Pearson New International Edition

Wireless Communications and Networks

William Stallings

Second Edition

ALWAYS LEARNING

PEARSON

Pearson New International Edition

Wireless Communications and Networks

William Stallings
Second Edition

PEARSON®

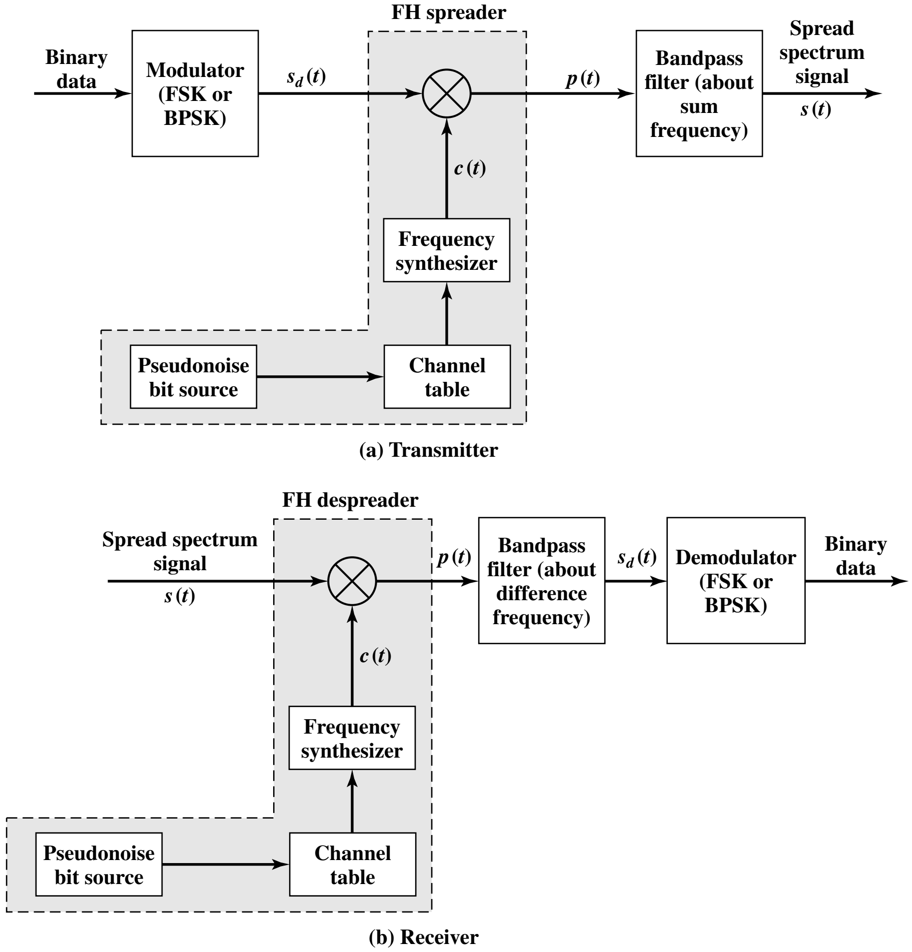

Figure 3 Frequency Hopping Spread Spectrum System

A typical block diagram for a frequency hopping system is shown in Figure 3. For transmission, binary data are fed into a modulator using some digital-to-analog encoding scheme, such as frequency-shift keying (FSK) or binary phase-shift keying (BPSK). The resulting signal $s_d(t)$ is centered on some base frequency. A pseudonoise (PN), or pseudorandom number, source serves as an index into a table of frequencies; this is the spreading code referred to previously. Each k bits of the PN source specifies one of the 2^k carrier frequencies. At each successive interval (each k PN bits), a new carrier frequency $c(t)$ is selected. This frequency is then modulated by the signal produced from the initial modulator to produce a new signal $s(t)$ with the same shape but now centered on the selected carrier frequency. On reception, the spread spectrum signal is demodulated using the same sequence of PN-derived frequencies and then demodulated to produce the output data.

Figure 3 indicates that the two signals are multiplied. Let us give an example of how this works, using BFSK as the data modulation scheme. We can define the FSK input to the FHSS system as:

$$s_d(t) = A\cos(2\pi(f_0 + 0.5(b_i + 1)\Delta f)t) \qquad \text{for } iT < t < (i + 1)T \tag{1}$$

where

A = amplitude of signal
f_0 = base frequency
b_i = value of the ith bit of data (+1 for binary 1, −1 for binary 0)
Δf = frequency separation
T = bit duration; data rate = $1/T$

Thus, during the ith bit interval, the frequency of the data signal is f_0 if the data bit is −1 and $f_0 + \Delta f$ if the data bit is +1.

The frequency synthesizer generates a constant-frequency tone whose frequency hops among a set of 2^k frequencies, with the hopping pattern determined by k bits from the PN sequence. For simplicity, assume the duration of one hop is the same as the duration of one bit and we ignore phase differences between the data signal $s_d(t)$ and the spreading signal, also called a chipping signal, $c(t)$. Then the product signal during the ith hop (during the ith bit) is

$$p(t) = s_d(t)c(t) = A\cos(2\pi(f_0 + 0.5(b_i + 1)\Delta f)t)\cos(2\pi f_i t)$$

where f_i is the frequency of the signal generated by the frequency synthesizer during the ith hop. Using the trigonometric identity[2] $\cos(x)\cos(y) = (1/2)(\cos(x + y) + \cos(x - y))$, we have

$$\begin{aligned} p(t) = 0.5A[&\cos(2\pi(f_0 + 0.5(b_i + 1)\Delta f + f_i)t) \\ &+ \cos(2\pi(f_0 + 0.5(b_i + 1)\Delta f - f_i)t)] \end{aligned}$$

A bandpass filter (Figure 3) is used to block the difference frequency and pass the sum frequency, yielding an FHSS signal of

$$s(t) = 0.5A\cos(2\pi(f_0 + 0.5(b_i + 1)\Delta f + f_i)t) \tag{2}$$

Thus, during the ith bit interval, the frequency of the data signal is $f_0 + f_i$ if the data bit is −1 and $f_0 + f_i + \Delta f$ if the data bit is +1.

At the receiver, a signal of the form $s(t)$ just defined will be received. This is multiplied by a replica of the spreading signal to yield a product signal of the form

$$p(t) = s(t)c(t) = 0.5A\cos(2\pi(f_0 + 0.5(b_i + 1)\Delta f + f_i)t)\cos(2\pi f_i t)$$

[2]See the math refresher document at WilliamStallings.com/StudentSupport.html for a summary of trigonometric identities.

Again using the trigonometric identity, we have

$$p(t) = s(t)c(t) = 0.25A[\cos(2\pi(f_0 + 0.5(b_i + 1)\Delta f + f_i + f_i)t) + \cos(2\pi(f_0 + 0.5(b_i + 1)\Delta f)t)]$$

A bandpass filter (Figure 3) is used to block the sum frequency and pass the difference frequency, yielding a signal of the form of $s_d(t)$, defined in Equation (1):

$$0.25A \cos(2\pi(f_0 + 0.5(b_i + 1)\Delta f)t)$$

FHSS Using MFSK

A common modulation technique used in conjunction with FHSS is multiple FSK (MFSK). MFSK uses $M = 2^L$ different frequencies to encode the digital input L bits at a time. The transmitted signal is of the form:

$$s_i(t) = A \cos 2\pi f_i t, \qquad 1 \le i \le M$$

where

$f_i = f_c + (2i - 1 - M)f_d$
$f_c =$ denotes the carrier frequency
$f_d =$ denotes the difference frequency
$M =$ number of different signal elements $= 2^L$
$L =$ number of bits per signal element

For FHSS, the MFSK signal is translated to a new frequency every T_c seconds by modulating the MFSK signal with the FHSS carrier signal. The effect is to translate the MFSK signal into the appropriate FHSS channel. For a data rate of R, the duration of a bit is $T = 1/R$ seconds and the duration of a signal element is $T_s = LT$ seconds. If T_c is greater than or equal to T_s, the spreading modulation is referred to as **slow-frequency-hop spread spectrum**; otherwise it is known as **fast-frequency-hop spread spectrum**.[3] To summarize,

Slow-frequency-hop spread spectrum	$T_c \ge T_s$
Fast-frequency-hop spread spectrum	$T_c < T_s$

Figure 4 shows an example of slow FHSS $M = 4$. The display in the figure shows the frequency transmitted (y-axis) as a function of time (x-axis). Each column represents a time unit T_s in which a single 2-bit signal element is transmitted. The shaded rectangle in the column indicates the frequency transmitted during that time unit. Each pair of columns corresponds to the selection of a frequency band based on a 2-bit PN sequence. Thus, for the first pair of

[3]Some authors use a somewhat different definition (e.g., [PICK82]) of multiple hops per bit for fast frequency hop, multiple bits per hop for slow frequency hop, and one hop per bit if neither fast nor slow. The more common definition, which we use, relates hops to signal elements rather than bits.

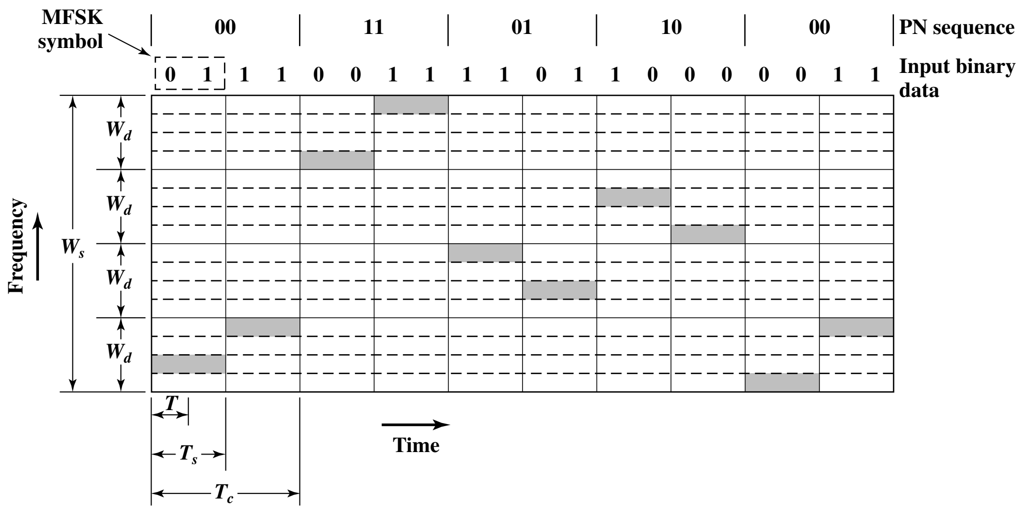

Figure 4 Slow-Frequency-Hop Spread Spectrum Using MFSK ($M = 4, k = 2$)

columns, governed by PN sequence 00, the lowest band of frequencies is used. For the second pair of columns, governed by PN sequence 11, the highest band of frequencies is used.

Here we have $M = 4$, which means that four different frequencies are used to encode the data input 2 bits at a time. Each signal element is a discrete frequency tone, and the total MFSK bandwidth is $W_d = Mf_d$. We use an FHSS scheme with $k = 2$. That is, there are $4 = 2^k$ different channels, each of width W_d. The total FHSS bandwidth is $W_s = 2^k W_d$. Each 2 bits of the PN sequence is used to select one of the four channels. That channel is held for a duration of two signal elements, or four bits ($T_c = 2T_s = 4T$).

Figure 5 shows an example of fast FHSS, using the same MFSK example. Again, $M = 4$ and $k = 2$. In this case, however, each signal element is represented by two frequency tones. Again, $W_d = Mf_d$ and $W_s = 2^k W_d$. In this example $T_s = 2T_c = 2T$. In general, fast FHSS provides improved performance compared to slow FHSS in the face of noise or jamming. For example, if 3 or more frequencies (chips) are used for each signal element, the receiver can decide which signal element was sent on the basis of a majority of the chips being correct.

FHSS Performance Considerations

Typically, a large number of frequencies are used in FHSS so that W_s is much larger than W_d. One benefit of this is that a large value of k results in a system that is quite resistant to noise and jamming. For example, suppose we have an MFSK transmitter with bandwidth W_d and noise jammer of the same bandwidth and fixed power S_j on the signal carrier frequency. Then we have a ratio of signal energy per bit to noise power density per Hertz of

$$\frac{E_b}{N_j} = \frac{E_b W_d}{S_j}$$

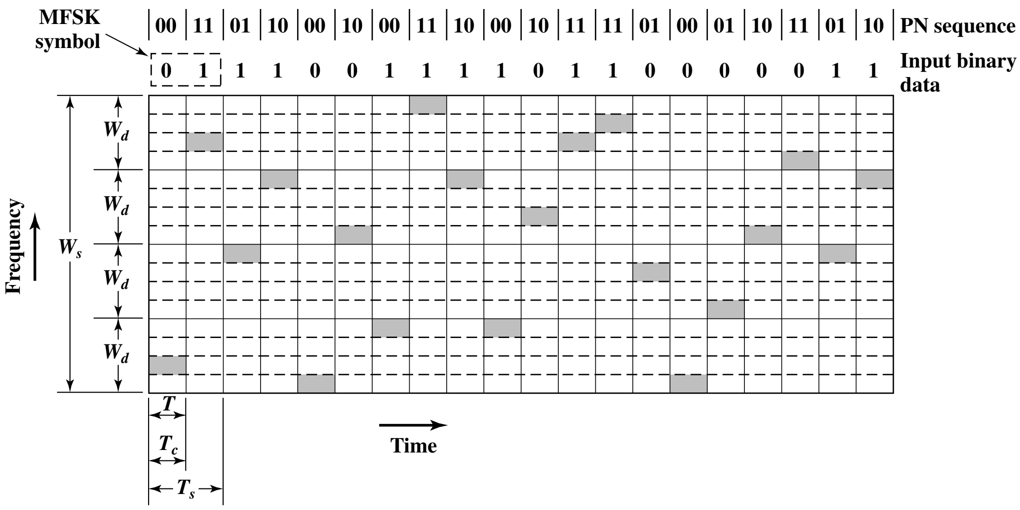

Figure 5 Fast-Frequency-Hop Spread Spectrum Using MFSK ($M = 4, k = 2$)

If frequency hopping is used, the jammer must jam all 2^k frequencies. With a fixed power, this reduces the jamming power in any one frequency band to $S_j/2^k$. The gain in signal-to-noise ratio, or processing gain, is

$$G_P = 2^k = \frac{W_s}{W_d} \tag{3}$$

3 DIRECT SEQUENCE SPREAD SPECTRUM

For direct sequence spread spectrum (DSSS), each bit in the original signal is represented by multiple bits in the transmitted signal, using a spreading code. The spreading code spreads the signal across a wider frequency band in direct proportion to the number of bits used. Therefore, a 10-bit spreading code spreads the signal across a frequency band that is 10 times greater than a 1-bit spreading code.

One technique for direct sequence spread spectrum is to combine the digital information stream with the spreading code bit stream using an exclusive-OR (XOR). The XOR obeys the following rules:

$$0 \oplus 0 = 0 \qquad 0 \oplus 1 = 1 \qquad 1 \oplus 0 = 1 \qquad 1 \oplus 1 = 0$$

Figure 6 shows an example. Note that an information bit of one inverts the spreading code bits in the combination, while an information bit of zero causes the spreading code bits to be transmitted without inversion. The combination bit stream has the data rate of the original spreading code sequence, so it has a wider bandwidth than the information stream. In this example, the spreading code bit stream is clocked at four times the information rate.

DSSS Using BPSK

To see how this technique works out in practice, assume that a BPSK modulation scheme is to be used. Rather than represent binary data with 1 and 0, it is more

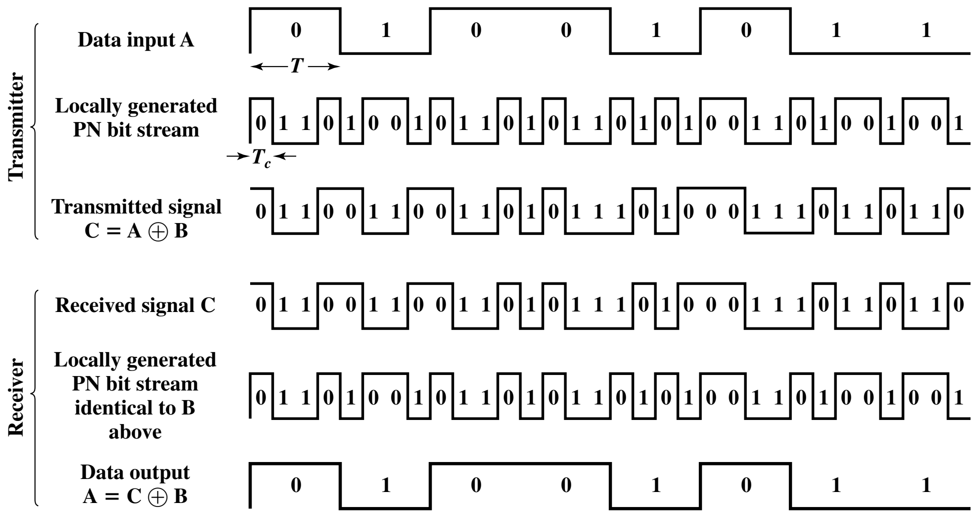

Figure 6 Example of Direct Sequence Spread Spectrum

convenient for our purposes to use +1 and −1 to represent the two binary digits. In that case, a BPSK signal can be represented as

$$s_d(t) = Ad(t)\cos(2\pi f_c t) \tag{4}$$

where

A = amplitude of signal
f_c = carrier frequency
$d(t)$ = the discrete function that takes on the value of +1 *for one bit time* if the corresponding bit in the bit stream is 1 and the value of −1 *for one bit time* if the corresponding bit in the bit stream is 0

To produce the DSSS signal, we multiply the preceding by $c(t)$, which is the PN sequence taking on values of +1 and −1:

$$s(t) = Ad(t)c(t)\cos(2\pi f_c t) \tag{5}$$

At the receiver, the incoming signal is multiplied again by $c(t)$. But $c(t) \times c(t) = 1$ and therefore the original signal is recovered:

$$s(t)c(t) = Ad(t)c(t)c(t)\cos(2\pi f_c t) = s_d(t)$$

Equation (5) can be interpreted in two ways, leading to two different implementations. The first interpretation is to first multiply $d(t)$ and $c(t)$ together and then perform the BPSK modulation. That is the interpretation we have been discussing. Alternatively, we can first perform the BPSK modulation on the data stream $d(t)$ to generate the data signal $s_d(t)$. This signal can then be multiplied by $c(t)$.

An implementation using the second interpretation is shown in Figure 7. Figure 8 is an example of this approach.