EE690: Research Project

Simulations of Spread Spectrum Communications with SystemView

University of New Haven, Winter 2001
Project Advisor: Dr. Aliane Bouzid
Student: Wiwat Tharateeraparb 888-80-6557

Table of Content

List of Figures

Chapter 1: Introduction to Spread Spectrum Systems

Spread spectrum is a special communications technique that deliberately uses much more RF bandwidth than necessary to transmit signal. A sophisticated receiver process the wide bandwidth RF signal to recover a much more lower bandwidth version of original signal, from which the desired data can be efficient extracted. This technology is of interest due to the noise-like properties of the transmitted signal and the several unique properties of the processing circuitry in receiver.

Spread spectrum was originally developed by military to improve the security of sensitive radio communications. The technology has been under active development by military and government agencies in the US and elsewhere for over 30 years. This technology offers three main advantages for secure radio communications:

- It is more difficult for the adversary to detect
- It is considerably harder to recover the information once the transmission has been detected.
- It is resistant to intentional jamming.

None of these general advantages of spread spectrum technology are of particular importance in wireless microphone systems, however. This new technology is being promoted by the Government as one means of providing wider access to radio communications by the general public and industry. One of these new policies is to allow use of unlicensed spread spectrum transmitters in the 902 to 928 and 2400 to 2483.5 MHz. Bands, provided that the equipment employed has been pre-approved by the Government. Within these frequency bands, approved spread spectrum equipment may be used by anyone, at any time, for any legal purpose, subject only to sime minor conditions. For these and other reasons, spread spectrum technology is being rapidly adopted for a wide range of private, commercial and industrial application.

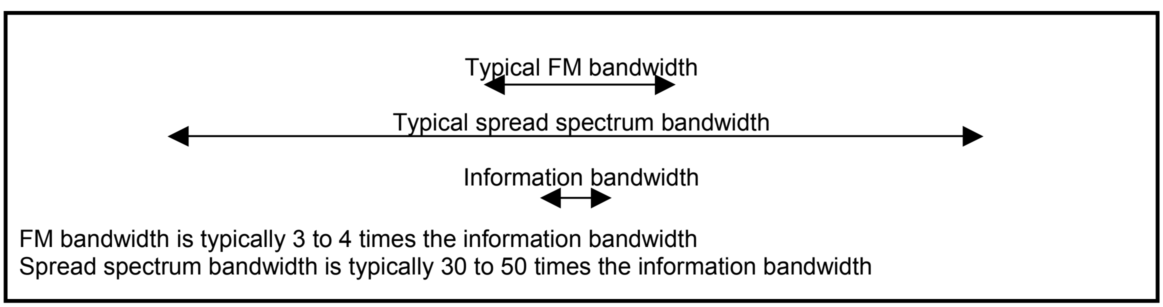


Fig.1 Spread spectrum bandwidth usage

In this research project should provide simulations of spread spectrum techniques via SystemView by Elanix. All files used to contribute the graphs and tables in this report are included along with the body of this report

Chapter 2: Spread Spectrum Techniques

Some of today Spread Spectrum Techniques are as follows:

1. Direct Sequence (DS)
2. Frequency Hopping (FH)
3. Hybrid of DS and FH
4. Time Hopping (TH)

The DS and FH Spread Spectrum techniques are the two basic types discussed in this project. Both FH and DS are currently implemented in today commercial communication solutions such as Cellular Telephony, Global Positioning Systems (GPS) and Wireless LAN, which mostly concern secure connections.

2.1 Direct Sequence Spread Spectrum (DSSS)

Another name for DSSS is Pseudo Noise Spread Spectrum (PN) and both of them pertain to the same techniques. DSSS makes use of the noise-like properties of pseudorandom digital sequences. The idea is to spread the energy contained in a narrow-bandwidth signal over a much wider bandwidth for transmission. This is accomplished by means of a phase modulator driven by the digital pseudorandom sequence. The phase modulator, such as BPSK, QPSK or M-ary QAM, distributes the energy in the original signal over a bandwidth proportional to the bit rate of the PN sequence. Thus, the amount of energy at any frequency within the bandwidth is quite small. Fig.2 shows the block diagram of spectrum estimation of a digital sequence and its PN modulated signal (dsss_0.svu) and Fig.5 describes some of the basics of DSSS. Starting from the left, the data source is modulated by a much higher bit rate PN sequence called “PN Code”. The nature of the code will be discussed in the later chapter.

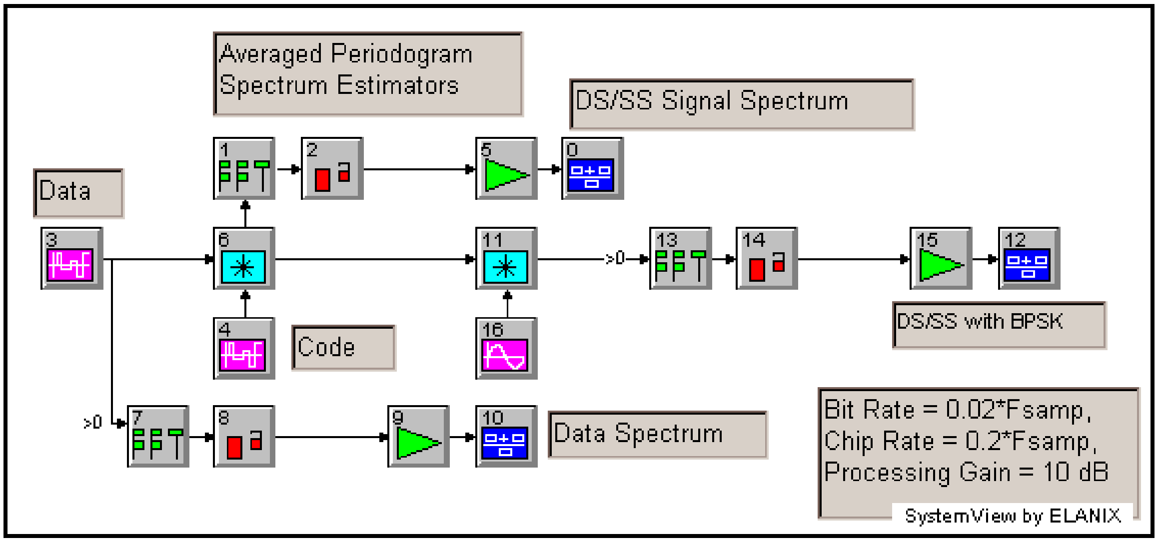


Fig.2 Averaged periodogram of Polar NRZ and its spread signal block diagram (dsss_0.svu)

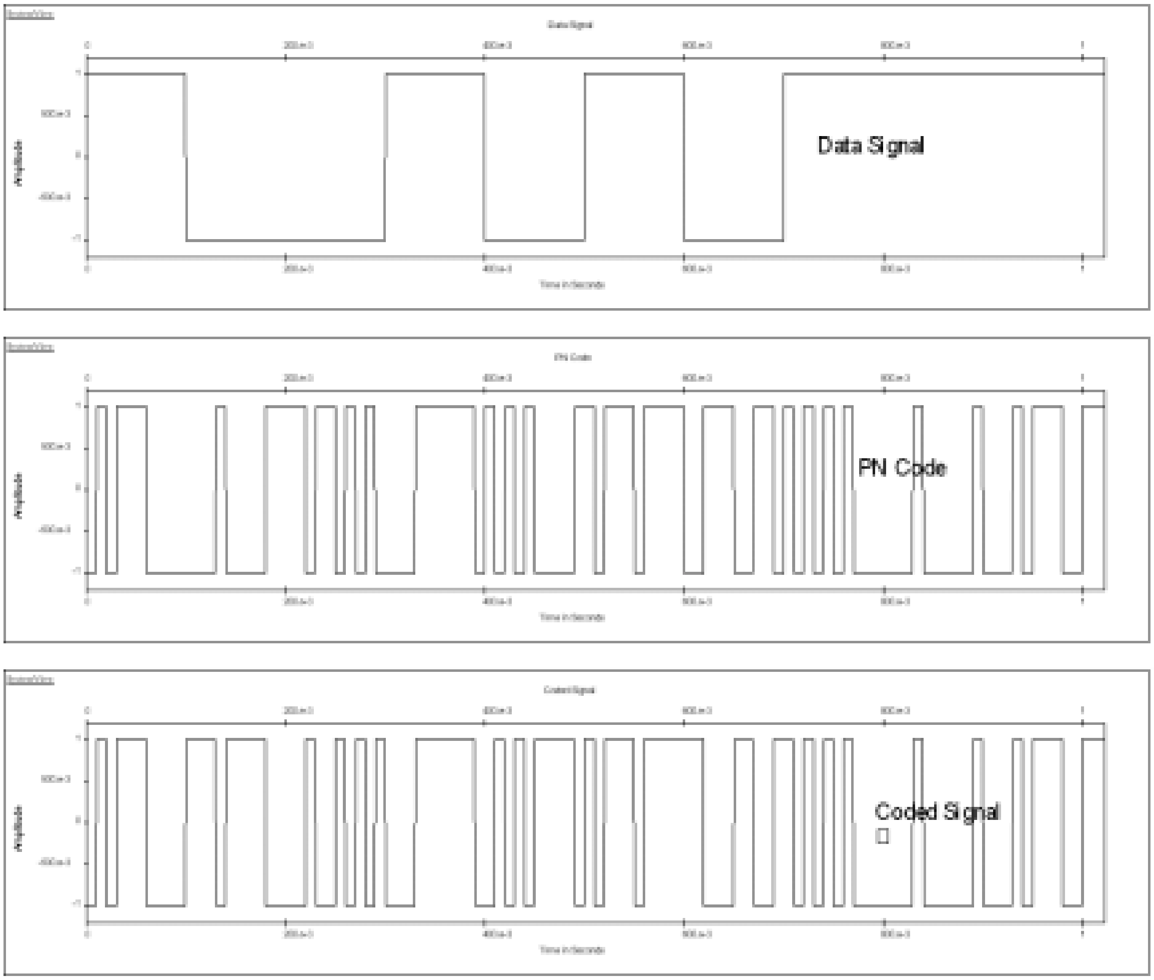


Fig.3 Example of DSSS signals

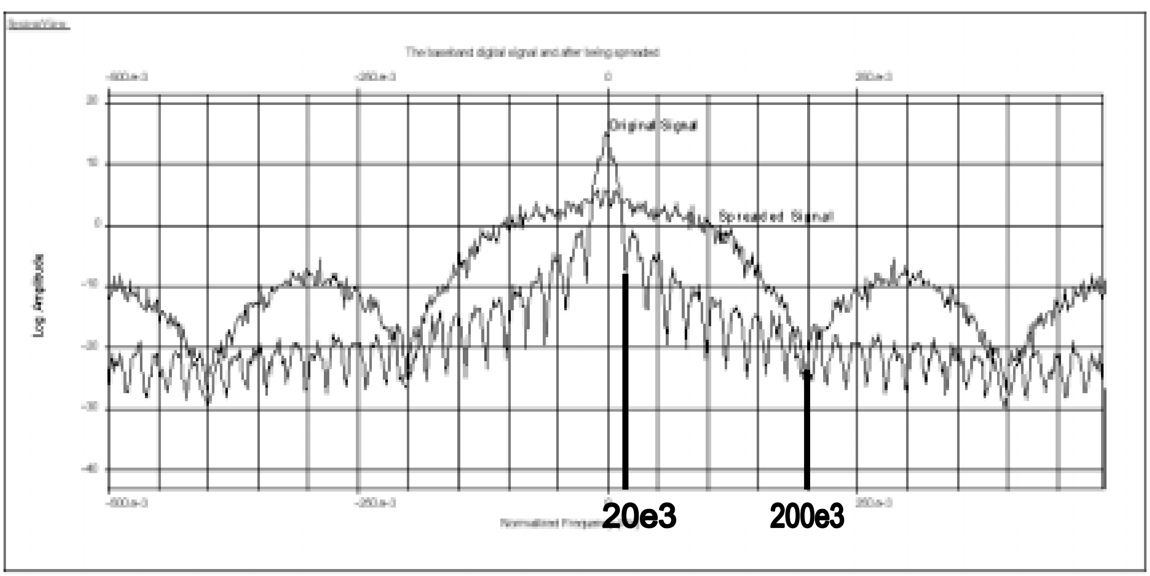


Fig.4 Example of spreading signal PSD in DSSS

Fig.3 is an example of a DSSS signal. The data is combined with the PN code which yields a result of coded signal which has much higher bandwidth. The spectra of DSSS signals are shown in Fig.4. However, the spectral estimation capabilities of the SystemView software analysis window is limited to FFT analysis of a single-window data record. To reduce the variance in the power spectrum density (PSD) estimation of random signals, the *averaged periodogram* [3, p.730] is useful. The SystemView file dsss_0.svu introduces the use of this method to estimate the PSD. In Fig.2 the original data rate 0.02 Hz. is multiplied by the PN code with the rate of 0.2 Hz. Note that in this system the sampling rate is set to unity. This simple DSSS system also yields a significantly important parameter called *Processing Gain* computed as the ratio of code rate to bit rate:

$$G_p = R_c/R_b = 0.2/0.02 = 10 \text{ dB} \quad (1)$$

Besides, by observing the Fig.4, the first nulls of signal spectra fall at 0.2 and 0.02 Hz. as a property of NRZ polar signals.

The next example is a simulation of the DSSS system with BPSK modulation with the present of interferences. We will investigate the effects of those interferences through the average error and Eye Diagram. The system is described as Fig.5.

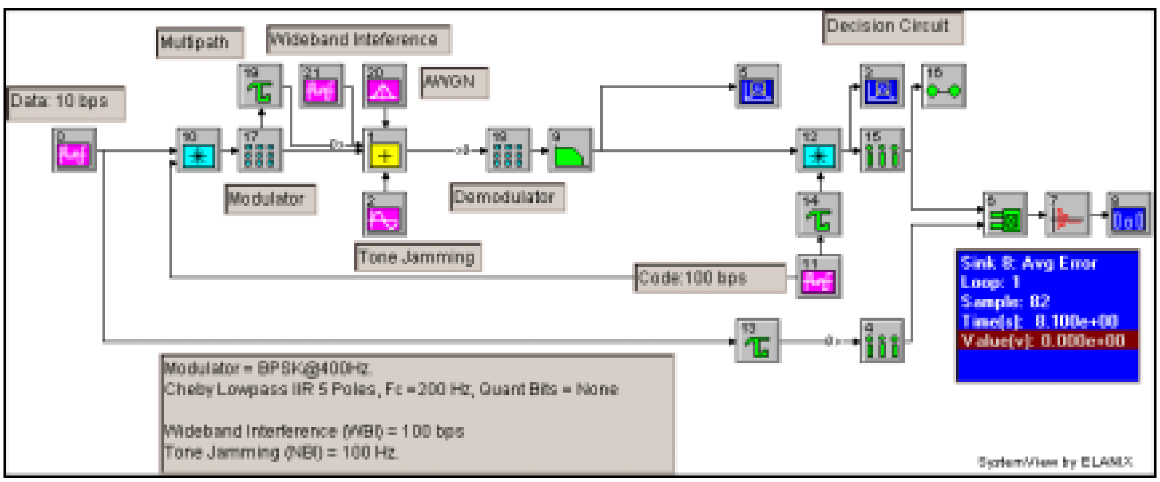


Fig.5 DSSS System (dsss_5.svu)

In Fig.5, the10 bps data is multiplied by the 100 bps code and modulated as a BPSK (Binary Phase Shift Keying) signal and transmitted. Once transmitted, the signal is subject to all sorts of impairments. It can bounce and come back on itself to produce mutipath interference. In addition, wideband interference (WBI) such as Additive White Guassian Noise (AWGN) and narrowband interference (NBI) such as single tone jamming can corrupt the signal through the transmission path. At the receiver, the received signal is demodulated and is filtered out by a Chebyshev LPF. Then the same identical PN code is used to decode or despread the signal in order to recover the baseband signal. Precisely, the decision circuit is used to determine the most likely digital level of the baseband signal and fed to the Error Detection at the end to calculate the average error. Theoretically, DSSS system can be described by these equations:

Designate symbols are as follows:

Original Signal : d(t)
PN Code : Pn(t)
Coded signal : d(t) x Pn(t)
NBI : Narrowband Interference (Tone jamming)
WBI : Wideband Interference (AWGN)
MP : Mutipath Interference
n : Other overall noise

Received signal : $y(t) = d(t)Pn(t) + NBI(t) + WBI(t) + n(t)$ (2)

Despread signal : $y'(t) = y(t) \times Pn(t)$
$= d(t)Pn(t)Pn(t) + NBI(t)Pn(t) + WBI(t)Pn(t) + n(t)Pn(t)$ (3)
$= d(t) + N(t)$

As mentioned earlier, the baseband signal is multiplied by PN code, say, Pn(t) to produce the product that is eventually fed to a balance modulator and transmitted. The received signal is a combination of all impairments. At the receive end, we again multiply by the synchronized Pn(t). Choosing a synchronized Pn(t) will lead to the product of Pn(t) x Pn(t) = 1. Thus, after this so-called dispreading, the received signal is the original signal plus the product of all these interferences, denoted by N(t) and the bandwidth of N(t) is much greater than the rate of d(t) by which graphically described in Fig.6.

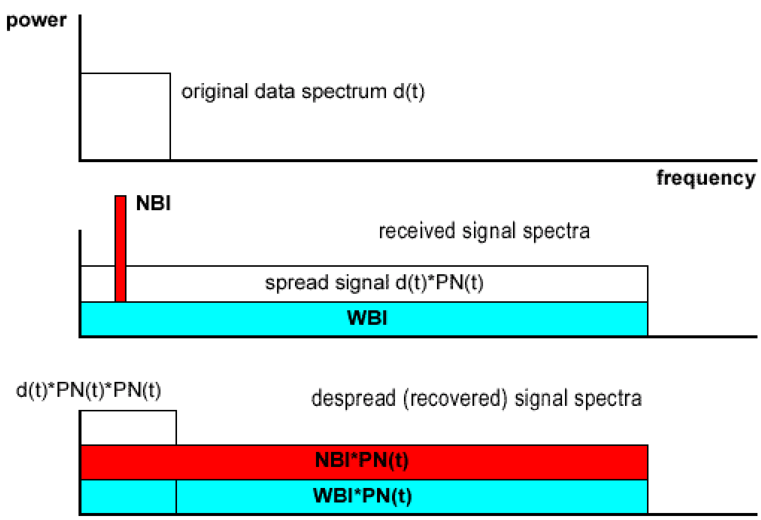


Fig.6 Frequency oriented DSSS processing

By simulating the block diagram in Fig.5 with the parameters as follows:

Parameters of the system in the Fig.5 , SystemView file: dsss_5.svu

Data rate = 10 bps, code rate = 100 bps, BPSK modulating carrier = 400 Hz., 1.414 V., Cheby Lowpass IIR filter 5 Poles, Fc =200 Hz. and interferences are:

WBI – another small cross-correlated 100 bps, low-level PN sequence represents an adjacent channel interference (ACI). Amplitude = 0.1 V.
Delay Token –represents a multipath signal generation. Delay = 0.5 msec.
AWGN – an Additive White Guassion Noise. Variance = σ^2 = 0.4.
NBI – a tone jamming. 100 Hz., 1 V.
So, the processing gain is 10 dB, Pj/Ps = 0 dB, Ps/ σ^2 = 4 dB, where Pj/Ps is the ratio of jamming power to carrier signal power and Ps/ σ^2 is a SNR.

By simulating in Fig.5, the Eye Diagram of the system is shown in Fig.6. along with the original data and received data.

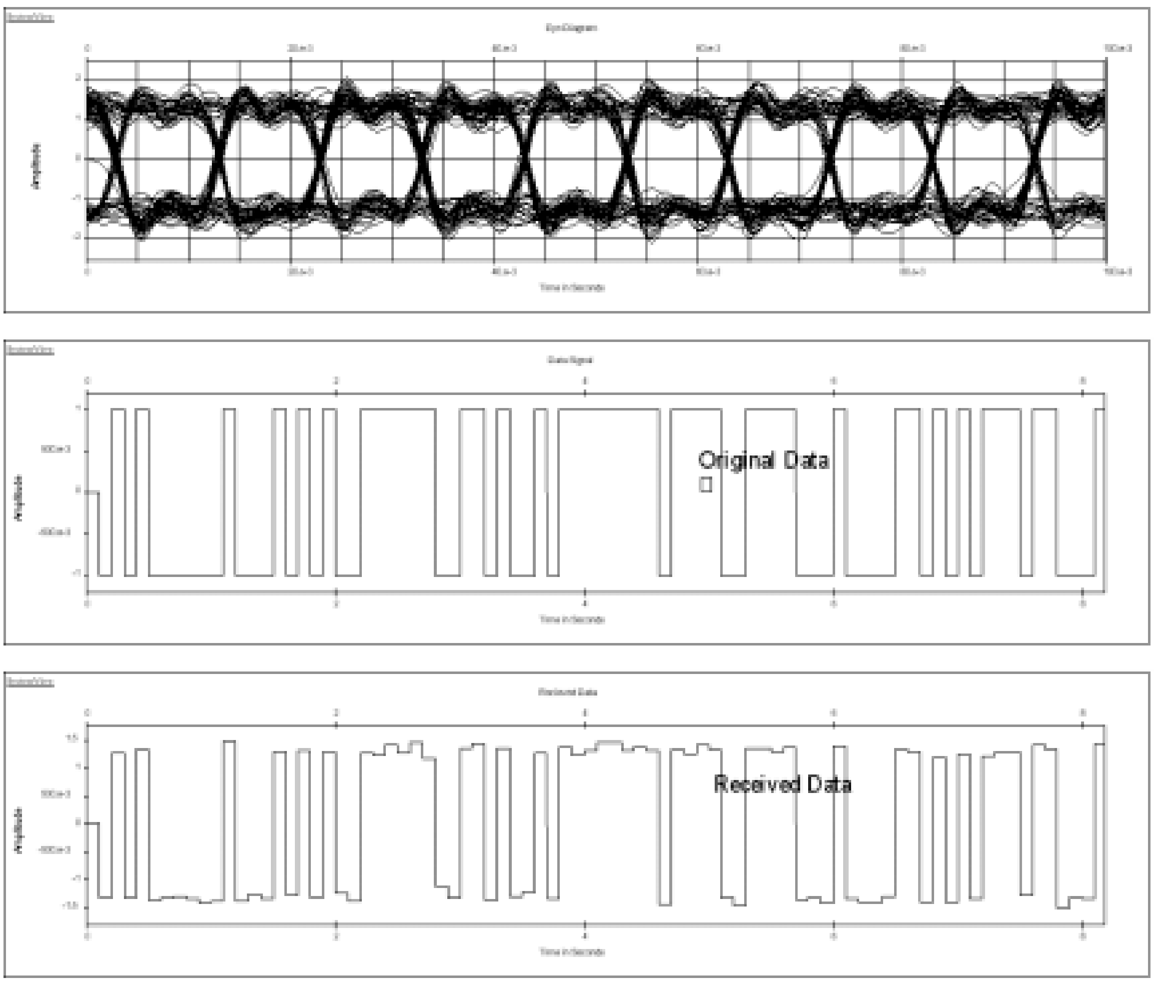


Fig.6 Eye Diagram of the signal after LPF, data and received data

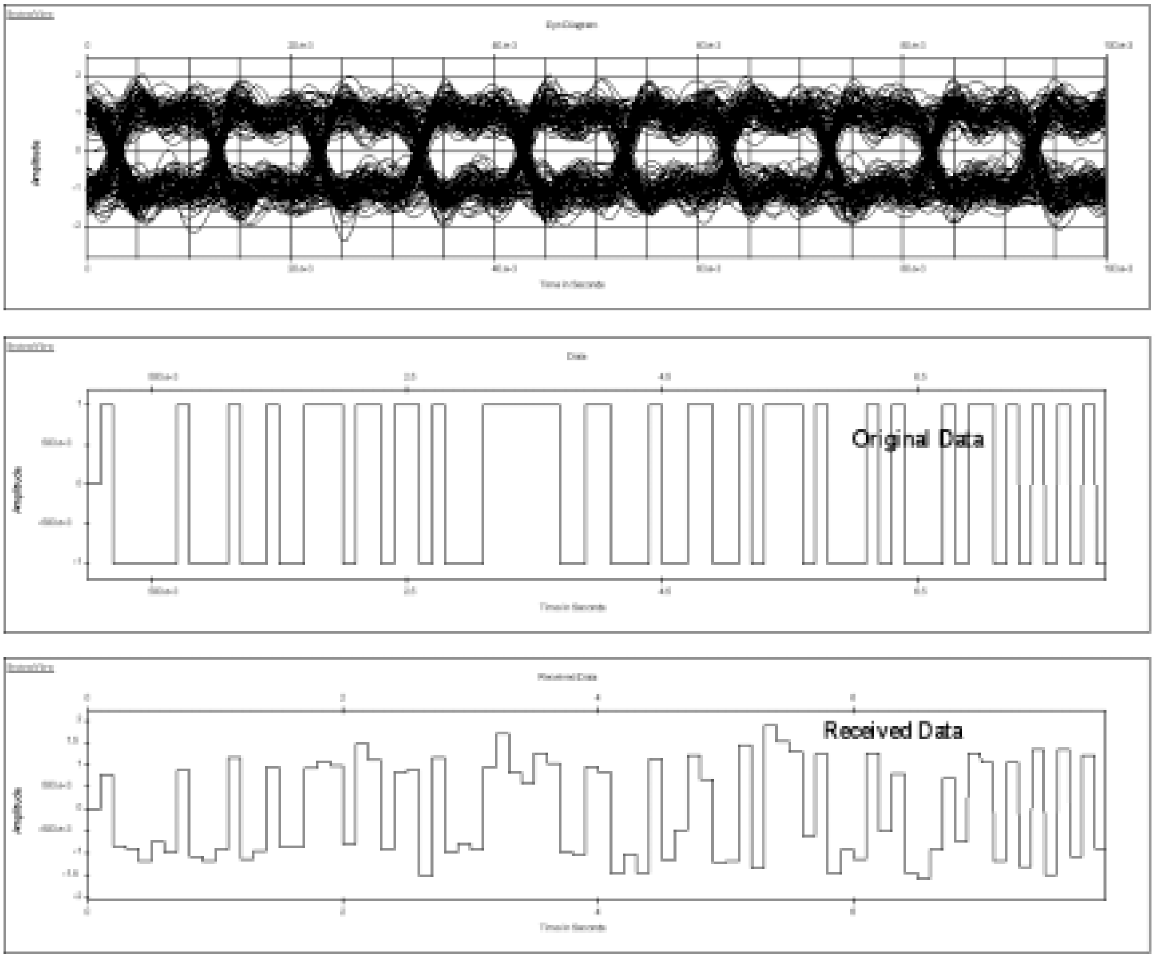


Fig.7 Effect of SNR against the received signal

Upon decreasing the SNR to 2 dB, the Eye Diagram is more likely to close than at the higher SNR and the received data is corrupted. The Eye Diagram in the SystemView simulation window, which is an output of a match filter, can be achieved by using a "*slice*" style of the data in the *Sink Calculator* window.

In the purpose of AWGN and probability error investigation, we disable other kinds of interferences and measure the probability error (P_E) versus the SNR and the result is given below.

Ps/σ^2 (dB)	***P_E***
-2 ($\sigma^2 = 1.59$)	0.067
-4 ($\sigma^2 = 2.51$)	0.195
-6 ($\sigma^2 = 3.98$)	0.302
-8 ($\sigma^2 = 6.31$)	0.345
-10 ($\sigma^2 = 10.00$)	0.393

At the end of the system in Fig.5, the original data is used to compare with the received data by an XOR operation to produce the average probability error (P_E). In Fig.8, one example is shown as the SNR = -4 dB and the P_E = 0.195.

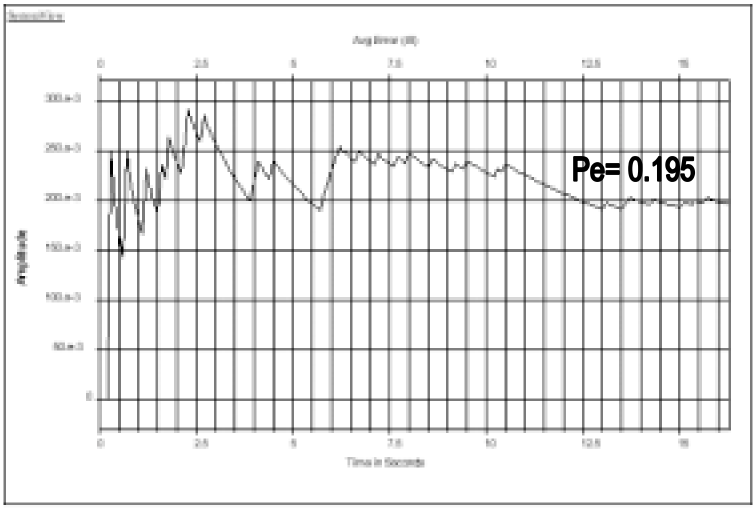


Fig.8 Average Error of the received data @ SNR = -4 dB

To reduce the average error of the received data, a suitable line coding and PN code play a significant role in data correction as well as the overall performance. There are several types of PN code used in DSSS, which are addressed in another chapter. A usual way to create a PN code is by means of at least one shift register.

The special features of DSSS are such as:

- Secure Communication and Jamming Resistance
- Multiple Access: several users on the same band
- Advantage of CDMA
- Resistance to mutipath fading: a signal attenuation in particular frequency range is shared among all users.

Which is mentioned in another chapter as applications and simulations.

The DSSS has the best performance in terms of jamming rejection and multipath immunity. However, it does suffer from the *near-far* problem. The discussion so far is assumed that the signals from all sources are received with the same signal power. When this is not true, we may encounter the near-far problem. The dispreading of a desired signal increases it strength by G_p compared to the residual signal level due to other unwanted signals. However, if an unwanted signal strength is strong due to the proximity of its transmitter and receiver, an the strength of the desired signal is weak due to the remoteness of its transmitter from the receiver, the undesired signal may still overpower the desired signal. This often necessitates adaptive power control techniques to overcome the near-far problem. In addition, the interference from multiple users, although small individually, adds up to degrade the system performance. If all the codes were strictly orthogonal or uncorrelated, this problem would not arise. Unfortunately, it is difficult to find a large number of codes that are strictly orthogonal, and we have to use many codes that are nearly orthogonal.

2.2 Frequency Hopping Spread Spectrum (FHSS)

As a matter of fact, Frequency Hopping Spread Spectrum technique is the simplest conceptually. In the transmitter, a frequency synthesizer is controlled by a microprocessor or a roughly equivalent controller device. The transmitter frequency is changed to a different pre-assigned channel several times per second. This process is called “hopped”, The order in which the pre-assigned channels are selected is pseudorandom. In other words, the channel order is seemingly random, but actually repeats itself at a defined interval.

At the receiver, another synthesizer and controller steps the receiver channel frequency thru the same list of pre-assigned channels is the same order as for transmitter. Once the transmitter and receiver are “synchronized the received signal will be “de-hopped”. That is, the recovered signal will be at a fixed frequency which can be demodulated to recover the information being sent.

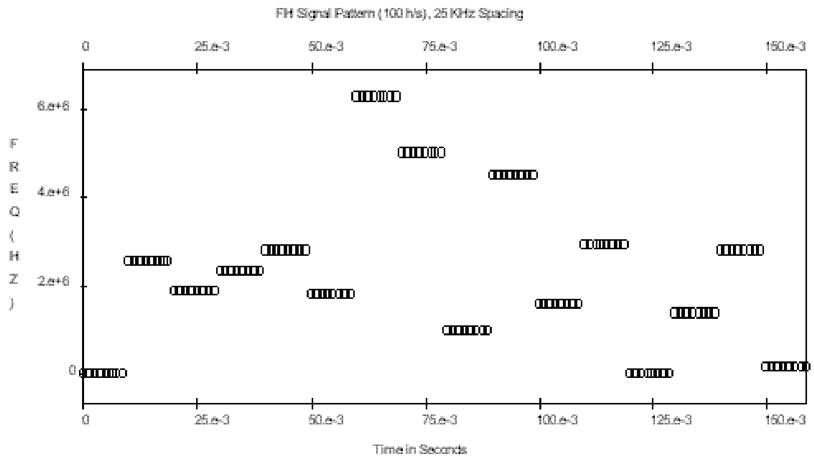


Fig.9 Frequency time diagram of FHSS

Fig.9 is an example of carrier hopping from one frequency to another at some rate. A jammer, of course, needs to know exactly where those frequencies are. The typical numbers of hop rates are in the order of about 100 hops per second and a channel spacing are something like 25 kilohertz. Other number is possible, but these are numbers used in tactical military systems.

Not like DSSS, FHSS is less affected by the near-far problem. If the near-interference is present, only a number of frequency-hops will be blocked instead of the whole signal. From the hops that are not blocked it should be possible to recover the original data.

Processing Gain = Ratio of the total hop BW to information BW = # channels (4)

Let us calculate the processing gain of a frequency hop FH system. The de-hopping operation does not change the signal power so the output power is equal to the input power. The output bandwidth after collapsing is that of the information bandwidth itself. The input bandwidth W is the total bandwidth over which the system's hopping from frequency to frequency. Substituting these terms into the basic equation yields result that the processing gain is the ratio of the total spread bandwidth to the data bandwidth equal the total number of channels over which the system is hopping. For the tactical systems, the data rate is 25KHz and the total bandwidth can be as high as 58MHz so the number of channels is 2320 or roughly about 34dB of processing gain.

<u>Parameters of the system in the Fig.10 , SystemView file: fhss_2.svu</u>

The essential tokens used to implement the digital frequency synthesizers in FHSS system are FSK modulators. By using 12 levels of PN code at the rate of 10 Hz., we get the hop rate of 120 hop/second and 12 channels of signal hopping. Data rate is 20 bps and modulated by 2-FSK of 50 Hz step frequency. During the transmission, the RF signal is interfered with an AWGN and a tone jamming. At the receiver, to synchronize the PN code we use the same code to generate another frequency synthesizer in order to dehop the signal. Some errors are introduced in the Fig.12.

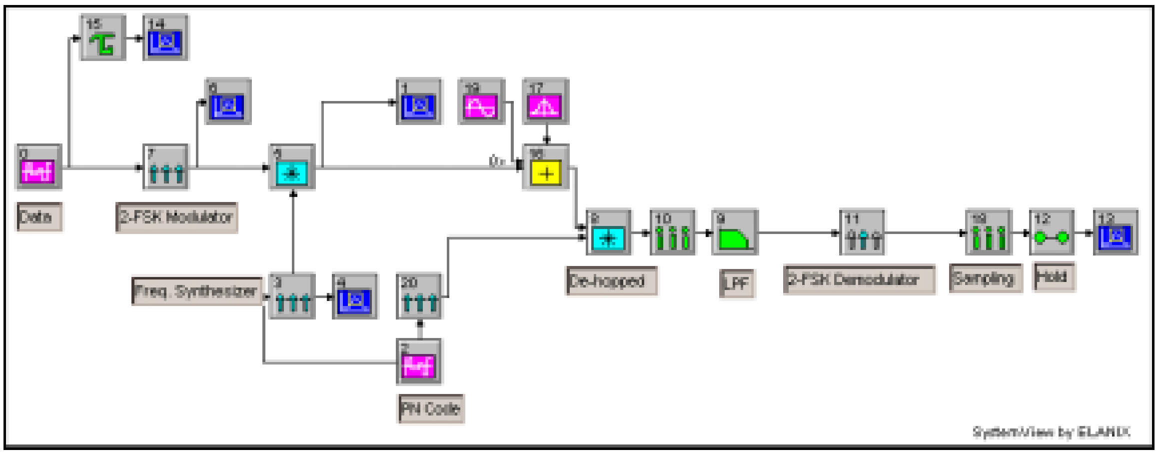


Fig.10 FHSS system (fhss_2.svu)

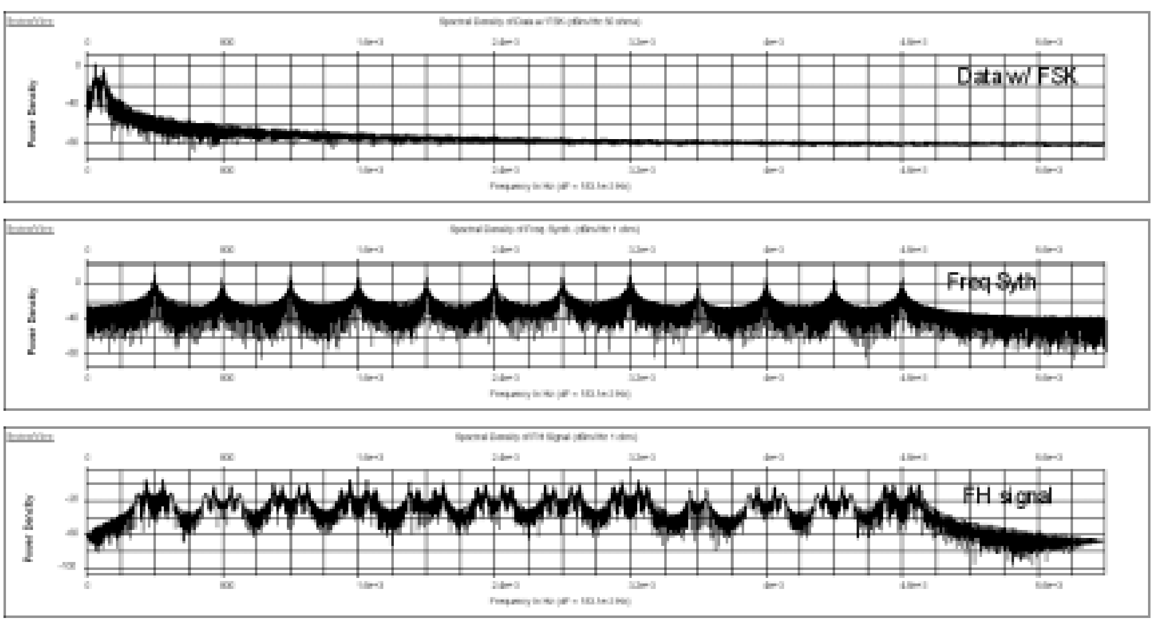


Fig.11 Spectra of (a) modulated signal, (b) Frequency synthesizer, (c) FHSS signal

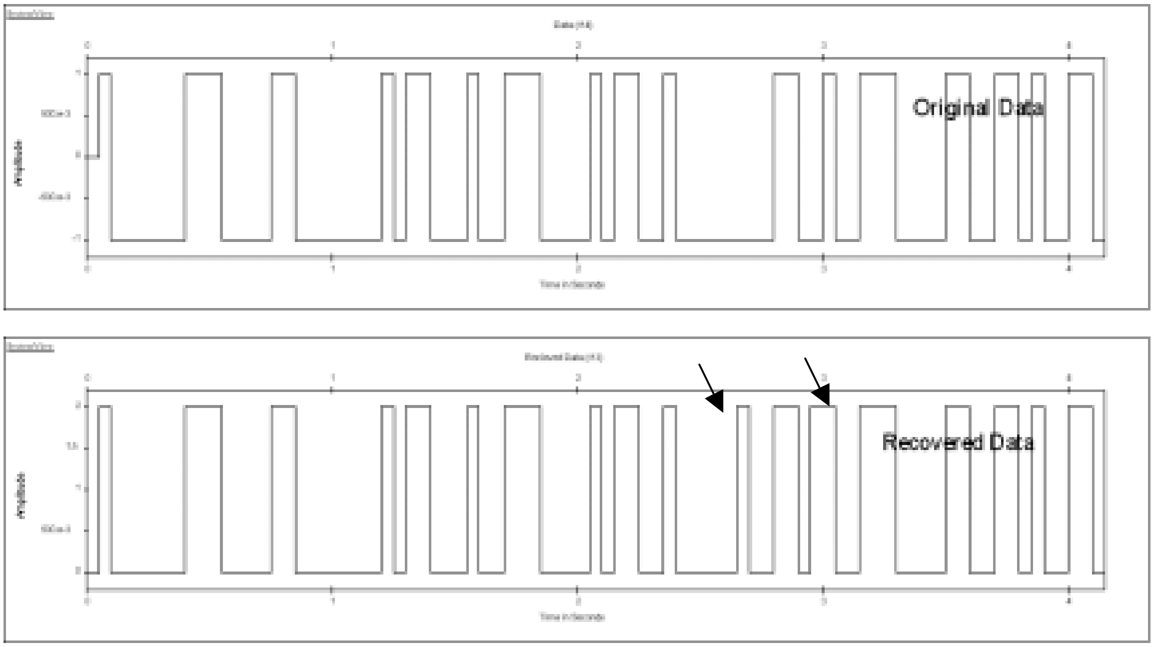


Fig.12 Original data and recovered data with the present of bit errors

Synchronization of the receiver with the transmitter is a complicated process. Even with sophisticated software algorithms and high performance hardware, the process can require significant time, during which no information is being received. Synchronization delays will be experienced upon initial turn-on and after any signal dropout.

Additional spread spectrum “channels” can be obtained by using different pseudorandom hopping sequences and the same channel frequencies. That is, the order in which the transmitter hops to new frequencies can be changed. If frequencies are shared or relatively close together, however, there will be severe interference each time two transmitters randomly switch to the same RF frequency.

So which of DSSS or FHSS is better? Unfortunately, there is no good answer to this question. The answer will depend on the nature of what are you trying to accomplish and the constraints under which you have to operate. For example, CDMA DSSS is used in IS-95 (discussed in later chapter) because they can control power as FHSS is used in army tactical radios because of near-far problem as mentioned earlier.

2.3 Summary of Spread Spectrum Techniques

Direct Sequence Systems

Advantages

- DSSS has the best noise and anti-jam performance.
- It is the most difficult to intercept.
- Flights well against multipath effect.

Disadvantages

- DSSS requires large bandwidth channel with relatively small phase distortions.
- Due to long PN codes, it requires long code acquisition time.
- Near-far problem exists.

Frequency Hopping Systems

Advantages

- It provides the greatest amount of spreading.
- It can be arranged to avoid portions of the bands occupied by other systems in order to avoid the channel fading.
- Not much affected by near-far problem.

Disadvantages

- It requires a complex frequency synthesizer in order to generate the hops.
- It requires error correction.

Chapter 3: Spread Spectrum Applications

This chapter will investigate the impairment resistant of Spread Spectrum system. The applications are:

- Multipath distortion resistant
- Narrowband Interference (NBI) and Wideband Interference (WBI) resistant
- CDMA and multi-access

3.1 Application to Multipath

The multipath interference occurs when the transmitted signal arrives at the receiver by two or more path of different delays. In radio links, the signal can be received by direct path between the transmitting and the receiving antennas and also by reflections from other objects, such as hills, buildings and so on. Consequently, the magnitude and the phase characteristics will cause linear distortion. If, for instance, the gains of two paths are very close, the signal received by the two paths can very nearly cancel each other at certain time or frequencies.

<u>Parameters of the system in the Fig.13 , SystemView file: multipath.svu</u>

Data rate: 2 Hz., code rate: 50 Hz., relative delay is 0.51 sec. and the gains are 0.7.

We have two simulations running side by side, one is using DSSS and one not. The transmitted signal is multiplied by gain as the delay signal representing the multipath signal bouncing from an object. Those signals are added together in typical multipath fashion. At the receiver, a matched filter is used to recover the original signal by setting the average window timing matched to the bit rate of the original data.

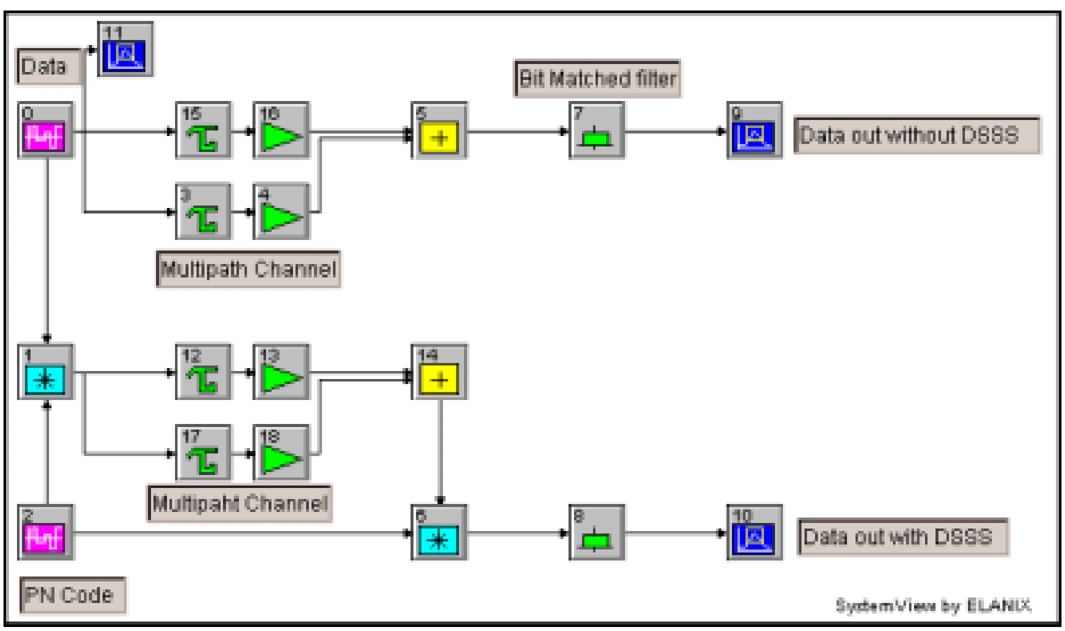


Fig.13 Multipath applications diagram (multipath.svu)

The top graph is the original data that was transmitted NRZ format. The middle one is the recovered signal after the matched filter without the use of PN code. Notice that two graphs do not agree very well. The bottom one is after the matched filter when the DSSS is applied to the transmitting process. Notice the near agreement between the recovered and the original signal in this case.

Note that in this system the delay and the gain are set in favor of the bouncing signal and the original signal are nearly cancel each other.

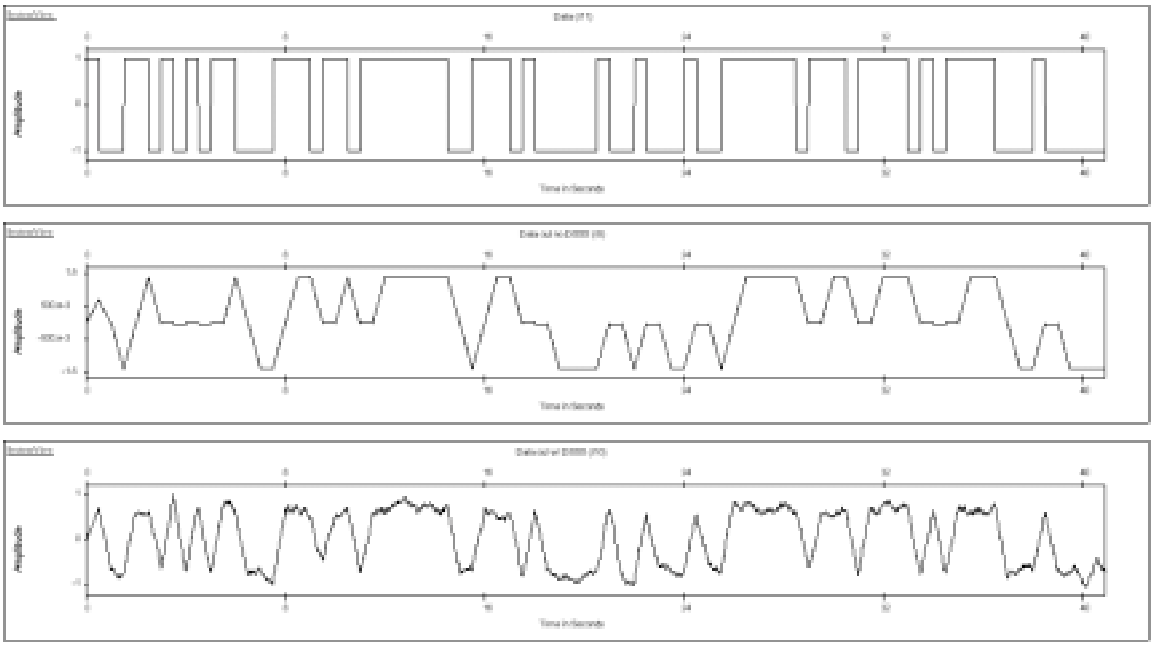


Fig.14 (a) Input data (b) Output data w/o DSSS (c) Output data w/ DSSS

3.2 Application to Narrowband (NBI) and Wideband (WBI) Interference

In this simulation, the DSSS signal is corrupted by AWGN and a tone jammer. The block diagram is shown in Fig.15 with normalized sampling frequency. The parameters are specified in the same picture. By the result, it shows that that pre- and post-despread power spectrum is shown in Fig.16. We can see that prior to dispreading the signal is almost totally immersed in the additive noise. Following the dispreading, the signal spectrum is compressed by a factor of 10 dB. Hence, the PSD increases allowing the main lobe to push out of the noise and the NBI.

<u>Fig.15 :SystemView file: dsss_1.svu (parameters in the figure)</u>

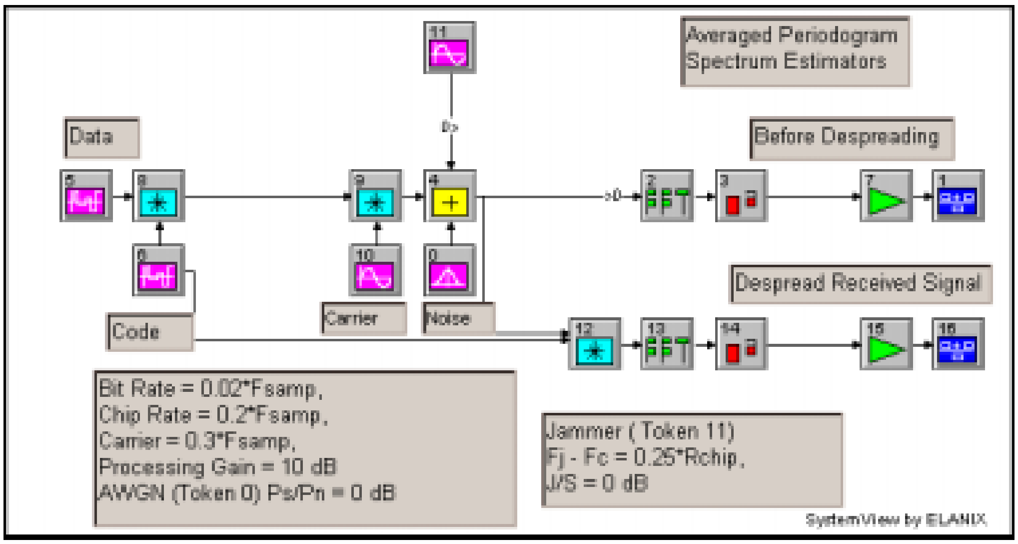


Fig.15 DSSS with the present of AWGN and NBI (dsss_1.svu)

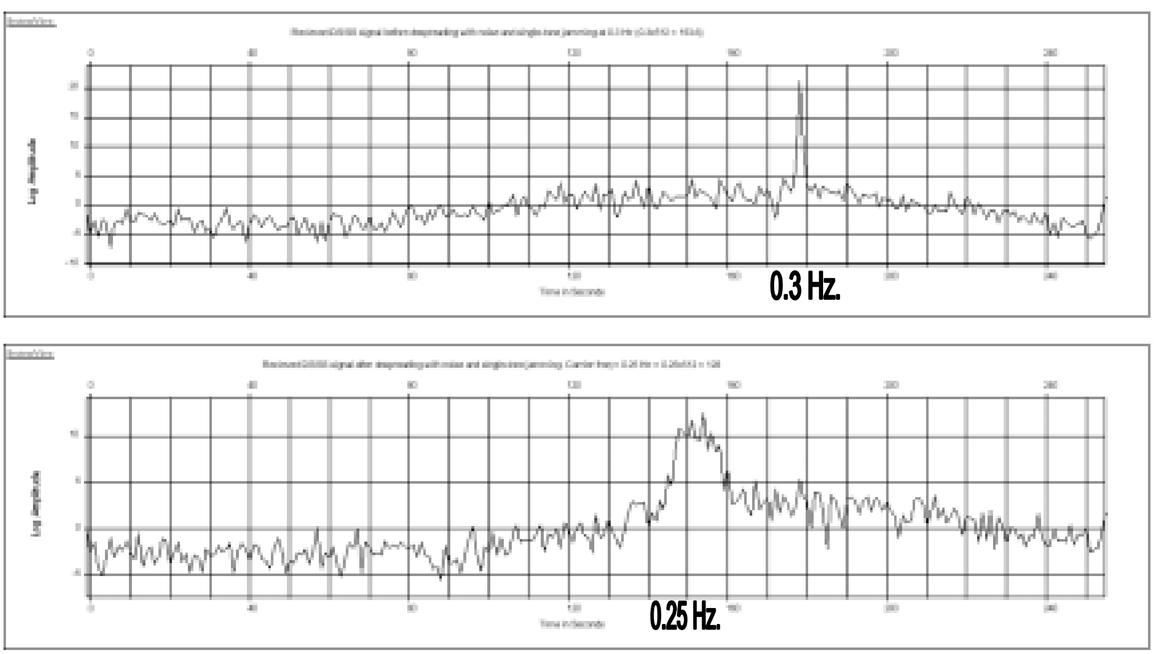


Fig.16 (a) Signal before despreading (b) after despreading

The next simulation provides an application of DSSS to narrowband interference such as a tone jammer. We have two systems running side by side, with and without the DSSS system. The impairment in the NBI and again we use matched filters to recover the data at the end.

<u>Fig.17 ; SystemView file: NBI.svu (parameters in the figure)</u>

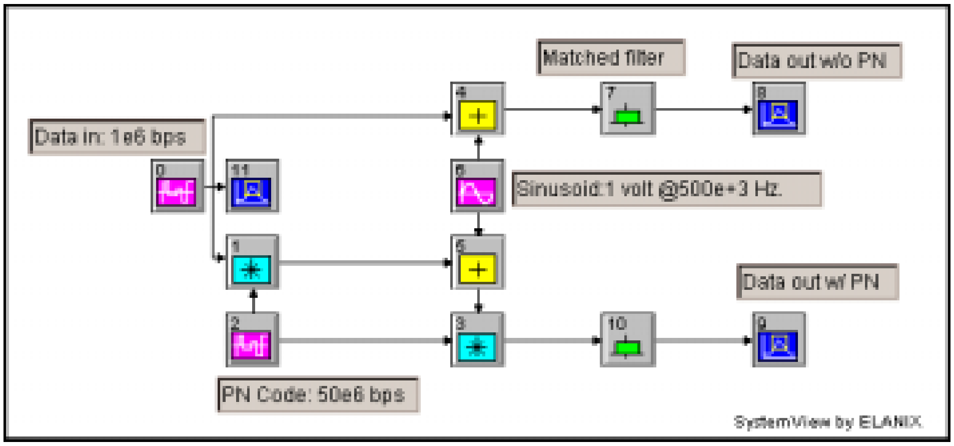


Fig.17. DSSS system with NBI (NBI.svu)

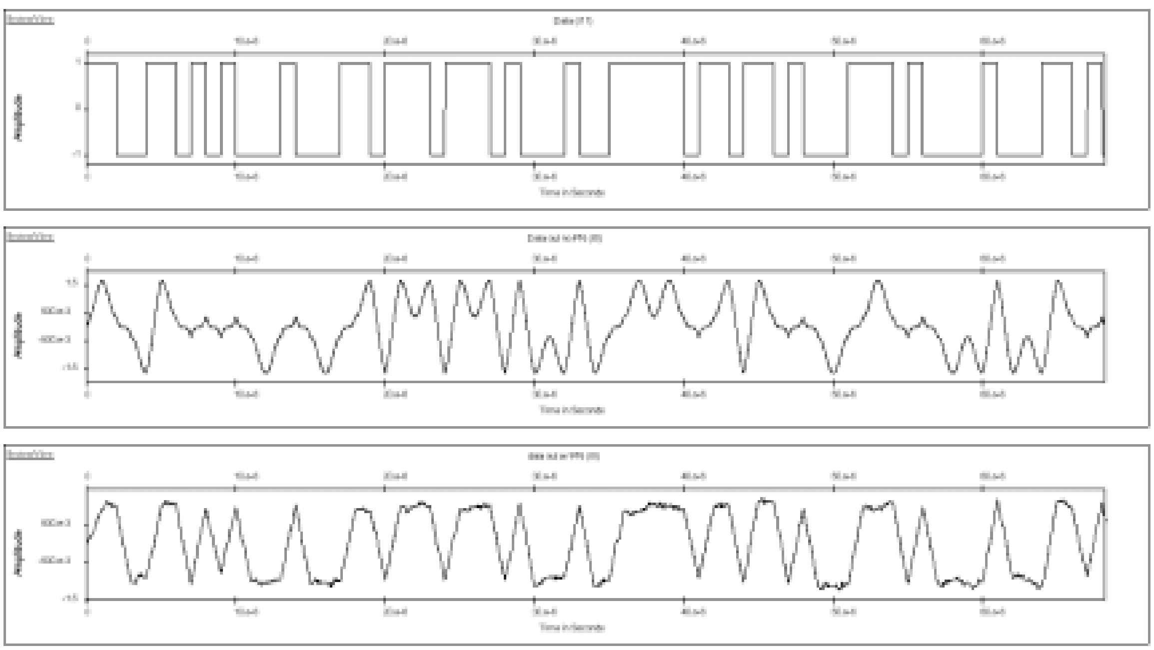


Fig.18 (a) Original data (b) Received signal w/o DSSS (c) with DSSS

Graphs show the results of matched filters to the two systems with and without the spread system. In Fig.18 (b) traces without the DSSS and we can see the ripple due to the oscillations of the NBI or the tone jammer still getting through the matched filter. Fig.18 (c) shows the data very neatly recovered using the DSSS.

3.3 Application to CDMA and Multi-Access

An important issue in wireless communication systems is multiple random access: communication link can be activated at any moment while several links can be active simultaneously. As multi-access and random-access are properties mainly determined by the chosen data communication technique it is important to keep these requirements in mind from the very beginning.

Code Division Multiple Access (Spread spectrum) is one of possible concepts to realize multiple-access other than FDMA and TDMA. In CDMA, a unique code is assigned to each user. This code is used to "code" the data message. AS codes are selected for low cross-correlation properties, all users can transmit simultaneously in the same frequency channel while a receiver is still capable of recovering the desired signal. Synchronization between links is not strictly required and so random-access is possible. A practical application at the moment is the cellular-cdma phone system IS-95. The simulation block diagram in Fig.19 shows a possible way to describe CDMA.

<u>Parameters of the system in the Fig.19 , SystemView file: CDMA.svu</u>

Data rate: data 1 = data 2 = 1 MHz., Code: PN 1 = 50 MHz., PN2 = 70 MHz. and matched filters' average windows are set in correspond to the data rate = 1e-6 seconds.

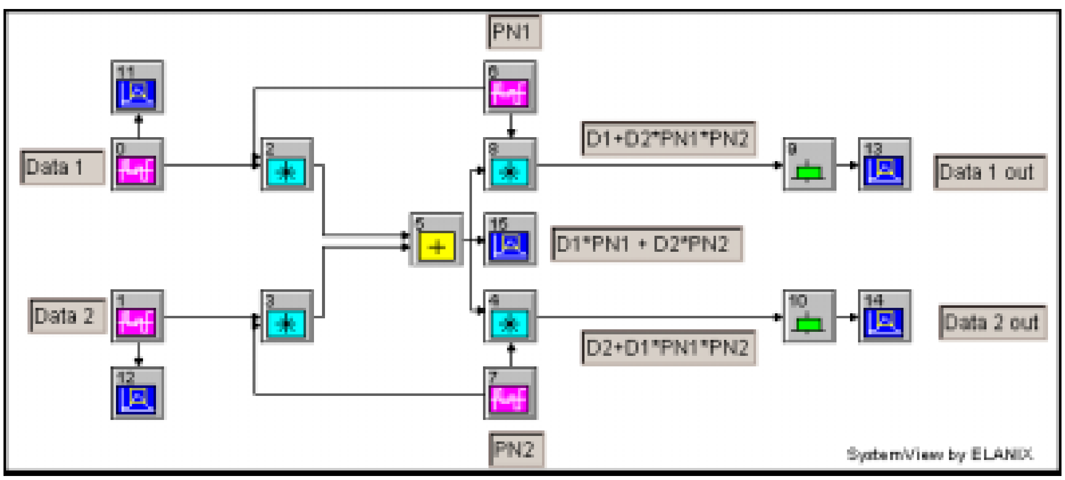
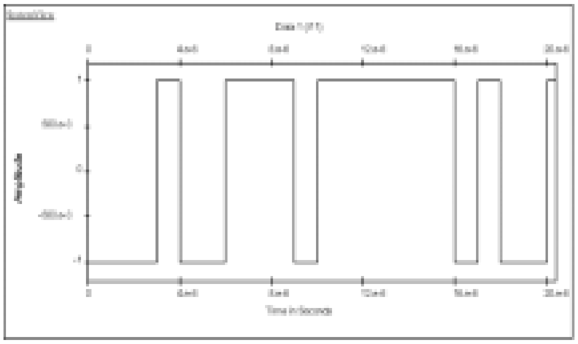
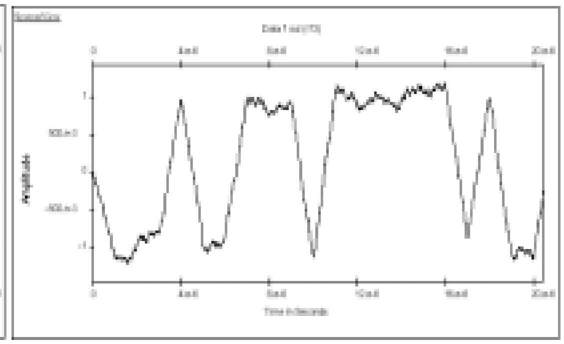


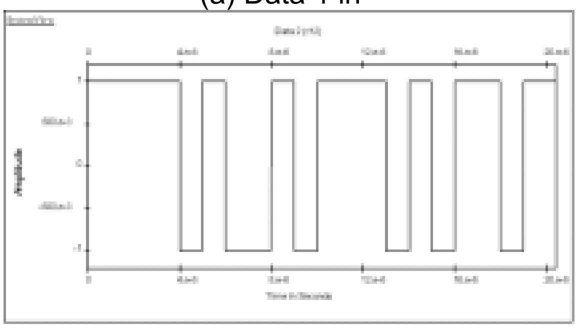
Fig.19 CDMA Application using DSSS (CDMA.svu)

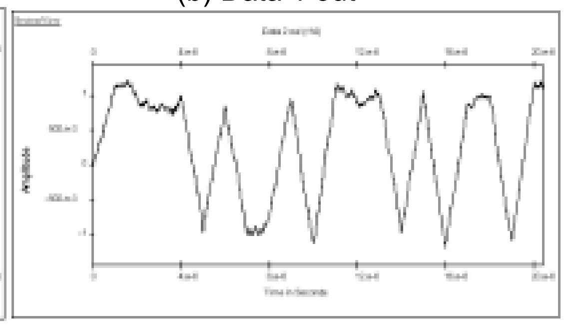
The result is shown in Fig.20. The left column is the original data. The right column is the waveforms of recovered signals of each user after despreading and eliminate the other signal through matched filtering. Clearly, we can see that the recovered signals resemble the original signal.





(a) Data 1 in (b) Data 1 out





(c) Data 2 in (d) Data 2 out

Fig.20 Data and recovered data by multiple access technique

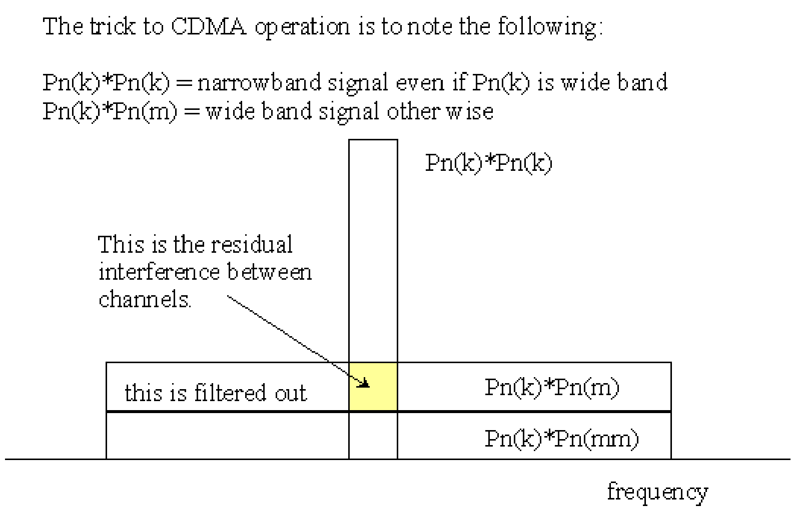


Fig.21 CDMA processing

Fig.21 shows the essence of CDMA processing with respect to a given user. The trick is to notice two facts. When the user as receiver multiples by his own code, the product of his code times the code of his intended target collapses the bandwidth because PN times PN is equal to 1. However, that same code when multiplied by the other user's code does not collapse (they are uncorrelated), that signal is still a wide band signal. As shown in the graphic, then you have a narrow band signal and a residual wide band signal. Subsequent matched filtering then recovers the signal that you desire.

We described CDMA or Code Division Multiple Access systems as one where more than one user is on the air at the same time. Each user has its own strength A and its own pseudorandom code PN. For two users as shown by the lower equation, the received signal is just the linear sum of the two individual components. As described, the receiver multiplies by his own PN and correlates to recover his signal. The result is his original signal plus a term that is a cross correlation between the original signal and the user from the other end. What we want to do is make that second term as small as possible. The name of the game is to make the product A_2, which is the amplitude of the second signal times R_{12} which is the cross correlation between the two codes, as small as possible.

$$S(t) = A_1 PN_1 + A_2 PN_2$$

$$\overline{S(t) PN_1} = A_1 N + \overline{A_2 PN_2 PN_1}$$

$$= A_1 N + A_2 N R_{12}$$

$$R_{12} = \frac{1}{N} \sum_{k=0}^{N-1} a_{1k} a_{2k} = \text{cross correlation}$$

TWO CHOICES:

- Make A_2 as small as possible. This is accomplished in IS-95 systems via power control.
- Make R_{12} as small as possible. This is accomplished by choosing codes with good cross correlation properties

Chapter 4: Spread Spectrum Codes

The importance of the code sequence to a spread spectrum communications is defined by the properties of the codes, in which are:

- Protection against interference: coding enables a bandwidth trade, for processing gain against interfering signals.
- Provision for privacy: coding enables protection of signals from eavesdropping to the degree that the code themselves are secured.
- Noise-effect reduction: Error-detection and –correction codes can reduce the effects of noise and interference.

Maximal sequences or maximal codes are, by definition, the longest codes that can be generated by a given shift register or a delay element of a given length before repeating itself. In binary shift register sequence generators, the maximal length is 2^n–1 chips, where n is the number of stage in shift register. A shift register sequence generator consists of a shift register working conjunction with appropriate logic, which feeds back a logical combination of the state of two or more of its stages to its input. Properties held by all maximal codes are:

- The number of ones in a sequence equals the number of zeros within one chip.
- The statistical distribution of ones and zeros is well defined and always the same.
- Autocorrelation of maximal code is –1.
- A modulo-2 addition (exclusive-or operation) of a maximal code with a phase-shifted replica of itself results in another different phase-shifted replica.
- Every possible state of a given n-stage generator exists at some time during the generation of a complete code cycle except all-zeros state does not normally occur and cannot be allowed to occur.

There are several different varieties of spread spectrum codes that could be used. We will list a few here. The most prominent of which are called *Pseudo Noise* or *PN* codes. A variant of those is the Gold codes used in the GPS systems. There are also *Kasami* codes and *Walsh* codes used in the IS-95, which are not strictly spreading codes but are listed here for completion. A subset of the Pseudo Noise codes used in radar is called the *Barker* codes.

4.1 Pseudo Noise Codes (PN)

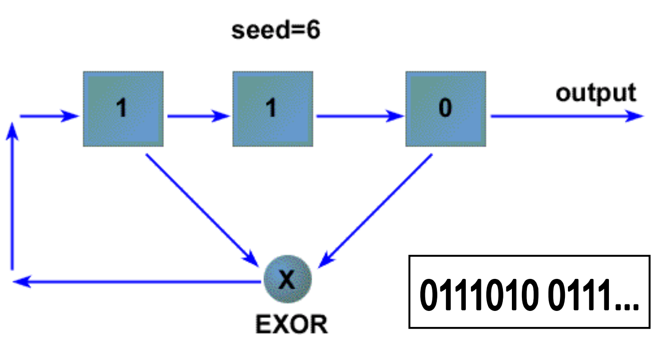


Fig.22 An example of a shift registers

The best way to describe PN codes is via an example. In Fig.22, we show a 3-stage PN generator, where the 3 boxes are shift register sequences initialized to the data [1 1 0], as shown. The output of the first & third generator are passed through an exclusive "or" and fed back to the input. At each clock tick, then the output sequence shown below, [0 111010 011...], is generated. It is easy to show by the example, that after the seventh click, the contents of the shift register are back to [1 1 0] and the process repeats itself again. Any binary code may be represent by a polynomial, where the degree of the polynomial is equal to the number of stages in the generating register. The terms in the polynomial (i.e., $x, x^2, x^4, \ldots, x^n$) represent the stages in the register, and the coefficients that are 0 or 1 determine which stages are include in the feedback. Note that a 1 is always included as x^0 stage with coefficient 1.

In Fig.22,codes are described via polynomial representing feedback connections:

$$\mathbf{p(x)=1+x+x^3}$$

End around auto-correlation: $R(T) = N \quad = 2^n-1 \quad T=0$
$\quad\quad = -1 \quad$ Otherwise,
With no particularly good cross correlation properties.

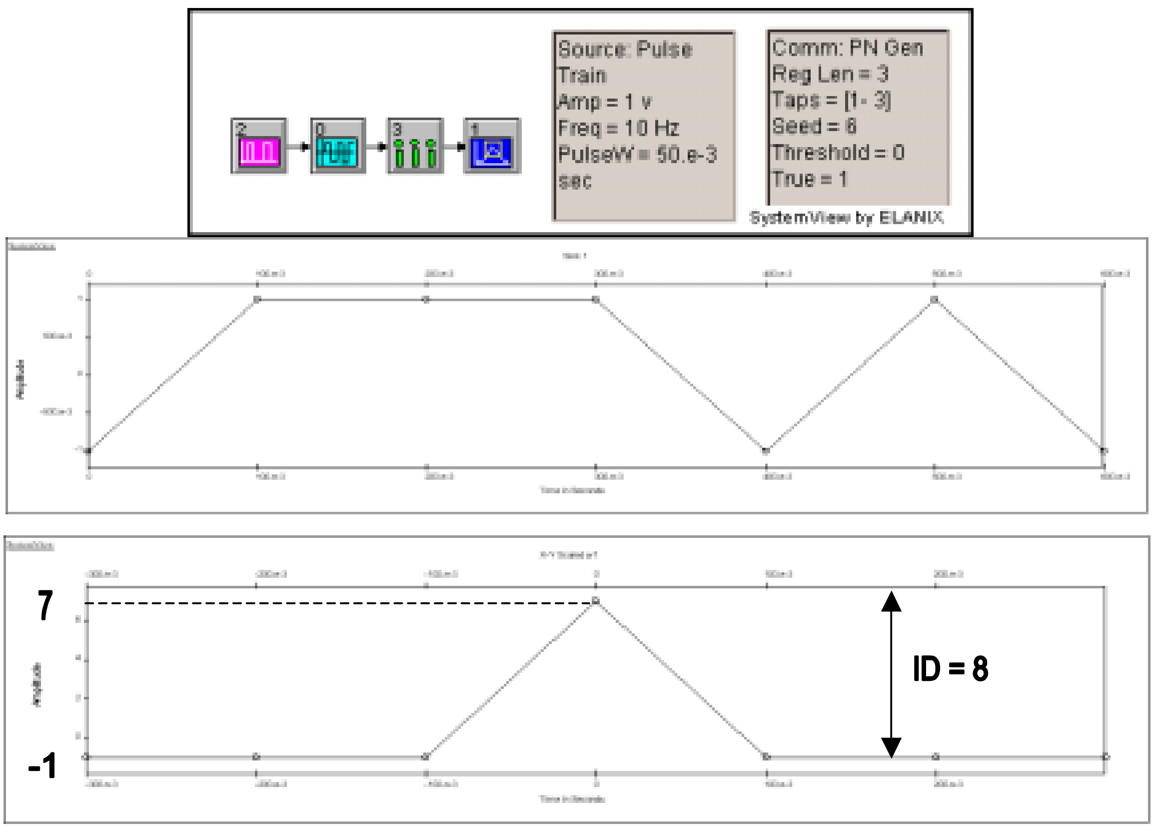


Fig.23 (a) SystemView simulation of Fig.22 (b) output sequence (c) its auto-correlation

As mentioned earlier, PN code has two levels of auto-correlation. In this case the auto-correlation is 7 or –1 and we call the different in correlation as *Index of Discrimination* (ID). In this case, $ID_{auto} = 8$.

4.2 Gold Codes

In general, PN codes have poor cross correlation properties. However, Gold and Kasami have shown PN code pairs exist that have a three-level cross correlation function, which is (-1, -t(m), t(m)-2), where:

$$t(m)=2^{(m+1)/2}+1 \quad m \text{ odd}$$
$$t(m)=2^{(m+2)/2}+1 \quad m \text{ even}$$

Such code pairs are called *preferred sequences*.

As we just said, ordinary maximal length sequences do not provide good cross correlation properties to make them very useful for CDMA applications. However, both Bob Gold & Kasami have shown that there are pairs of cross correlation PN codes that have three-level cross correlation functions defined by -1, t(m), t(m)-2, where m is the number stages of the shift register sequence. These sequences are called preferred pairs and give a much better cross correlation than the direct PN sequences.

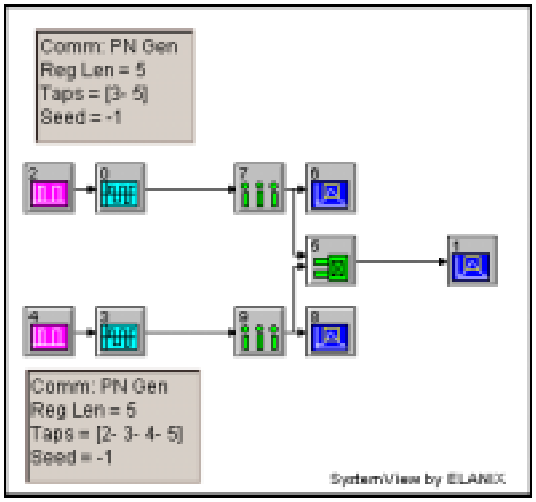


Fig.24 A set of two m-sequences to generate gold codes (goldcode.svu)

Shift		Agreement (***A***)		Disagreement (***D***)		**A - D**	
0	16	15	15	16	16	-1	-1
1	17	19	15	12	16	7	-1
2	18	15	11	16	20	-1	-9
3	19	11	11	20	20	-9	-9
4	20	19	11	12	20	7	-9
5	21	15	19	16	12	-1	7
6	22	19	11	12	20	7	-9
7	23	15	15	16	16	-1	-1
8	24	15	19	16	12	-1	7
9	25	15	19	16	12	-1	7
10	26	15	15	16	16	-1	-1
11	27	19	15	12	16	7	-1
12	28	19	15	12	16	7	-1
13	29	15	15	16	16	-1	-1
14	30	15	19	16	12	-1	7
15	31	19	15	12	16	7	-1

Note: The term ***A – D*** represents the cross correlation.

Fig.24 shows the simulation of two sequence generators with different (shown in the picture). Those sequences are [1111100011011101010000100101100] and [1111100100110000101101010001110] with respect to tap [3,5] and tap [2,3,4,5].

In this system, m = 5 and t(m) = 9 so the possible level of cross correlation is [-1, -9, 7]. The upper table shows all the cross correlation (circular) of the preferred sequences. The results of this simulation are depicted below in Fig.25.

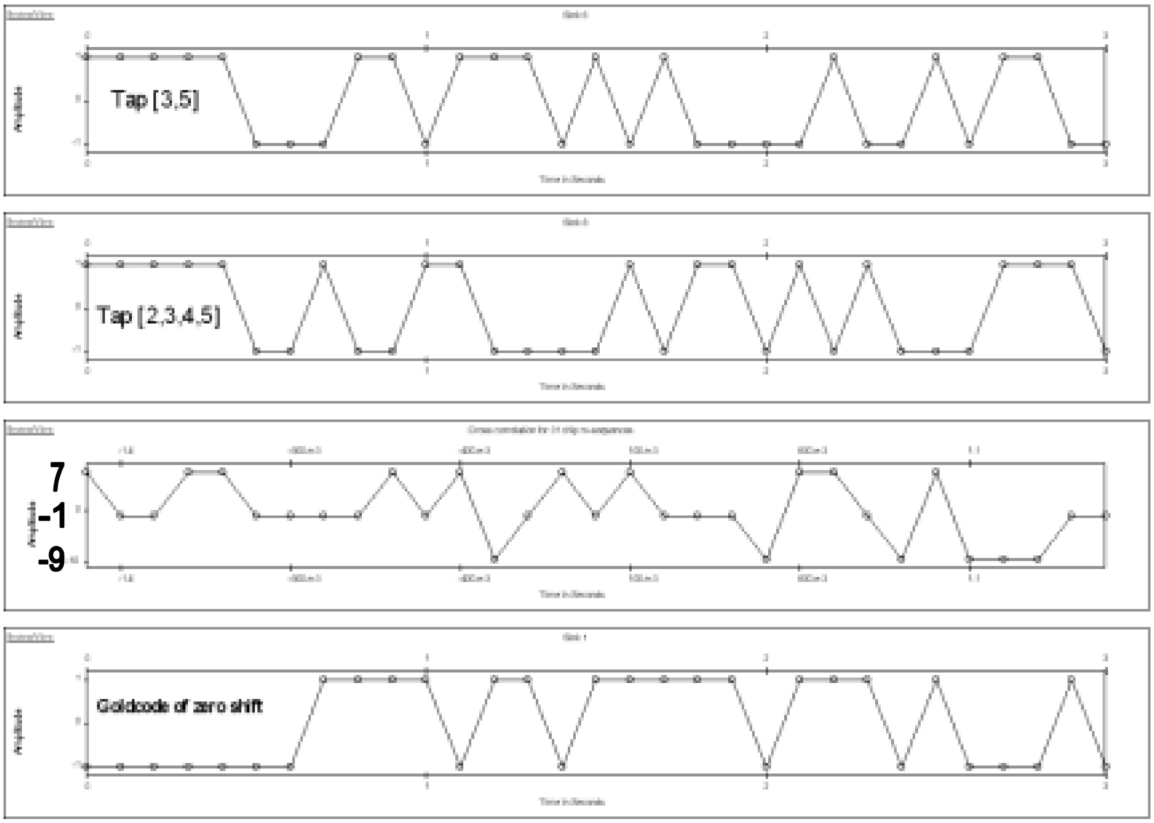


Fig.25 (a) sequence of tap [3,5] (b) tap [2,3,4,5] (c) cross correlation (d) Gold code

Continuing the idea of preferred pair codes, we would generate Gold codes at the end. We take two of the maximal length codes that are part of the preferred pair set and exclusive-or them together as shown in the block diagram. What we then do is we hold the initial conditions of one of the codes fixed while shifting the initial condition of the other code, thus rotating the second code with respect to the first code. Each of these shifts produces another member of the Gold codes set, as shown in the diagram for a five-stage maximal length sequence gold pair. Gold codes of 1023 or 10 stages are used in the GPS system. That is Gold codes are useful because of large number of codes they supply although only one pair of tap set requirement. For instance:

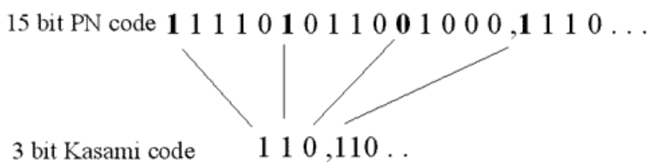
Zero-shift combination	Five-shift combination
1111100011011101010000100101100	1111100011011101010000100101100
1111100100110000101101010001110	0010011000010110101000111011111
0000000111101101111101110100010	1101111011001011111000011110011

This table summarizes the property of the Gold codes that we just described versus the standard PN maximal length codes. The first two columns are the number of shift register sequences and the length of the PN code maximal length. The third column is just a number of possible codes that are in existence. The fourth column lists the maximal cross correlation peak for standard PN code. The next column is the normalized cross correlation peak to the size of the code. The second to last column is the value t(m) that we described a minute ago, which describes the preferred pair. And the last column is the cross correlation between preferred pairs. Compare then, the size of the last column, which would be the R_{12} for Gold Codes with the second-to-last column, which would be the R_{12} or the standard maximal length sequences. You can see how much better the Gold code type sequences are.

m	n	No. of m sequences	Peak Cross Corr. ϕ_{max}	ϕ_{max}/n	t(m)	t(m)/n
3	7	2	5	0.71	5	0.71
4	15	2	9	0.60	9	0.60
5	31	6	11	0.95	9	0.29
6	63	6	23	0.36	17	1.27
7	127	18	41	0.32	17	0.13
8	255	16	95	0.37	33	0.13
9	511	48	113	0.22	33	0.06
10	1023	60	383	0.37	65	0.06
11	2047	176	287	0.14	65	0.03
12	4095	144	1407	0.34	129	0.03

4.3 Kasami Codes

$$2^m - 1 = \left(2^{m/2} - 1\right)\left(2^{m/2} + 1\right) \text{ integers if } m \text{ is even}$$




```
1 1 1 1 0 1 0 1 1 0 0 1 0 0 0     1 1 1 1 0 1 0 1 1 0 0 1 0 0 0
1 1 0 1 1 0 1 1 0 1 1 0 1 1 0     0 1 1 0 1 1 0 1 1 0 1 1 0 1 1
------------------------------    ------------------------------
0 0 1 0 1 1 1 0 1 1 1 1 1 1 0     1 0 0 1 1 0 0 0 0 0 1 0 0 1 1
```

Kasami codes are based on PN codes of length $L=2^m-1$ where m is an even integer. In this case, L is the difference of two squares where the two terms are both integers. Thus for m=4, 2^m-1 is equal to 15, which is the product 5x3. As shown in the diagram, we take the 15th stage PN code and take every 5th bit (1, 1, 0) and keep repeating it (1, 1, 0, 1, 1, 0, 1, 1, 0..). We then play that code against the original PN code and "exclusive or" them together bit by bit, much like we did with the Gold code as shown in the figure. We then, shift the shortened code by 1 bit and produce another member of the set. Kasami codes are used but they are not all that common.

4.4 Walsh Codes (IS-95)

Matrix Representation

$$H_2 = \begin{vmatrix} 0 & 0 \\ 0 & 1 \end{vmatrix}$$

$$H_4 = \begin{vmatrix} H_2 & H_2 \\ H_2 & \overline{H}_2 \end{vmatrix}$$

Orthogonality

$$< H_m(k) H_m(p) >= 0 \qquad k \neq p$$

Walsh codes are not strictly maximal length or PN type codes for spread spectrum operation. They are heavily used however for different reasons in the IS95 CDMA type system. The idea is to start with a 2x2 matrix, called H2, which represents two codes, the top row being H2[0] = [0,0] and the bottom code being H2[1] = [0, 1]. To build up a larger code, the individual matrixes are increased so that the matrix H2, H2, H2, and H2 inverse then build H4. This idea can be increased to get larger and larger code sets. The basic idea though is that each row of this matrix represents a string of bits 1s and 0s that are orthogonal to each other, as shown that *Hm*[*k*] • *Hm*[*p*] = 0 unless k is equal to p.

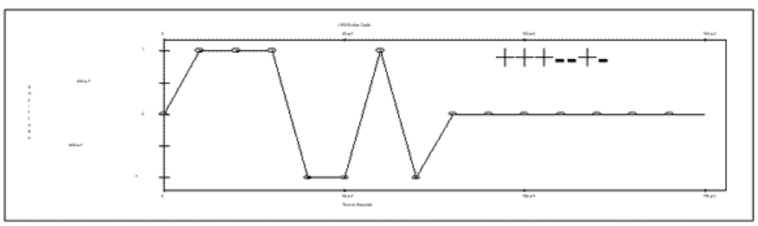
4.5 Barker Codes

Barker codes were originally developed for radar.
They are short codes, the longest being 13 bits.

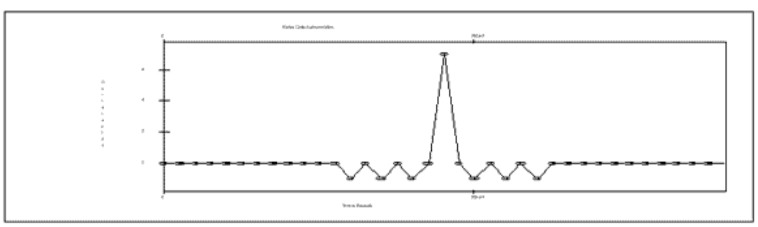
N		
2	+ -	or + +
3	+ + -	
4	+ - + +	or + - - +
5	+ + + - +	
7	+ + + - - + -	
11	+ + + - - - + - - + -	
13	+ + + + + - - + + - + - +	

One shot correlation sidelobes: 0, -1

Barker codes were originally developed for radar. They're actually a subset of a PN but they are short codes with a length up to 13 and the codes are listed above. The property that makes them popular for radar is what is called the one shot correlation, not the circular correlation. This function has side lobes (correlation coefficients) of 0 and minus 1. This is not the case of PN codes in general.



Barker Code



Autocorrelation

Fig.26 Barker code and its auto-correlation

This shows a basic 7-bit Barker code (+ + + - - + -) and the auto-correlation on a one shot basis which shows the main peak at the center and a 0 and -1 cross correlation. By one shot, we mean that the data comes in to the match filter and goes out the other end. The PN have -1 cross correlation properties but that is end around (circular) where the end bit is pushed back into the first in a circular fashion. These are two different types of correlation operations.

Chapter 5: Synchronization

Code synchronization is necessary in all spread spectrum systems because the code is the key to dispreading the desired information and to spreading any undesired signals. The overall requirements of frequency hopping and direct sequence systems with regard to synchronization are similar, but somewhat more latitude is possible in the frequency hopping. First of all, even if we know exactly what time the transmitters are synchronized to and can keep that time synchronization, the time delay due to the transit distance puts an offset that must be search out if we do not know the relative distance. Actually, FH systems are easier to synchronize than PN systems by orders of magnitude. This is due to the great different between hop rates (1000h/s) and PN chip rates (Mbps).

The "Sliding" Correlator

The simplest of all correlation techniques uses a "sliding" correlator, so called because the receiving system, in searching for synchronization, operates its code-sequence generator at a rate different from the transmitter's code generator. The effect is that two code sequences slip in phase with respect to each other. Fig.27 shows the SystemView block diagram used to implement the sliding correlator.

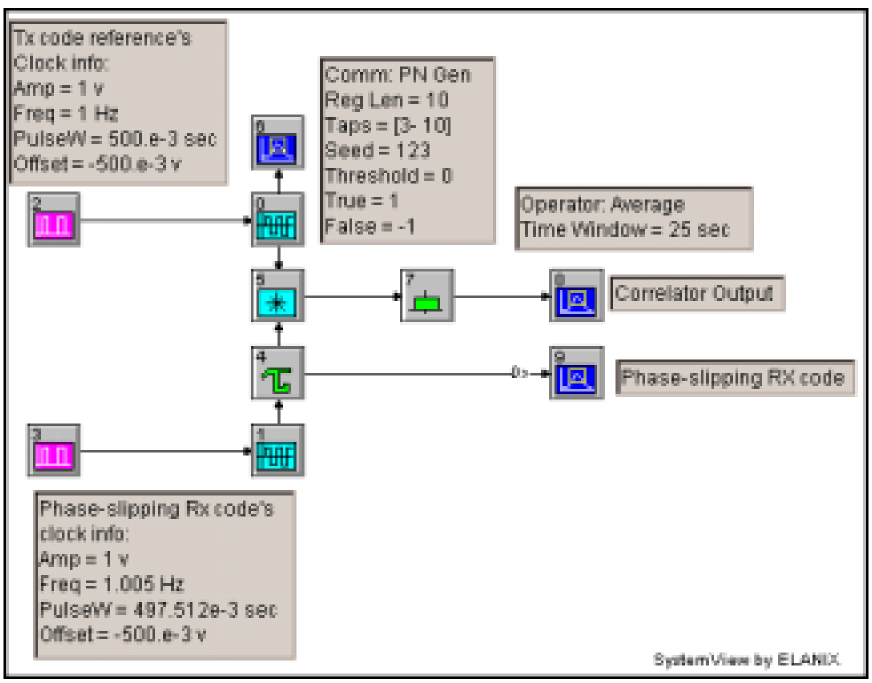


Fig.27 A Sliding corrleator (nsync.svu)

<u>Fig.27: SystemView file: nsync.svu (parameters in the picture)</u>

At the receiver, the receiver PN generator is initially set in a state that is outside the maximum uncertainty that it expects to find. This PN generator then is driven in a rate slightly higher than the nominal rate. Eventually, the received PN correlator catches up with the transmitted code and a correlation peak comes up and a signal is detected.

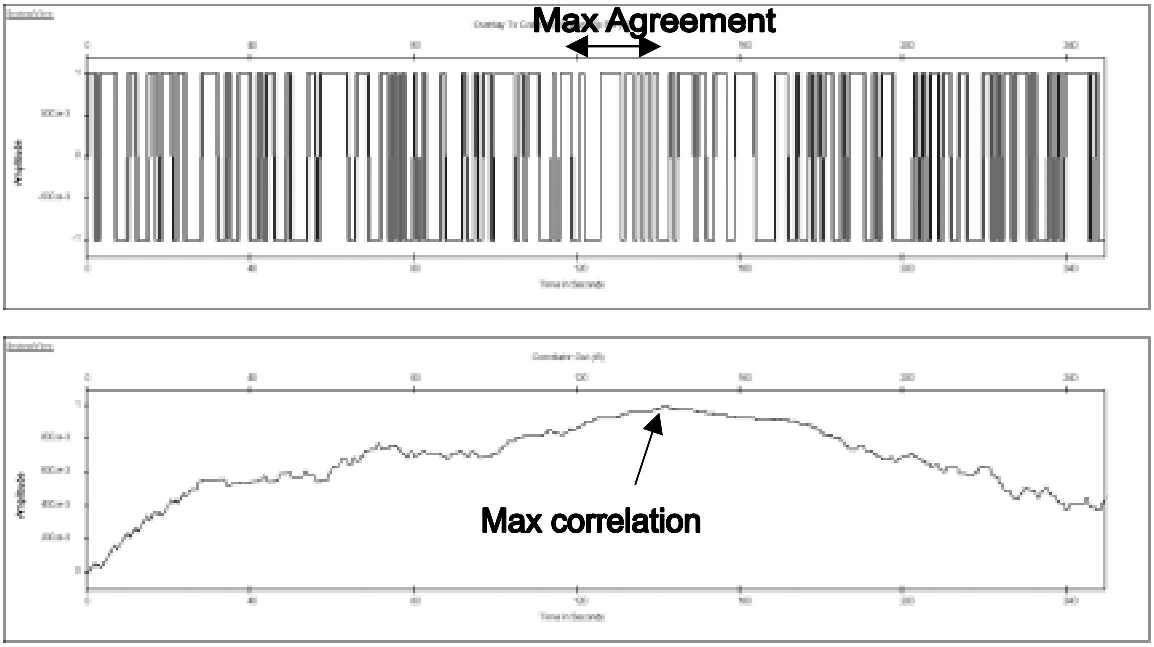


Fig.28 (a) overlay of Tx code and phase-slipping Rx code (b) correlator output

The top graph is the overlay of the transmitted code and the sliding PN code. Notice in the beginning and at the end, the two waveforms do not look like each other and are sort of offset. In the middle where the times match up, we have a nice clear overlay and the two agree. The graph below shows a correlation by averaging the correlation output over short periods of time (25 seconds in this simulation), and you can see the energy build up as the correlation comes in and then dies out as the correlation moves out.

The advantage of the sliding correlator is its simplicity in that nothing more is required than some way, on demand, of shifting the code clock of the receiver to a different rate. The difficulty for synchronization, however, is that when a large degree of uncertainty is encountered (both code-phase and code-frequency jitter), examination of all code-phase position is impractical because of the time involved. Recognition of synchronization, which must occur to stop the sliding or searching process at or near the point of synchronization, is limited in response time by the bandwidth of the system's post correlation receiver. It is possible under some circumstances for areceiving code-sequence generator to be offset by trillion of chips, which hours, days, or even years might be consumed in the initial synch process if the sliding correlator alone were used. As a rule, one can expect to search at a rate approximately equal to the data rate for which the receiver has been designated.

Bibliography

[1] Robert C. Dixon, "*Spread Spectrum Systems with Commercial Applications*," John Wiley & Sons Inc., Third Edition, 1994. ISBN 0-471-59342-7.

[2] B. P. Lathi, "*Modern Digital and Analog Communication System,*" Oxford University Press Inc., Third Edition, 1998. ISBN 0-19-511009-9.

[3] Alan V. Oppenheim, Ronald W. Schafer, "*Discrete-Time Signal Processing*," Prentice-Hall, Inc., Englewood Cliffs, NJ, International Edition, 1989. ISBN 0-13-216771-9