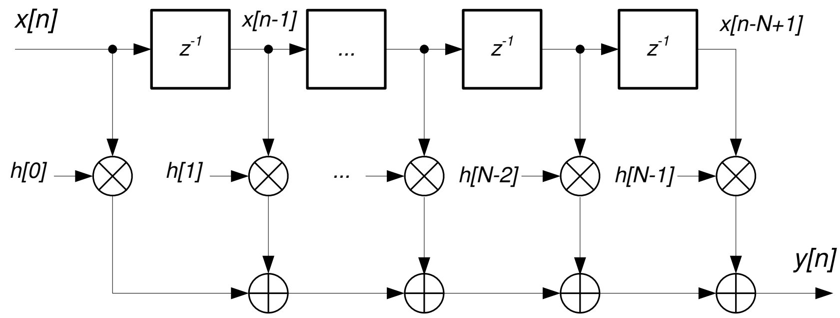


Fig. 2.19 Implementation of an FIR filter

If the impulse response $h[n]$ of the LTI system has a finite number of samples different from zero, the filter is *Finite Impulse Response* (FIR), otherwise it is IIR (*Infinite Impulse Response*). The operations to be computed to implement an FIR filter can be represented as in Fig. 2.19, where the blocks labeled z^{-1} implement the delay of their input sequence by one sample (unit-delay element), and where we have assumed that $h[n] = 0$ when $n < 0$ or $n \geq N$.

Once we have turned an analog signal into a digital sequence via sampling, and after we have possibly performed some digital processing on such (digital) signal (even simple storage on a digital medium such as a Compact Disc or a flash memory stick), it may be desired to reconstruct an analog signal form the resulting (retrieved) sequence. This reverse-sampling operation is called *interpolation* and in the practice it is implemented by a *Digital to Analog Converter* (DAC). The general form of an interpolated signal $\hat{x}(t)$ is

$$\hat{x}(t) = \sum_{n=-\infty}^{+\infty} x[n] \cdot p(t - nT_s) \tag{2.75}$$

where $x[n]$ is the sequence being interpolated, and $p(t)$ is the pulse shape that is specific of a particular interpolator. If $p(t)$ is the rectangular pulse in Fig. 2.20 (a), then we have a zero-hold interpolator that basically produces a sample-and-hold signal. If on the contrary $p(t)$ is the triangular pulse in Fig. 2.20 (b) we get a linear intepolator that joins consecutive values of the signal samples with straight line segments to produce the interpolated analog signal $\hat{x}(t)$. The frequency-domain counterpart of (2.75) is very simple:

$$\hat{X}(f) = P(f)\overline{X}(f) \tag{2.76}$$

Notice that both $\hat{X}(f)$ and $P(f)$ are FTs of analog signals, whilst $\overline{X}(f)$ is the FT of the sequence $x[n]$ to be interpolated.

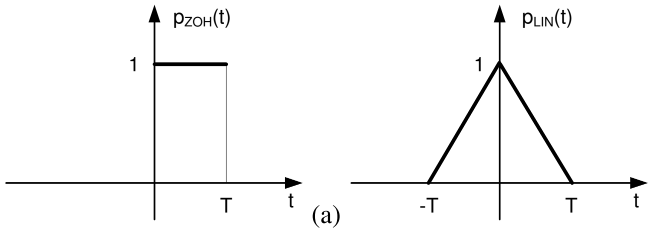


Fig. 2.20 Interpolating pulses: (a) ZOH interpolator; (b) linear interpolator

2.5.4 The sampling theorem

A rather fundamental question about signal sampling comes immediately to one's mind. Once an analog signal is sampled, and its samples are all collected and, say, stored, is it possible to fully recover such signal with no loss? At first sight the response is NO, since when converting a signal from time-continuous to time-discrete all that is in between samples appears to have been lost for ever. BUT... a glance in the frequency domain may give more hope. If the signal is bandlimited to B and we meet the Nyquist's condition $f_s \geq 2B$, we already know that we have no aliasing, and we "see" an undistorted replica of the spectrum of the analog signal in the spectrum of our sequence (the replica with $k = 0$ in the Poisson formula (2.70)). The real issue is how to recover such replica and get back to the analog domain. The answer is relatively simple: we are to use an appropriate interpolator that preserves the replica with $k = 0$ while canceling all of the others. Figure 2.21 explains that (the reference is again the analog signal spectrum shown in Fig. 2.18 (a)): we need an interpolator whose FT $P(f)$ is flat within the frequency interval $[-f_s/2, f_s/2)$ that contains the main replica with $k = 0$, and zero outside that band. Also, it has to compensate the factor f_s in Posisson's relation. In a word, we have to choose

$$P(f) = \frac{1}{f_s}\text{rect}\left(\frac{f}{f_s}\right) = T_s\text{rect}(fT_s) \tag{2.77}$$

Under Nyquist's condition and using this interpolator, it is apparent that $\hat{X}(f) = X(f)$, so that we can say that the issue of reconstructing the sampled signal is now solved. The interpolating pulse that corresponds to such $P(f)$ is trivially $p(t) = \text{sinc}(t/T_s)$, so that the relevant interpolation formula is

$$\hat{x}(t) = \sum_{n=-\infty}^{+\infty} x[n] \cdot \text{sinc}\left(\frac{t - nT_s}{T_s}\right) \tag{2.78}$$

that is called the *cardinal interpolator*. Since we known that $\hat{X}(f) = X(f)$, it is also clear that $\hat{x}(t) = x(t)$

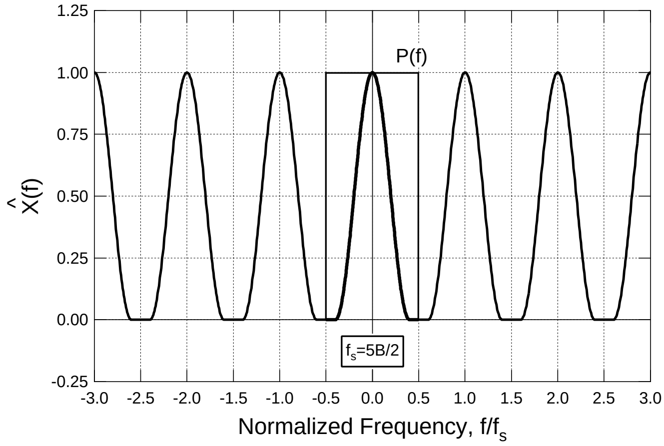


Fig. 2.21 Frequency-domain interpretation of cardinal interpolation

2.6 BASEBAND TRANSMISSION OF DIGITAL SIGNALS

2.6.1 NRZ and bandlimited PAM signals

The basic form of a baseband data signal that conveys digital information (as is encountered on wireline data communications) is that of a plain binary No Return to Zero (NRZ) digital signal:

$$x(t) = A \sum_{n=-\infty}^{+\infty} a[n] g(t - nT) \tag{2.79}$$

Here, A is the overall signal amplitude, $g(t)$ is a unit-amplitude rectangular pulse with time width T, $g(t) = \text{rect}[(t - T/2)/T)]$, and $a[n]$ is the sequence of digital data. For binary *antipodal* signals, $a[n]$ takes either the value -1 or +1, according to the particular sequence (*pattern*) of data to be transmitted. The values -1 and +1 are associated to a logical values 0 or 1, respectively, of the data bits. If the width of the data pulse $g(t)$ is smaller than T, the format of the data signal is called RZ (return to Zero) since the data pulse for each bit falls back to 0 before the next bit is sent out. It is apparent that the signal (2.79) is a repetition of pulses with different amplitude (polarity). The repetition rate $R \triangleq 1/T$ is called the *signaling rate*, *symbol rate* or *clock rate* and is measured in *symbols/s* or *Baud*. The data signal can be thought of as the result of an interpolation as in (2.75) of a digital sequence $a[n]$ coming with a sampling frequency R, with a basic interpolation pulse equal to the data pulse $g(t) = p_{ZOH(t)}$. Traditionally, the resulting signal format is called *Pulse-Amplitude Modulation* (PAM).

From the standpoint of the receiver, the specific value at time n of the n-th data symbol $a[n]$ is not known in advance (otherwise there would be no need to carry out any transmission). Therefore, the data signal is modeled as a random process and as such it has to be treated with the tools and approaches that we introduced in Sect. 2.3. The simplest assumption, that is actually well verified in the practice, is that $a[n]$ is a sequence of i.i.d. (independent identically distributed) random variables. In most cases, the values that $a[n]$ can take are also equiprobable.

Although we have implicitly assumed so in our first discussion of digital signals, it is not necessarily true that the data symbols $a[n]$ are binary. This is the reason why we call them "symbols" and not "bits". The symbols of the digital signals that are used in ISDN twisted pairs connections are *quaternary*, i.e., they can take the values $\{-3, -1, +1, +3\}$. Such values are selected according to a law that associates them to pairs of bits to be sent out, for instance $00 \rightarrow -3$, $01 \rightarrow -1$, $11 \rightarrow 1$, $10 \rightarrow 3$ (the so-called *mapping*). It is apparent that in this example a single data symbols carries the information of *two* bits at a time. Therefore, the *bit rate* R_b in the transmission (as measured in bit/s) is different from the symbols rate: $R_b = 2R$. In general, if the symbol $a[n]$ can take one of M values (that is, $a[n]$ belongs to an M-ary *alphabet*), the bit rate is $R_b = \log_2(M) \cdot R$. The simplest alphabet for M-PAM signals is

$$\mathcal{A} \equiv \{-(M-1), -(M-3), \ldots, -1, +1, \ldots, (M-3), (M-1)\} \qquad (2.80)$$

NRZ signals are the most elementary form of digital signals, but they are not used in telecommunications (they are only used for data communications in computer networks or data acquistion/storage equipment). The reason for this is related to the *bandwidth* of such signals.

2.6.2 Signals and spectra

How a data signal look like in the time domain, we already saw in the previous section. What is still to be understood is its spectral appearance and bandwidth occupancy. We find first the autocorrelation function of the baseband PAM data signal (2.79):

$$\begin{aligned} R_x(t,\tau) &= E\left\{x(t)x(t-\tau)\right\} \\ &= A^2 E\left\{\sum_{n=-\infty}^{+\infty} a[n]g(t-nT) \sum_{m=-\infty}^{+\infty} a[m]g(t-\tau-mT)\right\} \\ &= A^2 \sum_{n=-\infty}^{+\infty} \sum_{m=-\infty}^{+\infty} E\left\{a[n]a[m]\right\} g(t-nT)g(t-\tau-mT) \\ &= A^2 \cdot A_2 \sum_{n=-\infty}^{+\infty} g(t-nT)g(t-\tau-nT) \end{aligned} \qquad (2.81)$$

where A_2 is the average power of $a[n]$, $A_2 = E\left\{a^2[n]\right\}$, and where we took into account that $a[n]$ and $a[m]$ are zero mean and uncorrelated when $n \neq m$. We see that the data signal is not a WSS process, since the autocorrelation function still depends on t and not on τ only. We also see that $R_x(t,\tau)$ is a periodic function of the time

t for any time lag τ. When this happens, the process is said to be *cyclostationary*. The psd function of such process can be found by introducing the time-averaged autocorrelation function, where averaging is performed over the repetition period:

$$\rho_x(\tau) \triangleq \frac{1}{T}\int_{t=-T/2}^{T/2} R_x(t,\tau)dt \tag{2.82}$$

The psd function is now defined as the customary FT of $\rho_x(\tau)$. If we take back into consideration (2.81), we get

$$\begin{aligned} \rho_x(\tau) &\triangleq \frac{A^2}{T}\int_{t=-T/2}^{T/2} A_2 \sum_{n=-\infty}^{+\infty} g(t-nT)g(t-\tau-nT)dt \\ &= \frac{A^2 \cdot A_2}{T} \sum_{n=-\infty}^{+\infty} \int_{t=-T/2}^{T/2} g(t-nT)g(t-\tau-nT)dt \\ &= \frac{A^2 \cdot A_2}{T}\int_{-\infty}^{+\infty} g(t)g(t-\tau)dt \end{aligned} \tag{2.83}$$

We see that the (averaged) autocorrelation function of the random data signal $x(t)$ is proportional to the autocorrelation function of the deterministic signal $g(t)$. Taking the FT of $\rho_x(\tau)$ we get

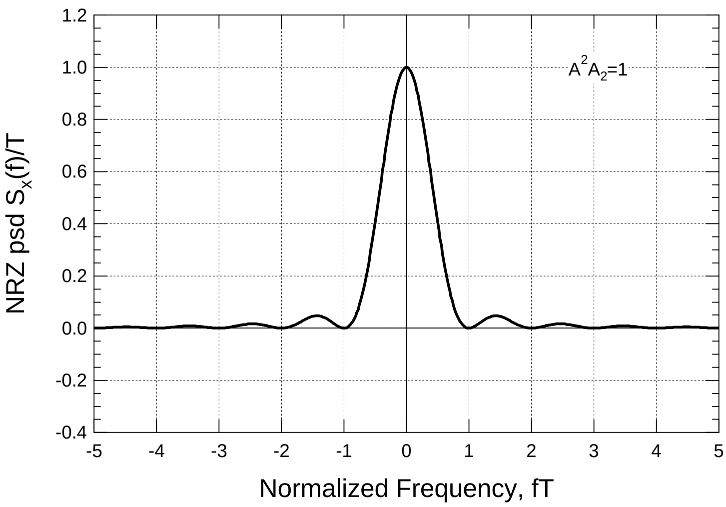
$$S_x(f) = \frac{A^2 \cdot A_2}{T}|G(f)|^2 \tag{2.84}$$

The final result of our long but instructive computation is that the psd function of a data signal is determined by the (squared) amplitude spectrum of the basic data pulse. As a consequence, the bandwidth occupancy of a digital signal is equal to the bandwidth of its elementary pulse. An NRZ signal with a full response rectangular pulse has a psd function

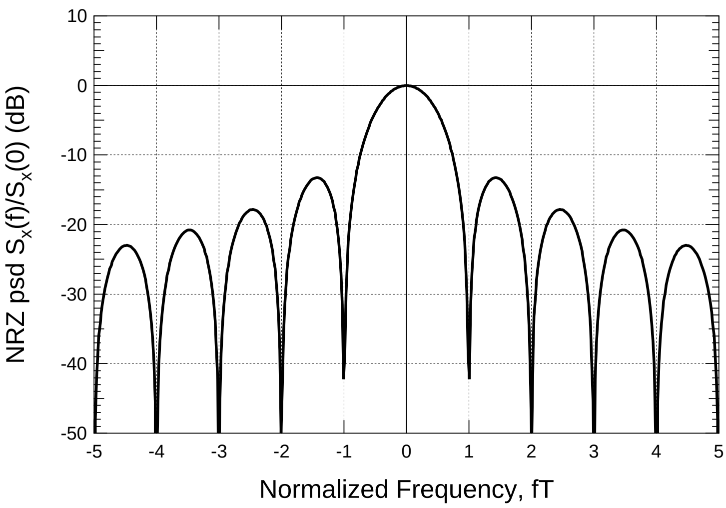
$$S_x(f) = A^2 \cdot A_2 T \text{sinc}^2(fT) \tag{2.85}$$

as represented in Fig. 2.22, whose bandwidth is infinite. A practical measure of its spectrum occupancy is the so-called *bandwidth at the first null* that is equal to $1/T$. On the other hand, we know that each physical medium (copper wire, optical fibre, radio band) has a bandwidth limitation. So the real problem in telecommunications is finding a "good", i.e., one that can support a high data rate R_b data signal when sent onto a bandlimited physical channel with bandwidth B. Filtering an NRZ signal is not efficient, since bandlimiting causes distortion and bad reception of data. The solution is resorting to an *intrinsically bandlimited* data signal. This can be done by using a bandlimited pulse $g(t)$ such as the Nyquist FRC pulse $g_N(t)$ (2.14), whose bandwidth is $B = R(1+\beta)/2$. The most popular bandlimited pulse shape is the *Square-Root Frequency Raised Cosine* (SRFRC) pulse whose FT is proportional to the square root of the FT of the Nyquist FRC pulse $g_N(t)$:

$$G(f) = T\sqrt{G_N(f)/T} = \sqrt{TG_N(f)} \tag{2.86}$$



(a)



(b)

Fig. 2.22 Power spectrum of an NRZ data signal. Linear scale (a) and log (dB) scale (b) whose waveform is

$$g(t) = \frac{\sin\left(\pi(1-\beta)\frac{t}{T}\right) + 4\beta\frac{t}{T}\cos\left(\pi(1+\beta)\frac{t}{T}\right)}{\pi\frac{t}{T}\left[1-16\beta^2\left(\frac{t}{T}\right)^2\right]} \tag{2.87}$$

and whose bandwidth is again equal to $R(1+\beta)/2$.

For any kind of basic pulse, integration of (2.84) gives the total power of the data signal as

$$P_x = \frac{A^2 \cdot A_2}{T}\int_{-\infty}^{+\infty} |G(f)|^2 df = \frac{A^2 \cdot A_2}{T}\int_{-\infty}^{+\infty} |g(t)|^2 dt = A^2 \cdot A_2 E_g/T \tag{2.88}$$

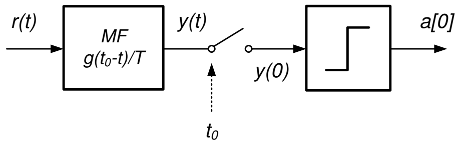
where E_g is the energy of the data pulse $g(t)$. In many cases, for instance the rectangular pulse of NRZ signaling, or the SRFRC pulse, the energy of the pulse turn out to be $E_g = T$, so that $P_x = A^2 \cdot A_2$. For a multilevel PAM signal with M-ary symbols in the alphabet $\{-(M-1), -(M-3), \ldots, -1, +1, \ldots, (M-3), (M-1)\}$ we get $A_2 = (M^2-1)/3$.

2.6.3 Detection of a PAM signal on the AWGN channel

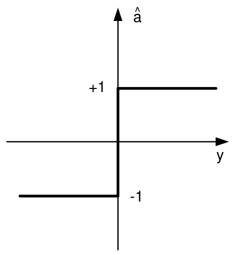
One of the fundamental problems in communications theory is the reliable detection of digital data signals in the presence of random noise. The simplest such case is detection in the AWGN channel that is typical of wireless communications over a line-of-sight link (as in satellite communications). We outlined such case in Example 7. Considering the simple case of a binary PAM signal with a T-energy pulse $g(t)$, the received signal is

$$r(t) = \sqrt{P_r}\sum_{n=-\infty}^{+\infty} a[n]g(t-nT) + w(t) \tag{2.89}$$

where $w(t)$ is AWGN with psd $N_0/2$ and P_r is the received signal power. The received energy per data symbol E_s is also $E_s = P_r T$. The detection problem is easily formulated: the receiver has to recover (or *regenerate*) the transmitted data stream upon observation of $r(t)$. Unfortunately, the presence of the random noise $w(t)$ hinders such function. When the particular *realization* of the random process is particularly unfavorable, it may happen that the regenerated datum $\hat{a}[n]$ is different from $a[n]$, and a *symbol* error is produced. "When" and "where" an error is produced is not predictable in advance, due to the random nature of noise and data. What we can evaluate upon knowledge of the statistical properties of the noise (and of the data as well) is the *probability* that a symbol error (or bit error) occurs. So the preeminent performance index of the communication link is the *Symbol Error Rate* (SER) or the *Bit Error Rate* (BER), where *rate* is the practical interpretation of *probability*. The detection problem is thus formulated as follows: *find the signal processing that we ought to apply to $r(t)$ to regenerate the data symbol stream $a[n]$ with the minimum SER.*



(a)



(b)

Fig. 2.23 Matched filter receiver for baseband PAM data signals (a) and nonlinear element characteristic (b)

This issue is not actually simple to solve, so let us tackle it step by step. Assume that we have to detect a *single* isolated datum $a[0]$ with its associated pulse $g(t)$, i.e., $r(t) = \sqrt{P_r}a[n]g(t) + w(t)$. This problem was solved back in the 50's with an elegant geometric interpretation based on expansion of $r(t)$ onto a Hilbert space of orthonormal functions. All in all, such solution is equivalent to the processing scheme shown in Fig. 2.23 (a) for PAM signals with equiprobable i.i.d. symbols, and is called the *matched filter receiver*. As was anticipated in Example 2.51, the signal is processed by an LTI filter matched to $g(t)$, i.e., with an impulse response[1] $h_{MF}(t) = g(t_0 - t)/E_g$, with an appropriate t_0. The resulting signal is

$$y(t) = \sqrt{P_r}a[0]g_r(t) + n(t) \tag{2.90}$$

where $g_r(t) = g(t) \otimes h_{MF}(t)$ is the filtered data pulse, and $n(t)$ is filtered zero-mean Gaussian noise with psd function $S_n(f) = N_0|H_{MF}(f)|^2/2$. The filtered signal is evaluated at the time instant t_0 to get the following "soft" (i.e., real-valued) sample

$$y(0) = \sqrt{P_r}a[0]g_r(t_0) + n(0) \tag{2.91}$$

For an NRZ pulse, $t_0 = T$, whereas for a symmetric SRFRC pulse, $t_0 = 0$. In both cases, $g_r(t_0) = 1$. Also, $n(0)$ is a Gaussian random variable with zero mean and variance

$$\sigma_n^2 = \frac{N_0}{2}\int_{-\infty}^{+\infty} |H_{MF}(f)|^2 df = \frac{N_0}{2E_g^2}\int_{-\infty}^{+\infty} |G(f)|^2 df = \frac{N_0}{2E_g} = \frac{N_0}{2T} \tag{2.92}$$

[1]The impulse response cited in the text is optimum for real-valued pulses $g(t)$. If the pulse is complex-valued, the impulse response of the matched filter is $h_{MF}(t) = g^*(t_0 - t)/E_g$

The soft output $y(0)$ (also called *decision variable*) is finally passed through a "hard detector" that implements the threshold nonlinearity shown in Fig. 2.23 (b) to change the continuous-amplitude sample $y(0)$ into the regenerated discrete-value symbol $\hat{a}[0]$. The probability of a symbol error (that, in our case of binary signaling, is equal to that of a bit error) is

$$SER = BER = \frac{1}{2}\Pr\{y(0) > 0 \mid a[0] = -1\} + \frac{1}{2}\Pr\{y(0) \leq 0 \mid a[0] = +1\}$$
$$= \frac{1}{2}\Pr\{-\sqrt{P_r} + n(0) \leq 0\} + \frac{1}{2}\Pr\{\sqrt{P_r} + n(0) \leq 0\}$$
$$= \Pr\{n(0) > \sqrt{P_r}\} = Q\left(\frac{\sqrt{P_r}}{\sigma_n}\right) = Q\left(\sqrt{\frac{2E_s}{N_0}}\right) \qquad (2.93)$$

where $Q(\cdot)$ is the familiar Gaussian integral function

$$Q(\alpha) = \frac{1}{\sqrt{2\pi}}\int_{-\alpha}^{+\infty} \exp(-\beta^2/2)d\beta \qquad (2.94)$$

With a computation only slightly more complex, it is easily found that the SER for generic M-PAM with alphabet (2.80) is

$$SER = 2\left(1 - \frac{1}{M}\right) Q\left(\sqrt{\frac{3}{M^2-1}}\sqrt{\frac{2E_s}{N_0}}\right) \quad \text{(M-PAM)} \qquad (2.95)$$

Instead of the ratio E_s/N_0 it is sometimes more expedient to express the SER/BER as a function of the ration between the received energy *per bit* E_b and the noise psd N_0. For M-PAM, $E_s = \log_2(M) \cdot E_b$ and so

$$SER = 2\left(1 - \frac{1}{M}\right) Q\left(\sqrt{\frac{3\log_2(M)}{M^2-1}}\sqrt{\frac{2E_b}{N_0}}\right) \quad \text{(M-PAM)} \qquad (2.96)$$

A variant of the matched filter receiver that is equally optimum on the AWGN channel is depicted in Fig. 2.24. Take back into consideration (2.90) that describes the output of the matched filter for one-shot data transmission. We have

$$y(t) = r(t) \otimes h_{MF}(t) + n(t) = \int_{-\infty}^{\infty} r(\alpha)h_{MF}(t-\alpha)d\alpha$$
$$= \frac{1}{E_g}\int_{-\infty}^{\infty} r(\alpha)g(t_0 + \alpha - t)d\alpha \qquad (2.97)$$

The filtered received signal is evaluated at $t = T_0$, so that the soft decision variable is ($E_g = T$)

$$y(t_0) = \frac{1}{T}\int_{-\infty}^{\infty} r(\alpha)g(\alpha)d\alpha = \frac{1}{T}\int_{-\infty}^{\infty} r(t)g(t)dt \qquad (2.98)$$

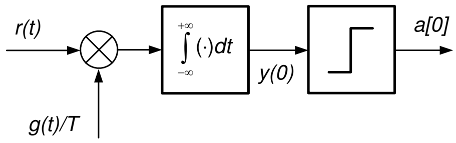


Fig. 2.24 Correlation receiver for baseband PAM data signals

This is just what is depicted in Fig. 2.24. The operation that is performed is the *correlation* between the received signal $r(t)$ and a local waveform that corresponds to transmission of a data bit equal to 1. The correlation receiver is just an alternative arrangement of the matched filter receiver, and so its BER performance is exactly the same.

The same results concerning the BER/SER of the matched filter/correlation receiver can be arrived at if we *normalize* equation (2.91). If we divide both the signal and the noise term (thus leaving the SNR unchanged) by the standard deviation $\sigma_n = \sqrt{N_0/2T}$ of $n(0)$ we get

$$y(0) = \sqrt{\frac{2E_s}{N_0}} a[0]g_r(t_0) + n_1(0) \tag{2.99}$$

where $n_1(0)$ is a zero-mean Gaussian random variable with *unit* variance. We may alternatively normalize by $\sqrt{P_r}$ obtaining

$$y(0) = a[0]g_r(t_0) + N(0) \tag{2.100}$$

where now $N(0)$ is a zero-mean Gaussian random variable with variance $\sigma_N^2 = N_0/(2E_s)$. We explicitly mention these alternative formulations since in the following we will use indifferently one of the three expressions (un-normalized (2.91), unit-variance (2.99), unit-amplitude (2.100)) that we introduced here.

Expression (2.93) for the BER/SER is called the *matched filter bound* and its appearance as a function of the E_b/N_0 ratio (that is basically a measure of the SNR experienced on every data symbol) is shown in Fig. 2.25. Why is this called a *bound*? Because this is what we would get for *one-shot* transmission of a *single* data symbol.Taking back into consideration the adjacent symbols, what we actually get at the output of the matched filter is

$$y(t) = \sqrt{P_r} \sum_{n=-\infty}^{+\infty} a[n]g_r(t - nT) + n(t) \tag{2.101}$$

that, evaluated at t_0 gives

$$\begin{aligned} y(t_0) &= \sqrt{P_r} \sum_{n=-\infty}^{+\infty} a[n]g_r(t_0 - nT) + n(t_0) \\ &= \sqrt{P_r}\, a[0] + \sum_{n \neq 0} a[n]g_r(t_0 - nT) + n(t_0) \end{aligned} \tag{2.102}$$

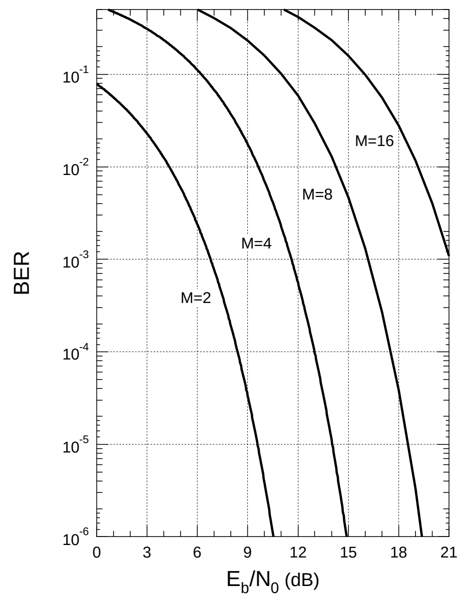


Fig. 2.25 BER of the matched filter receiver for M-ary PAM with no ISI

In addition to the data symbol to be detected and to the noise term as in the "one-shot" computation above, we also have an additional term (the intermediate one) that is due to the presence of trailing and leading symbols in the data stream, and is perceived as an interference of them on $a[0]$: the *InterSymbol Interference* (ISI). The ISI makes the BER degrade with respect to the value (2.93) given by the matched filter bound. Of course, not any pulse shape $g_r(t)$ arises ISI. If we have

$$g_r(t_0 - nT) \equiv 0 \; , \; n \neq 0 \tag{2.103}$$

then the ISI is no longer present, and we experience again the BER of the matched filter bound. The condition (2.103) is for instance actually satisfied by a (symmetric) NRZ pulse filtered by its matched filter. The resulting $g_r(t)$ is a triangular pulse as in Fig. 2.20 (b), and it is seen that its symbol-rate samples are all zero apart from the one at $t = t_0 = 0$. The same is true for the SRFRRC pulse, although is may not seem so at a first glance. To see this, take into consideration that the impulse response of the matched filter to an SRFRC is basically another SRFRC pulse, since $g(t)$ is even-symmetric. If we consider FTs, we have

$$\begin{aligned} G_r(f) &= G(f) \cdot H_{MF}(f) = G(f) \cdot G^*(f)/E_g \\ &= \sqrt{TG_N(f)} \cdot (1/T)\sqrt{TG_N(f)}/T = G_N(f) \end{aligned} \tag{2.104}$$

and so $g_r(t) = g_N(t)$. From (2.14) and from Fig. 2.7 it is easily seen that all samples of the Nyquist pulse $g_N(t)$ at $t_n = nT$ are null, apart from the one with $n = 0$ that is equal to 1. The same conclusion can be actually drawn with frequency-domain considerations. Take in fact again (2.103) with $t_0 = 0$. The sequence of symbol rate samples of $g_r(t)$ is such that $g_r(nT) = \delta[n]$. Therefore, taking the FT of both sequences and considering Poisson's relation (2.70)), we have

$$R \sum_{k=-\infty}^{+\infty} G_r(f - kR) = 1 \tag{2.105}$$

that is called the *Nyquist criterion* for the absence of ISI. If we replace G_r with G_N as above, we easily see that (2.105) is satisfied by virtue of the particular symmetry around $f = R/2$ of the roll-off region of $G_N(f)$ in Fig. 2.7, for any value of β.

We can now motivate the adoption of the SRFRC pulse in almost all bandlimited data communication systems currently in use: such pulse give rise to no ISI when matched-filter detected, combining optimal immunity against noise with the absence of ISI on a bandlimited channel.

2.7 MODULATION, DEMODULATION AND MODEM ARCHITECTURE

2.7.1 Linear I/Q digital modulation

It is widely known that a baseband signal like those encountered in the previous section is not suited for transmission over a wireless channel. Radio signals can be efficiently detected only when the size of the transmitting/receiving antenna is of the order of magnitude of the wavelength of the electromagnetic wave. Just to make an example, the wavelength of wave oscillating at a frequency f_0=100 kHz is $\lambda_0 = c/f_0$=3 km, thus making transmission of such a component really problematic.The solution to this is using a high-frequency carrier (for instance, in Wi-Fi wireless LANs the carrier frequency is f_0=2.4 GHz) and attaching to that carrier the digital data to be sent by *modulating* the amplitude and/or the phase of the carrier with the data bearing baseband signal much like what described in Sect. 2.4 concerning the general scheme of an I/Q modulator of Fig. 2.15. When we intend to perform a digital modulation, what we have to specify in that general scheme is how the I/Q baseband components of the bandpass signal are related to the digital data to be sent. In a sense, the different modulation schemes are different, specialized "front-ends" to the general I/Q modulator.

The simplest digital modulation is BPSK (Binary Phase-Shift Keying) wherein the Q component $x_Q(t)$ is null, and the I component is just an NRZ binary PAM signal like (2.79), so that the resulting complex envelope is

$$x(t) = A \sum_{n=-\infty}^{+\infty} a[n]g(t - nT) + \jmath 0 \quad \text{(BPSK)} \tag{2.106}$$

At each time instant, the In-phase carrier $\cos(2\pi f_0 t)$ is either multiplied by a positive value, or by a negative one, according to the polarity of the data to be transmitted.

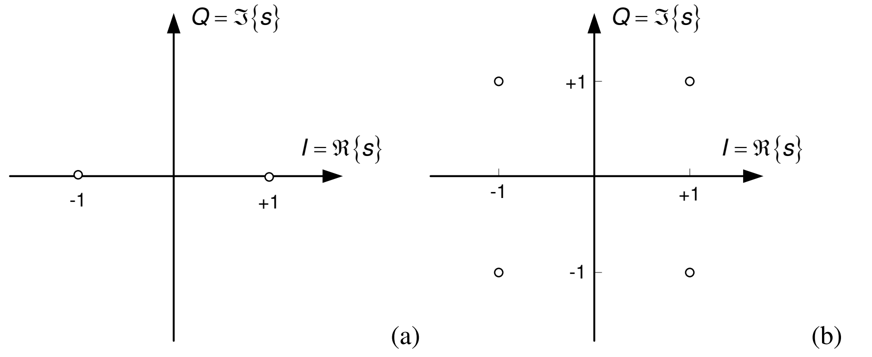


Fig. 2.26 Representation of BPSK (a) and QPSK (b) signals on the complex plane ($A = 1$)

Therefore, data modulation amounts to either not changing the phase of the carrier, or by *shifting* it by 180 degrees.

2.7.2 Signal constellations

Many variants of digital modulation exist. We stick here for simplicity to *linear* schemes, wherein the I and Q baseband components $x_I(t)$ and $x_Q(t)$, hence the complex baseband equivalent $x(t)$ of the modulated signal $x_{BP}(t)$, is obtained from the data bits with *linear* operations only. The very form of the BPSK signal (2.106) can be seen as a linear interpolation with pulse $g(t)$ of the digital stream $a[n]$. The simpler advancement to BPSK involves the use of the Q component that in BPSK is absent. To do so, we build an NRZ signal with a decimated version of the binary stream containing only even-index data $a[2m]$, and another NRZ signal with the odd-numbered bits $a[2m+1]$. Using the two signals as x_I and x_Q respectively, gives a *Quadrature Phase-Shift Keying* signal

$$x(t) = A \sum_{m=-\infty}^{+\infty} a[2m]g(t-mT) + \jmath A \sum_{m=-\infty}^{+\infty} a[2m+1]g(t-mT)$$

$$= A \sum_{m=-\infty}^{+\infty} \left(a[2m] + \jmath a[2m+1]\right) g(t-mT) \quad \text{(QPSK)} \tag{2.107}$$

At each time instant, the signal $x(t)$ takes one of four possible values given by $A(\pm 1 \pm \jmath)$ that lie on a circle in the complex plane with radius $\sqrt{2}A$, as represented in Fig. 2.26 (b). The four points have constant amplitude but four different phases, hence the denomination of quadri-phase signal. A BPSK signal on the contrary is characterized by two points only, both lying on the real axis as in Fig. 2.26 (a). It is apparent that for the same signaling rate $R = 1/T$, the two signals achieve different bit rate. BPSK has $R_b = R$, whilst with QPSK $R_b = 2R$, akin to what happens with a four-level PAM signal.

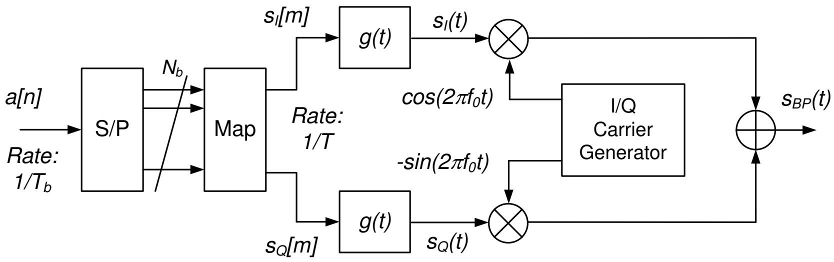


Fig. 2.27 Linear digital data modulator

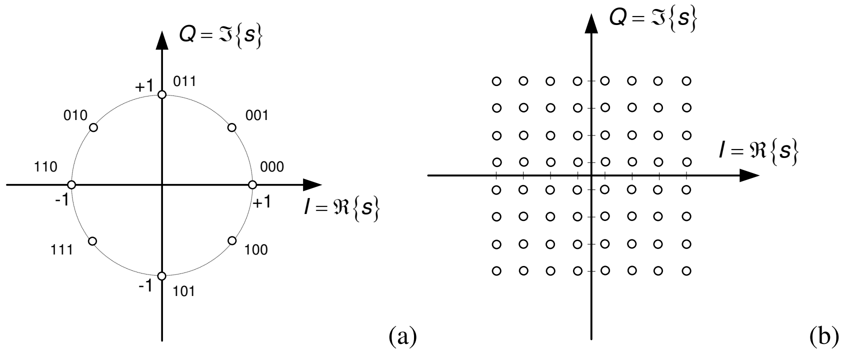


Fig. 2.28 8PSK (a) and 64-QAM (b) constellations - ($A = 1$)

The appearance of the QPSK signal in (2.107) suggests the general form for the "front-end" of a linear data modulator that is shown in Fig. 2.27. The input binary data stream $a[n]$ with a bit-rate R_b is segmented into *words* of N_b bits each, with a word rate $R_w = R_b/N_b$. Every word is used to identify a specific *symbol* in the complex plane selected in a set (the *alphabet*) of $M = 2^{N_b}$ elements. The representation of the symbols in the set is also called the *constellation* of the digital modulation. From the previous example, for QPSK we have $N_b = 2$ and $M = 4$. Higher-order M-PSK constellations with M equal to 8, 16 or 32 are characterized by M points evenly distributed on a circle in the complex plane. We show in Fig. 2.28 (a) an 8-PSK constellation where each symbol is labeled by the particular pattern of 3 bits (the value of the 3-bit word) that causes its own transmission. The correspondence between bit words and symbols in the constellation is the *mapping* that is implemented by the relevant block in Fig. 2.27. Upon generation of N_b bits, the mapper generates at time mT a new complex-valued symbol $s[m] = s_I[m] + \jmath s_Q[m]$ selected in the constellation. The symbol rate is $R = 1/T = R_b/N_b$.

As in baseband transmission, modulated signals need some form of bandlimiting to fit a particular bandwidth (centered on the carrier frequency f_0) that is assigned by some regulatory body. Such function is accomplished by bandlimiting the baseband equivalent of the signal prior to I/Q modulation. This is why the front-end in Fig.

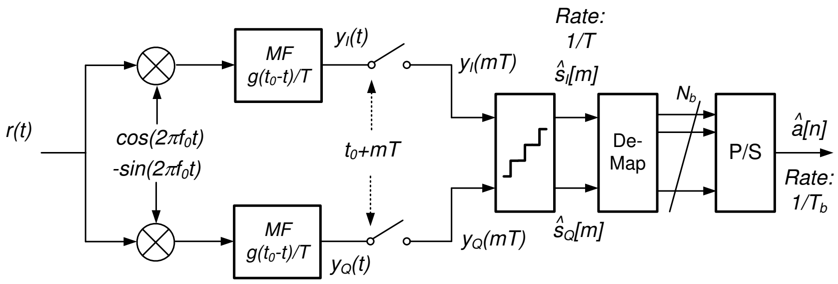


Fig. 2.29 I/Q Digital data demodulator

2.27 features a shaping filter (interpolator) with (bandlimited) pulse $g(t)$, usually a SRFRC function. The general form of the signal produced by the modulator is

$$s(t) = A \sum_{m=-\infty}^{+\infty} s[m]g(t - mT) \tag{2.108}$$

where $s[m]$ is the m-th complex symbol taken from a constellation to be specified, for instance the square 64-QAM constellation in Fig. 2.28 (b), that is produced by the composition of two 8-level PAM signals with alphabet $\{-7, -5, -3, -1, +1, +3, +5, +7\}$ on each component.

2.7.3 Demodulation of Linear I/Q modulated signals

Demodulation of a digital signal is straightforward if we appropriately combine (we would say "cascade") the I/Q demodulator as in Fig. 2.16 with matched filter detection (Fig. 2.23). In particular, the I/Q demodulator has to be followed by a "back-end" that reverse the operation of the front-end in Fig. 2.27, resulting in the schematic shown in Fig. **??**. The low-pass filter of the I-Q demodulator in Fig. 2.16 no longer appears since the double-frequency filtering function is performed by the matched filters. At the output of the signal samplers we get two "soft" values that represent a noise-corrupted version of the I/Q components of the transmitted symbol $s[m]$. If we collect a number of such samples and we report such values as dots on the complex plane, what we get is called a *scatter plot* and appears as in Fig. 2.30 for a 16-QAM constellation.

The "complex slicer" in Fig. 2.16 is the complex counterpart of the threshold detector for baseband PAM signals. The complex plane is partitioned into different zones, the so-called *decision regions* according to a decision rule, so that digital symbols are regenerated from the soft outputs $y(mT)$. Each region is labeled by a constellation symbol so that whenever $y(0)$ lies in a certain region, the "label" symbol is regenerated. The regenerated symbols $\hat{s}[m]$ are finally de-mapped to get back the N_b-bit word ($N_b = 4$ in the example) corresponding to the regenerated symbol. The function of the cascade of complex slicer and demapper is represented in Fig. 2.31,

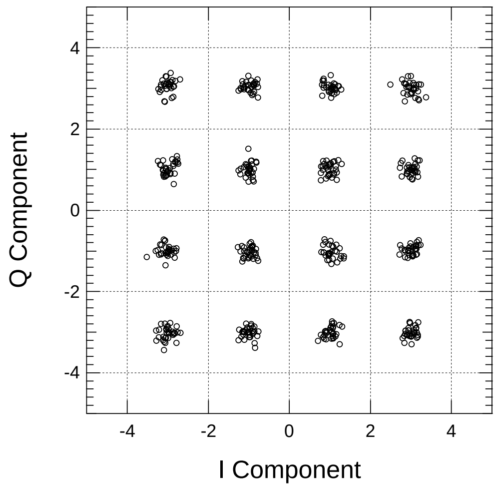


Fig. 2.30 Scatter plot of noisy 16QAM samples at the matched filter output

where we directly indicate the estimated N_b-bit word after demapping. The labels of the decision regions are directly the demapped bits.

When the noise component is large, $y(mT)$ may cross the boundaries of the decision region pertaining to $s[m]$, and a symbol error is produced: $\hat{s}[m] \neq s[m]$. The computation of the SER for a generic complex constellation is simple but tedious, so that we will not dwell here on such issue. A very tight bound, especially when the SER is small, is

$$SER \leq 4\left(1 - \frac{1}{\sqrt{M}}\right) Q\left(\sqrt{\frac{3}{2(M-1)}}\sqrt{\frac{2E_s}{N_0}}\right) \quad \text{(M-QAM)} \qquad (2.109)$$

2.7.4 Architecture of DSP-based Data Demodulators

How are the techniques for signal modulation and demodulation that were described in the previous Sections actually implemented in practical realizations? The current trend in modem design, both in a cheap Mobile Terminal (MT) of a cellular network (UMTS, GSM) and/or in an expensive Radio Base Station, is to perform as much signal processing in the digital domain as possible (modulation, demodulation, filtering etc.). In the receiver section of the modem (the most challenging to implement) we have basically two possible alternatives as far as the location of the ADC stage is concerned. The first approach, sketched in Fig. 2.32, is mainly pursued in low-power MTs. The RF received signal is converted to base-band using a conventional I/Q analog tuner followed by twin I/Q ADCs with sampling frequency f_s, and base-band digital signal processing for data detection. In the lowest-cost, lowest-power implementations, the

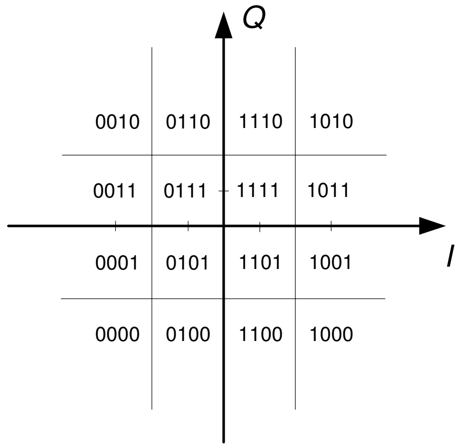


Fig. 2.31 Decision regions for a 16QAM signal with demapping

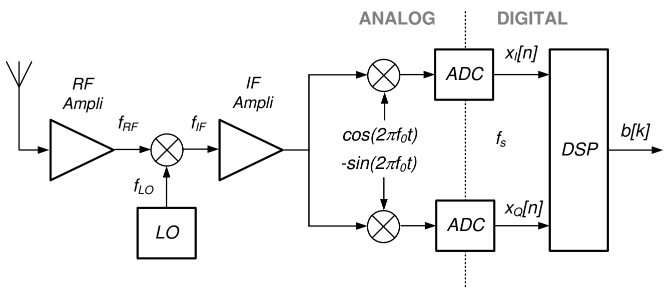
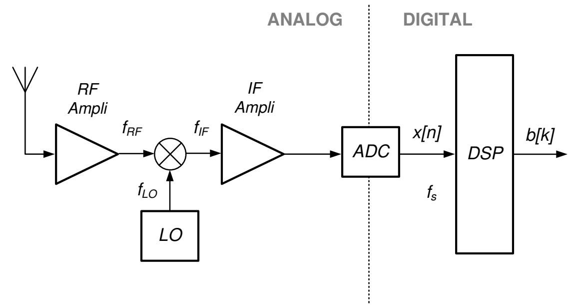


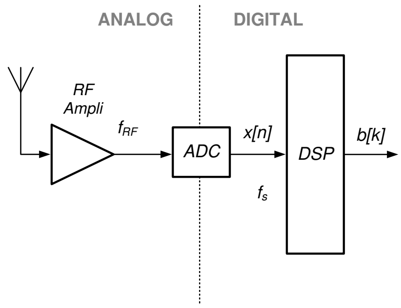
Fig. 2.32 I/Q Signal Demodulation with BaseBand sampling

first conversion stage to IF is absent, and the RF signal is directly brought to (I/Q) baseband (so-called *zero-IF* architecture).

The critical points of this arrangement are the possible amplitude imbalance of the two I and Q analog rails, as well as the imperfect quadrature between the two I/Q carriers used for IF to baseband conversion. A precision receiver with loose consumption or cost constraint is implemented according to the IF-sampling architecture shown in Fig. 2.33 (a), where the ADC stage is shifted towards the antenna. Incidentally, we observe that this is just the general scheme of the so-called *software-defined radios* (SDR), where all of the signal processing, apart from the initial RF-to-IF conversion is performed in the digital domain, and the DSP components are reprogrammable to a certain extent. With an SDR, changing the signal format to be handled just amounts to changing the software that drives the (programmable) DSP components, instead of changing a piece of dedicated hardware (a card, or the whole modem) as in conventional equipment.



(a)



(b)

Fig. 2.33 I/Q Signal Demodulation with IF sampling (a) or with RF sampling (b)

Coming back to Fig. 2.33 (a), the ADC operates directly on the IF signal, and the relevant digital output is still a bandpass signal on a different, digital, intermediate frequency $f_{DIF} = f_{IF} \pm k \cdot f_{sa}$ (k an integer). The analog front-end is simplified, and the task of base-band conversion is deferred to the digital section (with no issues of amplitude imbalance and imperfect quadrature). The main drawback of the IF-sampling approach is the tighter requirements for the ADC which must now handle faster, IF signals (instead of base-band signals as in Fig. 2.32). In particular, the converter rise and fall times, i.e., the time needed to "open" and "close" the gate of the sampling device shall be commensurate to the analog IF frequency, and turns out to be much shorter than those of the converters operating on the base-band signal. As a consequence, the IF-sampling ADC is in general more expensive and power-consuming than the two baseband converters in Figure 2.32. The "ultimate" version of a software-defined radio is shown in Fig. 2.33 (b), and is called a *direct-RF-conversion* receiver. The ADC is further moved towards the antenna, and the analog components are reduced to the bare minimum, in particular no IF conversion is needed. Of course, the issues related to the ADC cost and power consumption that were already mentioned regarding the IF-sampling architecture are here further exacerbated. Nonetheless, this kind of architecture is used in military equipment where special needs such as fast signal acquisition and detection, as well as total terminal reconfigurability, are of paramount relevance.