Ombega Edwin Matoke

ACKNOWLEDGEMENT

I give thanks and all glory to almighty God for his sufficient mercy, love grace and favor .He has also given me good health and wisdom during the study, design and implementation of this project. With him all things are possible

My sincerest appreciation to my project supervisor Dr. Vitalice K. Oduol who guided me in every step of this tedious process and provided me with useful information and motivation that helped me a great deal to shape this project

Also I thank all my lecturers, especially my examiner Dr. GSO Odhiambo, classmates and lab technicians for their support.

Lastly I would like to thank Timothy, Asenath, Raymond, Grace and Alfayo Ombega my entire family for their immense support, prayer and motivation. May God bless you abundantly.

DEDICATION

This project is dedicated to

My loving mum Grace , Dad Alfayo Bro Raymond, Job and Timothy being confident of this very thing that he who has begun a good work in you will carry it unto completion

ABSTRACT

Frequency hopping is the periodic changing of the carrier frequency of a transmitted signal. It is one of the spread spectrum modulation techniques in which a spectrum of data modulated carrier is widened by changing the carrier frequency in a pseudo random manner. Spread spectrum is signaling scheme in which the transmitted signal occupies a bandwidth that is much larger than the minimum typically required to send the information. The band spread is accomplished by means of a code which is independent of the data, and synchronized reception with the code at the receiver is used for dispreading and subsequent data recovery.

In frequency hopping spread spectrum (FH-SS) communication system, perfect communication succeeds only when the locally generated code sequence at the receiver is synchronized to the code sequence at the transmitter generator.

For synchronization, FH-SS receiver requires a replica of the code, with which the correct clock phase, in order to despread the signal. Typically this code must be derived from the received information signal. The process of synchronizing to the transmitter code consists of two steps.

Acquisition, also called coarse synchronization, involves searching throughout a region of time and frequency (chip carrier) in order to synchronize the received spread spectrum signal with the locally generated PN sequence.

Tracking or fine synchronization continuously maintains the best possible waveform alignment by means of a feedback loop.

In this report a detailed description of the frequency hopping spread spectrum (FH-SS) synchronization process (code acquisition and tracking) and ultimately a design of a synchronizing system frequency hopped (FH) signals. The synchronization system has been modeled and implemented using MatLab Simulink. Simulink is a model for simulating dynamic systems. It provides various tools one of which being the communications block set has been used in this project.

TABLE OF CONTENTS

CHAPTER ONE

1.0 INTRODUCTION

1.1 PROJECT OBJECTIVE

To describe the synchronization of FH spread spectrum communication systems. Design a synchronization circuit/ system and demonstrate that it works

1.2 SPREAD SPECTRUM COMMUNICATION SYSTEMS

Spread spectrum techniques have been extensively studied and successfully applied over the last fifty years. At initial stages of development, mainly due to their inherent resistance to intentional interference as well as their low probability of interception, spread spectrum techniques were firs considered for military communication. Spread spectrum can be found in present day and next generation mobile land communication systems (e.g. IS-95, Wideband-CDMA, cdma 2000) as well as in military communication systems, wireless LAN applications (e.g. IEEE-802.11, Hiperlan), wireless local loops and satellite communication systems. Today, in addition, spread spectrum systems are also found in a variety of other applications, such as radar and navigation systems, including the Global Positioning system (GPS) and the European Global Navigation Satellite System (GNSS), better know as Galileo.

Using spread spectrum techniques a signal that has already been modulated (whether by AM, FM or PM) is modulated again so that is spread over a much wider range of frequencies in such a way it does not interfere with any other signals that are using the same frequency band. The desired signal and the interfering signals are transparent to each other. If an AM signal with a bandwidth of 10 kHz is spread out to 1MHz, using the same total power, then only 1% of the new signal is in the old band and it has the power level 20dB down on the AM signal. A receiver is set to recover the AM signal will see the component of the spread signal as noise

In the design of a communication system, efficiency with which the system utilizes the signal energy and bandwidth are of prime importance. However, in some cases, it is necessary for the system to offer privacy and high security of information resisting external interference; operate at low power spectral density; and provide multiple access capability enabling a large number of users to share the same channel without external control. Communication systems following these concepts are ñspread spectrum systems.ò

The principle of spread spectrum communication is that the frequency spectrum of a data signal is spread using a code uncorrelated with that signal and as a result the bandwidth occupancy is much higher than the minimum required to transmit the signal.

As a digital signal modulation and multiplexing technique, spread spectrum (ss) technology spreads a transmitted radio signal across a frequency band(fig 1) wider than the minimum bandwidth required for transmission. Spread spectrum allows several users to transmit and receive on the same frequency within a given geographical location. By distributing a signal over a wide bandwidth, spread spectrum techniques allow more signals to be packed into a given bandwidth concurrently without interference from one another. Since the spread spectrum signal is spread across such a wide bandwidth, it appears as noise to a narrow band receiver.

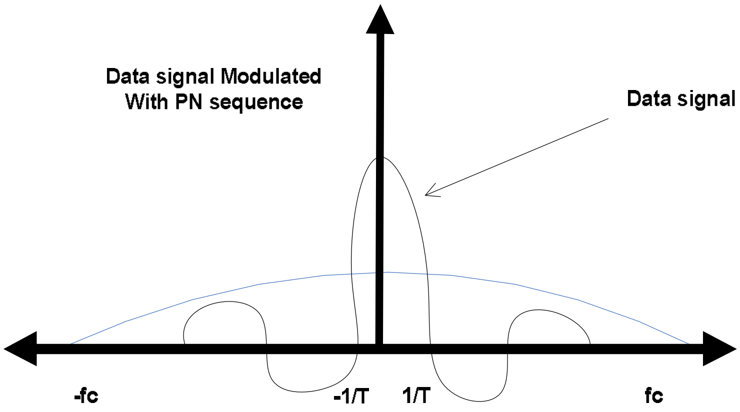


Figure 1: Data Signal and PN Modulated Signal In Frequency Domain.

A popular method for spreading is to multiply the user data signal with a fast code sequence, which mostly is independent of the transmitted data message. In particular, multiple users can share the same portion of the radio spectrum provided using distinct codes to distinguish their transmissions, thus achieving code division multiple access (CDMA).

One important measure of spread spectrum communications system is the processing gain, which is approximately the ratio between the bandwidth of the user signal and the transmitted bandwidth which is basically known as óspreading factorô. The processing gain determines the number of users that can be allowed in a system, the amount of multi-path effect reduction, the difficulty to jam or detect the signal etc.

For spread spectrum systems it is advantageous to have a processing gain as high as possible.

The distinct features of a spread-spectrum system are the two identical pseudorandom pattern generators: one interface with the modulator at the transmitter and the second interfaces with the demodulator at the receiver. These two generators produce a pseudo random or pseudo noise (PN) binary sequence. The PN sequence is used to spread the frequency band of the transmitted signal at the modulator and to dispread the received signal at the demodulator. The time synchronization of the PN sequence generated at the receiver with the PN sequence embedded in the received signal required for proper dispreading.

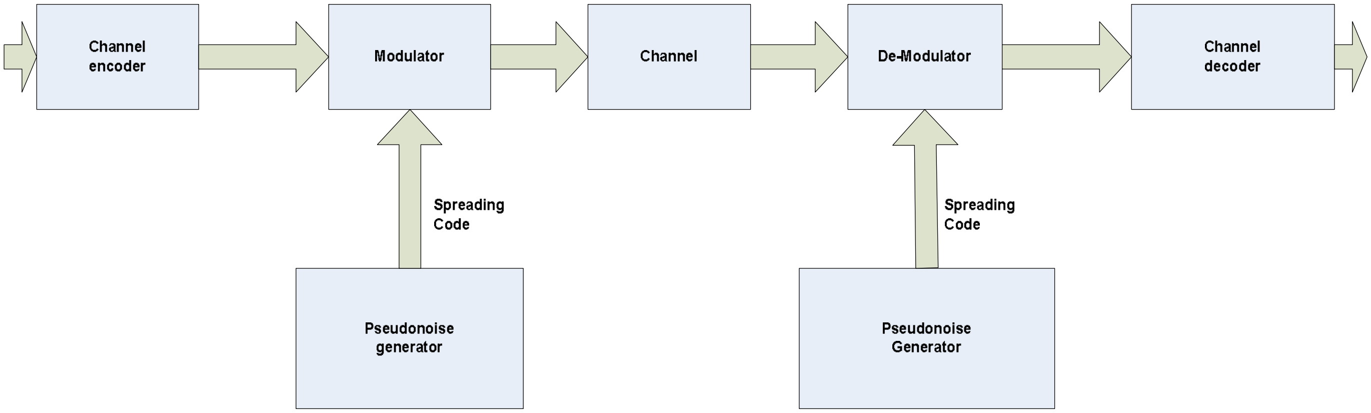


Figure 2: Spread Spectrum Communication System.

Under less ideal circumstances spread-spectrum techniques can have he following advantages over conventional modulation methods:

1) ***Low detect ability***: as the signal is spread over a large frequency band, the power spectral density becomes very small. This makes the signal difficult to detect as it submerges in the floor noise. Other communication systems may not suffer from this kind of communications.
2) ***Resource sharing***: multiple access or random access schemes, such as code division multiple access (CDMA), can be effectively implemented using spread spectrum technologies. In a nutshell, a large number of codes can be generated, so a large number

of users can be permitted to transmit. This kind of multiple access can operate without centralized control.

3) ***Robust***: spreading and dispreading makes the signal robust against interference. This also holds for multi-path self-interference.
4) ***Secure transmission***: without knowing the spreading code, it is difficult to recover the transmitted data. Moreover, as the spectral density is small, the signal may remain undetected.
5) ***Fading rejection***: as the bandwidth can be made much larger than the coherence bandwidth of the channel, the system is less susceptible to deep fades at particular frequencies.
6) ***License:*** segments of the radio spectrum exempt from federal licensing have been allocated from spread-spectrum communications. The 2.4-2.5 GHz ISM band isa license free band in most countries of the world. The user can operate type approved equipment in this band without a license.

Recognizing these advantages, the FCC in 1985 made a decision to allow the use of spread spectrum signals in the ISM bands with power levels set at 1W maximum. The FCC allows three types of spread spectrum signals in the ISM band, frequency hop (FH), direct sequence (DS). The FCC regulations have no provision for chirp spread spectrum in the ISM bands.

1.2.0 Spread Spectrum Techniques

Some of the modulation techniques employed in spread spectrum include:

1) Direct sequence (DS)
2) Frequency hopping (FH)
3) Time hopping

In time hopping technique one needs memory to store and re-time the received bits to recover the message since only a few or even one time slot per frame is used to carry the message. This demands extra hardware which renders the technique the least feasible means of implementing a spread spectrum modulation system. Standard modulation schemes such as FM and PCM which also spread the spectrum of an information signal do not qualify as spread spectrum.

Two basic types of spread spectrum signals for digital communications are of interest in this report; Direct-sequence spread-Spectrum (DS-SS) and frequency-hopping spread spectrum (FH-SS) techniques. Only frequency hopping spread spectrum (FH-SS) signals and systems are discussed and used for the design of this project.

1.2.1 Direct Sequence Spread Spectrum (DS-SS)

Direct spread spectrum (DS-SS) conducts the spread spectrum modulation before the RF modulation. Analogous to imprinting a digital address on a wireless signal, DS-SS imprints the digital address by logically multiplying the digital information signal by another, higher frequency digital signal referred to as the pseudo random noise, or PN signal. This PN signal provides a random bit stream that appears as noise when modulated onto an RF carrier. In reality, this PN signalôs bit stream pattern repeats itself over time. The 1s and 0s that constitute the PN signalôs bit stream pattern are called *chips*, where the frequency of the PN signal is referred to as the *chipping* rate

With spread spectrum, the original information signal (data signal) is multiplied by the PN signal, resulting in a new signal that spreads over a much larger bandwidth. Figure 3 depicts a block diagram of a generic DS-SS system with the data information processed by the pseudo noise generator.

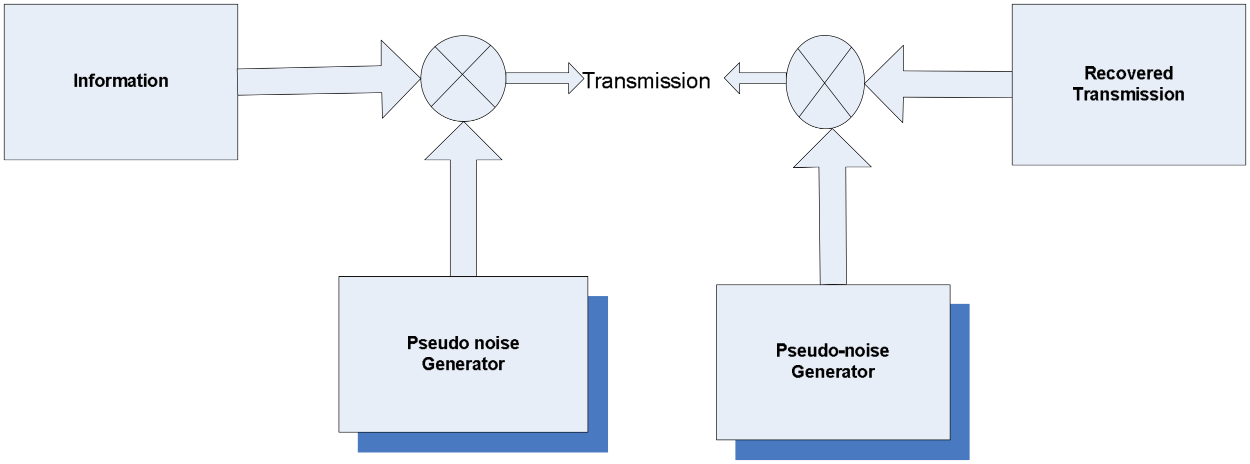


Figure 3: Block Diagram of Direct Sequence System

As the new signal spreads over a larger bandwidth, its power level drops, and the signal appears on a spectrum analyzer as noise. The original signal is retrieved from this noise signal by multiplying the noise signal by the exact same PN signal to restore the original information. When every user (radio terminal) has a different PN code, multiple users can transmit simultaneously.

With DS-SS, the information signal is modulated onto a carrier frequency (typically 2.4 GHz ISM band) with 11 possible channels spaced at 5 MHz using differential quaternary phase shift keying (DQPSK) modulation to provide data rates up to 2 Mbps. This is because the 2.4 GHz band will only effectively accommodate 3 non-overlapping channels, only 3 DS-SS networks can operate collocated. At the DS-SS receiver, the same spreading code is reapplied to the spread received power signal, and the wideband signal is narrowed.

The IEEE 802.11 standard specifies the use of barker codes for the chip sequence used in DS-SS systems. Barker codes are known to possess a good aperiodic correlation property, which simply means that due to the non-repetitive behavior of the code a matched filter correlator can easily identify the location of a barker code in a sequence of bits. The same properties that make barker codes good frame sync markers also make them good PN codes for spreading and dispreading DS signals.

For binary data input the data is multiplied with the PN sequence to produce a signal whose bandwidth is the same as that of the PN code

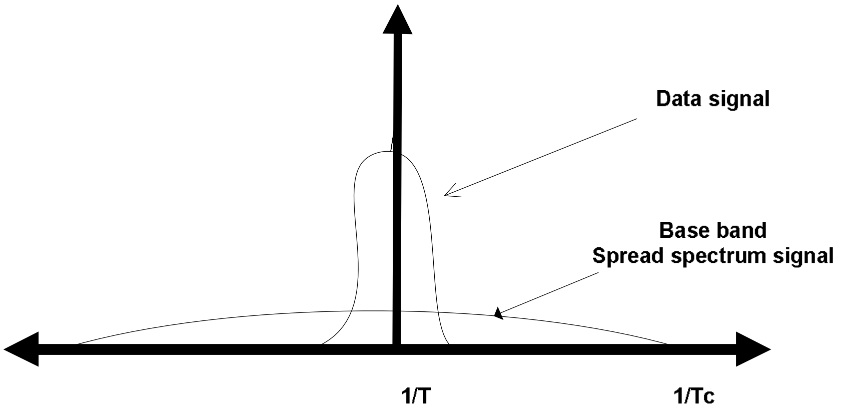


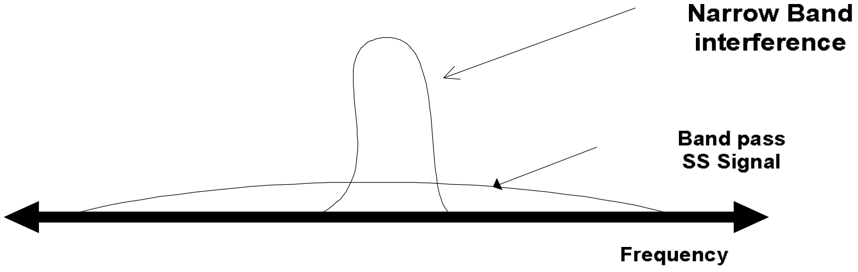
Figure 4: DS spread signal in frequency domain

The DS-SS signal is white noise-like. The amplitude and thus the power in the SS- signal transmitted is the same as in the original signal, due to increased bandwidth of the SS- signal the power spectral density is lower.

The spreading factor, $sf = \frac{BWss}{BW\ \inf}$ where BW_{ss} is the SS-signal bandwidth and the $BW_{\inf}$ is the original data (information) bandwidth.

Ideally, the spreading code should be designed so that the chip amplitudes are statistically independent of one another. The entire period of PN sequence consists of N time chips. In case of maximal linear PN generator, the value on N is 2^n-1, where n is the number of stages in the code generator. Another important reason for using PN generator to modulate a signal is the properties of the resulting signalôs autocorrelation function. It has a maximal value of one repeating itself every period.

In the presence of interference, the received signal Rx consists of the transmitted signal S(t) plus an additive interference I(t) (noise, other users and jammer). To recover the original signal S(t) is multiplied with a synchronized PN sequence. This helps in spreading the interference and thus decreases its power spectral density. If the transmitter PN sequence is not synchronized to the receiver sequence then the signal recovery is impaired.



Despreading at the receiver

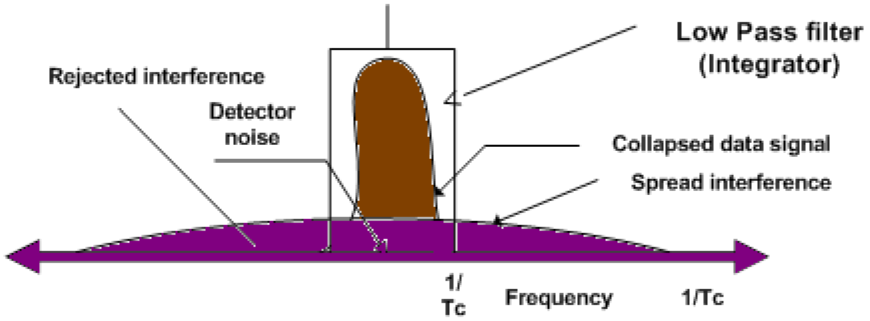


Figure 5: power spectral density: (a) in the channel and (b) at the receiver.

The PN sequence at the transmitter attenuates the signal between -1 and +1 . this attenuation is destroyed if the PN at the receiver is perfectly synchronized. Multiplication of the interfering signal by the PN code generated locally means that the spreading code will affect the interference just as did with the information bearing signal at the transmitter.

After the dispreading the data component is narrow band whereas the interference component is wide band.

1.2.2 Frequency Hopping Spread Spectrum (FH-SS)

Similar to DS-SS, the frequency hopping spread spectrum (FH-SS) uses a PN code. Unlike DS-SS, however, the PN code is not modulated onto the RF carrier. Rather, the PN code is used to determine a sequence of discrete frequencies, which become the RF carrier.

In this fashion, FH-SS forces the RF carrier to constantly hop from one frequency to another within a predetermined frequency range. This frequency hopping occurs in a random pattern with each new hop to a new frequency lasting only a fraction of a second, or dwell time. As with DS-SS , the FH-SS scheme spreads the original information signal over a wider bandwidth. Figure 6 provides a block diagram of a basic frequency hopping system. Note the added complexity associated with this system versus the direct system in fig 3.

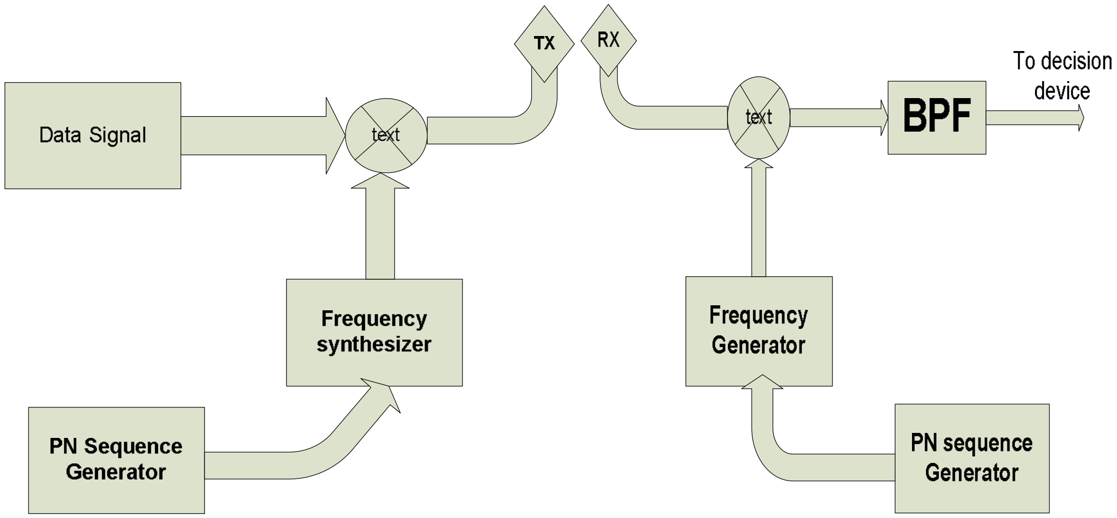


Figure 6: block diagram of frequency hopping system

With FH-SS, the carrier frequency (2.4 GHz with 79 possible channels spaced at 1 MHz) hops from channel to channel in a prearranged pseudo random sequence or ñhop patternò. There are 78 different hop patterns where the transmitterôs carrier frequency changes in accordance with the PN code sequence. The FCC regulations require a minimum hop rate of 2.5 hops/s or a channel dwell time of less than 400ms. Based on the probability of collisions where two FH-SS networks choose the same one of 79 possible channels at the same time, up to 13 FH-SS networks can be collocated without generating significant interference among themselves. A detailed description and analysis of the FH-SS communication system shall be handled in the next chapter

1.2.3 Comparison of DS-SS and FH-SS Techniques

Table 1 summarizes the salient features of DS-SS and FH-SS with the pros and cons for each SS technique denoted by a (+) or (-) respectively.

	DS-SS	**FH-SS**
Spectral Density Interference	**+ Reduced with processing gain** **+ Continuous spread of the T_x**	**+ Reduced with processing gain** **- Only the average Tx power is**

Generation	**power gives minimum interference**	**spread, giving less interference reduction.**
Transmission	**-Continuous, narrowband**	**-Discontinuous, narrowband**
Interference Susceptibility	**+Narrowband interference in the same channel can be reduced**	**- Narrowband interference in the same channel is not reduced** **+ Narrowband interference in a different channel has no influence.**
Modulation	**+BPSK and QPSK are very power efficient**	**-GPSK is less power efficient in narrowband operation**
Multi path	**+Rejection if bandwidth is wider than coherence delay of the environment (outdoor applications)** **-For a chip rate of Mcps, the chip period is 91 ns, corresponding with a wave of distance of about 30m(large for indoor applications)**	**- Some of narrowband channels are unusable** **+ Hopping makes transmission on useable channels possible**
Multiple signals	**-Only 3 collocated networks** **+Higher aggregate throughput**	**+Up to 13 collocated networks** **-Lower aggregate throughput**
Synchronization	**+Self synchronization**	**- Many channels to search**
Real time (Voice)	**+No timing constraints**	**-If a channel is jammed, the next available transmission time on a clear channel may be 400ms away**
Implementation	**-Complex base band processing**	**+Simple analog limiter/discriminator**
Power consumption	**-More power consumption due to higher speed and more complex processing**	**-More simple circuit**

In conclusion, this section can be summarized by noting that the spread spectrum provides improved security, resistance to jamming, efficient bandwidth sharing, and resistance to fading. In a low to medium interference environment, defined by one in which the interfering signal strength is below the jamming margin of the DS-SS, 100% of the direct sequence message will get through. In heavily congested environments where interfering signals exceed the jamming margin of the DS-SS, the direct radio will continue to function until the entire frequency band is jammed. DS-SS allows for higher data rates whereas FH-SS is more resistant to multipath. FH-SS is particularly useful where reliability and data integrity are more important that the timelines of data transmission. With DS-SS, the net throughput will always equal or exceed the FH-SS transmitter- but the expense of data reliability in conditions where hostile environment problems are encountered.

CHAPTER TWO

2.0 LITERATURE REVIEW

2.1 FREQUENCY HOPPING SPREAD SPECTRUM MODULATION

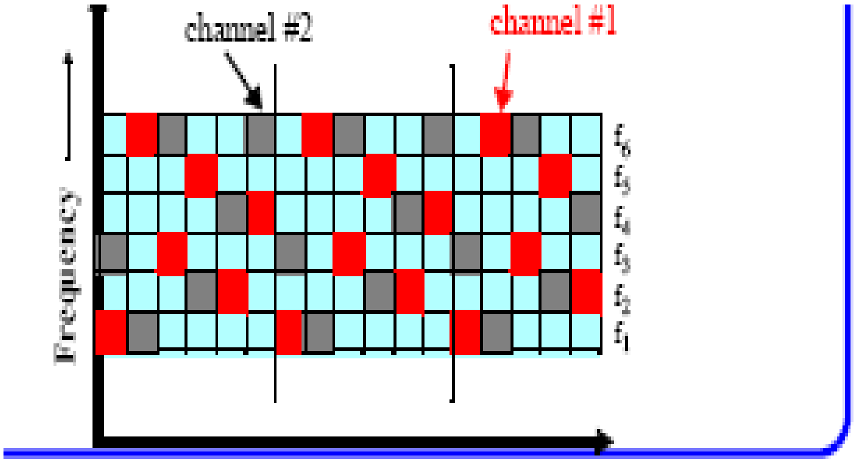
With the proliferation of radio communication systems and its traffic, interference problems are becoming of a major concern. A solution to this problem is to employ spread spectrum multiple access schemes, which enhance the capability of many users to share a finite amount of radio spectrum without interfering with one another significantly. One of the more popular spread spectrum technique used in communication systems is frequency hopping (FH). It is a digital multiple access scheme that uses different carrier frequencies in a pseudorandom fashion within a wideband channel.

In FH-SS, the signal itself is not spread across the entire large bandwidth; instead the wide bandwidth is divided into N sub-bands, and the signal ñhopsò from one band to the next in a pseudorandom manner. The center frequency of the signal changes from one hop to the next, changing from one sub-band to another. A single hop packet is first transmitted on a selected channel frequency and a new channel frequency is then selected to transmit the next packet at fixed intervals. As a result, the transmitted message is spread across the spectrum.

Frequency hopping (FH) systems drive a frequency synthesizer with a pseudo- random sequence of numbers spanning the range of the synthesizer with a pseudo-random to achieve spreading of the carrier. In basic form this technique, data is usually frequency-shift-keyed onto the spreading carrier.

The signal is recovered using the receiver that is hopping between frequencies in synchronization with the transmitter. Any form of interference in one frequency will only succeed in knocking out a few bits. In a code division multiple access FH-SS system, each user is assigned a distinct FH code sequence so as to allow many users to share the same bandwidth simultaneously with minimal interference.

The figures 7 (a) and 7 (b) below show an example of how information is transmitted using frequency hopping signal.



ivastava

Figure 7 (a): bandwidth divided into frequency sub-bands

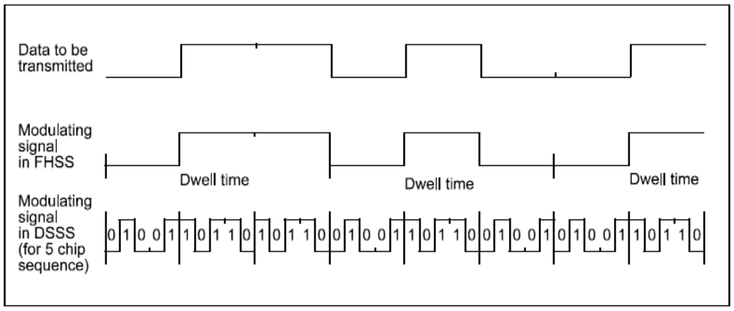


Figure 7 (b): a frequency hopping signal

In this example, 8 frequency sub-bands are allocated to be used by the FH signal. Typically there are 2k carrier frequencies, forming 2k channels, where k is the number of bits of the sequence to be used (PN source). A PN sequence is used to shift the carrier frequency of the FSK signal pseudo randomly at a hopping rate Rh. Usually, the width of each sub-band is set so that the amount of signal that overlaps with adjacent sub-bands is minimal, and is thus approximately the bandwidth of the original data signal, and that is the spacing between the carrier frequencies.

In FH-SS, the transmitter operates at one channel at a time for a fixed time interval. (for example, IEEE 802.11 standard uses 300-ms interval). During the interval, some number of bits is transmitted using some encoding scheme (M-FSK). The sequence of channels used is dictated by the spreading code. Both transmitter and receiver use the same code to tune into a sequence of channels synchronization.

Two different kinds of FH-SS are often use; slow FH and fast FH. In slow FH-SS, several bits are sent for each hop, so the signal stays in a particular sub-band for a long time relative to the data rate. In a fast-SS, the reverse is true. The signal switches sub-bands for a very short time relative to the data rate.

In the fast FH case, the performance of the system with respect to white Gaussian Noise is not changed, as in the DS-SS case. The noise power seen at the receiver is approximately the same as that in the un-hopped case, since each sub-band is approximately the same size as the original data signalôs bandwidth. And just as in the DS-SS case, the effect of a jammer is decreased by the spreading of the signal. Here, if we again assume that jamming signal j (t) is distributed uniformly over the entire band, it is clear that the only portion of the jamming signal that affects the data is the data is the part within the sub-band, and thus the jamming signal is reduced by the factor of the processing gain Gp which is given by:

$Gp = \frac{BW_{RF}}{BW_{\text{inf}\,0}} = \frac{Nf_b}{f_b} = N$, where BW_{RF} is the bandwidth that the signal has been increased to, BW_{info} is the minimum bandwidth necessary to transmit the information or the data signal, N is the number of sub-bands and f_b is the width of the sub-band(equal to information band width)

Thus in the frequency hop case, the protection afforded is equal to the number of frequency bands used. However, in this case the best way for a jammer to disrupt the signal is not to spread his power equally over the entire band, but to concentrate his power among a few bands. In this case, the jammer is more effective, because he can assuredly disrupt certain bits of data. The probability of a bit being in error is then given by p=J/N, where j is the number of channels selectively jammed, and N is the number of frequencies available to the hopper, which is essentially the probability that the jammer ñguessedò which frequencies to jam correctly. This can still give high bit error rates (BER). For example, a possible scenario would be where the jammer jams 10 out of possible1000 frequencies, giving a highly unacceptable BER of 10^{-2}

2.1.1 Spreading

The pseudo-random algorithm in the PN generator at the transmitter generates a random, district hopping sequence that controls the frequency synthesizer to change carrier frequencies. This generator is synchronized with one at the receiver, which will create the same hopping pattern as the transmitter so that during communication, both the transmitter and the receiver hop from channel to channel together. The PN sequence is used here to determine the hopping sequence. So in order to transmit the signal, the data must be modulated up the center frequency of the band determined by the PN sequence. Therefore, the structure of the transmitter is shown in figure 8.

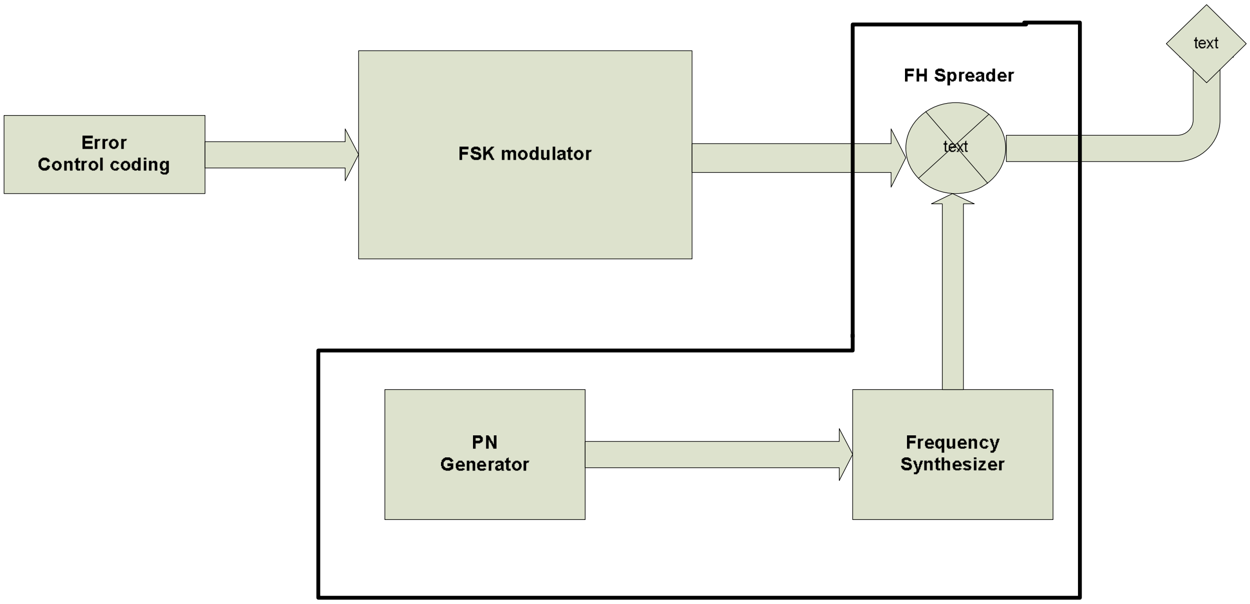


Figure 8: FH-SS transmitter

FSK modulation is applied to the data bits at the front end of the transmitter. The narrowband, modulated signal is then mixed with the frequency output of the frequency synthesizer to generate a wideband FH-SS signal. In other words, the frequency synthesizer acts as a modulator that determines the instantaneous carrier hop frequency of the FH-SS signal. The resulting signal is centred on some base frequency. The PN sequence is generated using a pseudo-noise generator; this code is used to spread the signal.

The data signal is modulated up to transmit frequency by the frequency produced by the frequency synthesizer, which takes as its input the output of the PN sequence generator (the band pass filter is also required at the output of the transmitter- it was omitted to simplify the structure). Upon transmission of the first packet, the frequency synthesizer changes the frequency corresponding to the pseudo random algorithm at the generator, and the next data packet will be transmitted at this new frequency.

Let k bits of the PN sequence specify one of the carrier frequencies i.e. at each successive interval (each k bits of PN sequence) a new carrier frequency is selected. This signal is then modulated by the signal from the initial modulator to produce a signal with a same shape, but now centered at different frequency.

For am M-FSK signal, M=2k is the number of frequencies used to encode the digital input, k bits at a time. For M-FSK, $S(t) = A\cos(2\pi f_c) + 2\pi m\Delta ft$ where f_c is the carrier frequency, æf is the difference frequency which is equal to f_b, m=1,23é .M, and K=bits per symbol.

For FH-SS signal is translated to a new frequency every T_h seconds by modulation with the FH-SS carrier for transmission. For data rate of R, the symbol rate1/T_s where T_s=K/R. for slow frequency hopping T_h > Ts since for the slow hop system the hopping rate is smaller that the message bit rate. For fast-frequency hopping Th < Ts. In fast frequency hopping the PN sequence is generated fast enough such as to allow the hop to occur in a shorter time than the duration of the message symbol.

The total M-FSK bandwidth is BWm=N.æf. if k is the number of bits in a block of the PN code, the bandwidth, BWss, of the FH-SS signal becomes:

$$BW_{ss} = 2k.BW_m$$

Each k bits of the PN code is used to choose one of the N channels.

The higher the value of k the more resistant to interference the system is. The power of the interfering signal (noise, other users, jammer etc) is divided by a factor of that is a function of the extended bandwidth of transmission at the receiver. The message is subjected to error correcting codes before it is stretched. Error correction codes allow for the reconstruction of any hops that were destroyed by interference and thus recover the information contained in them. The frequency multiplier at the input of the system is used to increase the bandwidth of the system and thus increase the processing gain of the FH signal.

A frequency hopping (FH) system obtains its wideband modulation characteristics by switching its narrowband signal over a wide range of frequencies in time. For a Frequency Hopping system processing Gain (Gp) is defined as the ratio between the instantaneous bandwidth of each hop (narrowband signal) and the overall bandwidth of the transmission channel.

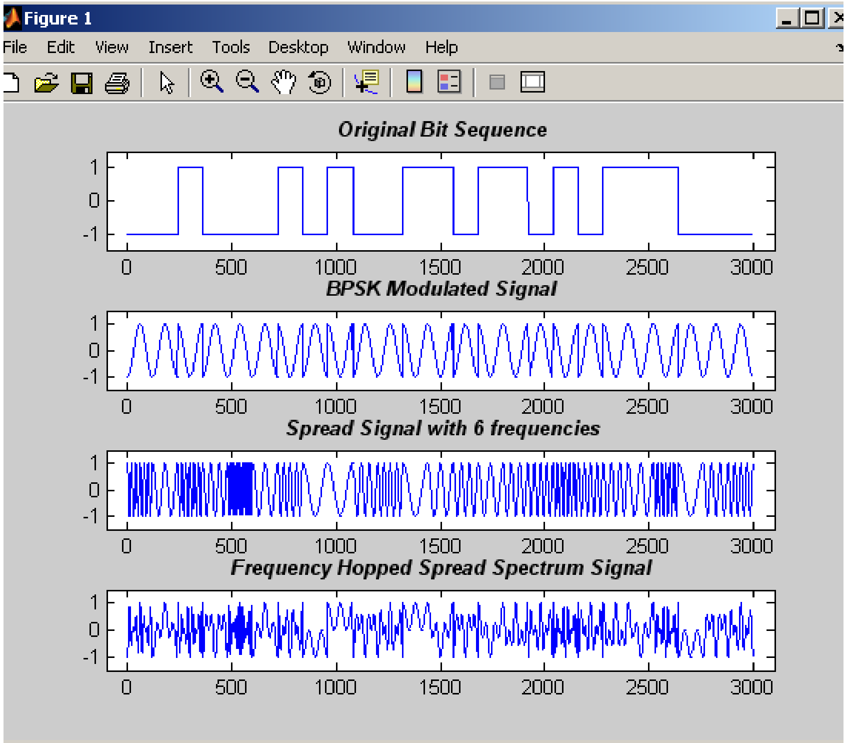


Figure 9: Frequency hopped spread spectrum signal

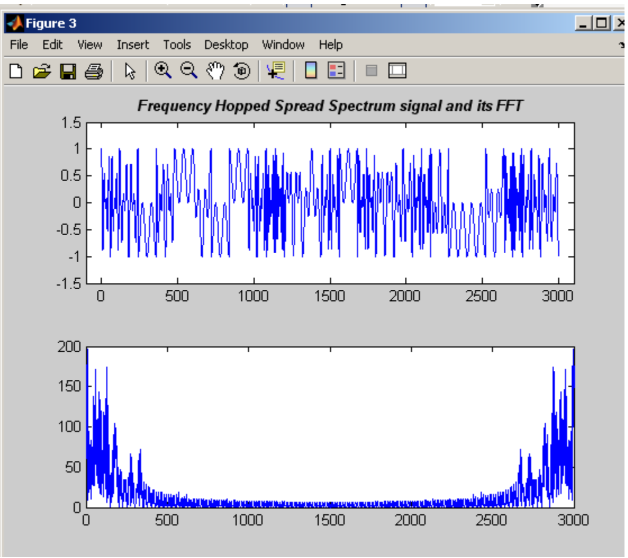


Figure 3
File Edit View Insert Tools Desktop Window Help
Frequency Hopped Spread Spectrum signal and its FFT
1.5
1
0.5
0
-0.5
-1
-1.5
0
500
1000
1500
2000
2500
3000
200
150
100
50
0
0
500
1000
1500
2000
2500
3000

Figure 10: frequency hopped spread spectrum signal with it s FFT

2.1.2 Despreading

One of the basic operations in a FH-SS receiver is the removal of the ñcodeò which was used by the transmitter to spread the data-message. This operation is known as dispreading. Dispreading is performed by combining the received signal with the same code which was used by the transmitter to code the data. To enable a low Bit Error Rate (BER), the local-code should be aligned with the received signal. An alignment error results is a loss of signal to noise ratio (SNR). Obtaining this code-alignment is the code-synchronization process.

An important observation is that at the beginning of a new PN-code also a new symbol and a new frequency-hop start. Obtaining PN-code synchronization therefore implies symbol and FH-sequence synchronization.

In a FH-SS system, the receiver structure, figure 11, is simply the reverse of the transmitter. The frequency synthesizer demodulates the signal down to an intermediate frequency (or baseband if desired), then the signal is filtered so only the desired data signal is passed through, finally the signal is decoded.

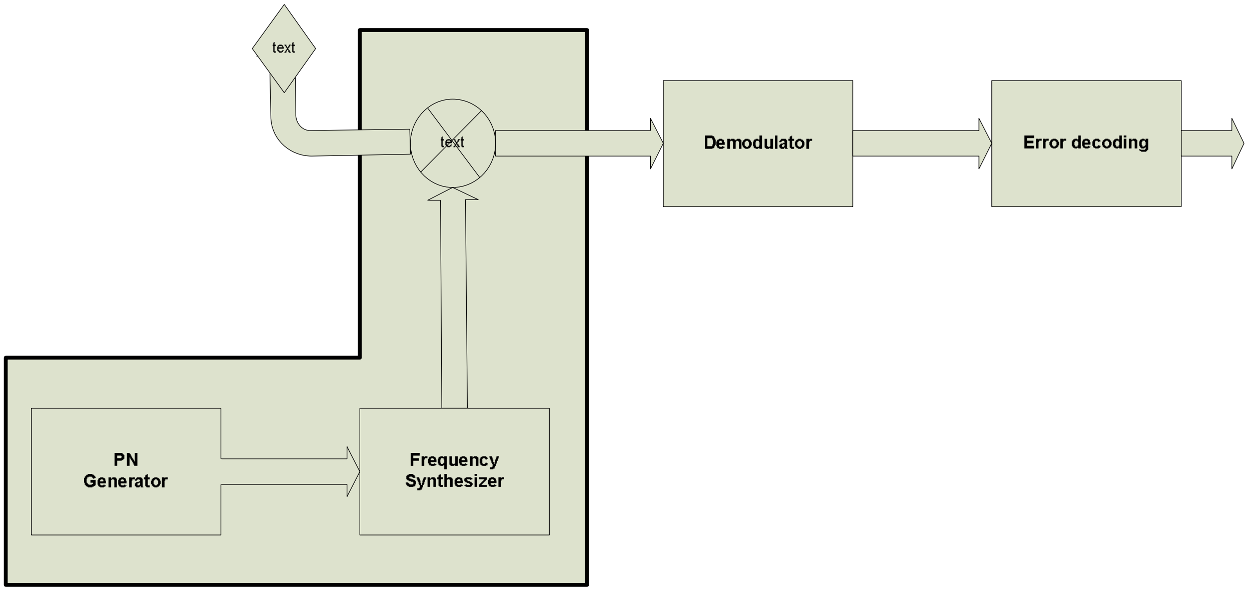


Figure 11: FH-SS receiver structure.

2.1.3 Advantages of using FH-SS

Implementation of FH-SS technique is extremely advantageous, especially in spectrums that are subjected to a lot of interface. Consider the case in which an interfering signal occupies a center frequency channel channel. If all the data of a user were also transmitted on this frequency, then all of the data would be lost at the receiver. If instead, the user employed FH-SS, then the number of lost packets would reduce down to one. This illustrates one of the significant advantages where RF interference is reduced since the transmitted message is hopped in a pseudorandom manner from one frequency to another. When many users use a FH spectrum, the level of interference increases. However as long as the number of users is significantly less than the number of channel frequencies, the interference level and subsequently rate of packet loss is minimal.

A FH-SS system is able to provide a high level of security for communications since each user is assigned a distinct hope code, which generates a unique FH hop sequence. Intelligible reception of the transmitted signal is only possible if the receiver knows the frequency hop sequence used.

Multipath interference is drastically reduced, as the delayed time versions of the transmitted packets will have low correlation with the original packets and will thus appear as another uncorrelated sequence, which is ignored by the receiver, also since each user is assigned a distinct code sequence, the probability that packets of the same channel frequency from two or more users will collide at a particular time is low. This results in reduced multiple-access interface and therefore a FH-SS system is less sensitive to the near-far effect.

2.1.4 Interference in FH-SS systems

The spectrum of white noise is flat and covers all the hop frequencies evenly. Therefore the received SNR of the FH-SS system is additive white Gaussian noise is the same as of the same system type of the system but without frequency hopping. While the SNR does not improve due to spreading the spectrum the signal to interference ratio does. Spread spectrum system is much more resistant to interference than a non-spread spectrum system. In frequency hopping narrow band interference affects single hops while wide band interference power is spread among all the hops in the given operating frequencies.

Although FH-SS reduces multipath and multiple-access interference significantly by assigning a distinct FH code sequence to each user area or area network to share a large bandwidth, it is still susceptible to data loss. This occurs when too many users or area networks utilize the same spectrum simultaneously or delayed versions of the transmitted signal interfere with the original signal at the receiver.

Multipath interference

The presence of the reflecting objects and scatterers in the communication channel create a constantly changing environment that result in multiple versions of the transmitted signal arriving at the receiving antenna. This is illustrated in figure 12, where a FH-SS signal sequence, S (t) from one user transmitted through a multipath channel. The signal at the receiver is a result of the combination of three sequences (one original and the other two delayed versions).

When packets of delayed versions of the transmitted FH-SS signal sequence happen to be using the same frequencies as the packets of the original sequence happen to be using the same frequencies as the packets of the original sequence at the same time, they will collide with each other and subsequently be lost. The number of packet collisions can be described by the *autocorrelation* function of the FH sequence. The autocorrelation function determines how much similarity a frequency hop sequence has compared to a time-shifted version of itself. A high level of similarity will result in more packets collisions.

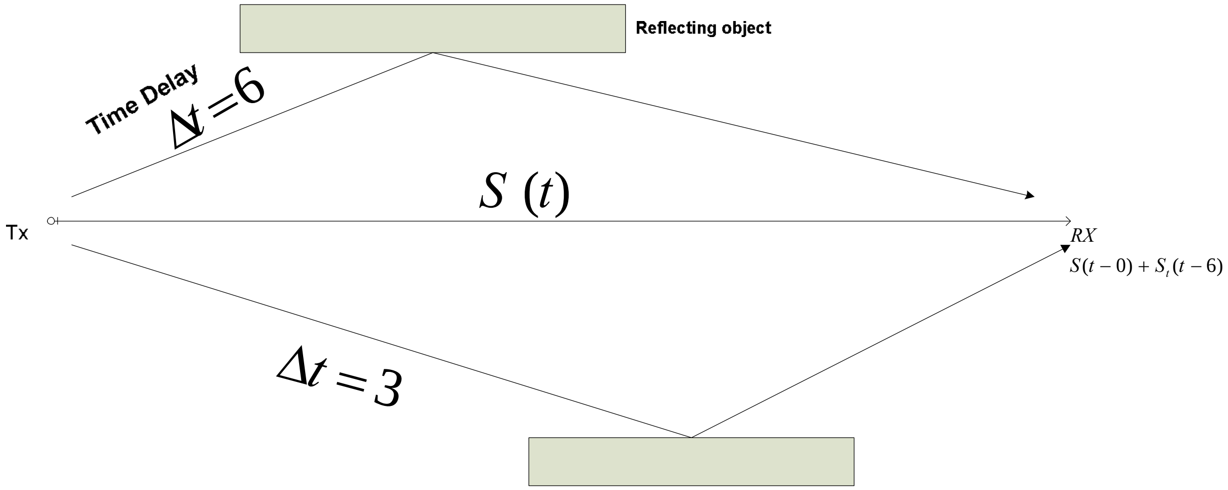


Figure 12:multi-path effects in a communication system.

Multiple-access interference

When two or more users (each user is assigned a distinct FH code sequence) access a communication channel, the resulting signal at the intended receiver may consist of several transmitted from some undesired users. This is illustrated in figure 13, where two FH-SS signal sequences S_1 (t) and S_2 (t) from two different users are transmitted through a multipath channel

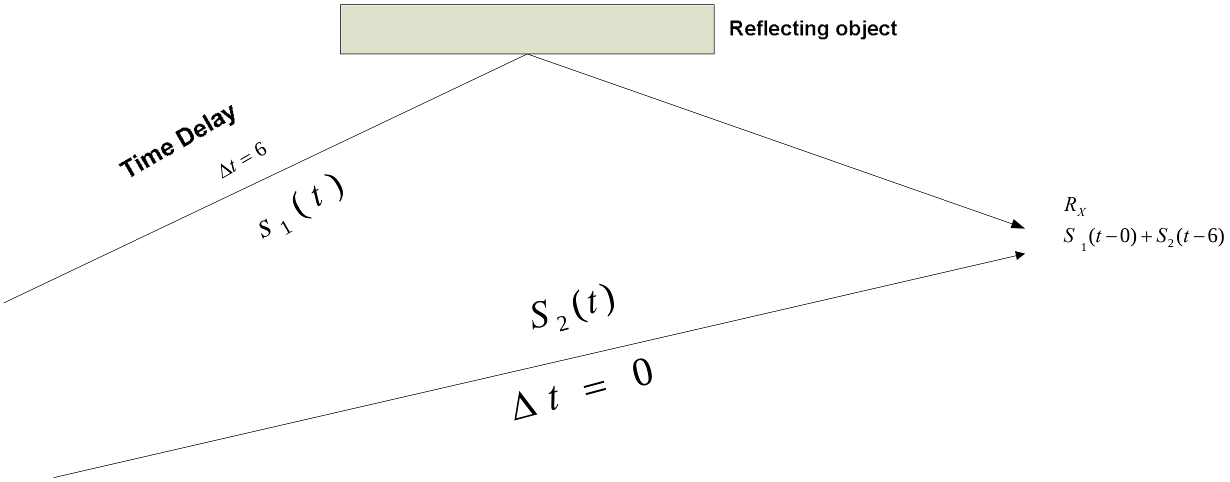


Figure 13: Multiple access effects in a communication system.

When packets of a transmitted FH sequence from an undesired user happen to be using the same frequencies as the packets of the desired user at the same time, packet collision occurs. The number of packet collisions occurring in this manner can be. Described by the *cross-correlation* function, which determines the similarity between the two frequency hop sequences from different users. A high level of similarity corresponds to a larger number of packet collisions.

Other forms of interferences include the effects of narrow band and wideband interface. Consider an FH-SS signal power spectrum shown below in figure 14.

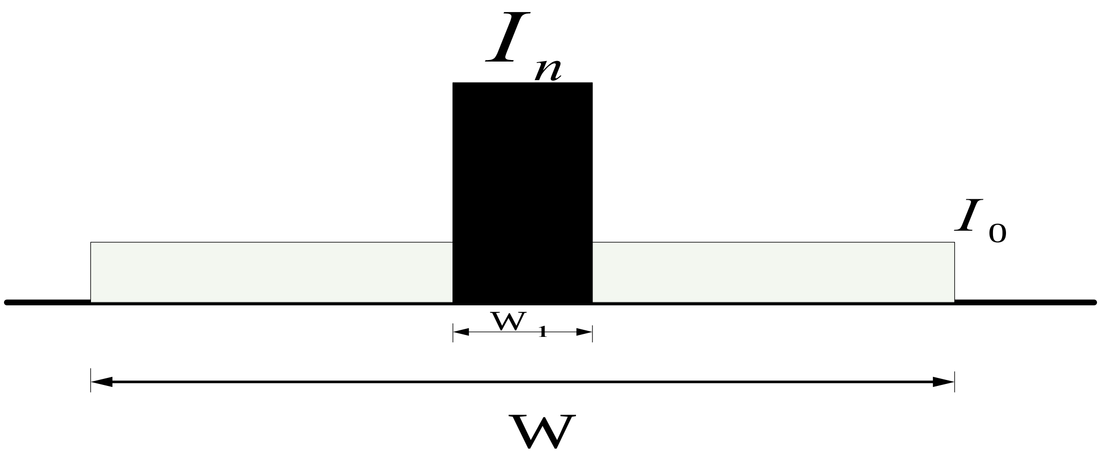


Figure 14(a):

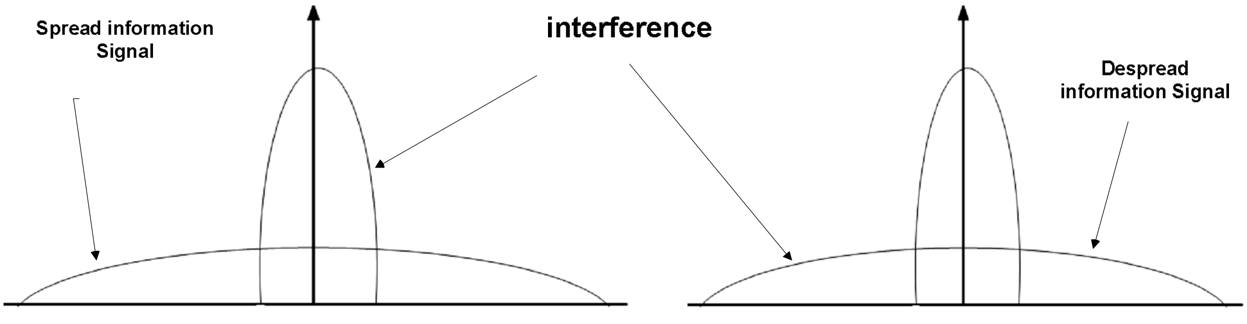


Figure 14(b):

Figure 14 :(a) and (b) Illustrates The Effects Of Narrow Band And Wideband Interference.

In figure 14 (a), the following assumptions are made:

- The narrowband and wideband interference signals are assumed to have rectangular power spectrum
- The bandwidth of the wideband interference is assumed to be the same as the transmission bandwidth W (its height is I_0).
- The bandwidth w1 of the narrowband interference signal is assumed to be narrower then the bandwidth at each hop and the spectral density of the wideband interface is $I_{n.}$

Thus the received interference power is expressed as follows:

$\mathbf{I_{av}=I_nW_1=I_0W}$

The same power is assumed for both narrowband and wideband interference.

Wideband interference:

Wideband interference in FH-SS is spread among all hops in the given operating frequencies. In this case, the signal to noise ratio for wideband interference is expressed as:

$$SNR_{wb} = \frac{E_b}{I_0} = \frac{W}{R_b} x \frac{P_{av}}{I_{av}}$$

Where R_b is the bit rate, P_{av} is the average power of the transmitted signal and I_{av} is the average power of the interfering signal.

As it can be observed from the above equation, the signal to noise ratio is $G_p = \frac{W}{R_b}$ times larger than it would be if the SS technique was not used. This result also has implications on the calculation of BER is G_p times larger thus significantly decreasing the error rate. This result also demonstrates the possibility of overlaying FH-SS over existing systems which is an attractive feature especially in commercial applications (due to ñthinningò spectrum resources).

Narrowband interference

In FH-SS narrow band interference affects single hops in the given operating frequencies. To analyze the effects of narrowband interference, the interference signal is assumed to affect only one hop and that its spectrum is flat over the hop bandwidth. For example the interference bandwidth of one-bit-per-hertz modulation would be $W1 = \frac{W}{G_p} = R_b$ and the height $In = \frac{I_{av}}{W_1}$.

Therefore, the signal to noise ratio per bit for the narrowband interference is: $SNRnb = \frac{E_b}{I_n} = \frac{W_1}{R_b} x \frac{P_{av}}{I_{av}}$ it is equal to the SNR of the system without FH-SS.

Thus, FH-SS does not provide any protection against interference assuming that the interfering signal ñhitsò a hop, but since it is assumed that other hops remain unaffected, and then the average signal to interference ratio is:

$$\overline{SNRnb} = \frac{NEb}{In} = \frac{NW_1E_b}{I_{av}} = \frac{WE_b}{I_{av}} = \frac{E_b}{I_0} = \frac{W}{R_b} x \frac{P_{av}}{I_{av}}$$

This is the same as the SNR for wideband interference. The average probability of error over all hops (assuming interference with one hop at a time) shows that if the inrefering signal follows the hops exactly, FH-SS does not reduce the effects of narrow band interference. This scenario is very unlikely in a commercial application and therefore FH-SS can be quite effective in fighting interference especially if used in combination with error detection and correction schemes.

2.2 PN SEQUENCES

A Pseudo-random Noise (PN) sequence is a sequence of binary numbers, e.g. ±1, which appears to be random; but is in fact perfectly deterministic. The sequence appears to be random in the sense that the binary values and groups or runs of the same binary value occur in the sequence in the same proportion they would if the sequence were being generated based on a fan- "coin tossing" experiment. In the experiment, each head could result in one binary value and a tail the other value. The PN sequence appears to have been generated from such an experiment. A software or hardware device designed to produce a PN sequence is called a PN generator.

Pseudo-random noise sequences or PN sequences are known sequences that exhibit the properties or characteristics of random sequences. They can be used to logically isolate users on the same frequency channel. They can also be used to perform scrambling as well as spreading and dispreading functions.

The reason we need to use PN sequences is that if the code sequences were deterministic, then everybody could access the channel. If the code sequences were truly random on the other hand, then nobody, including the intended receiver, would be able to access the channel. Thus, using a pseudo-random sequence makes the signal look like random noise to everybody except to the transmitter and the intended receiver.

Pseudo-random Noise sequences have the following characteristics:

- It is not random but it looks random for the user who doesn't know the code.
- It is deterministic, periodical signal that is known to both the transmitter and the receiver. The longer the period of the PN spreading code, the closer the transmitted signal shall be to a truly random binary wave, and the harder it is to detect.
- It has statistical properties of sampled white-noise.

2.2.1 PN SEQUENCE GENERATION

A PN generator is typically made of N cascaded flip-flop circuits and a specially selected feedback arrangement as shown in figure 12. The flip-flop circuits when used in this way are called a shift register since each clock pulse applied to the flip-flops causes the contents of each flip-flop to be shifted to the right. The feedback connections provide the input to the left-most flip-flop. With N binary stages, the largest number of different patterns the shift register can have $is2^N$. The all-binary-zero state, however, is not allowed because it would cause all remaining states of the shift register and its outputs to be binary zero. The all-binary-ones state does not cause a similar problem of repeated binary ones provided the number of flip-flops input to the modulo 2 adder is even. The period of the PN sequence is therefore 2^N -1.

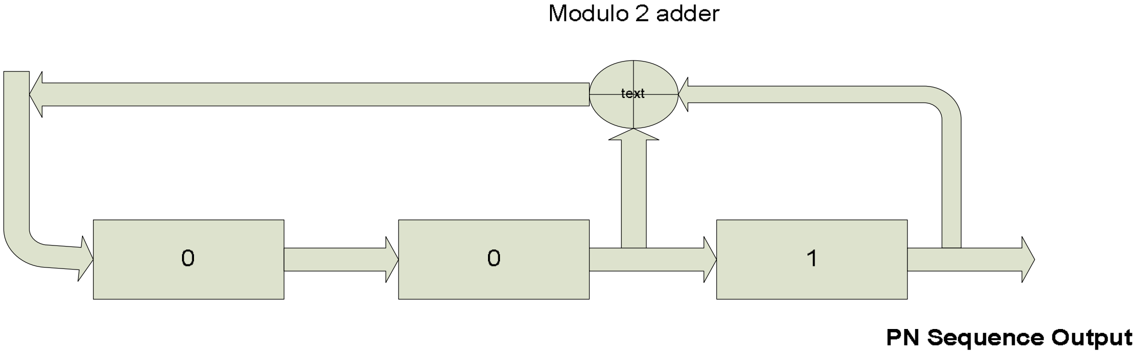


Figure 15: PN sequence generator

For example, starting with the register in state 001 as shown, the next 7 states are TOO, 010, 101, 110, 111, Oil, and then 001 again and the states continue to repeat. The output taken from the right-most flip-flop is 1001011 and then repeats. With the three-stage shift register shown, the period is 2^3 -I or 7.

The flip-flops that should be tapped-off and fed into the modulo 2 adder are determined by an advanced algebra that has identified certain binary polynomials called primitive irreducible or unfavorable polynomials. Such polynomials are used to specify the feedback taps. For example, IS-95 specifies the in-phase PN generator shall be built based on the characteristic polynomial: $P(x) = x^{15} + x^{13} + x^9 + x^8 + x^7+x^5+1$

Picture a 15-stage shift register with the right-most stage numbered zero and the successive stages to the left numbered 1, 2, 3,..., 14. Then the exponents less than 15 in the equation above tell us that stages 0, 5, 7, 8, 9, and 13 should be tapped and summed in a modulo 2 adder. The output of the adder is then input to the leftmost stage.

2.2.2 PROPERTIES OF PN SEQUENCES

PN signals are deterministic signals. They can be shown to have the following three properties:

- *Balance property:* The number of binary zeros and that of binary ones of a PN code are different only by one. When modulating a carrier with a PN sequence, one-zero balance ensures good spectral density properties (equally spreading the energy over the whole frequency band).

- *Run Property:* A run is a sequence of a single type of binary digits i.e. the number of times the binary ones and binary zeros repeat in groups or runs appear in the same proportion they would if the sequence were actually generated by a coin tossing experiment. For an N bit code, there would be N zero (or one) runs. 1/2 of these runs would be of length one, 1/4 of length 2, 1/8 of length 3 and so on.
- *Correlation Property:* The correlation value of two N-bit sequences can be obtained by counting the number of similar (N_s) and dissimilar (N_d) bits and inserting them into the following equation: $P = (1/N) * (N_s - N_d)$

3.1. Autocorrelation

The origin of the name pseudo-noise is that the digital signal has an autocorrelation function which is very similar to that of a white-noise signal: impulse-like. The autocorrelation function *R(Ű)* , of a PN period of sequence is defined as the number of agreements less the number of disagreements in a term by term comparison of one full period of the sequence with a cyclic shift (position Ű) of the sequence itself.

R(Ű) =∫*C(t)C(t* + Ű*)dt* = 0; if Ű≠ 0 there shouldn't be any correlation.

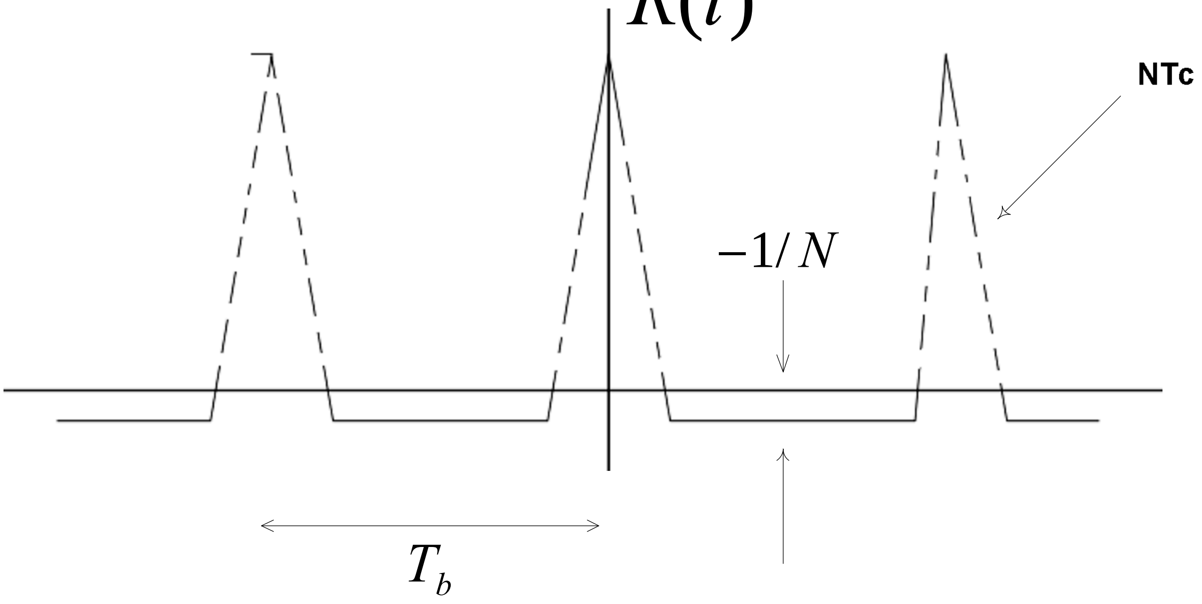


Figure 16: Autocorrelation Function of PN Sequence

For PN sequences the autocorrelation has a large peaked maximum (only) for perfect synchronization of two identical sequences (like white noise).The synchronization of the receiver is based on this property.

The autocorrelation function will be very small in the regions between the peaks if N is very large. By using shift register codes PN sequence can be made very large.

For example, if the clock frequency is 10 MHz, which means that chip time is 10^{-7} seconds, and the shift register with 41 stages is used, the resulting is:

$N=2^{41}$ ï 1 = 2.199 x l0^{12}

N x T_c =2.199 x l0$^{12\ x}$ 10^{-7} = 2.199 x 10^5 sec

This is an equivalent of about 2.5 days. The longer the period of the PN spreading code, the closer the transmitted signal will be to a random binary wave, and therefore the harder it is to detect.

3.2. Cross-correlation:

Cross correlation describes the interference between codes C_1 (t) and C_j (t). The Cross-correlation function for the periodic PN sequences C_1 (t) and C_j (t) describes the number of agreements less the number of disagreements in a bit by bit comparison of one full period of the sequence C_1 (t) with a cyclic shift (position Ű) of the sequence C_j (t).

R_c (Ű) = ∫C_1 *(t)* C_J *(t* + Ű)*dt* = 0 for orthogonal codes

In CDMA, since multiple users occupy the same RF bandwidth and transmit simultaneously, there is no interference between users after despreading and the privacy of communication of users is protected if the codes used are orthogonal. If they are not perfectly orthogonal the cross-correlation between user codes introduces performance degradation (increased interference) which limits the maximum number of simultaneous users.

2.2.3 TYPES OF PN SEQUENCES

1) **M-Sequence**

A Linear Feedback Shift Register is used to generate this type of PN code. A shift register is linear if the feedback function can be expressed as a modulo-2 sum, A simple shift register has all the feedback signals returned to a single input of a shift register.

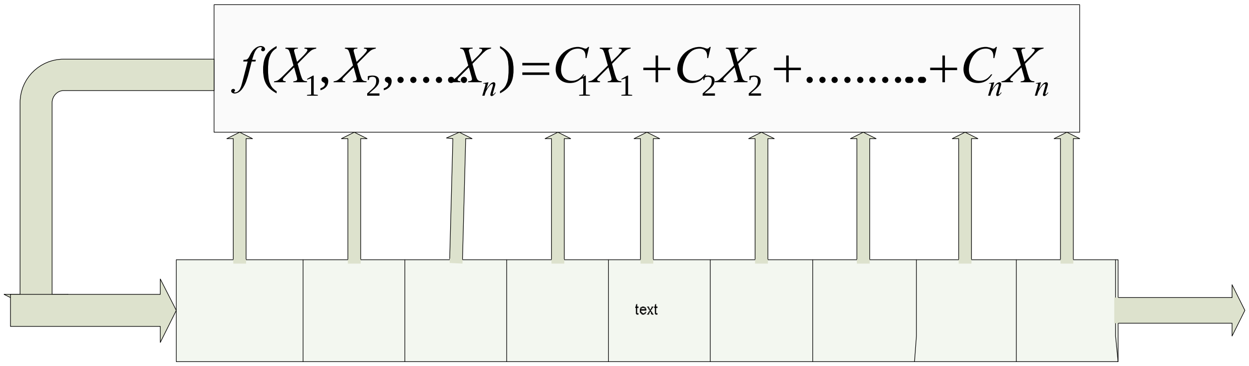


Figure 17: Simple Linear Feedback Shift Register

The feedback function:

f($X_{1,}X_2$,é .X_n) + C_1X_1 + C_2X_2 + é é   + C_nX_n

is a modulo-2 sum of the contents Xi of the shift register cells with Q being the feedback connection coefficients.

A LFSR with L flip-flops produces sequences that depend upon the register length L, feedback tap connections and initial conditions. When the period of the of the sequence is N_c= 2^L-1 the PN sequence is called a maximum-length sequence.

…a_i, a_{i+1}, a_{i+2}……….

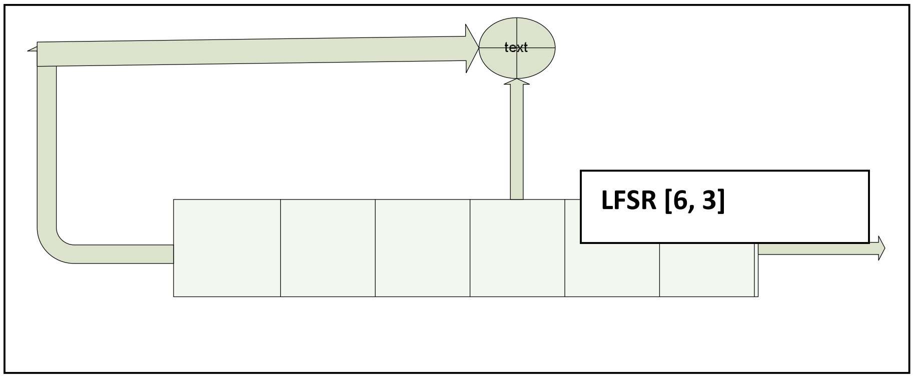


Figure 18: Maximum-length sequence

An m-sequence generated from a LFSR has an even number of taps. The contents of the register, the bits tapped for the feedback function, and the output of the feedback function together describe the state of the LFSR. With each shift, the LFSR moves to a new state. (There is one exception to this; when the contents of the register are all zeroes, the LFSR will never change state.) For any given state, there can be only one succeeding state. The reverse is also true: any given state can have only one preceding state.

A state space of an LFSR is the list of all the states the LFSR can be in for a particular tap sequence and a particular starting value. Any tap sequence will yield at least two state spaces for an LFSR. (One of these spaces will be the one that contains only one state ð the all zero one.)
The state of an LFSR that is L bits long can be any one of 2^L different values. The largest state space possible for such an LFSR will be 2^L - 1 (all possible values minus the zero state). Because each state can have only once succeeding state, a LFSR with a maximal length tap sequence will pass through every non-zero state once and only once before repeating a state.

One corollary to this behavior is the output bit stream. The period of an LFSR is defined as the length of the stream before it repeats. The period, like the state space, is tied to the tap sequence and the starting value. As a matter of fact, the period is equal to the size of the state space. The longest period possible corresponds to the largest possible state space, which is produced by a maximal length tap sequence. (Hence "maximal length")

Maximal length sequence determination:

To determine the taps that produce m-sequence the characteristic polynomial comes into play. The polynomial:

$f(x)=X^{m}+ C_m X^{m-1} +.... C_2 X + C_1$

is called the characteristic polynomial used to define the periodic sequence generated by the LFSR.

This polynomial describes the linear recurrence relation of the PN generator and can be analogous to the *linear difference equations.* Just as with the linear differential equations with constant coefficients, the nature of their solutions depends on solution of an associated polynomial (the auxiliary polynomial).
It turns out that the maximum possible period of the sequence can be generated only if f(x) is a primitive polynomial.(irreducible).It is primitive if and only if f(x) irreducible and the least positive integer n such that f(x) divides X^n-l is n= 2^{m} -1.

Properties of m-sequence:

1. *Balance:* For an m-sequence there is one more ' 1' than '0' in a full period of the sequence. Since all states but one-the all-zero state are reached in an m-sequence, there must be 2^{L-1} "ones" and 2^{L-1} ñones and 2^{L-1} ñzerosò.
2. *Autocorrelation:* The autocorrelation function of the m-sequence is ð 1 for all values off the chip phase shift τ, except for the [-1 ,+1] chip phase shift area , where the correlation varies linearly from the -1 to a value equal to the sequence length- 2^L-1= *N*. The autocorrelation peak increases with increasing length of the m-sequence.

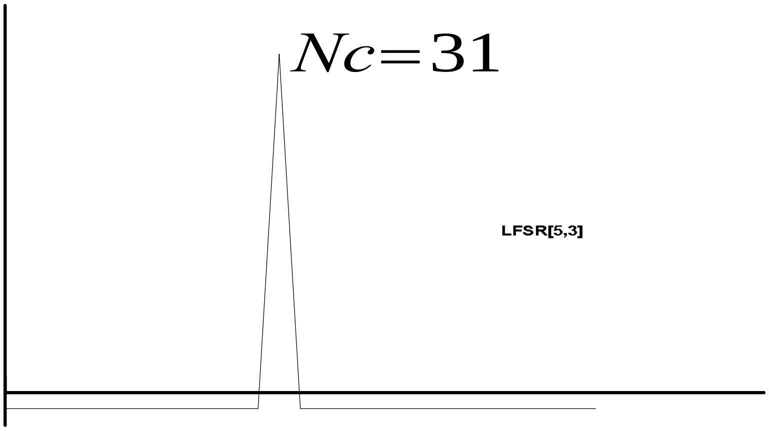


Figure19: Autocorrelation Function of m-sequence

3. .*Cross-correlation.*

The cross-correlation between different codes is very poor therefore when large numbers of transmitters using different codes are to share a common frequency band (multiuser environment), the code sequences must be carefully chosen to avoid interference between users.
An example of the cross correlation between the sequences generated by LFSR[5,3]and LFSR[5,2] is shown in Figure 22.

2) Gold Codes

Gold code sequences are useful because a large number of codes with the same length and with controlled cross correlation properties can be generated, although they require only one pair of feedback tap sets.
These are achieved by modulo-2 addition of two m-sequences with the same length. The code sequences are added chip by chip bisynchronous clocking. Because the m-sequences are of the

same length, the two code generators maintain the same phase relationship, and the codes generated are of the same length as the base codes which are added together, but are non-maximal. With this type a large number of codes (whose cross-correlation can be controlled) can be generated thus useful in a multi-user environment not good autocorrelation property.

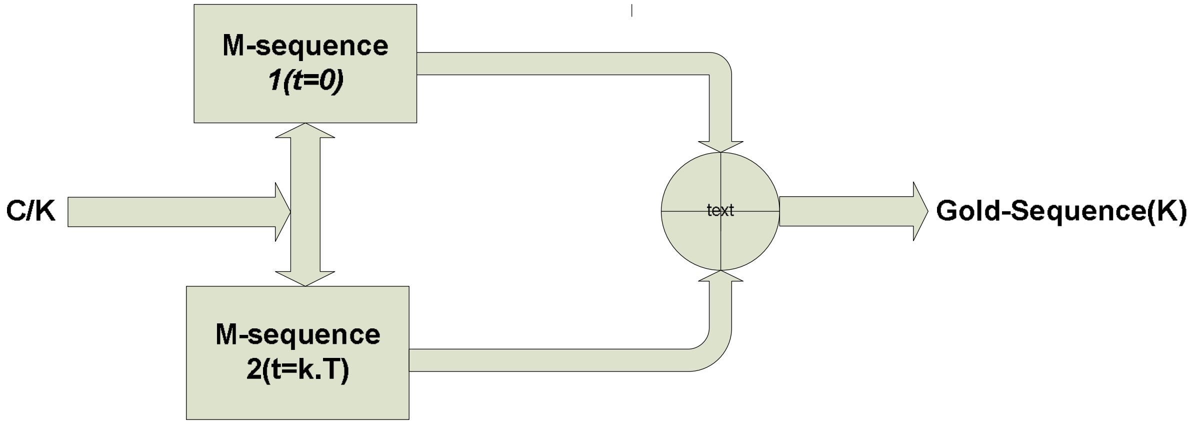


Figure 20: Gold sequence generation

3) Barker Codes

This type gives codes with different lengths (determined by the number of registers used) and similar autocorrelation properties as the m-sequence. This type of sequence is not necessarily balanced.

CHAPTER THREE

3.0 PN CODE SYNCHRONIZATION

In order for the Spread spectrum communication system to operate normally it requires that the locally generated PN sequence at the receiver be synchronized to that at the transmitter generator in both its frequency and phase. The code synchronization process in Frequency Hopping Spread Spectrum happens in following two phases:

Acquisition ð this is the coarse code synchronization process. It is the initial phase where the receiver recognizes the transmitter. This involves searching throughout a region of time and frequency (chip carrier) in order to synchronies the received spread spectrum signal with the locally generated PN sequence. The objective of this stage is to resolve the code phase error to within certain bounds which can be further reduced by the tracking-stage.

Tracking - This is also called a fine-tuning (synchronization).This phase happens upon successful acquisition phase. In the tracking phase, the transmitter and the receiver need to be in continuous synchronization until data transmission is complete. Tracking continuously maintains the best possible waveform fine alignment by means of a feedback loop. This is essential to achieve the highest correlation power and thus highest processing gain (SNR) at the receiver.

A receiver typically consists of acquisition circuits, tracking circuits, and demodulator circuits. Figure 24 shows the basic synchronization system of a FH-SS receiver. In this system, the incoming signal is first locked into the PN signal generator using the acquisition circuit, and the kept in synchronism using the tracking circuit, finally the data are demodulated.

The main sources of synchronization uncertainty are: Time uncertainty due to:

- Propagation delay
- Relative clock shifts
- Different phase between transmitter and receiver (carrier, PN sequence).
- Frequency uncertainty due to relative velocity between transmitter and receiver which affects the carrier frequency

Figure 24: Functional diagram of the FH-SS synchronization system

3.1 ACQUISITION

As mentioned before, the objective of initial code- acquisition is to achieve a coarse synchronization between the receiver and the transmitted signal. This involves searching throughout a region of time and frequency (chip carrier) in order to synchronize the received spread spectrum signal with the locally generated PN sequence.

The acquisition search space is set of all possible relative shifts of the local code with respect to the received signal. This search-space is divided into search-cells. The process of acquisition is identifying the so-called sync-cell, which is the cell that corresponds to a situation in which the receiver is synchronized.

Searching a single cell takes a so called dwell-time or integration-time. After this dwell-time the power at the output of the data-detector is calculated. This power-level is used as a decision variable to select the sync-cell. A common feature of all acquisition methods is that the received signal and the locally generated PN are first correlated with a coarse time step (normally $T_C/2$) to produce the measure of similarity between the two. This measure is then compared with a *threshold value* to decide if the two signals are in synchronism. If they are a verification algorithm is started. To prevent false locking, it is necessary to dwell for sometime to test synchronism. For proper synchronization, a peaked autocorrelation is demanded of the PN sequence. There are a number of ways used to acquire a signal. Before we look at these schemes there are two important measures that determine the 'performance' of an acquisition-scheme:

- The false-alarm probability. This is the chance that acquisition is declared at a wrong cell
- The detection probability. This is the chance that if there is acquisition, there shall be detection also. The usual way to tackle the serial-search acquisition problem is as follows: After examining a cell the power-contents of that particular cell is calculated. If this power exceeds a certain threshold, detection probability will be low. The following are some of the ways to search for acquisition:

3.1.1 Serial Search

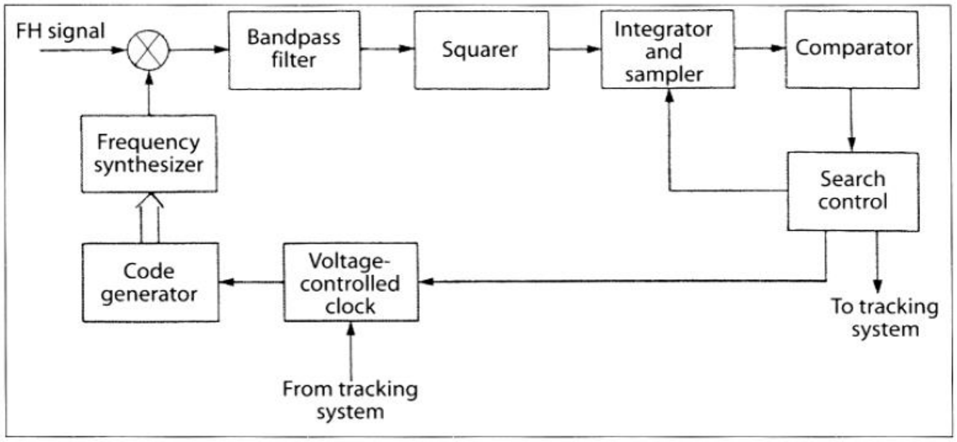


Figure 21; Serial acquisition architecture

The first method for acquisition is known as serial search. This method uses a single correlator and searches the cells sequentially. A clear disadvantage is that it takes long since a large number of cells are analyzed sequentially to find the sync-cell. The circuit complexity for serial search is low, however the penalty time associated with a miss is large. Therefore there is need to select a larger integration (dwell) time to reduce the miss probability. This, together with the serial searching nature, gives a large overall acquisition time (i.e., slow acquisition). To summarize**,** a serial-search strategy is slow, but cheap in terms of resource usage.

3. 1. 2 Parallel Acquisitions

Another way to find acquisition is by applying parallel search: examine more cells at the same time. A number of correlators operate in parallel which causes the acquisition time to decrease. It also increases complexity to analyze the power contents of the parallel stages. The required amount of computational power easily grows than beyond the available resources. A parallel-search strategy is fast, but expensive in chip-area and required computational power. Consider the figure below:

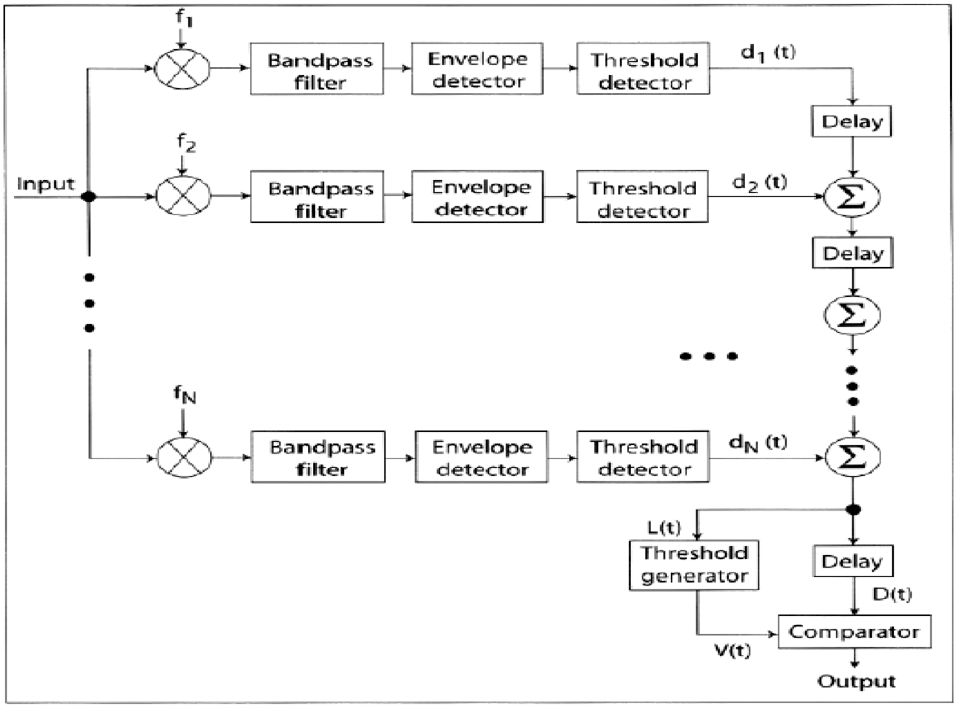


Figure 22: Parallel acquisition architecture

- Here, the incoming signal is sent down N different paths, where N is the number of hop frequencies available.
- In each path, the incoming signal is multiplied by one of the N frequencies, filtered, squared (to get the power), and then delayed by an amount depending upon the location of that frequency in the hop order. For example, if the sequence is $\{f_1, f_2, f_3, \ldots f_n\}$, the signal correlated with f_1 will be delayed by N hops, while the signal correlated with f_n will be delayed by one hop.
- In this way, at any given time *T*, the input to the summer will be the power of the data signal correlated with the entire hop sequence starting at time *T - N.* Thus, whenever the value of the output of the summer is greater than a certain threshold, lock has been acquired.

If N is large (which is usually the case), we see that this structure can become unnecessarily costly. Again, a trade-off between cost and length of time before lock is acquired can be managed.

3.2 TRACKING

Tracking commonly maintains the best possible waveform fine alignment by means of a feedback loop. This is essential to achieve the highest correlation power and thus highest processing gain (SNR) at the receiver.

- A typical tracking system for FSK/FH-SS is as shown in the figure 23(a) below together with wave forms as shown in figure 23 (b).
- Although acquisition has occurred we shall assume, for the purpose of discussion that, there is still an error Űseconds between transitions of the incoming signals frequencies and the locally generated frequencies,
- The bandpass filter is sufficiently wide to pass the product signal $V_p(t)$ when $V_1(t)$ and $V_2(t)$ are at the same frequency f_1 but sufficiently narrow to reject $V_p(t)$ when $V_1(t)$ and $V_2(t)$ are at different frequencies f_1 and f_{i+1}
- Thus the output of the envelope detector $V_d(t)$ is unity when $V_1(t)$ and $V_7(t)$ are at the same frequency and is zero when they are at different frequencies.
- From figure 29(a), we see that $V_g\ (t) = V_d(t)V_c(t)$ and is a three-level signal. This three-level signal is filtered to form a dc voltage which, in this case, presents a negative voltage to the VCO.

It is readily seen that when $V_2(t)$ has frequency transitions which precede those of the incoming waveform $V_1(t)$, the voltage into the VCO will be negative, thereby delaying the transition, while if the local waveform frequency transitions occur after the incoming signal frequency transitions, the voltage into the VCO will be positive, thereby speeding up the transition. The role of the tracking circuit is to keep the offset r small. However, even a relatively small r can have a major impact on the probability of error of the received data.

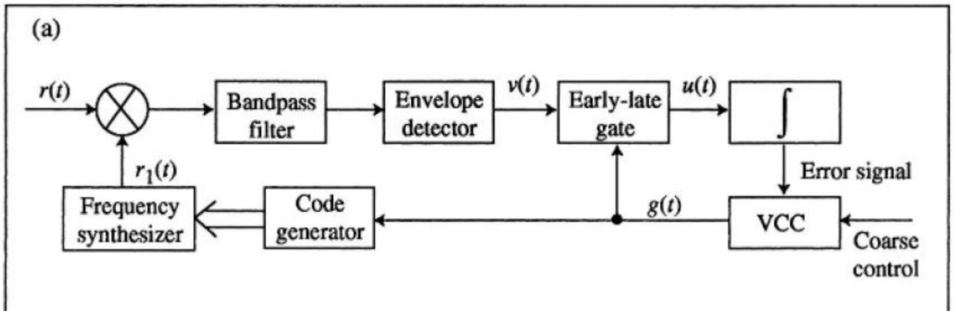


Figure 23(a): Tracking Loop for FH-SS Signals

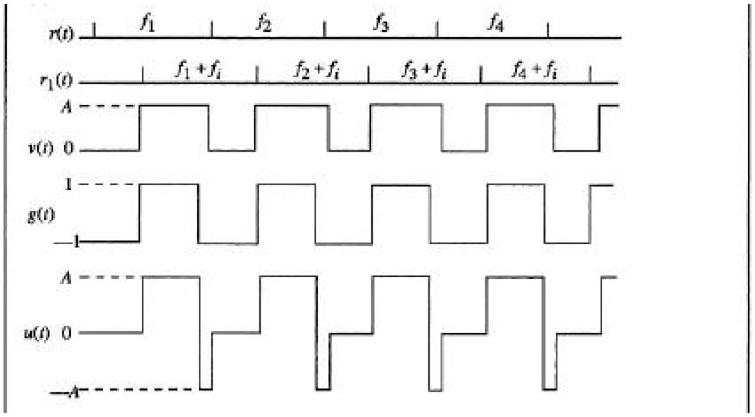


Figure 23(b): Waveforms for tracking an FH-SS Signal

3.2.1 The Phase Lock Loop (PLL)

Usually, a phase lock loop (PLL) circuit is used to synchronize an output signal, which is usually generated by an oscillator, with a reference or input signal in frequency as well as in phase. In the synchronized state, the difference (error) between the reference and the oscillator output is zero or at least very small. So it is called 'locked'.

The whole circuit consists of three main parts. They are phase & frequency detector (PFD), loop filter (LP) and voltage control oscillator (VCO). The diagram of the PLL is shown in Figure 24.

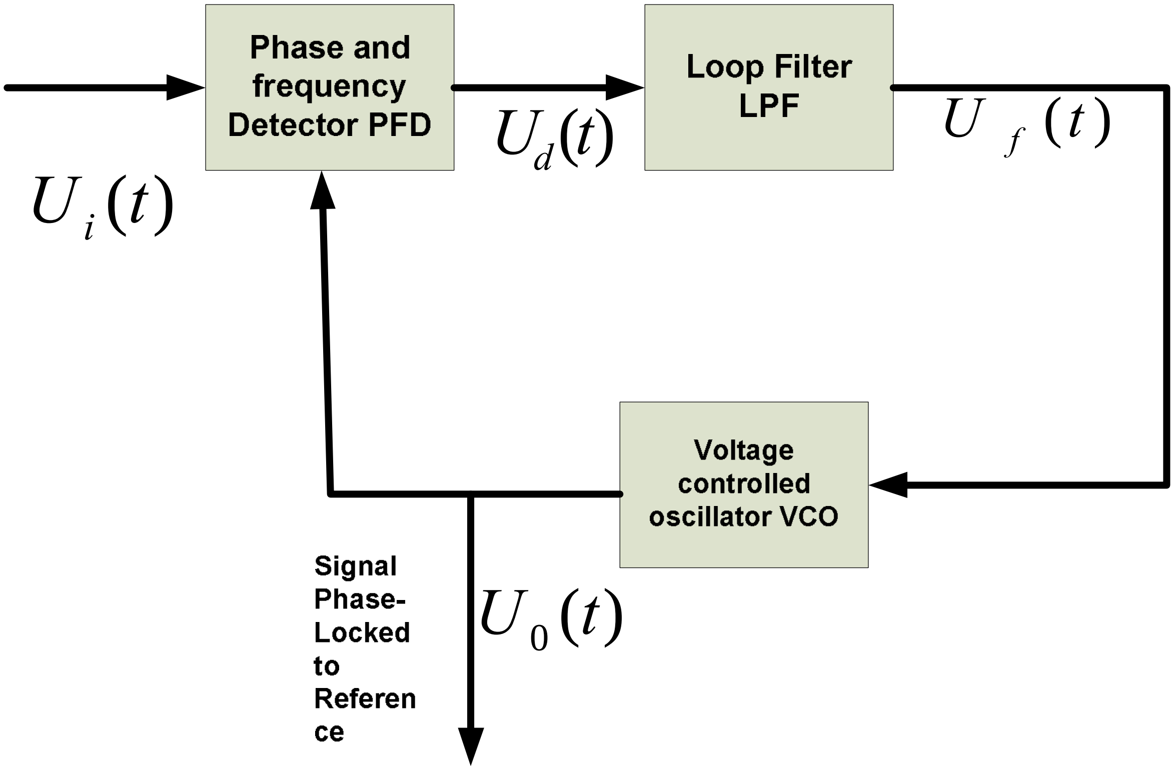

Phase and frequency Detector PFD
Loop Filter LPF
Voltage controlled oscillator VCO
$U_i(t)$
$U_d(t)$
$U_f(t)$
$U_0(t)$
Signal Phase-Locked to Referen ce

Figure 24: PLL schematic diagram

The input (or reference) signal: $U_i(t)$

The frequency of the input signal $U_i(t) f_i$

The output signal of the VCO: $U_o(t)$

The frequency of the output signal $U_o(t)$: f_o

The output signal of the phase and detector: U_d (t).

The output signal of the loop filter: $U_f(t)$

Operation principle

According to the block diagram shown in figure 30 above, we can clearly explain how this PLL circuit works.

First, we assume that the frequency of the input signal f_i and the frequency of the output f_o are the same. That means there is no phase error in this PLL. So the output of phase and frequency detector $U_d(t)$ must be zero. Therefore, the output signal of loop filter should be a constant value. This constant voltage is to be used as the input to the voltage controlled oscillator. Constant input voltage to the VCO will result in constant frequency of the output signal of that VCO. Thus, the output frequency of $U_o(t)$ is stable locked at the value of f_o , which is equal to the input frequency $f_{i.}$

This is the condition that permits the VCO to operate at its centre frequency. Thus the circuit is locked!

If the phase error is not zero initially, the PFD would develop a nonzero output signal $U_d\ (t),$ which represents somehow the phase difference between the input signal and output signal of VCO. So the output of the loop filter $U_f\,(t)$ will not be a constant value. It changes on a basis of the output of PFD, because different input voltage as input to VCO will change output frequency of VCO to make that output frequency become closer to the reference. After some time, when the output frequency of VCO becomes the same as the input frequency (signal) i.e. synchronized, we refer the circuit as stable and locked! Once in lock, the VCO frequency is identical to the input signal except for a finite phase difference. This net phase difference is necessary to generate the corrective error voltage to shift the VCO frequency from its quiescent value to the input signal frequency f_i and thus keep the PLL in lock. This self-correcting ability of the system also allows the PLL to track the frequency.

The range of frequencies over which the PLL can maintain lock with an input signal is defined as the "lock range" of the system. The band of frequencies over which the PLL can acquire lock with an incoming signal is called "Capture range" of the system and is never greater than the "lock range". Another means for describing the operation of the PLL is to observe that the phase detector is in actuality a multiplier circuit that mixes the input signal with the VCO signal. This mix produces sum and difference frequencies $f_i \pm f_{0}.$ When the loop is in lock, the VCO duplicates the input frequency so that the difference frequency $f_i - f_o$ is zero, hence the output of the phase comparator contains a dc component. The low pass filter removes the sum frequency component $f_i + f_o$ but passes the dc component, which is then amplified and fed back to the VCO. Notice that when the loop is in lock, the difference frequency component is always dc, so the lock range is independent of the band edge of the low pass filter.

Consider now the case where the loop is not yet in lock. The phase comparator again mixes the input and VCO signals to produce sum and difference frequency components. If the difference frequency is above the cut off frequency of the low pass filter, no signal is transmitted around the loop and the VCO remains at its initial free-running frequency. As the input frequency approaches that of the VCO, the frequency of the difference component decreases and approaches the band edge of the low pass filter. Now some of the difference component is passed, which tends to drive the VCO towards the frequency of the input signal. This in turn decreases the frequency of the difference component and allows more information to be transmitted through the low pass filter to the VCO. This is essentially a positive feedback mechanism, which causes the VCO to snap into lock with the input signal.

With this mechanism in mind, the term, "Capture range" can again be defined as the frequency range centered about the VCO initial free-running frequency over which the loop can acquire lock and track the input signal once lock has been achieved. The capture range is a measure of how close the input signal's frequency must be to that of the VCO to acquire lock. The "capture range" can assume any value within the lock range and depends primarily upon the band edge of the low pass filter together with the closed loop gain of the system. It is this signal-capturing phenomenon which gives the loop its frequency selective properties.

When the loop is in lock, the difference frequency component on the output of the phase comparator (error voltage) is dc and will always be passes by the low pass filter. Thus, the lock range is limited by the range of the error voltage that can be generated and the corresponding VCO frequency deviation produced. The lock range is essentially a dc parameter and is not affected by the band edge of the low pass filter.

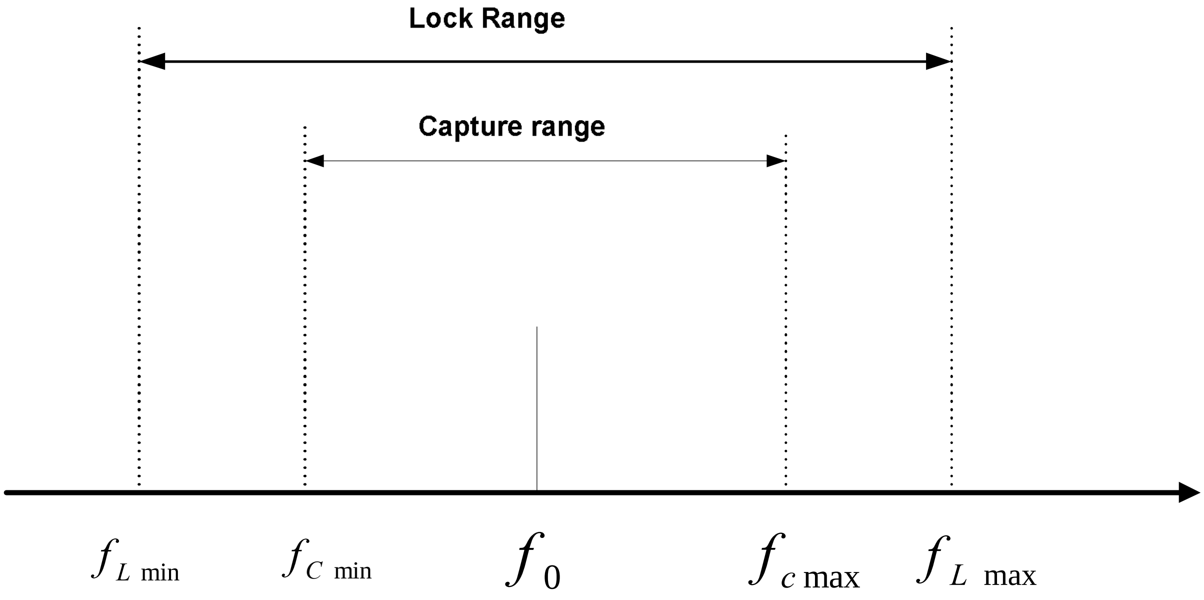


Figure25: Lock and Capture range of a VCO

There are four types of PLL:

- The LPLL (linear PLL)
- The DPLL (digital PLL)
- The ADPLL (all-digital PLL)
- The SPLL (software PLL)

CHAPTER FOUR

4.0 DESIGN & IMPLEMENTATION OF A SYNCHRONIZATION SYSTEM

Component description:

4.1 Phase and Frequency Detector (PFD)

The first component in the design **is** the phase and frequency detector. The output of the PFD depends on both the phase and frequency of the inputs. This device is modeled in Mat lab simulink as shown below.

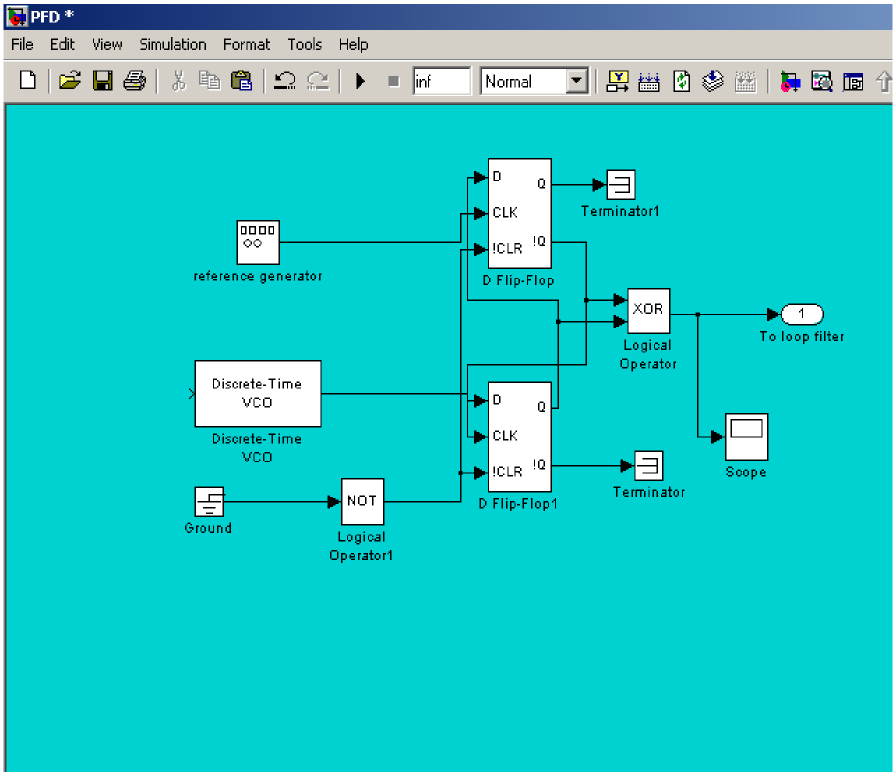


Figure 26: Phase and Frequency detector simulink model

This device can be used in conjunction with a counter to output the phase error. It consists of a Logical operator which performs the XOR operation; the multi-input XOR gate performs an addition- modulo-two operation as mandated by the IEEE Standard for Logic Elements and two D flip-flops. The D flip-flop block has the following characteristic table:

Q(t)	D(t)	Q(t-next)
0	0	0
0	1	1
1	0	0
1	1	1

Q (t-next) refers to the output on a clock pulse (CLK) rising edge and when the chip is enabled (!CLR = approx 0). The D flip-flop transfers "data" into a memory element (flip-flop) on each clock pulse (CLK). The chip enable input signal! CLR, is sometimes given the designation G (for gate) to indicate that this input enables the gated latch allowing data entry into the flip-flop.

4.2 Loop Filter.

The second component in the design is the loop filter. In the simulink design, the loop filter is a low pass filter type designed and implemented in Butterworth configuration. The input to this device must be a sampled-based scalar signal. The magnitude response of a Butterworth filter is maximally flat in the passband and monotonic overall. The design filter order is 3.

The output of the loop filter was observed to produce a constant amplitude waveform as shown below when the system was in its locked or synchronized state.

Figure 27: Simulation showing the Loop filter Output at locked or synchronized state.

4.3 Voltage Control Oscillator:

The Voltage-Controlled Oscillator (VCO) block generates a signal whose frequency shift from the quiescent frequency parameter is proportional to the input signal. The input signal is interpreted as a voltage. This block uses a continuous-time integrator. The input and output signals are both sample-based scalars.

Figure 28: Simulation showing the VCO Output

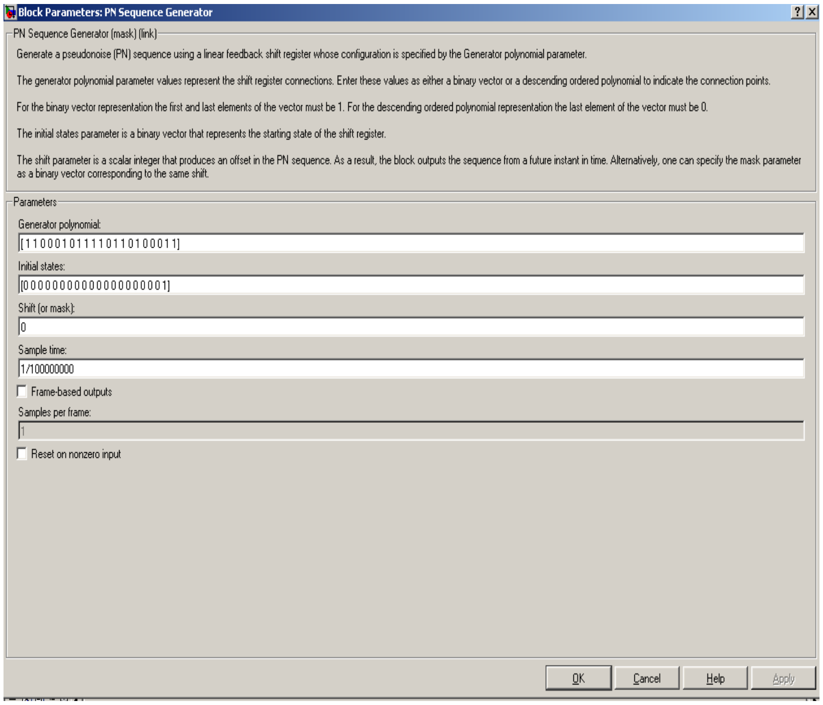
4.4 PN Sequence Generators:

The PN Sequence Generator block generates a sequence of pseudorandom binary numbers. The PN Sequence Generator block uses a shift register to generate sequences. All registers in the generator update their values at each time step according to the value of the incoming arrow to the shift register. The adders perform addition modulo 2. The Generator polynomial parameter can be specified using either of these formats:

- A vector that lists the coefficients of the polynomial in descending order of powers. The first and last entries must be 1. Note that the length of this vector is one more than the degree of the generator polynomial.
- A vector containing the exponents of z for the nonzero terms of the polynomial in descending order of powers. The last entry must be 0.

  For example, [100000101] and [8 2 0] represent the same polynomial, $p(z) = 1 + z^{-6} + z^{-8}$. The Initial states parameter is a vector specifying the initial values of the registers. The Initial states parameter must satisfy these criteria: All elements of the Initial states vector must be binary numbers. The length of the Initial states vector must equal the degree of the generator polynomial. At least one element of the Initial states vector must be nonzero in order for the block to generate a nonzero sequence. That is, the initial state of at least one of the registers must be non-zero. An external signal can be used to reset the values of the internal shift register to the initial state by selecting the Reset on nonzero input check box. This creates an input port for the external signal in the PN Sequence Generator block. The way the block resets the internal shift register depends on whether its output signal and the reset signal are sample-based or frame-based.

The following are the design specifications for the PN sequence generator.



Block Parameters: PN Sequence Generator

PN Sequence Generator (mask) (link)

Generate a pseudonoise (PN) sequence using a linear feedback shift register whose configuration is specified by the Generator polynomial parameter.

The generator polynomial parameter values represent the shift register connections. Enter these values as either a binary vector or a descending ordered polynomial to indicate the connection points.

For the binary vector representation the first and last elements of the vector must be 1. For the descending ordered polynomial representation the last element of the vector must be 0.

The initial states parameter is a binary vector that represents the starting state of the shift register.

The shift parameter is a scalar integer that produces an offset in the PN sequence. As a result, the block outputs the sequence from a future instant in time. Alternatively, one can specify the mask parameter as a binary vector corresponding to the same shift.

Parameters

Generator polynomial:
[1 1 0 0 0 1 0 1 1 1 1 0 1 1 0 1 0 0 0 1 1]

Initial states:
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1]

Shift (or mask):
0

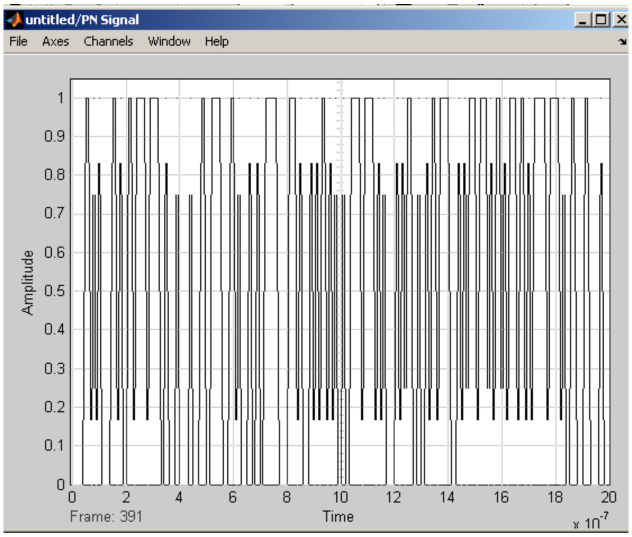
Sample time:
1/100000000

☐ Frame-based outputs

Samples per frame:
1

☐ Reset on nonzero input

Figure 29: Simulation of the PN Code generator showing the Code sequence.



4.5 Frequency Synthesizer:

The frequency synthesizer takes in the signal from the PN code generator as its input. Since the pseudo-random code produced provides a given set of discrete frequencies, the required frequency range is obtained from a variable voltage controlled oscillator (VCO) of the frequency generator, whose output is corrected by comparison with that of a reference source i.e. the PN sequence. A fractional-N frequency synthesizer was used in this case. The output of the frequency synthesizer is a multi-hop frequency signal that is to spread and/ or despread the information signal.

CHAPTER FIVE

5.0 SIMULATION AND ANALYSIS OF RESULTS

5.1 SIMULATION

The simulation begins with the generation of a frequency hopping signal using a pseudo random (PN) code sequence which determines the hopping frequency. The pseudo random (PN) code sequence is obtained by using the PN sequence generator simulink block set model, the output of this generator is the reference or input signal to the frequency synthesizer. An FSK modulated data signal is thus spread using the frequency hopping modulation technique by multiplying with the output signal of the frequency synthesizer. The channel adds white noise to the FH signal. At the receiver the signal received is despread and the information signal is recovered by correlating with the locally generated signal from the VCO. This is known as synchronization. The simulink models are shown below.

The simulink model shown below is the transmitter generating a frequency hopping signal. It consists of the information signal source, a frequency shift keying modulator, the PN code generator and the frequency synthesizer. The output of this model goes to the channel where white noise is added

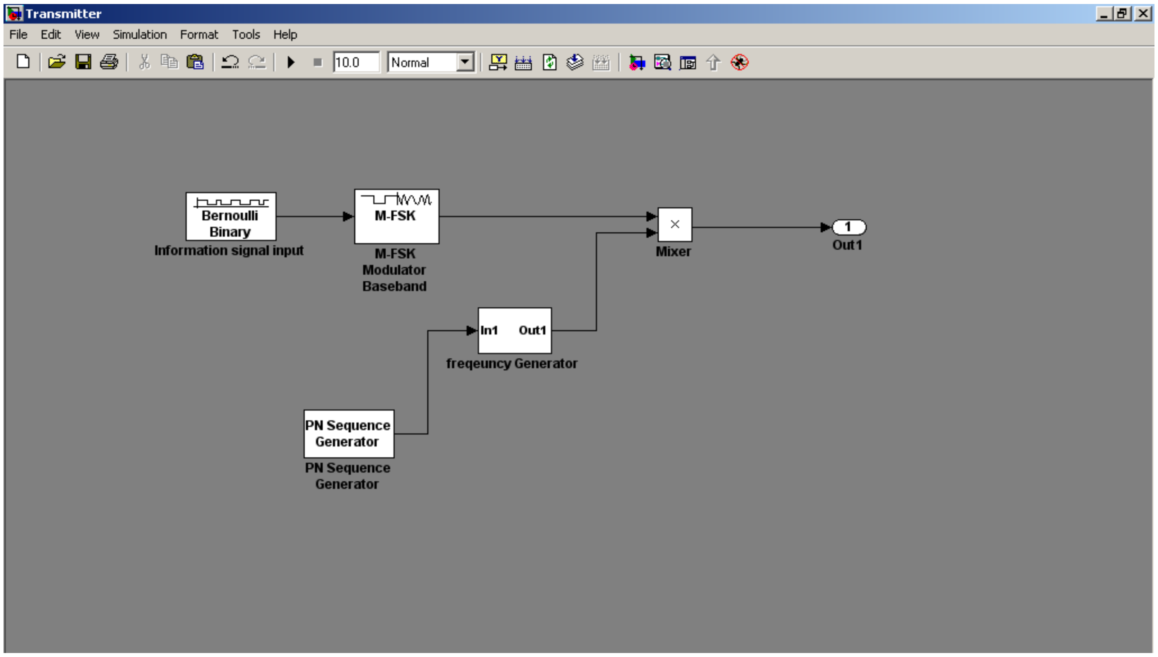


Figure 30: Frequency Hopping Signal Generation Simulink Model

The simulink model shown below is the synchronization system for FH-SS. It consists of a PN code generator, a frequency synthesizer, a despreader, a phase/frequency detector, a loop filter and a voltage control oscillator. At the input port 1 is the FH signal; this signal is despread by multiplying with the output signal of the frequency synthesizer. The PN code generator generates the same code sequence as the one that was used to spread the signal. The PFD detects the frequency and phase of the signal input to it and correlates with that locally generated by the VCO. The VCO starts by running at the quiescent frequency set to it and based on the error signal it acquires and tracks the frequency of the input reference signal hence obtaining synchronization.

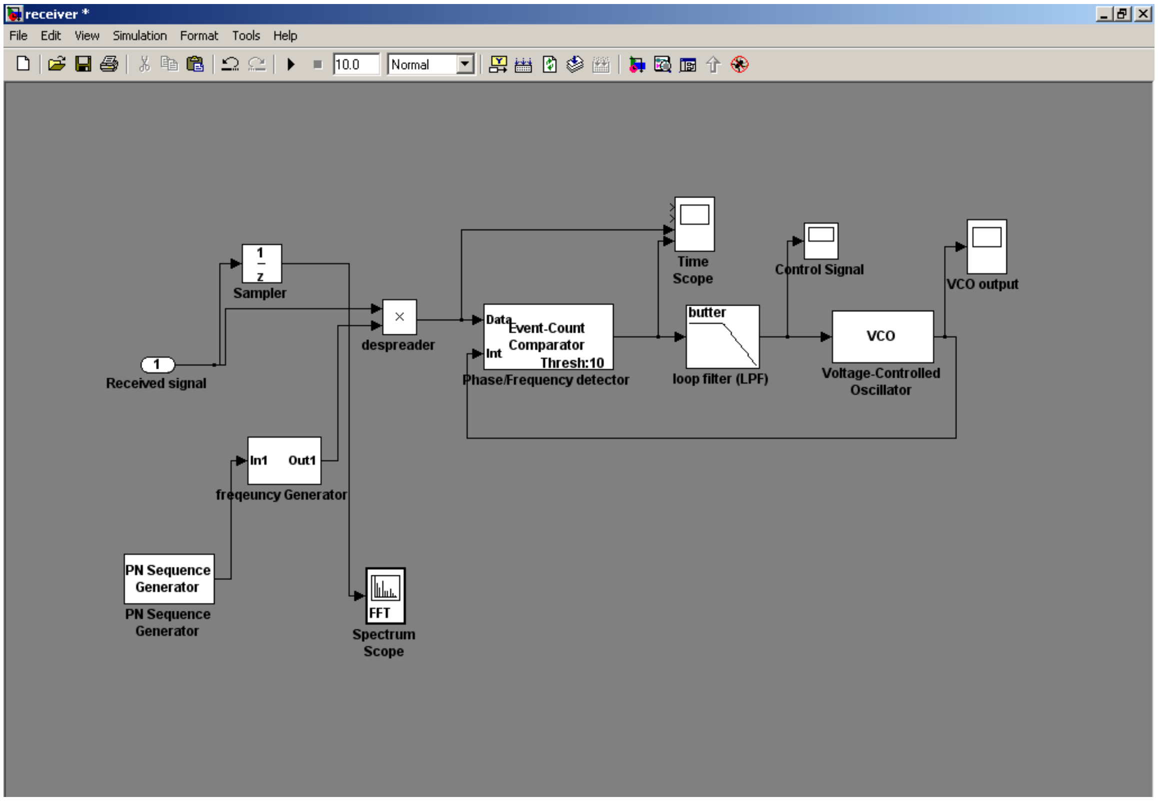


Figure 31: FH-SS Receiver Synchronization system Simulink model

5.2 RESULTS AND ANALYSIS

The FH spread signal spectrum is shown below. It has characteristics similar to that of white noise power spectrum, thus without the spreading code the jammer or hacker of a Frequency Hopping system will just be observing noise like spectrum.

Figure 32: Simulation results showing the FH Spread signal spectrum.

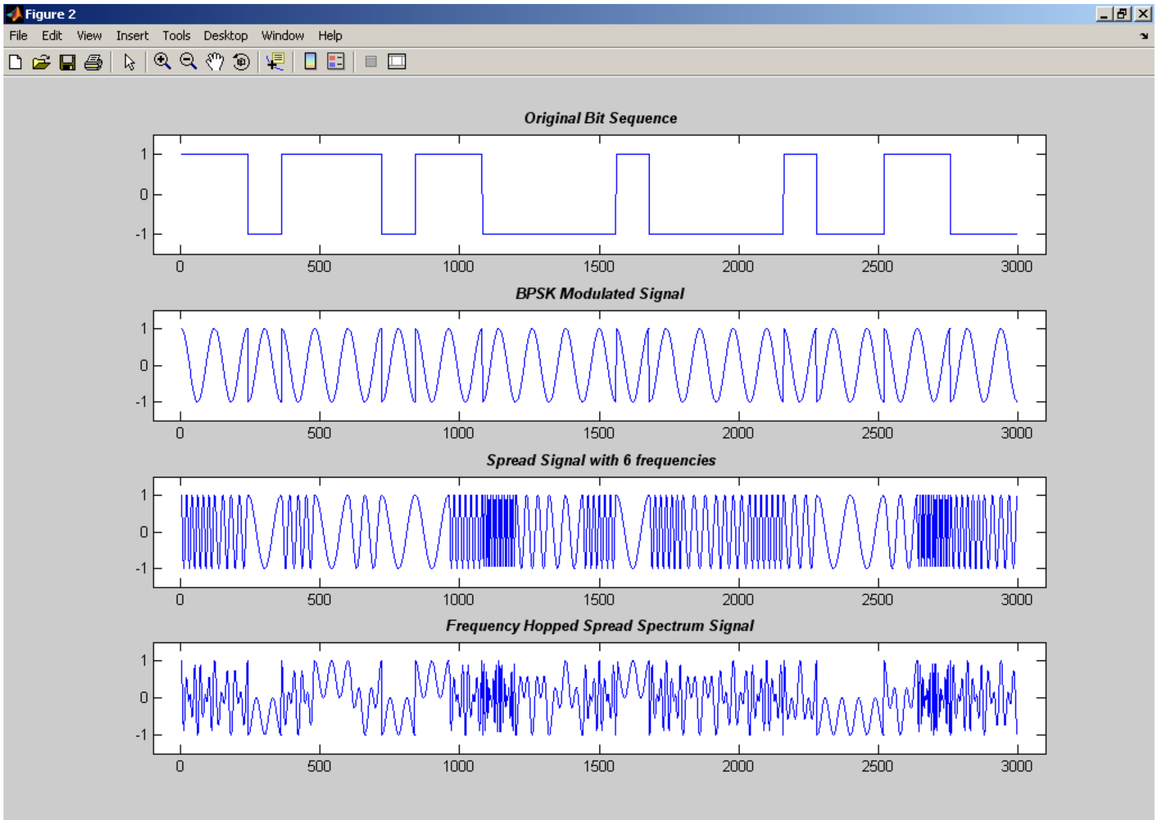


Figure 2
File Edit View Insert Tools Desktop Window Help
Original Bit Sequence
BPSK Modulated Signal
Spread Signal with 6 frequencies
Frequency Hopped Spread Spectrum Signal
1
0
-1
0
500
1000
1500
2000
2500
3000

The frequency synthesizer simulation results shown below were satisfactory. The results show the reference signal being the PN code sequence. The divided synthesized signal is the output of the frequency synthesizer VCO. These results were the same for both the transmitter and the receiver frequency synthesizers.

For the frequency hopping receiver synchronization Simulink model, the input signal and the VCO output signal were observed to synchronize as expected from the simulation of the model. The loop filter output control signal as shown in figure 45 was a constant amplitude dc signal when synchronization had been achieved. It was observed from the frequency estimators that at synchronization, the input and the VCO output signals were of the same frequency.

CHAPTER SIX

6.0 CONCLUSION AND RECOMMENDATION

It is worth noting that for a successful synchronization of Frequency Hopped signals, there is need to have the timing information of the transmitted signal in order to despread the received signal and demodulate the despread signal. Therefore the process of acquiring the timing information of the transmitted Frequency Hopping spread spectrum signal and the sequential tracking of the signal is essential in the design and implementation of a synchronizing system for FH-SS.

Frequency Hopping Spread Spectrum technique provides improved security, resistance to jamming, efficient bandwidth sharing, and resistance to fading, it is employed to combat interference and allow simultaneous users to share the finite radio spectrum without interfering with one another.

This report has provided a summary of the work performed during the research, design and implementation of a synchronization system for FH-SS. This project has been a success in describing the acquisition and tracking of frequency hopping signals. In synchronization, the concepts of PLL techniques have been widely exploited; more precisely the PLL model has been used to synchronize an output generated by the oscillator with the reference input signal in frequency and in phase. The linear feedback shift registers (LFSR) technique has been used in the PN code sequence generation.

As a recommendation the project could be improved by generating PN code sequences of the maximum possible length for a fixed degree of the generator polynomial of the PN code generator block set. Also the use could be made of the error correcting codes to improve the performance of Frequency Hopping Spread Spectrum communication system model.

REFERENCES

1. Raymond L. Pickholtz, Donald L. Schilling, Laurence B. Milstein. *"Theory of Spread Spectrum Communications — A Tutorial,'"* IEEE Transactions on Communications, Vol.COM-30, May 1982, pp. 855-884.
2. Robert Dixon. *Spread Spectrum Communications.,* Second Edition, John Wiley and Sons, New York, 1984.
3. Roland E. Best. "Phase-Locked Loops: Design, Simulation and Application" McGraw-Hill, 1997.
4. J.G. Proakis, *Digital Communications, 2nd* Ed. New York: McGraw-Hill, 1995.
5. M. K. Simon and A. Polydoros, "Coherent detection of frequency hopped Quadrature modulations in the presence of jamming. *IEEE Trans. Commun.,* vol. COM-29,pp. 1644-1668, Nov. 1981.
6. R. A. Scholtz, 'The origins of spread-spectrum communications,' This issue, pp. 822-854.
7. M.K Simon 'Spread Spectrum Communications hand book', Me Graw-Hill, Inc. 1994
8. Viterbi, Andrew J. 'CDMA Principles of Spread Spectrum Communications.' Reading Mass. Addison- Wesley, 1995.
9. Http/www. Math works, com.
10. SIMULINK User's Guide, The Mathworks, Inc. 24 Prime park way, Natick,
11. Mass (August 1995).

APPENDIX

ACRONYMS

ADPLL	All Digital Phase Lock Loop
BER	Bit Error Rate
BPSK	Binary Phase Shift Keying
CDMA	Code Division Multiple Access
DPLL	Digital Phase Lock Loop
DQPSK	Differential Quaternary Phase Shift Keying
DS-SS	Direct Sequence Spread Spectrum
FCC	Federal Communication Commission
FH-SS, FH	Frequency Hopping Spread Spectrum, Frequency Hopped
FM	Frequency Modulation
G-FSK	Gaussian Frequency Shift Keying
GNSS	Global Navigation Satellite Systems
GPS	Global Positioning Systems
IEEE	Institute of Electrical and Electronic Engineers
ISM	Industrial, Scientific & Medical
LFSR	Linear Feedback Shift Registers
LPLL	Linear Phase Lock Loop
M-FSK	M-ary Frequency Shift Keying
PCM	Pulse Code Modulation
PFD	Phase and Frequency Detector
PLL	Phase Lock Loop
PN	Pseudo-random Noise
RASE	Rapid Acquisition by Sequential Estimation
RF	Radio Frequency
SPLL	Software Phase Lock Loop
SS	Spread Spectrum
SNR	Signal to Noise Ratio
VCO	Voltage Control Oscillator

CODE:

```
%%%%%%%%%%%%%%%%%%%%%%
% Frequency Hopping Spread Spectrum
%%%%%%%%%%%%%%%%%%%%%%

% Generation of bit pattern
s=round(rand(1,25));    % Generating 25 bits
signal=[];  % empty array for signal
carrier=[]; % empty array for carrier
t=[0:2*pi/119:2*pi];     % Creating 60 samples for one cosine
for k=1:25                       % making the digital signal or data signal
    if s(1,k)==0
        sig=-ones(1,120);    % 120 minus ones for bit 0
    else
        sig=ones(1,120);     % 120 ones for bit 1
    end
    c=cos(t);   % cos(t) is my carrier
    carrier=[carrier c];% concatenation
    signal=[signal sig];
end
subplot(4,1,1);  %simple ploting
plot(signal);
axis([-100 3100 -1.5 1.5]);
title('\bf\it Original Bit Sequence');

% BPSK Modulation of the signal
bpsk_sig=signal.*carrier;   % Modulating the signal by using point by point
multiplication
subplot(4,1,2);
plot(bpsk_sig)
axis([-100 3100 -1.5 1.5]);
title('\bf\it BPSK Modulated Signal');

% Preparation of 6 new carrier frequencies
t1=[0:2*pi/9:2*pi];
t2=[0:2*pi/19:2*pi];
t3=[0:2*pi/29:2*pi];
t4=[0:2*pi/39:2*pi];
t5=[0:2*pi/59:2*pi];
```

```
t6=[0:2*pi/119:2*pi];
c1=cos(t1);
c1=[c1 c1 c1 c1 c1 c1 c1 c1 c1 c1 c1 c1];
c2=cos(t2);
c2=[c2 c2 c2 c2 c2 c2];
c3=cos(t3);
c3=[c3 c3 c3 c3];
c4=cos(t4);
c4=[c4 c4 c4];
c5=cos(t5);
c5=[c5 c5];
c6=cos(t6);

% Random frequency hops to form a spread signal
spread_signal=[];
for n=1:25
   c=randint(1,1,[1 6]);
   switch(c)
      case(1)
         spread_signal=[spread_signal c1];
      case(2)
         spread_signal=[spread_signal c2];
      case(3)
         spread_signal=[spread_signal c3];
      case(4)
         spread_signal=[spread_signal c4];
      case(5)
         spread_signal=[spread_signal c5];
      case(6)
         spread_signal=[spread_signal c6];
   end
end
subplot(4,1,3)
plot([1:3000],spread_signal);
axis([-100 3100 -1.5 1.5]);
title('\bf\it Spread Signal with 6 frequencies');

% Spreading BPSK Signal into wider band with more frequencies
freq_hopped_sig=bpsk_sig.*spread_signal;
subplot(4,1,4)
```

```
plot([1:3000],freq_hopped_sig);
axis([-100 3100 -1.5 1.5]);
title('\bf\it Frequency Hopped Spread Spectrum Signal');

% Expressing the FFTs to know about the frequency content of the signal
figure,subplot(2,1,1)
plot([1:3000],freq_hopped_sig);
axis([-100 3100 -1.5 1.5]);
title('\bf\it Frequency Hopped Spread Spectrum signal and its FFT');
subplot(2,1,2);
plot([1:3000],abs(fft(freq_hopped_sig)));
```