Filtering Techniques for Mitigating Microwave Oven Interference on 802.11b Wireless
Local Area Networks
by
Lorenzo M. Lorilla

ABSTRACT

With the increasing popularity and assimilation of wireless devices into the everyday
lives of people, the issue of their feasibility for coexisting with other radio frequency
(RF) devices arises. Particularly strong interferers for the IEEE 802.11b standard are
microwave ovens, since both operate at 2.4 GHz. The interference mitigation techniques
all exploit the differences between the interference and the signal, since the former is
sinusoidal in nature while the latter can be viewed as noise. The first mitigation filter
operates in the frequency domain and filters the received signal's Fast Fourier Transform
(FFT) sequence by detecting and removing peak sinusoidal components over the flat 3-
dB bandwidth of the signal. The second is a Least Mean Square (LMS) Adaptive filter
that produces an estimate of the interference through a recursive approximation method
and subtracts it out from the received signal. The third and last is the Adaptive Notch
Filter (ANF) which implements a lattice structure and has a time-varying notch frequency
parameter that converges to and tracks the frequency of the interference in the received
signal. The three filters are shown to produce improvements in the bit error rate (BER)
and frame error rate (FER) performance of the receiver under various relative strengths of
the signal with respect to the interference.

Acknowledgements

First and foremost, I would like to thank the Lord God for giving me the strength, determination and persistence in taking on and completing this thesis. I want to thank my supervisor at Mobilian, Rob Gilmore for making this whole thesis possible, for believing in my abilities and having enough faith in me to fund the project, for providing inspiration, motivation and much needed support that came in every possible form. I also would like to extend my appreciation to my thesis adviser Professor Charlie Sodini, for giving valuable insight and feedback on my research and for making sure that I never compromised the quality of my work.

I would also like to thank Eladio Arvelo, for brainstorming the idea for the thesis, for providing excellent advice from the beginning to the very end, and for answering the myriad questions I had even after I had left San Diego. The same heartfelt appreciation goes out to Warit Wichakool, Allan Lum, Andrew Huang, Oguz Silahtar, and Peter Agboh, who all took the time off work to listen to my ideas and guide me along the way. I also want to thank Darren Orloff and Mark Erikson for the technical help in setting up my experiments.

Lastly, I would like to thank my family and friends for being there with me throughout this whole process, for making sure I got my work done while keeping things in perspective!

Table of Contents

Chapter 1: Introduction

1.1. Introduction

In recent years, the wireless LAN market has emerged as one of the fastest-growing sectors of the communications industry. Wireless networks have found their way into a whole range of industries, such as healthcare, manufacturing, and transportation. With wireless connectivity, doctors can access patients' information at their bedside and update their medical records on the fly, while retailers can roam about their warehouses and manage their inventories and record transactions as they happen.

The IEEE 802.11 standard-based wireless LAN market is a $1 billion industry that is expected to grow five-fold over the next four years. Over this time period, wireless networks are expected to gain a strong presence in public places such as airports, hotels, and coffee shops, where an estimated 21 million users will be connecting to 41,000 wireless public access points across the country [1]. In the near future, wireless connectivity will be ubiquitous throughout cities and towns; projects like the Wireless Athens Group Zone (WAGzone), a wireless network covering over 24 city blocks in Athens, Georgia, are already pioneering the assimilation of wireless technology into communities [2].

Corporate firms currently consume the largest share of the wireless LAN market, and are 'unwiring' their office buildings in order to provide wireless Internet access throughout their facilities. Wireless connectivity provides a big boost to productivity in enterprise environments, since employees are given the ability to access and exchange information whenever they need to, wherever they are.

Although wireless LANs in public places are expected to increase in numbers, analysts predict the home market to experience the biggest growth over the next few years [3]. More home users will eventually replace their conventional internet connections with wireless enabled laptops and personal digital assistants (PDAs) that will allow them to surf the Internet or check e-mail from anywhere at home. Furthermore, home entertainment devices such as video pads and game console controllers will eventually be replaced by their wireless counterparts as more home applications are found for wireless technology.

As wireless devices increasingly crowd personal spaces such as homes and offices, the issue of their coexistence with other radio frequency (RF) devices arises. Since there are specific allotted bands for the operation of various RF devices, interference with each other is inevitable. Although the IEEE 802.11 wireless standard provides some inherent level of resistance to various types of interference, it does not give sufficient protection from other RF devices that produce strong interference. Rather, an active stance to mitigating the interference needs to be taken.

1.2. Problem Definition

The IEEE 802.11 standard outlines the Medium Access Control (MAC) and Physical (PHY) Layer specifications of a local area network, and offers several different implementations at the physical layer. The 802.11b standard, which has recently been gaining the widest acceptance among the various 802.11 extensions, employs Direct Sequence Spread Spectrum (DSSS) technology and operates in the Industrial, Scientific, and Medical (ISM) band from 2.4 to 2.4835 GHz.

12

Since the band is unlicensed, many other radio frequency devices also operate in the same frequency range and can interfere with 802.11b networks. Wireless LANs in home and office environments are bound to encounter interference from same-band devices such as microwave ovens and the new generation of cordless phones which operate at 2.4 GHz. The interference may degrade the networks' performance and even make communication impossible.

Although microwave ovens are designed to confine radiation inside their cooking areas, their seals and door panels wear out with time and allow microwaves to leak out. These emissions have been known to severely interfere with wireless LAN devices.
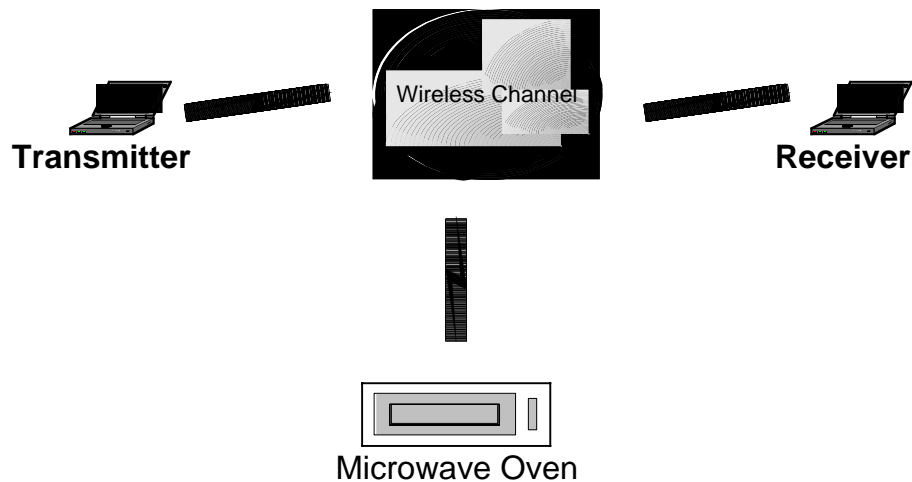
Figure 1.1. A basic illustration of the interference problem. The microwave oven leaks energy which interferes with signals at the receiver and forces errors in demodulation and detection.

The interference corrupts transmitted packets by forcing errors in demodulation and bit detection at the receiving end. A cyclic redundancy code (CRC) field in each packet frame allows the receiver to detect the presence of bit errors; errors in particular fields of the packet frame will cause the receiver to drop the packet and fail to send an

acknowledgement to the transmitter that the packet was received intact. If the transmitter does not receive an acknowledgement from the receiver, then it assumes that the packet was discarded and attempts to resend it at a later time.

If the interference is persistent, it may take several retransmissions before a packet is correctly received, and data throughput is reduced down between the two communicating stations. In some cases, the interference may be too pervasive that packets will never be received intact despite repeated transmissions, and as a result the connection between the two stations effectively dies.

Fortunately, the interference caused by microwave ovens can be characterized; studies have shown that microwave oven emissions are periodic in nature and exhibit somewhat predictable behavior. By exploiting these distinct properties, it may be possible to develop filters that will distinguish them from 802.11b signals and effectively cancel them out.

1.3. Thesis Focus

This thesis focuses on devising and analyzing different methods of mitigating residential microwave oven interference on IEEE 802.11b systems. The first phase of the research involved carrying out experiments in order to characterize microwave oven interference and assess the degree of throughput degradation that they cause on 802.11b systems.

The next phase consisted of developing the entire software simulation environment in Matlab which included a fully functional 802.11b system model at the physical layer (PHY), where an arbitrary data payload could be formatted into a packet,

transmitted as a waveform and demodulated at the receiver. The system model followed specifications outlined in Mobilian's High Level Design (HLD) system document.

In addition to modeling the system, the wireless channel through which the packet travels through was modeled. The channel included background noise along with the microwave oven interference, which was modeled using the data gathered from the experiments. The 802.11b system model was integrated with the wireless channel and interference model to produce the simulation environment.

The third phase involved the development of three different interference mitigation filters. The filters fall under two categories:

- Transform Domain Filtering (TDF) – filtering of the interference in the frequency domain, using its Fourier transform.
- Adaptive Filtering – Tracking, reconstructing, and subtracting or notching out the interference using time-varying filters.

Simulations were carried out on the filters in order to assess how well each performed depending on the relative strength of the signal to the interference and background noise. The research is concluded by a comparative analysis on the performance of the different methods and their feasibilities.

1.4. Thesis Outline

The thesis starts off with Chapter 2, which provides an overview of the IEEE 802.11b standard. Chapter 3 briefly discusses the 802.11b system and wireless channel models that were developed for the simulations, while Chapter 4 details the microwave oven interference tests, interference measurements and simulation model.

The succeeding three chapters present the interference mitigation techniques: Chapter 5 presents the Transform Domain Filter, Chapter 6 covers the Least Mean Square (LMS) Interpolation Filter, and Chapter 7 presents the Adaptive Lattice Notch Filter. Chapter 8 describes the simulations carried out on the filters and presents the results and a comparative analysis of their performance. Chapter 9 summarizes the research, cites shortcomings and lays the groundwork for possible future research on this particular problem.

Chapter 2: The IEEE 802.11b standard

2.1. Introduction

The IEEE Executive Committee created the 802.11 working group in 1990 to address the need for a wireless LAN standard. In 1997, the group presented the IEEE 802.11 standard, which defined the medium access control (MAC) and physical layer (PHY) specifications. The standard specifies a series of protocols for handling mobile, portable, or fixed-location users, and sets up parameters for such a network.

The standard was set to operate in the unlicensed 2.4GHz Industrial, Scientific, and Medical (ISM) band and provides data rates of 1 megabit per second (Mbps) and 2 Mbps, using either frequency hopping spread spectrum (FHSS), direct sequence spread spectrum (DSSS) technology or infrared modulation.

2.2. The IEEE 802.11b Standard

Higher payload data rate extensions were made to the physical layer of the standard in 1999, creating the 802.11a and 802.11b standards. The 802.11a standard operates at the 5GHz Unlicensed National Information Infrastructure (U-NII) band, which is composed of three sub bands with different power requirements, and employs orthogonal frequency division modulation (OFDM), allowing it to attain data rates of up to 54 Mbps.

The 802.11b standard, otherwise known as the Wireless Fidelity or "Wi-Fi" standard, operates in the 2.4GHz range and uses Complementary Code Keying (CCK) modulation to achieve payload data rates 5.5 and 11 Mbps in addition to 1 and 2 Mbps.[1]

The standard implements direct sequence spread spectrum (DSSS) technology, a technique which spreads the data signal over a much wider bandwidth than its own for transmission. Although this is inefficient in terms of bandwidth usage, it compensates by providing a *processing gain* which increases the receiver's resistance to interference.

Although the 802.11b standard uses lower data rates than the 802.11a standard, it has a much longer range than the latter. 802.11b devices have an indoor range of up to several hundred feet, while 802.11a devices have ranges in the order of tens of feet since they are required to have the same output power of 20 dBm as 802.11b but operate at a higher frequency, which results in greater path loss for any distance.

2.3.   Overview of the MAC Layer

The IEEE 802.11 is required to appear as any other wired IEEE 802 LAN to the higher-level logical link control (LLC) layer. Thus, the MAC sublayer is given the functionality to handle station mobility. These services include association and disassociation with a mobile station's coverage area, authentication of a station, and distribution of information. Other services of the MAC sublayer include data fragmentation, frame error checking, and power management.

The MAC uses carrier sense multiple access with collision avoidance (CSMA/CA), which senses the channel for other activity and avoids collisions by waiting

---

[1] An optional feature of the standard is using Packet Binary Convolutional Coding (PBCC) instead of CCK modulation. However, this was not within the scope of this thesis.

for a random period of time before transmitting a packet. In line with this scheme, the MAC also implements frame receipt acknowledgement: the receiving station, upon the receipt of a packet that is intact, sends an acknowledgement packet (ACK) back to the transmitting station. Acknowledgement packets are used primarily to detect packet collisions or other types of interference during transmission.

2.4. The 802.11b Physical Layer

The interference mitigation techniques explored in this thesis are signal processing-based and are implemented in the physical layer (PHY) of the 802.11b system. The PHY is the lowest of the seven hierarchical layers specified in the Open Systems Interconnect (OSI) reference model, and performs services requested by the MAC sublayer. The layer regulates the transmission and reception of packets through the wireless medium. Its main set of functions includes processing packet frames, encoding bits, modulating them onto a waveform, and establishing and terminating connections to the wireless medium.

2.4.1. PHY Sublayers

The physical layer is composed of two sublayers, namely the Physical Layer Convergence Procedure (PLCP) and the Physical Medium Dependent (PMD) sublayer. The PLCP acts as the interface between the MAC sublayer and PMD sublayer. The MAC sublayer hands down MAC Protocol Data Units (MPDU) to the PLCP for transmission. The PLCP packages the MPDUs into packet frames called PHY Protocol Data Units

(PPDU) and passes them on to the PMD sublayer for modulation and transmission. The

reverse processes occur at the receiver.

MAC Sublayer	
Physical (PHY) Layer	PLCP Sublayer
	PMD Sublayer
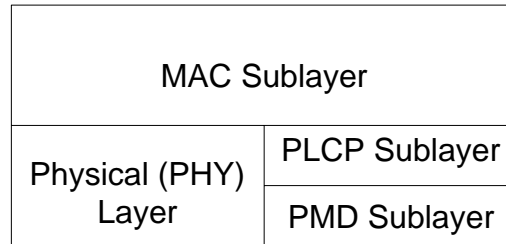
Figure 2.1. A hierarchical view of the MAC and PHY layer. The PLCP sublayer interfaces the PMD sublayer with the MAC layer.

2.4.2. PLCP Frame Formats

The standard defines two PLCP frame formats, *long* and *short*. All WLAN

devices are required to support the long format, while support for the short format is

optional. The short format was created to minimize overhead and maximize network

throughput. Both types of frames consist of three parts: the preamble, header, and the

payload, otherwise known as the PLCP service data unit (PSDU), which is the MPDU at

the physical layer. The two formats slightly differ in their preambles and headers, as well

as in the data rates that can be used for payload transmission.

2.4.2.1.   The Long PLCP Format

The long PLCP format is illustrated in Figure 2.2. The preamble consists of two

fields, namely the synchronization (SYNC) field which is a 128-bit long sequence of 1's,

and the start frame delimiter (long SFD) field which is a 16-bit long code sequence. The

SYNC field is used by the receiver to detect the presence of the signal and find the proper

timing synchronization, while the long SFD field identifies the packet as having a long

format. The whole preamble is a total of 144 bits long and is transmitted at 1 Mbps.

longSYNC 128 bits	longSFD 16 bits	SIGNAL 8 bits	SERVICE 8 bits	LENGTH 16 bits	CRC 16 bits

long PLCP Preamble 144 bits at 1 Mbps	long PLCP Header 48 bits at 1 Mbps	PSDU Variable at 1,2, 5.5, or 11 Mbps
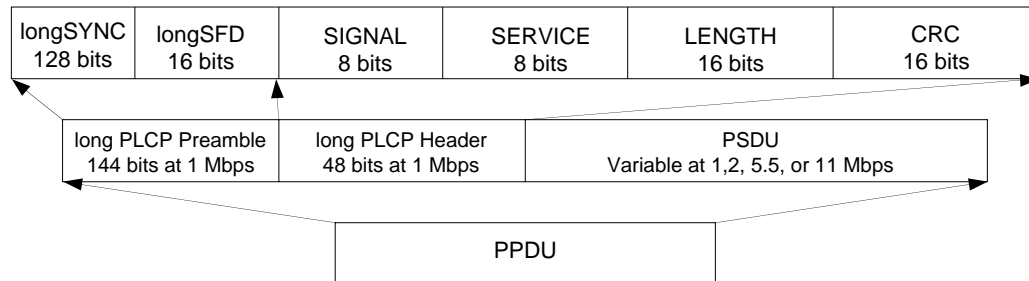
PPDU

Figure 2.2. The Long PLCP PPDU Format. Source: [4]

The header is 48 bits long and consists of four fields, starting with an 8-bit

SIGNAL field, which indicates the data rate being used for the payload (PSDU), an 8-bit

SERVICE field reserved for miscellaneous purposes, a 16-bit LENGTH field which

indicates the length of the payload portion in microseconds, and a 16-bit CCITT[2] Cyclic

Redundancy Check (CRC) field, which is a checksum derived from the first 3 header

fields and indicates the integrity of the header. The checksum is recomputed on the 3

fields at the receiver and is compared with the CRC field. If the checksum equals the

CRC field, the header fields are valid and the payload is demodulated. Otherwise, the

packet is dropped.

For this format, the header is also transmitted at 1 Mbps. The payload, handed

down from the MAC layer, can be transmitted using any of the four data rates and can be

up to 2346 bytes in length [4].

---

[2] CCITT stands for the Comite Consultatif Internationale de Telegraphie et Telephonie, a Geneva-based international committee that recommends telecommunications standards, including audio compression standards and modem speed and compression. The group recently changed its name to the ITU-T (International Telecommunications Union-Telecommunication).

2.4.2.2.   The Short PLCP Format

The short frame format differs slightly from the long format in several respects. The SYNC field is a shorter 56-bit sequence of 0's, while the SFD field is a time-reversed copy of the long SFD field. Similar to the long PLCP format, the preamble is transmitted at 1 Mbps. While the header format is the same, it is sent at 2 Mbps. Furthermore, the 1 Mbps data rate is not used for payload transmission in this format.

shortSYNC 56 bits	shortSFD 16 bits	SIGNAL 8 bits	SERVICE 8 bits	LENGTH 16 bits	CRC 16 bits

short PLCP Preamble 72 bits at 1 Mbps	short PLCP Header 48 bits at 2 Mbps	PSDU Variable at 2, 5.5, or 11 Mbps
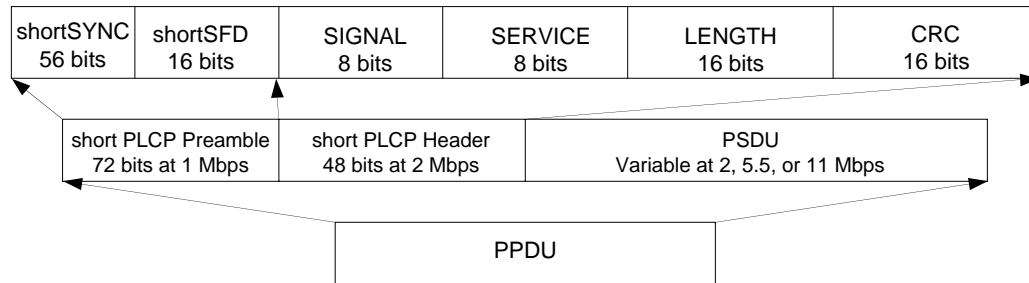
PPDU

Figure 2.3. The Short PLCP PPDU Format. Source: [4]

2.5.   Channel Encoding

After the packet is generated, the bits are passed through a scrambler (see Appendix A.1) and are channel-encoded onto a symbol constellation. Different channel-encoding schemes are used to achieve each of the four data rates.

Differential Binary Phase Shift Keying (DBPSK) is used for 1 Mbps, while a similar scheme, Differential Quadrature Phase Shift Keying (DQPSK), produces a data rate of 2 Mbps. Although both encoding schemes use the same symbol constellations as their non-differential counterparts, they determine the succeeding output symbol by mapping the input bit/s to phase changes from the current output symbol.

The higher data rates of 5.5 and 11 Mbps employ 8-chip complementary code keying (CCK) for modulation. CCK is a coded form of Quadrature Phase Shift Keying (QPSK), where every block of incoming bits is mapped to four phase values: ϕ_1, ϕ_2, ϕ_3 and ϕ_4. Each set of the 8 complex–valued chips correspond to one output symbol, where each chip is uniquely defined by a particular subset of the phase values and is of the form $\pm e^{j(\sum \phi)}$.

The optional packet binary convolutional coding (PBCC) scheme can be used in place of CCK encoding, but is not within the scope of this thesis. For a more detailed explanation of each modulation scheme, refer to Appendix A.2.

2.6.   Direct Sequence Spread Spectrum (DSSS) Technology

A key feature of the 802.11b standard is its use of direct sequence spread spectrum technology. Spread spectrum technology, as its name implies, is a technique that spreads the signal over a wide range of the spectrum for transmission.

There are two different implementations of spread spectrum, namely Frequency Hopping Spread Spectrum (FHSS) and Direct Sequence Spread Spectrum (DSSS). FHSS transmits a narrowband signal at a carrier that 'hops' from one frequency to another in a hopping pattern determined by a pseudo-noise (PN) sequence. In effect, a wide range of the spectrum is used for transmission.

In DSSS, a completely different concept of spreading is performed: the PN sequence is multiplied with each symbol coming out of the encoder. Since the PN sequence is clocked at a higher frequency than the symbols, the resulting signal has a higher bandwidth than the original.

In the 802.11b standard, direct-sequence spreading is performed on symbols coming from the DBPSK and DQPSK encoders. No spreading is done for 5.5 and 11 Mbps; the symbol rates of the CCK encoders are configured such that their outputs already occupy the bandwidth as that of the spread signals.

2.6.1. Barker/Complementary Codes

A special pseudo-noise sequence called a Barker code is used for spreading. The code consists of 11 'chips', where each chip takes a value of +1 or –1. The code is defined as:

$$+1,-1,+1,+1,-1,+1,+1,+1,-1,-1,-1 \qquad (2.1)$$

The Barker code, along with the symbols produced in CCK encoding, belong to a larger group known as polyphase complementary codes, which have very desirable properties for digital communications. Complementary codes possess good Euclidean distances between each other in their symbol constellation, which results in low bit error rates in multipath environments [5]. Furthermore, the codes have periodic autocorrelative vector sums that peak at the zero shift and have low values everywhere else.[3] Figure 2.4 shows the autocorrelation function of the Barker code.

---

[3] The periodic autocorrelative vector sum is the autocorrelation function of a periodic sequence of the polyphase code.
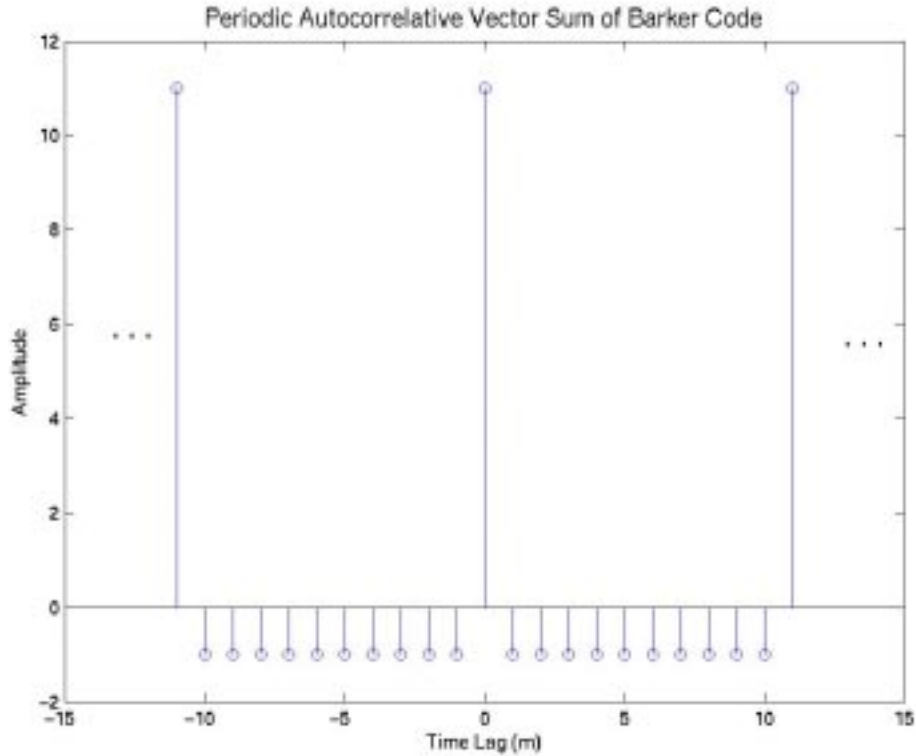
Figure 2.4. The periodic autocorrelation function of the Barker code repeats with the length of the Barker code. Peaks can be observed for zero-shifts, while a -1 can observed for all other shifts.

2.6.2. Spreading

To spread the signal, the Barker code is multiplied with each symbol coming out of the DBPSK and DQPSK modulators. The chipping rate is set at 11 MHz, while the symbol rate is at 1 MHz; one whole period of the Barker sequence coincides with one symbol period. Since the symbols are modulated onto the Barker sequence, which has a higher symbol rate, the signal's bandwidth is spread from 1 MHz to 11 MHz. Figure 2.5 shows an example of a bit stream being DBPSK encoded and spread using a Barker sequence.
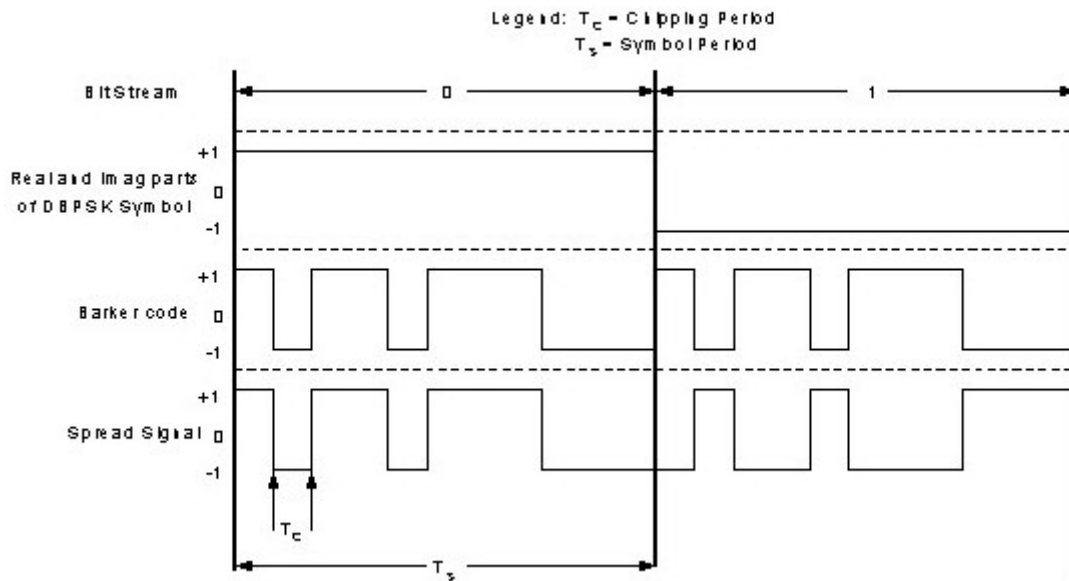
Figure 2.5. The bits are DBPSK modulated. The symbols are then spread using the Barker code that produces the spread signal at the bottom. The symbol rate (T_s) is 11 times the chipping rate (T_c). Source: [6].

2.6.3. Despreading

The reverse process of despreading is performed by sampling and correlating the received signal with the Barker code. The receiver slides the signal and computes the deterministic correlation value of the overlapping portion with the Barker code. Given the nature of the code's periodic autocorrelative vector sum, a peak correlation value will appear if it is aligned directly on top of a spread signal, while lower values will be observed for other shifts.

During the *acquisition phase*, the receiver uses the despreader output to determine the proper timing synchronization of the signal. That is, it tries to find the right time shift for correlating in order for successive demodulation to be performed only at multiples of the time shift.
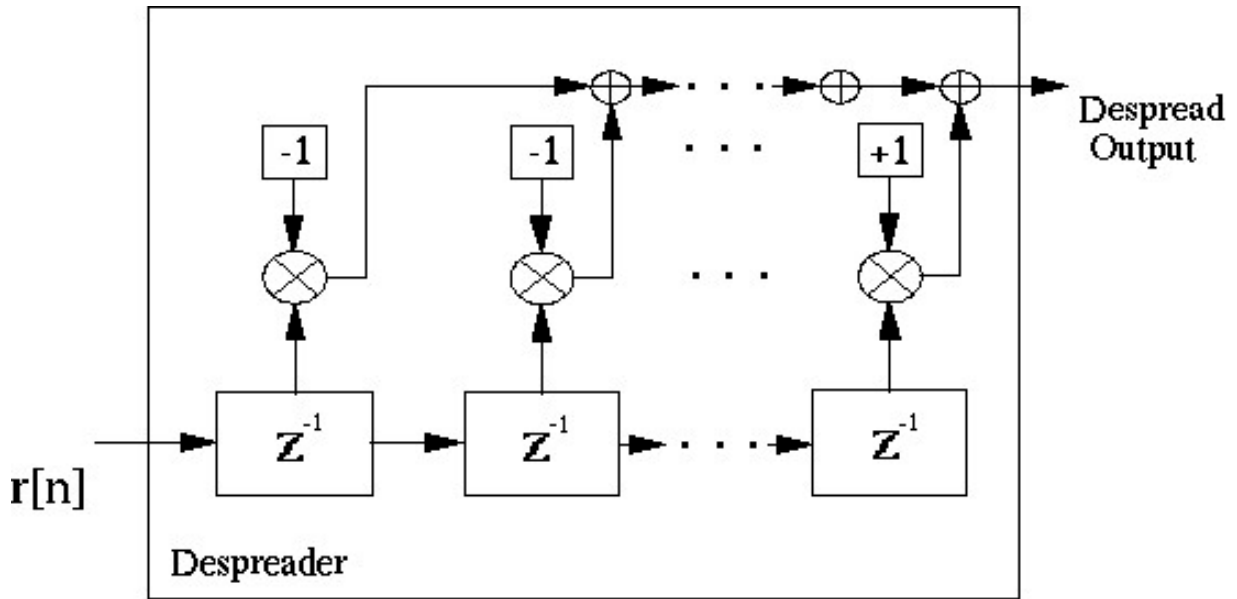
Figure 2.6. The despreader correlates the received signal with the reverse Barker code, since the first chip comes first in time.

2.6.4. Processing Gain

The key advantage that a DSSS system provides is additional resistance to interference as a result of spreading the signal. The amount of spreading it performs is measured by the system's *processing gain*. The processing gain, measured in decibels (dB), is defined as the ratio of the spreading bandwidth to the original signal bandwidth. This can simply be measured by the ratio of the number of chips per symbol, R_c, to the number of bits encoded in one symbol, R_b. The equation is normalized by a factor of 10.

$$G_p = 10\log_{10}\left(\frac{R_c}{R_b}\right) \quad (2.2)$$

The processing gain is an important measure of the system since it gives an indication of the system's resistance to interference; the higher the processing gain, the better the chances of recovering a signal in the presence of a jammer or noise.

Let $d[n]$ be the symbol sequence to be transmitted, and $p[n]$ the sequence of concatenated Barker codes used to spread it. The spread signal can be expressed as $s[n] = d[n]p[n]$.[4] Without loss of generality, assume that $s[n]$ is the transmitted signal; the pulse-shaping stage is bypassed for now. Assuming that the signal passes through an AWGN channel with noise $\zeta[n]$ and an additional narrowband jammer $j[n]$, the signal at the receiver, $r[n]$, can be expressed as:

$$r[n] = s[n] + j[n] + \zeta[n] \qquad (2.3)$$

As the despreader correlates $r[n]$ with the Barker code, $d[n]$ is recovered while $j[n]$ is spread or whitened. The correlation process increases the relative strength of the data signal $s[n]$ to $j[n]$, since the latter's power density is decreased because it is spread over a wider bandwidth. Also, despreading will have no effect on AWGN, since it is by definition already white. The first step in correlating involves multiplying $r[n]$ with $p[n]$:

$$p[n]r[n] = d[n]p[n]p[n] + \zeta[n]p[n] + j[n]p[n] \qquad (2.4)$$

$$p[n]r[n] = d[n] + \zeta[n]p[n] + j[n]p[n] \qquad (2.5)$$

---

[4] Assume that each Barker code chip takes one time instant, and thus each DBPSK/DQPSK symbol lasts for 11 time instants.

We note that the last term on the right in equation 2.5 indicates that the jammer is spread by the Barker code, while $d[n]$ is recovered since $p[n]p[n]=1$.

From the definition of G_p in equation 2.2, it can be inferred that the various data rates will have different processing gains. The processing gains given by the four data rates are listed in the table 2.1.

Data Rate (Mbps)	Number of Chips per symbol	Number of Bits per symbol	Processing Gain (dB)
1	11	1	10.41
2	11	2	7.40
5.5	8	4	3.01
11	8	8	0

Table 2.1: The processing gains of the different data rates are listed. As the data rate increases, there is less spreading which results in a smaller processing gain.

Since CCK encoding packs more bits into one symbol (see Appendix A.2), it gives a smaller processing gain for 5.5 Mbps, and no gain at all for 11 Mbps. However, CCK modulation is a coded form of QPSK, and provides a coding gain of around 1.6 dB at a bit error rate (BER) of 10^{-5}.[5]

2.7. Pulse-Shaping Baseband Filter

In order to create the actual baseband waveform that will be upconverted to 2.4GHz and transmitted, the symbols are passed through a pulse-shaping filter. That is, each symbol, having a value of either +1 or -1, is modulated onto the pulse (i.e. impulse

---

[5] The coding gain can be measured graphically by comparing the ratio at certain points of the BER vs. SNR coded and uncoded curves (CCK and QPSK, respectively).

response of the filter). To ensure no intersymbol interference (ISI), adjacent symbols are spaced apart by the first zero-crossing of the pulse.

The 802.11b standard specifies a square-root raised cosine filter (SQRRC), which has a frequency response that is unity over the passband and decays to zero in the transition band as a square-root cosine. The frequency response is defined as:

$$
H(j\omega) = \begin{cases} 1 & |\omega| < \omega_c(1-\alpha) \\ \sqrt{\dfrac{1+\cos\left(\dfrac{\pi\left(|\omega|-\omega_c(1-\alpha)\right)}{2\alpha\omega_c}\right)}{2}} & \omega_c(1-\alpha) < |\omega| < \omega_c(1+\alpha) \\ 0 & |\omega| > \omega_c(1+\alpha) \end{cases} \tag{2.6}
$$

where $\omega_c = \dfrac{\pi}{T_s}$ and $\dfrac{1}{T_s}$ is the sampling rate of the input set at 11 MHz. As a result, the 3-dB bandwidth of the filter is $B = \dfrac{1}{2T_s} = 5.5MHz$. The roll-off factor α of the filter was set at $\alpha = 0.3$.

Ideally, the frequency response is bandlimited, resulting in an infinite impulse response. However, the actual implementation used is in discrete-time, and thus the filter has a finite impulse response (FIR). The new impulse response is basically a 'windowed' version of the ideal response, resulting in side lobes that appear in the magnitude plot of the filter. The magnitude response of the filter is plotted in Figure 2.7.
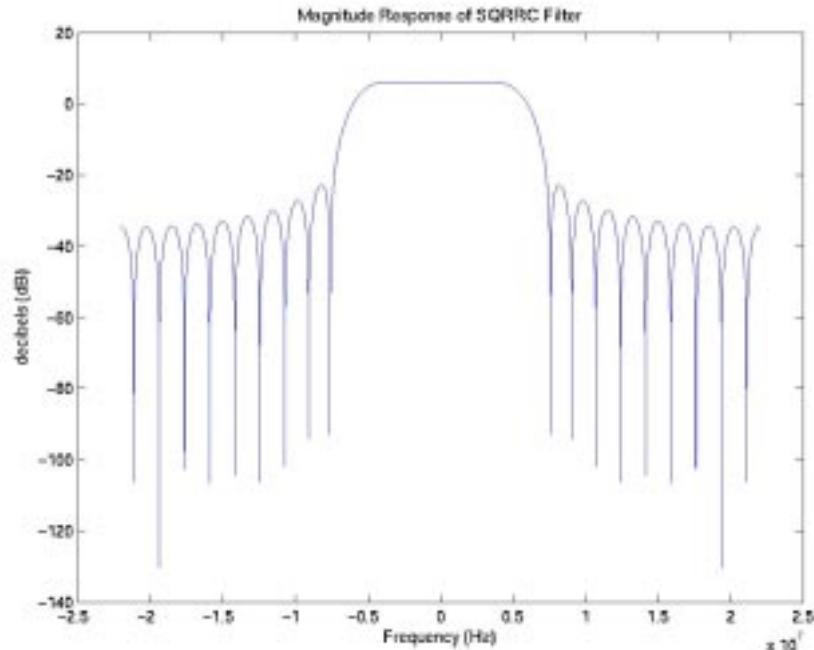
Figure 2.7. The magnitude plot of the pulse-shaping filter. The side lobes are a result of the windowing of the actual impulse response into the FIR version. The magnitude is flat over the 3-dB bandwidth of the filter.

2.8. The 2.4 GHz Industrial, Scientific & Medical (ISM) Band

The Federal Communications Commission (FCC) is an independent government agency that is responsible for regulating communications in the United States over various transmission media. The agency allocates frequency bands for different types of radio frequency devices and outlines their specifications for operation in each band in the Code of Federal Regulations.

The 2.4-2.4835 GHz band is unlicensed and can be used by any radio frequency (RF) device under the authorization of the FCC. The band is generally allocated for a variety of industrial, scientific and medical (ISM) equipment, which is where it derives its name.

In this band, RF devices are required to use spread spectrum technology. DSSS devices are required to have a minimum 6-dB bandwidth of 500 kHz [7]. Their maximum output power is set at 1 Watt, while their peak spectral power density cannot exceed 8 dBm in any 3 kHz band during transmission. Direct sequence devices also have a minimum processing gain requirement of 10 dB, which is a measure of the spreading factor of the signal's bandwidth.

DSSS Specifications		
Specification	Limit	Unit
Minimum 6-dB Bandwidth	500	kHz
Maximum Output Power	1	Watt
Peak PSD in any 3 kHz Band	8	dBm
Minimum Processing Gain	10	dB

Table 2.2: Specifications for Direct Sequence Spread Spectrum systems at 2.4 GHz.

The 802.11b standard has 14 operating channels in the ISM band, starting from channel 1 at 2412 MHz and moving up in 5 MHz increments up to channel 14 at 2484 MHz. The first 11 channels are allowed by the Federal Communications Commission (FCC) in North America. Use is more limited to other countries such as France, which uses channels 10 and 11, and Japan which uses only channel 14 [4].

2.9.   Summary

The physical layer (PHY) of the 802.11b standard was reviewed in this chapter, since the interference mitigation filters operate at this layer. The standard operates at the 2.4 GHz ISM band and uses three different modulation techniques, namely DBPSK, DQPSK and CCK encoding to achieve the four data rates it offers.

Direct-sequence (DS) spreading using a Barker code is applied for 1 and 2 Mbps, which provides larger processing gains for better narrowband interference resistance. Although no DS spreading is performed on the CCK symbols, the symbol rate is set such that they occupy the same bandwidth as the spread symbols.

Chapter 3: The 802.11b System and Wireless Channel Models

3.1. Introduction

A basic 802.11b network model consists of two stations communicating with each other through a wireless channel, where one station is set to be the *transmitter* and the other as the *receiver*. The transmitter generates a packet and transmits it at a carrier frequency around 2.4GHz, while the receiver demodulates the received waveform and reproduces the packet bits. The wireless channel can be modeled as a filter with a linear time-varying system response $H_1(f,t)$. The channel also corrupts the transmitted signal with additive white Gaussian noise (AWGN) and, for the purpose of this thesis, microwave oven interference. Similarly, the interference is filtered through a separate channel response $H_2(f,t)$. Figure 3.1 illustrates the topology of this basic network model.
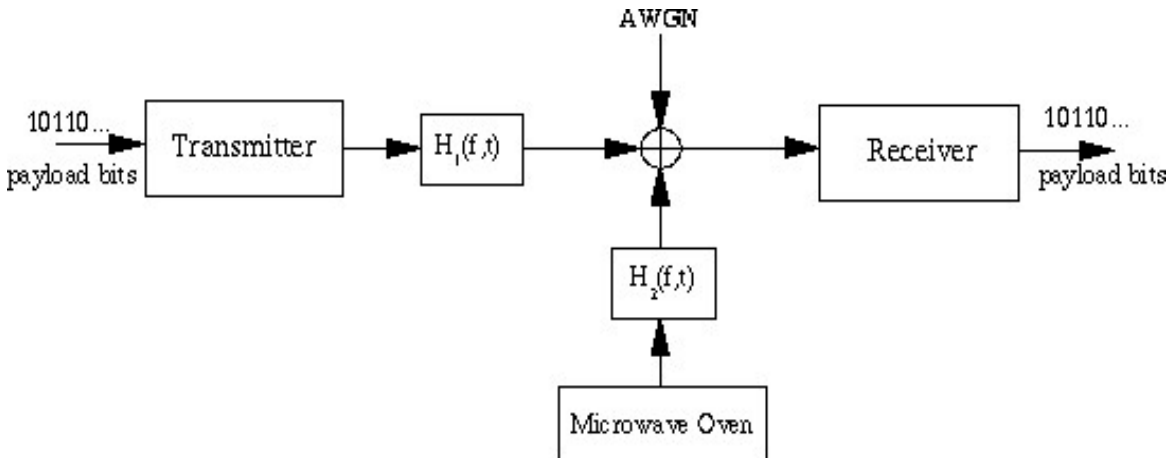
Figure 3.1. The wireless channel can be modeled as a linear, time-varying filter with additive white Gaussian noise (AWGN) and microwave oven interference.

3.2. The AWGN Channel with Microwave Oven Interference

In simulating wireless networks, simplifications to the wireless channel model are usually made as needed; simulations utilizing the model in Figure 4.1 are computationally expensive and were unfeasible given the computing resources for this thesis.

Instead, an equivalent *discrete-time* baseband simulation model was used, where the signal, noise and channel filter responses were shifted down by the carrier frequency to baseband and converted to discrete-time at the sampling frequency of the pulse-shaping filter [8]. This model was implemented in Matlab and used for the simulations, where the channel responses were assumed to be that of all-pass LTI filters with unity gain. That is, $H_1(f,t) = H_2(f,t) = 1$.[6] The model simplifies to an AWGN channel with microwave oven interference. Figure 3.2 shows the wireless channel model used for the simulations.
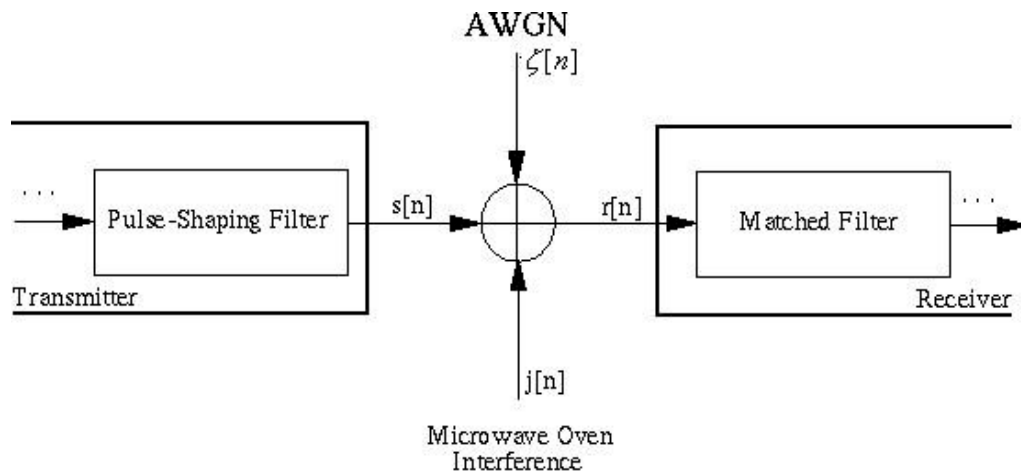
Figure 3.2. The channel model used was an AWGN channel with microwave oven interference. The model is in discrete-time, so the transmitted waveform is the output of the pulse-shaping filter, while the signal at the receiver is directly inputted to the matched filter.

---

[6] Essentially, this implies that Rayleigh fading is not included in the model.

With the model in Figure 3.2, the signal $r[n]$ at the receiver can be simply

expressed as $r[n] = s[n] + j[n] + \zeta[n]$, where $s[n]$ is the 802.11b waveform, $j[n]$ the

microwave oven interference, and $\zeta[n]$ the noise. With these simplifications, the

performance of the receiver can be directly assessed as a function of the relative strength

of the transmitted signal to the interference and noise.

3.3. Transmitter Model Architecture

The transmitter model is responsible for taking payload bits, packaging them into

a PLCP frame format and producing the baseband waveform for transmission. The

payload (MPDU), frame format, packet length and payload data rate to be used are

received from the MAC layer, and were set as model parameters for the simulations. The

802.11b transmitter model is illustrated in Figure 3.3.

The preamble and header are first generated according to the specified

parameters. Since these come first in time in the packet, the multiplexer (MUX) selects

these bits before the payload. The bits are then passed through the scrambler (see

Appendix A.1) and are modulated accordingly, depending on the data rate to be used.

The encoders produce complex symbols that are represented by two identical chains, the

in-phase and quadrature-phase, which carry their real and imaginary parts, respectively.

The DBPSK and DQPSK encoders have symbols coming out at 1 MHz and are

processed by the spreaders to produce an output at a higher frequency of 11 MHz. On the

other hand, CCK encoders have output symbol rates of 1.375 MHz, which translates to a

clocking frequency of also 11 MHz since there are 8 chips per symbol. The succeeding

MUX only selects the output of the spreader for 1 and 2 Mbps and directly selects the

CCK encoder outputs for 5.5 and 11 Mbps.

The multiplexer outputs are then pulse-shaped by the square-root raised cosine

filters (SQRRC) to generate the baseband waveform at 44 MHz.[7] Based on the wireless

channel model defined in Figure 4.2, this is where the transmitter model ends; the

baseband waveform is then passed through the channel. In an actual transmitter, however,

the waveform is converted to continuous time by the Digital-to-Analog converter (DAC),

up-converted to the carrier frequency at 2.4 GHz and transmitted.

Figure 3.3. The 802.11b transmitter model diagram. The transmitter model only includes the signal processing blocks included in the dashed line, or up to the input of the Digital-to-Analog Converter (DAC).

---

[7] The period between adjacent waveform samples is $T_s = \dfrac{1}{44MHz}$ seconds..

3.4.   Receiver Model Architecture

After encountering noise and microwave oven interference, the transmitted signal

makes its way to the receiver, which performs the reverse operations to produce the

payload bits. Figure 4.4 shows the receiver structure. In an actual receiver, the received

signal is at the carrier frequency and has to be downconverted and sampled into discrete-

time by the Analog-to-Digital (ADC) converter.

In line with the discrete-time baseband model, these blocks can be abstracted

away and the received signal is directly passed as input to the matched filter, which is

also an SQRRC filter. The matched filter maximizes the signal-to-noise ratio (SNR) and

removes out-of-band noise.

Upon detecting the presence of the received signal, the receiver begins the

acquiring phase, where it attempts to find the right timing synchronization for sampling

the signal. While in this mode, the despreader correlates the matched filter output with

the reverse Barker code until the *preamble detector* determines the peak correlation value

and timing synchronization of the sequence.[8] Once the preamble detector locks onto the

signal, the FIFO-Dispatcher samples the signal accordingly and passes the output onto

the demodulator, where the samples undergo detection at the DBPSK demodulator to

produce the preamble bits.

After passing through the descrambler, the preamble bits are scanned by the SFD

detector until it identifies a valid SFD pattern. Depending on the PLCP format, it informs

the appropriate demodulator for the header to produce the header bits. The header

detector then computes the CRC check on the first three fields and determines their

---

[8] If the stream of symbols were successively modulated onto the SQRRC pulse for transmission, the proper
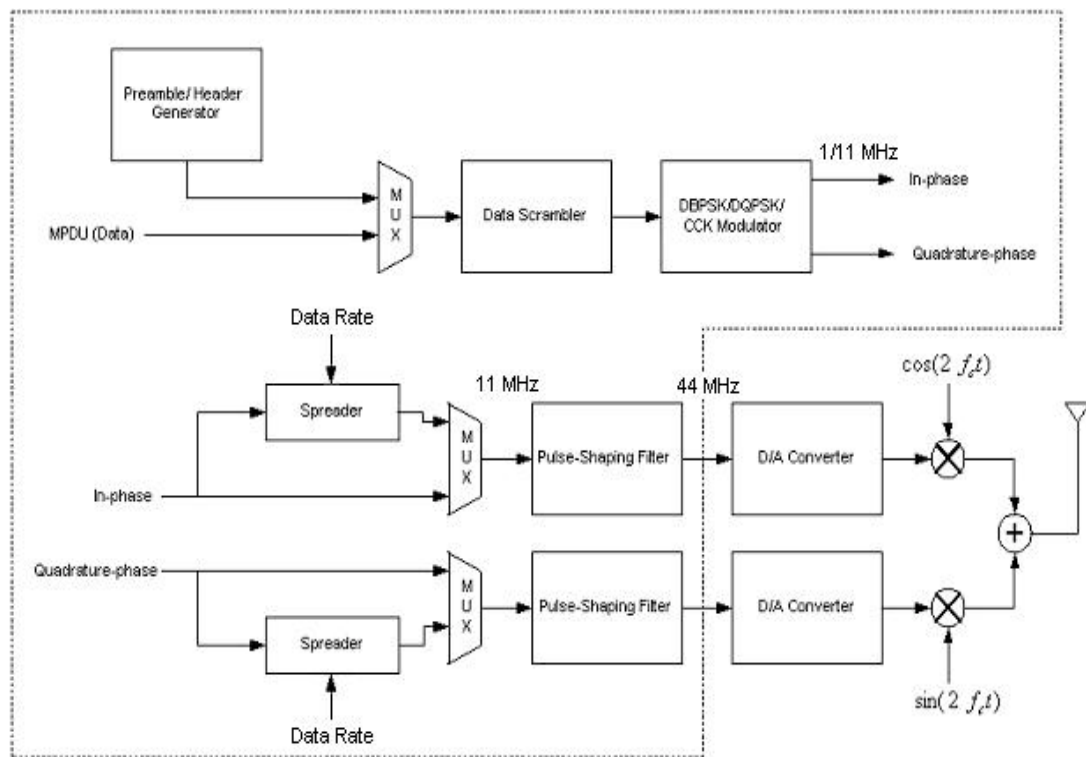timing synchronization is at the peak of each (shifted) pulse.
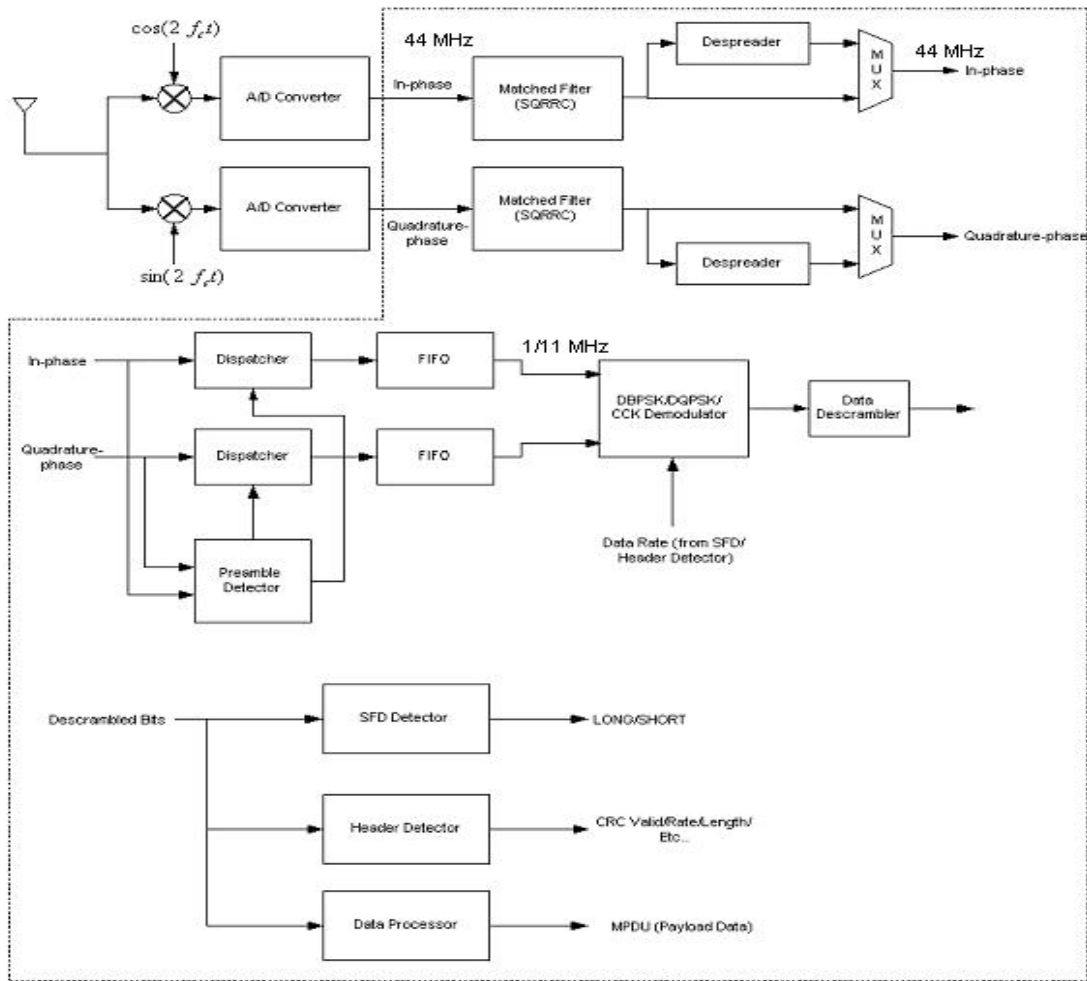
Figure 3.4. The 802.11b receiver model diagram. The receiver model only includes the signal processing blocks included in the dashed line, at the output of the Analog-to-Digital Converter (ADC).

integrity. If the fields are valid, it calls the appropriate demodulator for the payload. The

payload bits are passed back up to the MAC layer for processing.

3.5.   Model Limitations

The simplification of the channel frequency responses was a limitation of the

model that may be included in future work. A wireless channel frequency response

usually models the *multipath* property of the channel, where the signal takes a number of

different paths with associated attenuations and delays; this results in what is known as multipath fading.

One issue with including this in the model is the fact that there is no typical channel response, especially for indoor channels. The channel response is dependent on factors such as the orientation of the transmitter and receiver with the various objects that occupy space in the environment. Consistent with the channel model, the receiver model does not include a Rake receiver, which combines attenuated and delayed copies of the signal resulting from the multipath environment.

Furthermore, since a discrete-time model is used, both the transmitter and receiver do not account for any nonlinear effects introduced by the front-end analog circuitry, such as the low-pass filters, variable gain amplifiers (VGA) and power amplifiers. These nonlinearities are circuit-dependent and occur under certain operating conditions (e.g. high temperature).

3.6. Summary

The overall network model presented in this chapter operates at baseband, and assumes an AWGN wireless channel for the 802.11b signal with the addition of microwave oven interference; no Rayleigh fading is modeled. The transmitter and receiver models incorporate the basic signal processing operations performed in 802.11b transceivers. Operating at baseband, both bypass conversion to continuous time and the upconversion to the carrier frequency, and instead interface with the channel through the pulse-shaping and matched filter, respectively.

Chapter 4: Microwave Oven Interference Characterization

4.1. Introduction

Microwave ovens are widely used in business and residential environments where they can potentially interfere with 802.11b devices since both operate in the 2.4 GHz ISM band. Microwave ovens have built in mechanisms that are designed to prevent radiation from leaking out. Each oven is required by the US Food and Drug Administration (FDA) to have at least two safety interlock switches, which turn the oven off whenever the door is opened and prevent it from running until the door is closed [9]. Furthermore, seals surround the entire perimeter of the door to confine the radiation while the oven is running.

However, microwaves can manage to leak out of the oven due to basic wear and tear, manufacturing defects, broken or missing door glass, or food-particle build up around the door seals. This chapter attempts to quantify the adverse effects of microwave oven interference on 802.11b networks, as well as fully characterize and model the emissions.

4.2. Microwave Oven Operation

As its name implies, a microwave oven heats up and cooks food through the use of microwave radiation. The molecules in the food absorb electromagnetic energy and are jostled back and forth at 5 billion times per second. The high kinetic energy of the molecules is converted into heat for cooking. Microwaves can only penetrate food up to a

depth of 1_ inches; inner molecules that are not reached absorb heat from the outer layers through conduction.
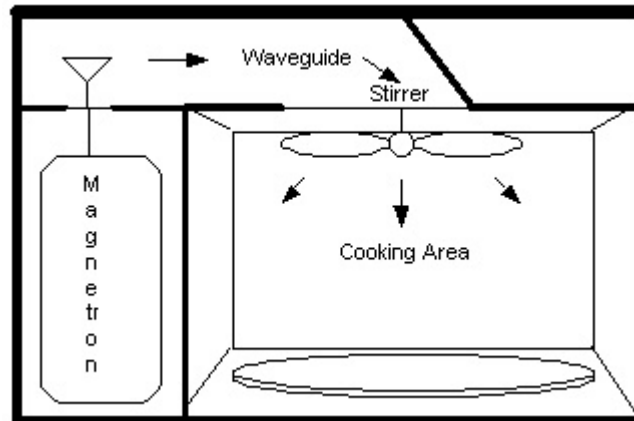
Figure 4.1. The microwaves are produced in the magnetron and channeled to the cooking area, where the stirrer evenly disperses them.

At the heart of the oven's operation is the magnetron tube, which is used to generate the microwaves. The tube consists of a hollow iron cylinder and an inner filament tube, which act as an anode and a cathode, respectively. The magnetron uses the combined forces of orthogonal electric and magnetic fields to generate a rotating electron cloud that oscillates in resonant cavities, formed by vanes that extend inward from the anode's wall.

These cavities form the equivalent of several high-Q resonant inductive-capacitive LC circuits, which are connected in parallel by physically linking alternating vanes together [10]. With the rotating electron cloud exciting these resonant cavities, the magnetron produces a high-frequency electromagnetic wave at 2.45 GHz.
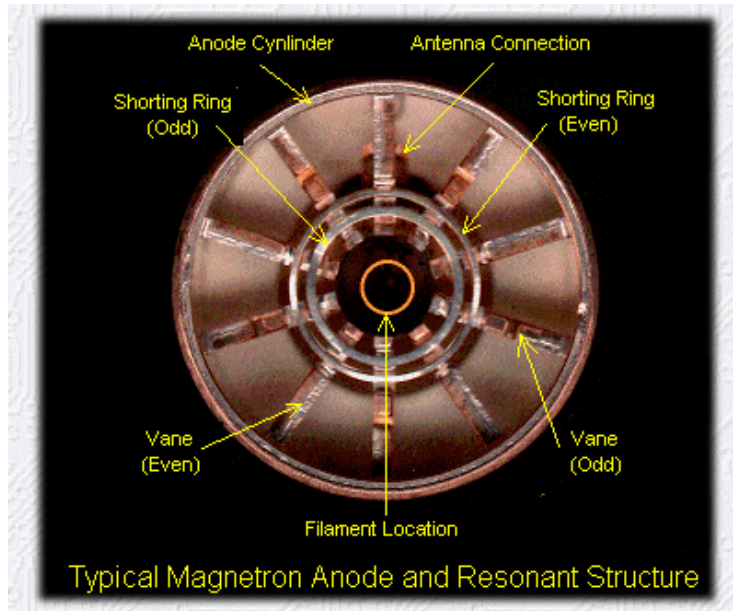
Figure 4.2. A cross-sectional image of a typical magnetron tube. The shorting rings connect alternating vanes. The electron cloud is produced in the gap between the filament and anode, and the antenna channels the radiation to the cavity. Source: [11]

From the magnetron tube, an *antenna* directs the generated microwaves to a *waveguide*, which is a hollow metal enclosure that channels the energy into the oven's cavity. A slowly rotating metal fan called a *stirrer* evenly disperses the waves throughout the cooking area.

4.3. Microwave Oven Safety Standards

Residential microwave ovens are classified as consumer ISM equipment, and are required by the FCC to operate at a nominal frequency of 2.45 GHz with a tolerance level of ±50 MHz. Although they can have unlimited radiated energy within their operating frequency range, the radiation must be confined as much as possible to their cooking areas. Emissions must meet a field strength limit defined by their operating power (Watts) at a distance of 300 meters away [9]:

$$Field\ strength\ limit = 25\sqrt{\frac{power}{500}}\ \frac{\mu V}{m} \qquad (4.1)$$

Furthermore, the field strength of the emissions cannot exceed 10 μV/m at a distance of 1600 meters from the oven. Table 4.1 summarizes these specifications.

Residential Microwave Oven Specifications		
Specification	Limit	Unit
Operating Frequency	2450 ± 50	MHz
Emission Field Strength @ 300 m	$25\sqrt{\dfrac{power}{500}}$	μV/m
Emission Field Strength @ 1600 m	10	μV/m

Table 4.1. The Sharp Household Microwave Oven's specifications as listed in the plaque inside the oven cavity.

In addition to specifications set by the FCC on microwave ovens, the US Food and Drug Administration (FDA) also set limits on the power density of microwave emissions, which are listed in Chapter 21 of the Code of Federal Regulations. Microwave ovens manufactured after 1971 must meet this requirement [12].

$$p = \frac{v_{\text{field-rms}}^2}{377\,\Omega} < 1\ \text{mW/cm}^2 \qquad (4.1)$$

The equivalent plane-wave power density p of the emissions, defined in equation 4.1, must be less than 1 mW/cm^2 at a distance of 5 cm or more from the oven's surface.[9]

---

[9] In order to make measurements for the equivalent plane-wave density, a 275-milliliter load of water, at a temperature of 20 degrees Celsius, placed in a 600-milliliter beaker with a diameter of 8.5 cm must be used and set in the middle of the oven cavity.

The equivalent plane-wave power density is defined as the square of the root-mean-square field strength divided by the impedance of free space, which is 377 ohms [12]. The field strength can be directly measured using an RF Field Strength Meter.

4.4. Emissions Characterization

In order to develop an accurate model of the interference for the simulations, the emissions were measured and fully characterized in a second set of experiments. Properties of the emissions, such as their frequency-sweeping behavior and duty cycle for example, were determined from analyzing interference measurements in both the time and frequency domain.

4.4.1. Experiment Tools

Due to resource and time constraints, a single microwave oven was used as the subject for the experiments. Although tests on various ovens would have provided more data for characterizing the interference, it was assumed that the properties of emissions over different microwave ovens would be fairly consistent since magnetron tube technology is standard throughout the industry. The specifications of the microwave oven are listed in Table 4.2.

The oven, being a few years old, was expected to have substantial radiation leakage. Furthermore, its operating frequency of 2450 MHz directly overlaps with the ISM band used by the 802.11b standard, indicating that there may be a potential interference problem if there is leakage. In the original verification tests on the product, a maximum emissions power of -30 dBm was measured at a distance of 3m away.

Microwave Oven Subject	
Description	Sharp Household Microwave Oven
Model Number	R409CK
Output Power	1100 Watts
Operating Frequency	2450 MHz
Power Supply	120 VAC, 60 Hz Single Phase
Date of Manufacture	August, 1999
FCC ID	APYDMR0121
Maximum Measured Emissions at 3m from oven door. Source [13]	-30 dBm

Table 4.2. The Sharp Household Microwave Oven's specifications as listed in the plaque inside the oven cavity.

In order to obtain frequency domain measurements of the emissions, an Agilent E4404B ESA-E Series Spectrum Analyzer was used. A spectrum analyzer displays the shows the power density of a signal over a specified frequency range. An antenna placed directly in front of the oven door captures the emissions from the microwave oven and sends it directly to the spectrum analyzer.

Agilent E4404B ESA-E Series Spectrum Analyzer Settings		
Parameter	**Value**	**Unit**
Center Frequency	2.45	GHz
Frequency Span	200	MHz
Reference Level	0	dB
Vertical Division Level	10	dB
Video Bandwidth	1	MHz
Resolution Bandwidth	1	MHz

Table 4.3. Spectrum Analyzer settings used for emission measurements.

The spectrum analyzer has *center frequency* and *frequency span* parameters that define the center and boundary frequencies of the display, respectively. Given that the microwave oven has an operating frequency of 2.45 GHz, as listed in Table 4.2, the center frequency parameter of the analyzer was set to 2.45 GHz, while the frequency span

was initially set to 200 MHz, which was assumed to be wider than the range covered by the emissions. A complete list of the settings is listed in Table 4.3.

Time domain measurements were recorded using an Agilent E4804A VXI Mainframe/Vector Signal Analyzer (VSA). Similarly to the spectrum analyzer, the vector signal analyzer has *center frequency* and *frequency span* parameters, which are used for down-conversion and baseband filtering the measured signal, respectively. The VSA down-converts the signal by the specified center frequency, low-pass filters it within the frequency span and samples it with a set frequency of 47.5 MHz. [10]
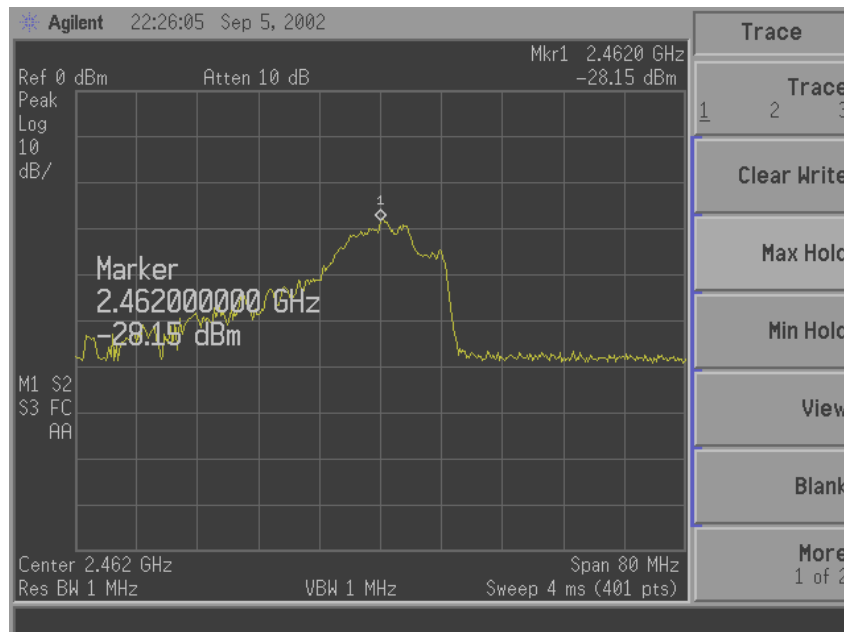
Figure 4.3. The spectrum analyzer was set at max-hold mode in order to display the maximum power of the emissions over the frequency range. Most of its power was observed to be in the 2460-2470 MHz range.

---

[10] A Nyquist sampling frequency of almost 5 GHz is needed in order to sample a signal centered at 2.45 GHz without any aliasing, thus producing a significant number of samples for saving the information. However, the same signal information can be saved more efficiently by downconverting it by 2.45 GHz, and sampling it with a rate that is twice its baseband bandwidth.

Preliminary measurements of the spectrum analyzer, shown in figure 4.3, indicate that the emissions had their highest energy around 2460-2470 MHz. As a result, the interference tests, discussed in section 4.6, used channel 11 (2462 MHz) for communication because of its direct frequency overlap with the emissions. Furthermore, the peak power of the emissions was measured to be -28 dBm at the oven door, which is consistent with the measurements conducted in the original tests.

The center frequency of the VSA was likewise set at 2462 MHz in order for its output to emulate the interference signal at the input of the receiver model. The frequency span was set by the VSA itself to its maximum value of 37.5 MHz. A recording time of 45 ms was used since it was found to be long enough to capture several periods of the emissions cycle. Table 4.4 lists the relevant settings of the VSA.

Agilent E8408A VXI Mainframe Vector Signal Analyzer		
Parameter	Value	Unit
Recording Time	45	ms
Center Frequency	2462	MHz
Frequency Span	37.5	MHz

Table 4.4. Vector Signal Analyzer settings used for emissions measurements.

4.4.2. Experiment Setup

For both spectrum analyzer and VSA measurements, a 1 Liter load of water was placed in the oven cavity, while antennas linked to the two were placed directly in front of the oven door. The oven was turned on and left running for 5 minutes; measurements were made only after 3 minutes in order to eliminate any possible transient properties that the magnetron tube may have had while it was warming up.

4.4.3. Results and Analysis

The results of analysis on the emissions were consistent with that of previous research on microwave oven interference [14, 15 & 16]. The emissions did not appear to be random, but rather deterministic and predictable. As the experiments revealed, microwave oven emissions are of a frequency-sweeping, sinusoidal nature and have a duty cycle. Figure 4.5a displays a time domain plot of the interference while figure 4.4b shows its spectrogram, both over a 45 ms window.[11] The following observations were made based on the diagrams.

a. *Periodic* - Observing both figures, it can clearly be seen that the emissions are of a periodic nature, where a period T lasts approximately 16.7 ms, which is also equivalent to one period of the 60 Hz power supply (i.e. $T = 16.7$ ms $= 1/60$). This property is further explained in part b.

b. *Duty Cycle*– It can be noted that the interference is only active for approximately half of each period; it has a 45% duty cycle to be exact. This can be attributed to the power supply of the magnetron tube, which is a high-voltage doubler circuit. The power supply consists of a high-voltage capacitor and diode that supply current to the magnetron tube only at the negative phase of each power supply period, thus only operating it for half of each cycle or approximately 8.35 ms.[12] Graphical measurements of the signal plot and spectrogram both indicate that the emissions last for 7.5 ms of each cycle.

---

[11] A spectrogram plots the energy distribution of a signal across the spectrum over time.
[12] The actual measurement of the emissions is 7.5 ms, although this was determined graphically.
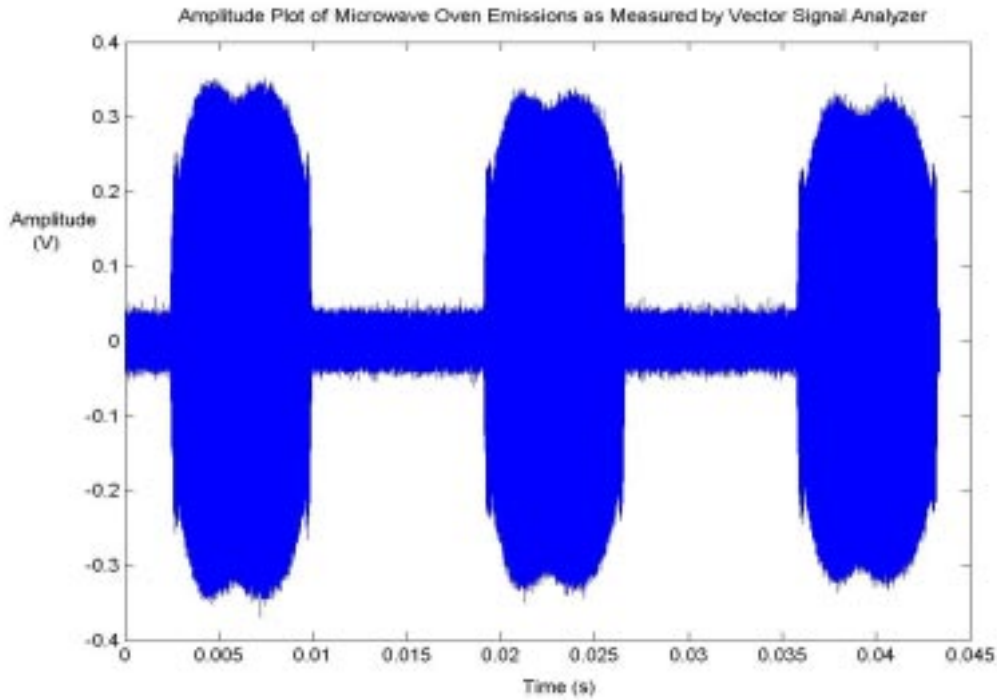
(a)
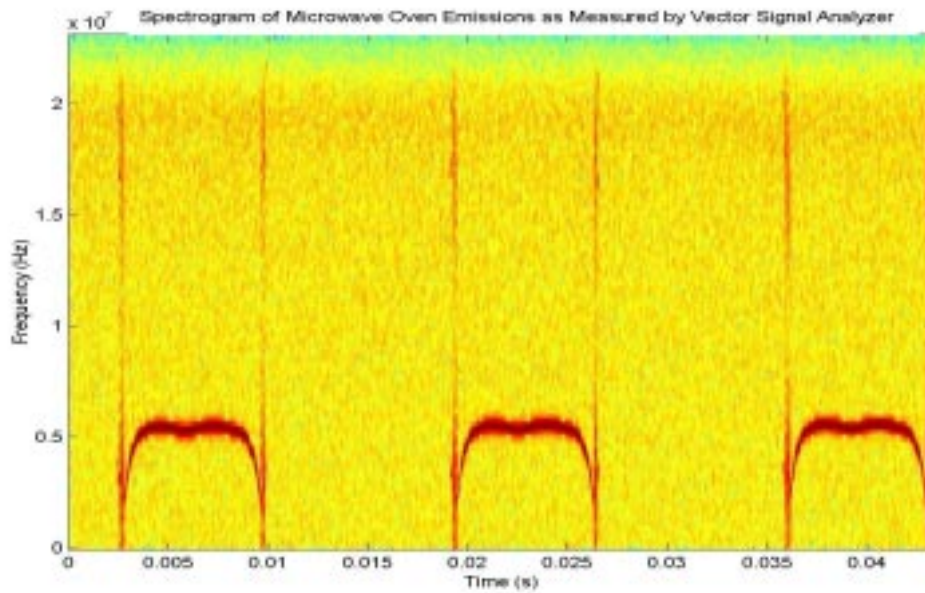
(b)

Figure 4.4: (a): Time domain measurement of microwave oven emissions over a 45 ms time frame. (b): Spectrogram of the measured emissions. It can be seen in both plots that the emissions are periodic in nature with a period of 16.7 ms. The spectrogram in figure (b) shows that the power of the jammer at any point in time is concentrated in the red narrow band, indicating it is sinusoidal. The band is observed to have a 'U' shape, indicating it sweeps in frequency. In this case, 0 Hz corresponds to 2462 MHz, since this plot shows the interference after downconversion.

c. *Frequency-sweeping Sinusoid* – The spectrogram displays the frequency content by using colors; a darker color such as red corresponds to a higher energy relative to a lighter color, such as yellow. The distinct red line in the spectrogram shows that at any point in time, the signal power is concentrated in a very narrow band, suggesting it is of a sinusoidal nature.[13]

Given this observation, it can also be noted that the interference sweeps in frequency, as illustrated by the inverted "U" shape of the spectrogram; it sweeps up in a parabolic fashion from 0 Hz to 5.5 MHz, settles at that *center frequency* for a period of time and sweeps back to zero in a similar fashion. [14]

The sweeping times as well as the *center frequency dwell time*, hereby defined as the approximate time spent at the center frequency, were estimated based on observing the spectrogram in figure 4.4b. Table 4.5 presents the measurements.

Interference Transient and Center Frequency Times		
Specification	Value	Unit
Sweep Up Time	1.2	ms
Sweep Down Time	1.2	ms
Center Frequency Dwelling Time	5.1	ms
Center Frequency	5.5	MHz
Center Frequency Variation	250	kHz
Total Emission Time	7.5	ms
Period	16.7	ms
Duty Cycle	45	Percent

Table 4.5. Measured time and frequency properties of the microwave oven emissions. The sweeping and center frequency times were estimates produced based on graphical observations.

---

[13] Ideally, a sinusoid in the frequency domain is represented as an impulse at its center frequency, which has an infinitely narrow bandwidth.
[14] This is at 'baseband', or equivalently after downconversion by 2462 MHz. The sweeping actually peaks at 2462+5.5 = 2467.5 MHz.

The spectrogram also shows that there are variations, albeit minor, in the center frequency of the interference; the center frequency was measured to vary by as much as 250 kHz during each cycle.

The interference bandwidth was measured at 25 μs intervals over the 5.1 ms time frame when the interference was at its center frequency in order to localize it in time and avoid observing small frequency variations as wider bandwidths.

Instantaneous Bandwidth		
Attenuation from Peak	Bandwidth	Unit
3 dB	114.39	KHz
6 dB	193.58	KHz
20 dB	497.15	KHz

Table 4.6. The maximum measured bandwidth of the emissions over a 25 us interval. The time interval was chosen such that the center frequency would effectively be stationary over that time interval, and small variations would not affect bandwidth measurements.

Table 4.6 lists the *maximum* measured 3, 6, and 20 dB-bandwidths of the interference found over all the intervals. The particular interval started at the 90th μs of the measurements and lasted for 25 μs.

d. *Wideband Smear at Emission Edges* – A wideband smear across the entire spectrum can be observed immediately before and after the frequency-sweeping portion of the emissions in every cycle. The power of each smear was measured to be 5.75 dB weaker than the average power of the jammer for one cycle, indicating that most of the energy of the jammer is concentrated in the purely sinusoidal portions.

e. *Amplitude Envelope* – Finally, by observing the time domain plot at the top of Figure 4.3a, it can be seen that the amplitude envelope of the emissions for each power-cycle are consistent. The envelope appears to be symmetric with respect to the midpoint of the interference, or at 3.5 ms into the active phase of the cycle. The envelope starts at zero and approaches the maximum amplitude, after which it takes a slight dip. The time-reversed shape of the envelope is then observed for the rest of the cycle.

4.5. Emissions Modeling

Although the actual measured interference could have been exported to Matlab for the simulations instead of developing an interference model, a considerable amount of noise was introduced into the measured signal due to the high noise floor of the VSA. Unfortunately, it was impossible to filter out all the noise from the emissions, and using the actual measurements would have reduced the set SNR noise value, which was a controlled variable as discussed in Chapter 8. To resolve this problem, a microwave oven interference model with the same set of characteristics was developed.

4.5.1. Model Properties

The model incorporated the essential features of the interference. The sinusoidal nature of the interference was first set as a basis for the model, after which the frequency-sweeping behavior, approximated as quadratic function, was added accordingly. A similar amplitude envelope was shaped onto the waveform, and the basic cyclical nature of the interference was accounted for when it was added to a packet in the simulations.
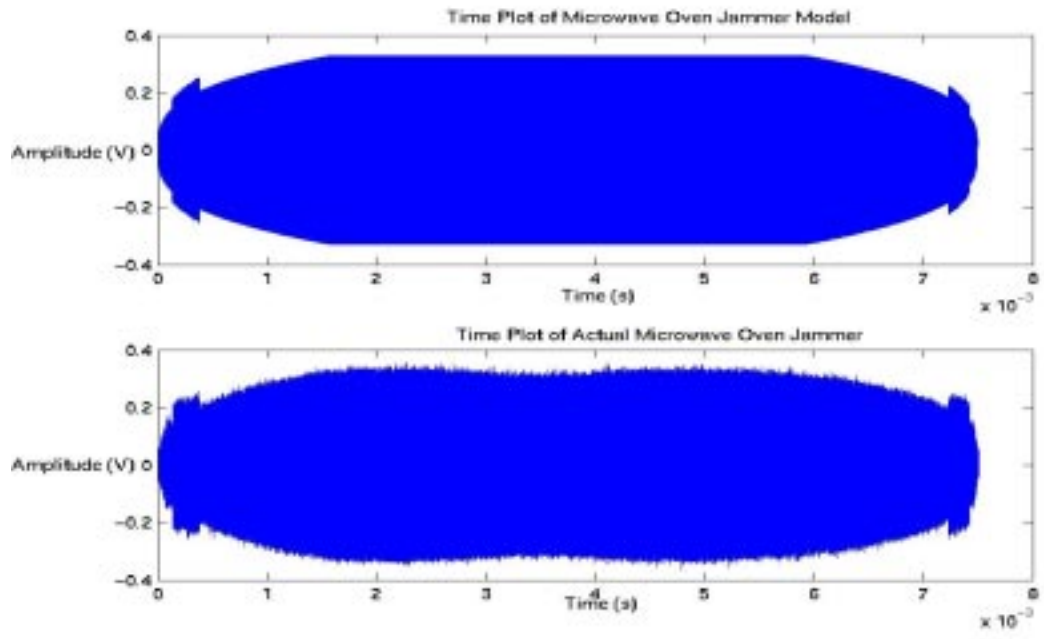
(a)
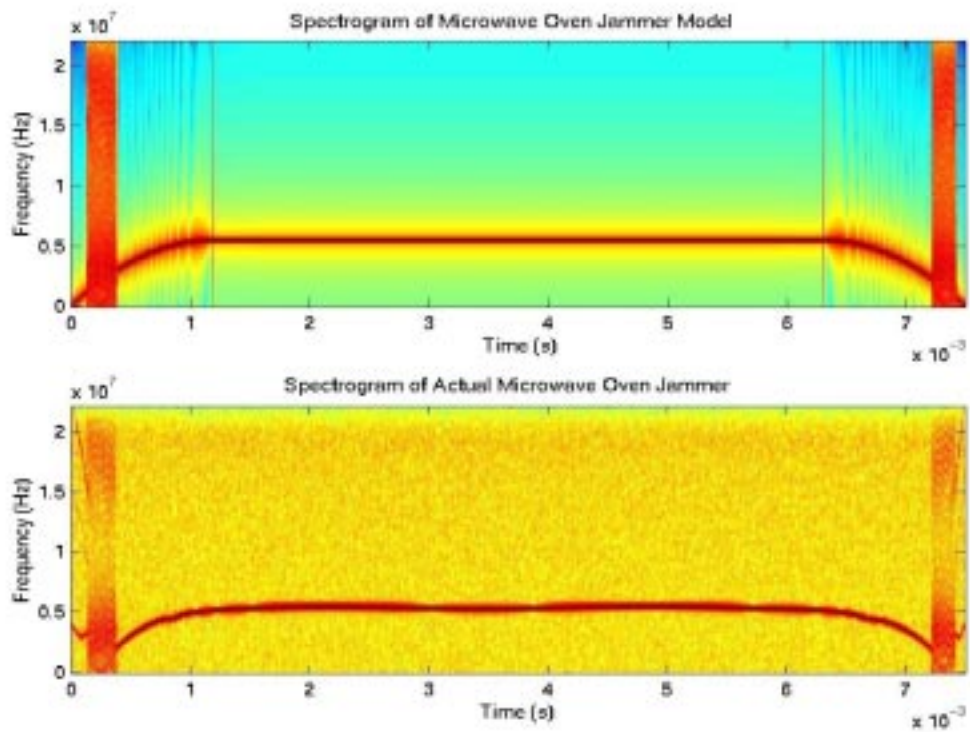
(b)

Figure 4.5. a.) Time-domain plots of the jammer model (top) and microwave oven emissions (bottom). b.) The corresponding spectrograms of the jammer model (top) and microwave oven emissions (bottom).

The figures in the previous page superimpose the interference model with the actual radiation measurements; figure 4.5a shows a time-domain comparison between the model and the actual emissions over one cycle, while figure 4.5b shows their corresponding spectrograms. Figure 4.6 below compares the magnitude plots of their respective Fourier transforms.
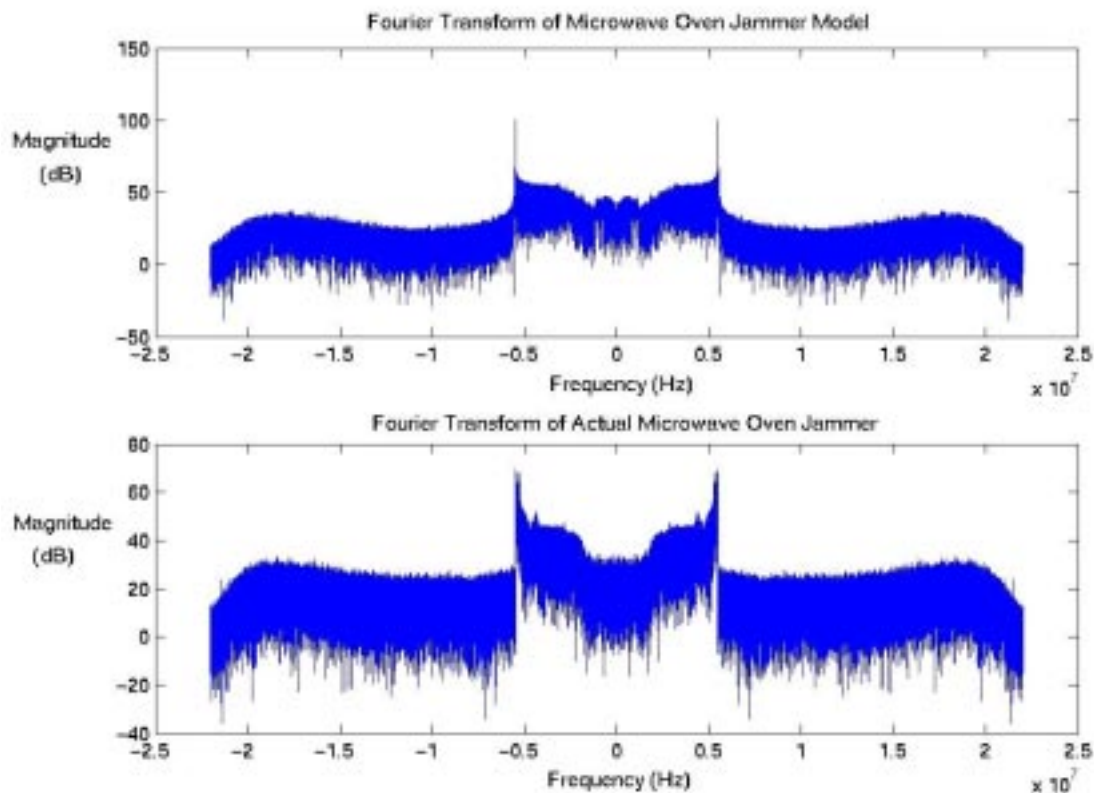
Figure 4.6. The corresponding Fourier transforms of the jammer model (top) and the microwave oven emissions (bottom). Note that although the two have different magnitudes, these do not make a difference in the simulations since they are scaled according to the desired signal-to-jammer (SJR) ratio.

4.5.2. Model Limitations

Although the model provides a noise-free jammer representation, it also has limitations and inaccuracies. These inaccuracies were assumed to have not affected the performance of the interference mitigation techniques, since they depended on features

57

included in the model. The following are several jammer characteristics that were not accounted for in the model:

a.) *Amplitude Envelope Portion* – In creating the model, a constant-amplitude sinusoid was first created with the same frequency-sweeping property as the emissions. The sinusoid was then amplitude modulated by a *cube-root* function, which closely resembled the actual envelope, until the amplitude was supposed to plateau after approximately 1.2 ms.

However, a dip in the amplitude was introduced at the point where the pulse modulation ends, creating a noise smear across the whole spectrum, represented by the thin line at the end of the frequency-sweeping portion of the model in figure 4.5b. Furthermore, the slight dip of the amplitude in the middle of the active phase, evident in the time-domain plot in figure 4.5a is not accounted for in the model.

b.) *No Center Frequency Variation* – The center frequency was fixed at 5.5 MHz and did not account for the slight frequency variations that were observed.

4.6.   Throughput Degradation Measurements

In order to gauge the effects of microwave oven radiation on an 802.11b communications link, an interference experiment was carried out to determine the degree of data throughput degradation. The experiment consisted of repeatedly uploading a randomly generated data file from a laptop onto a desktop computer through an access

point (AP) in the proximity of a running microwave oven, while measuring the average data rate of the link.

As discussed in section 2.4, each of the four data rates has a corresponding processing gain depending on its channel-encoding scheme, and thus each will be affected differently in the presence of interference. All four data rates were tested in the experiment.

4.6.1. Experiment Hardware & Software

A Hewlett-Packard Omnibook 2100 laptop equipped with an Orinoco Mobility 11 Mbps 802.11b Wireless LAN card was used as the mobile station. The card, which offered WEP-64 encryption, was configured to use each of the four Wi-Fi data rates. A Hewlett Packard Brio desktop computer running Windows NT was used as the receiving station, while a Samsung Magic LAN (SWL-3300) Access Point was used as the base station and controlled through its software manager.

To measure the effective data rate for each trial, the Chariot v4.3 software suite, an industry-standard program that measures the throughput between specified nodes in a network was used. The main console which featured the control settings was installed on the desktop computer, while a client program ran on the laptop.

4.6.2. Experiment Setup

The experiments were carried out in a wide-open office room with dimensions of approximately 20'x20'. The contents of the room were typical office furniture and equipment. The microwave oven, access point and laptop were all placed on platforms

that were 27" high. The arrangement was set such that the laptop always had a direct line of sight with the access point. Figure 4.7 illustrates the experiment setup.
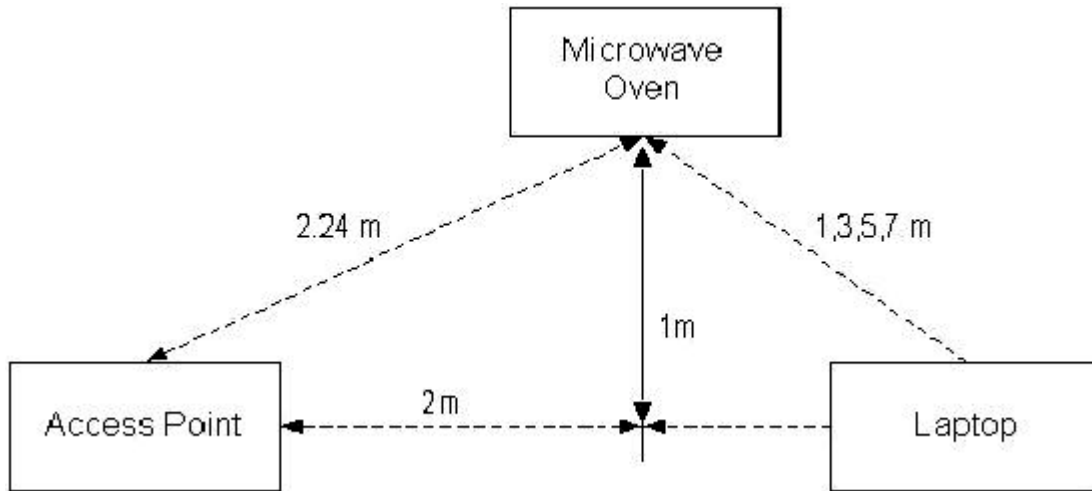
Figure 4.7. A schematic diagram of the experiment setup. The microwave oven was fixed at 2.24 m away from the access point, while the laptop's distance from the oven was varied. Four different distances were tested: 1,3,5 and 7m.

The laptop was set at distances of 1, 3, 5 and 7m from the oven for each data rate test. There were two tests for each case: a controlled test with the microwave oven off, and an experimental test with the oven running. For the experimental tests, the microwave oven was turned on at full power with a 1 Liter load of water in its cavity.

In each trial, the Chariot suite ran a test script that uploaded a 10 MB file 30 times from the laptop to the desktop using a TCP/IP connection. The laptop was the only station associated with the AP, and was set to communicate on channel 11 (2462 MHz) since the oven emissions were observed to be around the same band (discussed in section 4.4.1). Chariot provided estimates of the throughput (in Mbps) after each file transfer, and gave an overall average at the end of the test.

4.6.3. Results & Analysis

The experiment results, listed in table 4.7, show that the interference caused considerable throughput degradation and in some cases made the file transfers impossible.

On average, the interference cut throughput in half for 5.5 and 11 Mbps, while it practically brought throughput down to zero for most cases at 1 and 2 Mbps. In the latter case, Chariot terminated testing prematurely because TCP connections could not be established due to the interference; throughput was virtually brought down to zero.

Data Rate (Mbps)	Distance (m)	Average Throughput w/out Microwave Oven Interference (Mbps)		Throughput Ratio (%)	TCP Connection: Ok or Failed to Connect
		With	Without		
1	1	0.78	0.35	44.87	Ok
	3	0.77	0	0	Failed
	5	0.78	0	0	Failed
	7	0.78	0	0	Failed
2	1	1.47	0.78	53.06	Ok
	3	1.47	0	0	Failed
	5	1.46	0.83	56.85	Ok
	7	1.47	0	0	Failed
5.5	1	3.24	1.7	52.47	Ok
	3	3.25	1.93	59.38	Ok
	5	2.99	1.41	47.16	Ok
	7	3.23	1.43	44.27	Ok
11	1	4.88	2.39	48.98	Ok
	3	4.45	2.08	46.74	Ok
	5	3.23	2.73	84.52	Ok
	7	4.68	2.21	47.22	Ok

Table 4.7. The results show that for the higher data rates of 5.5 and 11 Mbps the throughput is slowed down by a factor of 2 on average. The throughput ratio is defined as the percentage of the new data rate (in the presence of interference) to the data rate without interference.

It can be observed that this is what happened for greater distances of the laptop from the oven and access point, since there were smaller Signal-to-Jammer (SJR) power ratios.

One anomaly in the results can be observed: the interference test successfully finished at a distance of 5 m using 2 Mbps, but a connection could not be established at a closer distance of 3 m, where a stronger SJR would be expected. One possible explanation would be that the exact arrangement of the components and furniture in the room for that test provided better reception for the laptop for the 5m trial.

Furthermore, the results might seem to contradict the notion of the processing gain as discussed in section 2.6.4; the performance degradation was worse for the lower data rates, which supposedly had higher processing gain and thus more resistance to interference. The phenomenon can be explained by the fact that it takes less time to transmit a packet at 11 and 5.5 Mbps than at 1 and 2 Mbps: over the same time frame, 8 bits are encoded in the former for every encoded bit in the latter. With shorter packets, the probability of getting jammed with the interference is smaller, since a greater number of packets will 'fit right in' between active phases of the emissions, thus avoiding the interference.

4.7. Summary

Several important findings were discovered about microwave oven emissions. The emissions were found to be of a frequency-sweeping, sinusoidal nature, with a duty cycle of 50%, consistent with the operation of the magnetron tube. Based on these measurements, a model of the interference was developed for use in the simulations. Furthermore, the emissions were found to degrade the throughput of an 802.11b

communications link, and more severely for the lower data rates. The results of the

interference tests suggest the need for possible interference mitigation at the receiver.

Chapter 5: Transform Domain Filtering

5.1. Introduction

A transform domain filter processes its input in the frequency domain. The filter computes the input's transform and performs filtering operations on it, after which it performs the inverse transform operation to produce the time-domain output signal. The Fourier transform is commonly used due to its simplicity and ease of computing, although other transforms such as Lapped and Wavelet transforms have also been used in similar narrowband interference problems on PN spread spectrum systems [17,18]. For discrete-time signals, the Discrete Fourier Transform (DFT) can be efficiently computed using the Fast Fourier Transform (FFT) algorithm.[15]

Figure 5.1. A schematic diagram of the Transform Domain Filter.
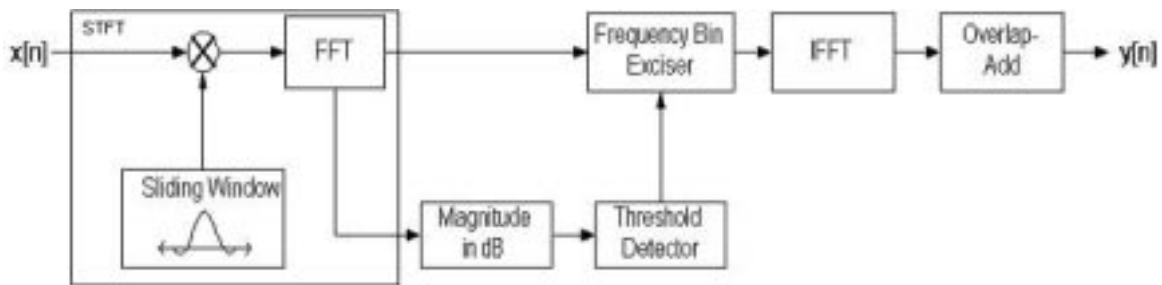
Due to exploitable frequency domain differences between 802.11b signals and microwave oven interference, a transform domain filter can be effectively used to cancel out the latter. The proposed transform domain filter for microwave oven interference is

---

[15] The FFT is used interchangeably to describe the algorithm and the actual DFT computed by the algorithm.

illustrated in figure 5.1. Essentially, the filter converts the input to an FFT sequence, performs interference mitigation using a thresholding algorithm to detect and remove the jammer, and computes the IFFT to produce the output.

5.2.   Integration with Receiver

The transform domain filter is positioned between the Analog-to-Digital Converter (ADC) and the matched filter, located in the front-end section of the receiver chain. In an actual receiver, the filter will take in the sampled signal from the ADC and perform jammer detection and excision before the signal is processed by the SQRRC matched filter. Figure 5.2 shows its proposed placement in the receiver chain.
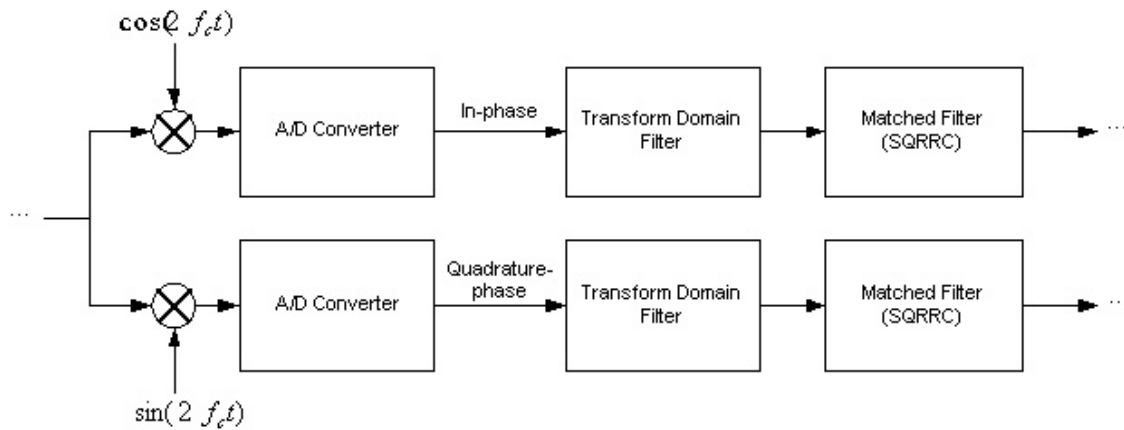
Figure 5.2. The Transform Domain Filter is placed in between the ADC and Pulse-Shaping matched filter in the receiver.

5.3.   Principles of Operation

An 802.11b signal derives its spectral characteristics from its pulse-shaping filter, which modulates the channel encoder's output symbols to produce the baseband

waveform. The standard specifies a Square Root-Raised Cosine (SQRRC) filter, which has a frequency response that is flat across its 3-dB bandwidth (see section 2.7). Figure 5.3 shows the magnitude plot of an 802.11b signal with an arbitrary bit payload.
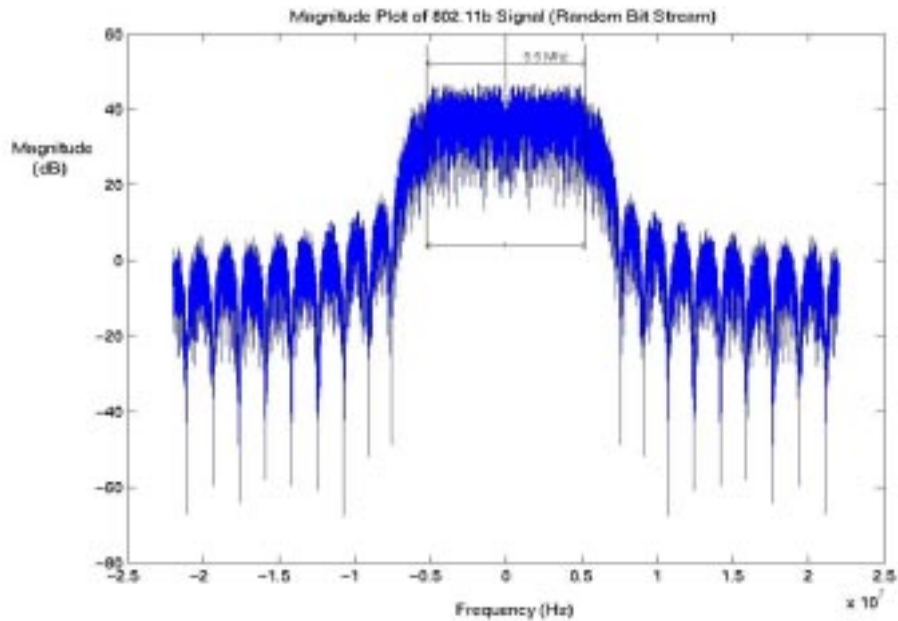
Figure 5.3. The magnitude plot of an 802.11b signal with an arbitrary payload. takes the shape of the frequency response of the SQRRC filter, discussed in section 2.7.

On the other hand, microwave oven emissions occupy a very narrow instantaneous bandwidth due to their sinusoidal nature (see section 4.6d). While a single-tone sinusoid will appear as a narrow spike or as an impulse in the frequency domain, the Fourier transform of a frequency-sweeping sinusoid will appear as a wide 'blob' over its sweeping range; the transform of the sinusoid is 'smeared' over the band since its frequency moves in time. Figure 5.4 illustrates the magnitude plot of the signal with the microwave oven interference sweeping over a range of approximately 2 MHz over its duration.
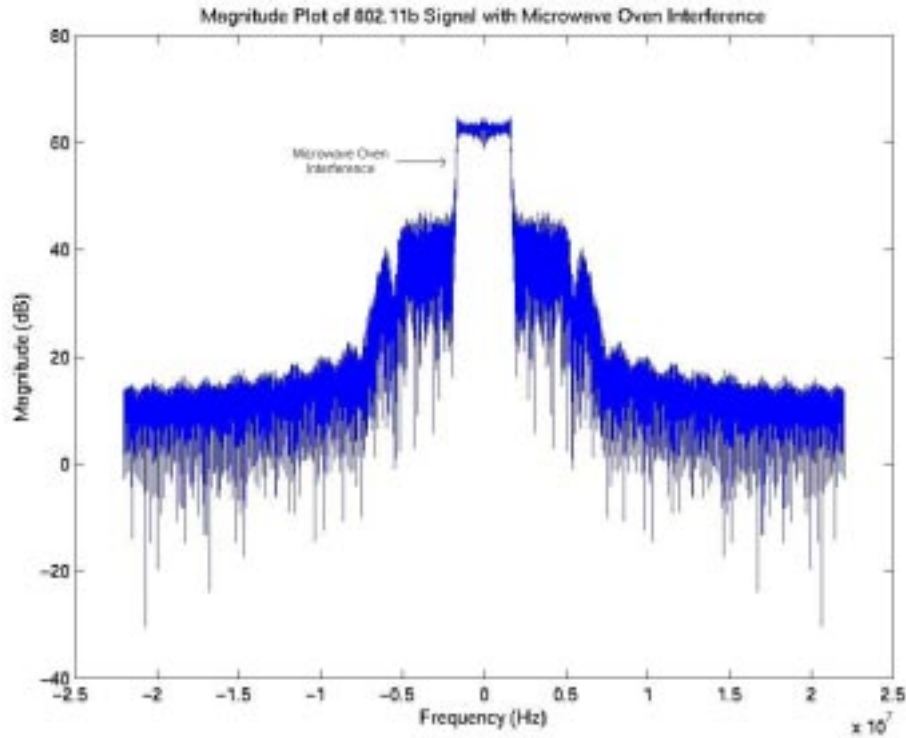
Figure 5.4. The magnitude plot of an 802.11b signal with microwave oven interference, the latter represented by the 'blob' in the middle rising to 65 dB.

The filter removes the interference by detecting frequency bins in the FFT sequence which the jammer occupies and setting them to zero. Since these bins are also occupied by the signal, the jammer must have as small a frequency spread as possible to minimize the loss of the signal. Because it is known that the jammer is sinusoidal and thus has a narrow instantaneous bandwidth, the filter localizes the frequency of the jammer in time by computing for the Short-Time Fourier Transform (STFT) of the received signal.

As seen in the schematic in figure 5.1, the STFT computation is a two-step process, consisting of multiplying the received signal x[n] with a sliding finite-length window w[n] and computing the FFT on one windowed portion at a time. The filter processes each windowed portion separately.

68

$$X(r,k) = \sum_n w[r-n]x[n]e^{-j2\pi nk/N} \qquad (5.1)$$

Equation 5.1 defines the STFT for the sequence $x[n]$, where k is the frequency bin number, r is the time-shift of the window, and N is the length of the window. Viewed in the frequency domain, the multiplication of $w[n]$ and $x[n]$ translates to a convolution of their transforms, as defined in equation 5.2.

$$X(r,k) = \frac{1}{2\pi} \int_{-\pi}^{\pi} W(e^{j\Omega}) X(e^{j(k-\Omega)}) e^{-j\Omega r} d\Omega \qquad (5.2)$$

The windowing process can be seen as taking a "snapshot" of the jammer: if the window is short enough, the jammer's frequency can be assumed to be stationary for that period of time. Thus, the jammer's frequency is localized in time by the STFT. The effect of the windowing process on a frequency-sweeping sinusoid is illustrated in figure 5.5, where a Hamming window was multiplied with the received signal at an arbitrary time interval. It can be clearly seen that over the windowed portion, the sinusoid has a stationary frequency.

With the STFT reducing the interference to a single-tone jammer, the differences between the signal and interference are clear; a received 802.11b signal with a relatively strong single-tone sinusoidal jammer will have a magnitude plot that is flat for most of the data signal's 3-dB bandwidth except for a sharp peak at the jammer's frequency. This relationship allows the jammer to be distinguished from the rest of the signal in the frequency domain.
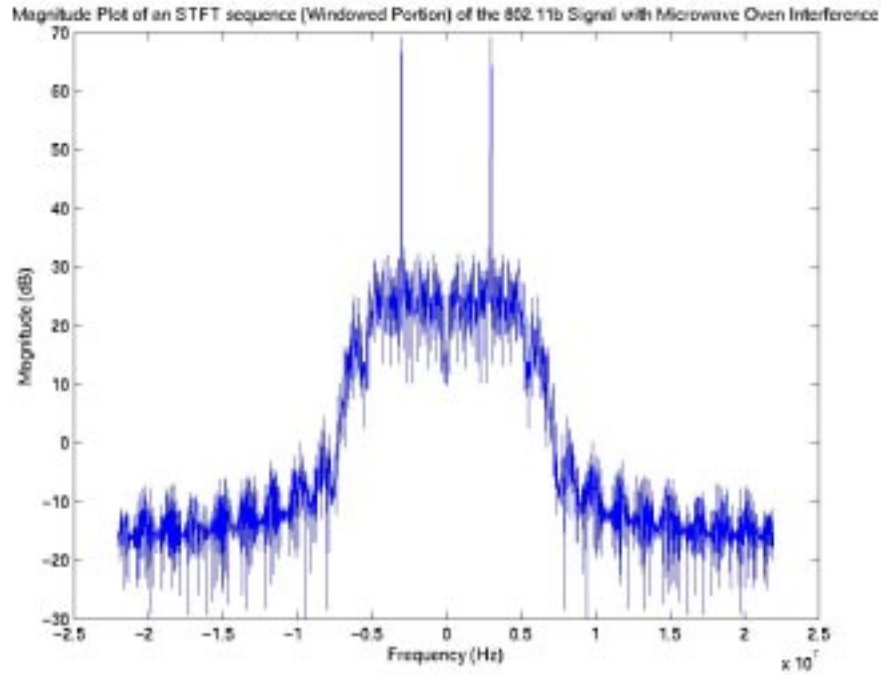
Figure 5.5. A magnitude plot of the windowed 802.11b signal with microwave oven interference. Since the window was confined to a small time interval, the jammer frequency is stationary during this period, as seen by the sharp spike concentrated at 3 MHz.

After performing the STFT, the magnitude of the transform in decibels is taken and a threshold is computed based on the power of the 802.11b signal across its 3-dB bandwidth. The decibel was the preferred unit since it allows the filter to focus only on significant changes in the magnitude. An *envelope detector* scans the STFT sequence and identifies frequency bins that exceed the threshold, which are expected to be those of the jammer. The *frequency bin exciser* is then notified to set all those bins to zero. The exciser also notches an additional 3 bins on each end of the range of those bins in order to remove any jammer-occupied bins that may not have met the threshold. Although a portion of the signal will always be notched out with the jammer, the algorithm can be shown to perform well with a good thresholding and excision scheme.

70

After the filtering is performed, the Inverse-FFT (IFFT) is computed on the transform, resulting in the jammer-free windowed signal. Since shifted copies of the sliding windows overlap with each other in time, an *overlap-add* operation is performed on the outputs such that the sum of the shifted windows is unity. This guarantees that the outputs corresponding to signal samples that were included in two different windows are combined accordingly.

5.3.1. Window Types and Parameters

Five different windows were considered for the filter, namely the Rectangular, Bartlett, Hanning, Hamming and Blackman windows. Two important criteria were used in order to choose the appropriate window for the filter [19]:

    a. **Main Lobe Width** – Each of the five windows has a bandlimited transform, and the width of the main lobe was the first consideration. The width is defined as the range from zero radians to the first zero crossing of the main lobe on either side. A smaller width yields finer spectral resolution of the jammer; the jammer's frequency spread is confined to a smaller number of FFT bins. This can be directly seen from equation 5.2, which shows that the STFT is the convolution of the Fourier transforms of $x[n]$ and $w[n]$. A smaller main lobe width for $W(e^{j\Omega})$ will result in a smaller bandwidth for the jammer in $X(r,k)$. This is desirable because it reduces the overlap of frequency bins occupied by the jammer and the signal, so that a minimal amount of the signal will be excised along with the jammer.

b. Side-Lobe Attenuation – Each window's frequency response has side-lobes that are not excised along with the main lobe of the jammer. This is because they have a certain degree of attenuation (in the order of tens of decibels) from the main lobe and will not meet the set threshold. Thus, in order for the side-lobes to interfere minimally with the rest of the signal, it is desirable for them to have as much attenuation relative to the main lobe as possible.

Table 6.1 summarizes the properties of the windows. Note that the width of the main lobe is a function of the window's length, *N*.

Window	Peak Side-Lobe Attenuation (dB)	Approximate Width of Main Lobe (radians)
Rectangular	-13	$4\pi/(N+1)$
Bartlett	-25	$8\pi/N$
Hanning	-31	$8\pi/N$
Hamming	-41	$8\pi/N$
Blackman	-57	$12\pi/N$

Table 5.1. Each window has its own peak side-lobe attenuation and main lobe bandwidth. Source: [19]

It can be observed that as the main lobe width increases, the side-lobe attenuation also increases. Based on the considerations mentioned above and on the properties listed in table 5.1, the Hamming window was chosen since it provided a good tradeoff between the side-lobe attenuation and main lobe width. Although the width of its main lobe is twice the size of that of the rectangular window, its side-lobe attenuation is 28 dB less, which reasonably compensates for the width.

Although the rectangular window might have seemed like a good candidate, it was not compatible with the thresholding algorithm, since it completely isolated one

windowed portion from the next, resulting in different thresholds for adjacent sections. This in effect left sections of the jammer at the edges of adjacent windows.

A Hamming window $w[n]$ of length N is defined as:

$$w[n] = \begin{cases} 0.54 - 0.46\cos(2\pi n / N), & 0 \leq n \leq N-1 \\ 0, & otherwise \end{cases} \qquad (5.3)$$

A Hamming window with length $N = 1838$ is shown in figure 5.6. This value for N was derived to be the optimal value, as will be explained in section 5.2.4.
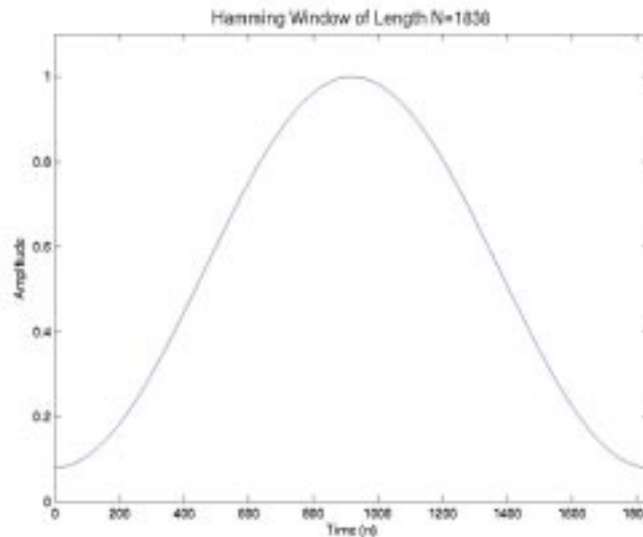
Figure 5.6. Plot of Hamming Window with length $N = 1838$ samples.

The Hamming window is shifted by $N/2$ samples every time it is multiplied by the input signal, creating overlap between adjacent copies of the window. The shifts are chosen such that the overlapping regions add up to unity in order to avoid any distortion

73

of the input aside from the excision itself. The summation is performed by the overlap-add operation at the output of the IFFT operation.

Figure 5.7. Shifted copies of the window $w[n]$ add up to unity (dashed line).

5.3.2. Optimal Window Length

The optimal window length minimizes the frequency spread of the jammer in each STFT. A window of length N samples has a duration of $\Delta t = \dfrac{N}{F_s}$ seconds, where F_s

is the sampling frequency set at 44 MHz.[16] During this time, the jammer will sweep a small frequency range Δf if the window multiplies it in its frequency-sweeping phase.

The value Δf depends on the rate at which the jammer sweeps, defined as its velocity v. Since the sweeping was observed to be of a parabolic nature, v is constantly changing and cannot be determined at any point. A reasonable approximation of v can be obtained by assuming that linear sweeping occurs over the relevant time frame. With this said, we can define $\Delta f \equiv v \Delta t$.

Based on the emission measurements listed in table 4.5, the jammer frequency was assumed to linearly ramp up from 0 to 5.5 MHz in 1.2 milliseconds, resulting in a velocity of $v = \dfrac{5.5 MHz}{1.2 ms} = 4.58 \dfrac{GHz}{s}$. The frequency spread of the jammer s, given as the frequency range that the jammer occupies, can be expressed as:

$$s = 2B + \Delta f \text{ , where } B = \text{width of main lobe of window in Hertz [18]} \qquad (5.4)$$

Substituting $\Delta f = v \Delta t = \dfrac{vN}{F_s}$ and converting B into Hertz, $B = \dfrac{8\pi}{N} \times \dfrac{F_s}{2\pi} = \dfrac{4F_s}{N}$

results in:

$$s = \frac{8F_s}{N} + \frac{vN}{F_s} \qquad (5.5)$$

Solving for the value N_{opt} that gives the minimum spread s, we get:

---

[16] The sampling frequency of 44 MHz is used at the output of the pulse-shaping filter and input to the matched filter, as discussed in Chapter 4.

$$\min_{N}\left\{s = \frac{8F_s}{N} + \frac{vN}{F_s}\right\} \rightarrow \frac{ds}{dN} = -\frac{8F_s}{N_{opt}^2} + \frac{v}{F_s} = 0 \qquad (5.6)$$

$$N_{opt} = F_s\sqrt{\frac{8}{v}} = 44x10^6\sqrt{\frac{8}{4.58x10^9}} \approx 1838.33 \qquad (5.7)$$

$$N_{opt} \equiv 1838 \text{ samples} \qquad (5.8)$$

Thus, a Hamming window with length 1838 samples was used for the filter.

5.3.3. Thresholding Algorithm

The thresholding algorithm drew its reference on the assumption that the STFT samples within the 3-dB bandwidth of the 802.11b signal formed a random distribution of points. A reliable threshold can be derived from the signal magnitude across the bandwidth since the SQRRC filter's frequency response in this range is relatively flat.

The algorithm computes the average magnitude μ of the samples within the 3-dB bandwidth, as well as their standard deviation, σ. The threshold l is defined as the magnitude level that is α standard deviations above μ, where α is a scalar multiplier. Any frequency bins that have a magnitude greater than l are set to zero.

$$l = \mu + \alpha\sigma \qquad (5.9)$$

The multiplier α determines the efficacy of the threshold: a low value may cause significant parts of the signal to be notched out in the absence of a jammer, while a high value may cause the jammer to be left in even after excision. Thus, the optimal value of α would be the smallest possible value that does not cause any notching out of the signal in

the absence of a jammer. This in turn also corresponds to the best value for jammer detection since the lower the value of α, the more discriminating the threshold is against sharp spikes.

For a normal distribution of samples with mean μ and standard deviation σ, a value α_i is associated with the probability that an observation will be made outside the range $\mu \pm \alpha_{i\_}$. For example, there is a 5% probability that an observation will be made above or below 2 standard deviations from the mean ($\alpha = 2$).

Since in this case the nature of the sample distribution was not known, α_{opt} was determined empirically. A parametric simulation was set up to test a range of values of α. The values considered ranged from 1.5 to 3.0. The simulation parameters are listed in table 5.2.

Experiment Parameters	
Data rate	11 Mbps
PLCP Format	Long
Number of Packets/value of α	200
Signal-to-Noise Ratio (SNR)	8.25 dB
Payload Size	1024 Bits
Payload Type	Random
Microwave Oven Jammer	None

Table 5.2. The simulation parameters for determining the best value for α.

Without any excision, an FFT followed by an IFFT operation on a *real*-valued sequence results in a measured 0.79 dB drop in SNR, since the IFFT operation produces a slightly inaccurate *complex* output from which only the real part is taken; the imaginary

part accounts for the SNR loss.[17] Thus, the maximum SNR for the simulation without performing any excision is approximately 7.46 dB, compared to an original SNR of 8.25 dB. The simulation results are listed in table 5.3.

It is evident from table 5.3 that as α increases, the output SNR asymptotically approaches the highest possible value of 7.46 dB. The output SNR values plateau starting at $\alpha = 2$; the increase in output SNR is marginal (approximately 0.01 dB) at higher values. Thus, $\alpha = 2$ was chosen as the value for the thresholding algorithm.

Threshold Experiment Results			
α	BER	FER	Output SNR (dB)
1.5	0.0002	0	5.9
1.65	0	0	6.8
1.8	0	0	7.31
2	0	0	7.43
2.2	0	0	7.43
2.4	0	0	7.44
3	0	0	7.44

Table 5.3. The simulation parameters for determining the best value for α

5.4. Interference Mitigation Example

The Transform Domain Filter was simulated on a packet corrupted with the microwave oven jammer and noise. The packet was modulated at 11 Mbps, used a Long PLCP format and had a payload of 2048 randomly generated bits. The signal-to-jammer (SJR) ratio was set to –21 dB, giving the jammer superior energy over the 802.11b signal, while the signal-to-noise (SNR) was set at 8.25 dB. Figure 5.7 plots the magnitude of the signal with AWGN before encountering the microwave oven jammer.

---

[17] This is a result of rounding of numbers in Matlab.
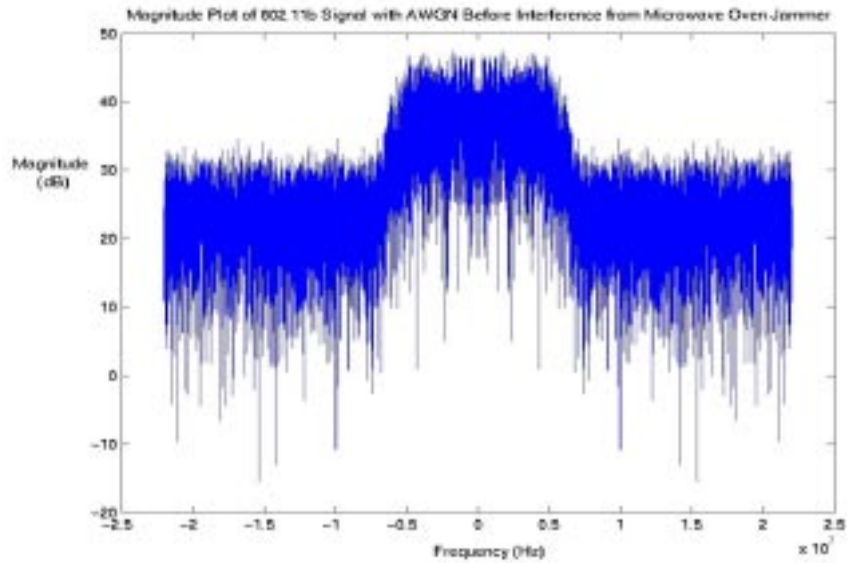
Figure 5.7. The magnitude plot of 802.11b signal with AWGN before encountering the microwave oven emissions.

The microwave oven interference, at the beginning of its frequency-sweeping

phase, was then added to the signal. The presence of the microwave oven jammer can be

clearly seen in the received signal's magnitude plot in figure 5.8. The jammer occupies a

'blob' approximately 3 MHz wide on each side which rises to approximately 21 dB

above the rest of the signal. Again, the frequency spread of the jammer is much wider

than its actual instantaneous bandwidth due to its frequency-sweeping behavior.

Figure 5.8. Magnitude plot of the 802.11b signal with AWGN and the microwave oven jammer, represented by the 'blob' in the middle that rises to 65 dB.

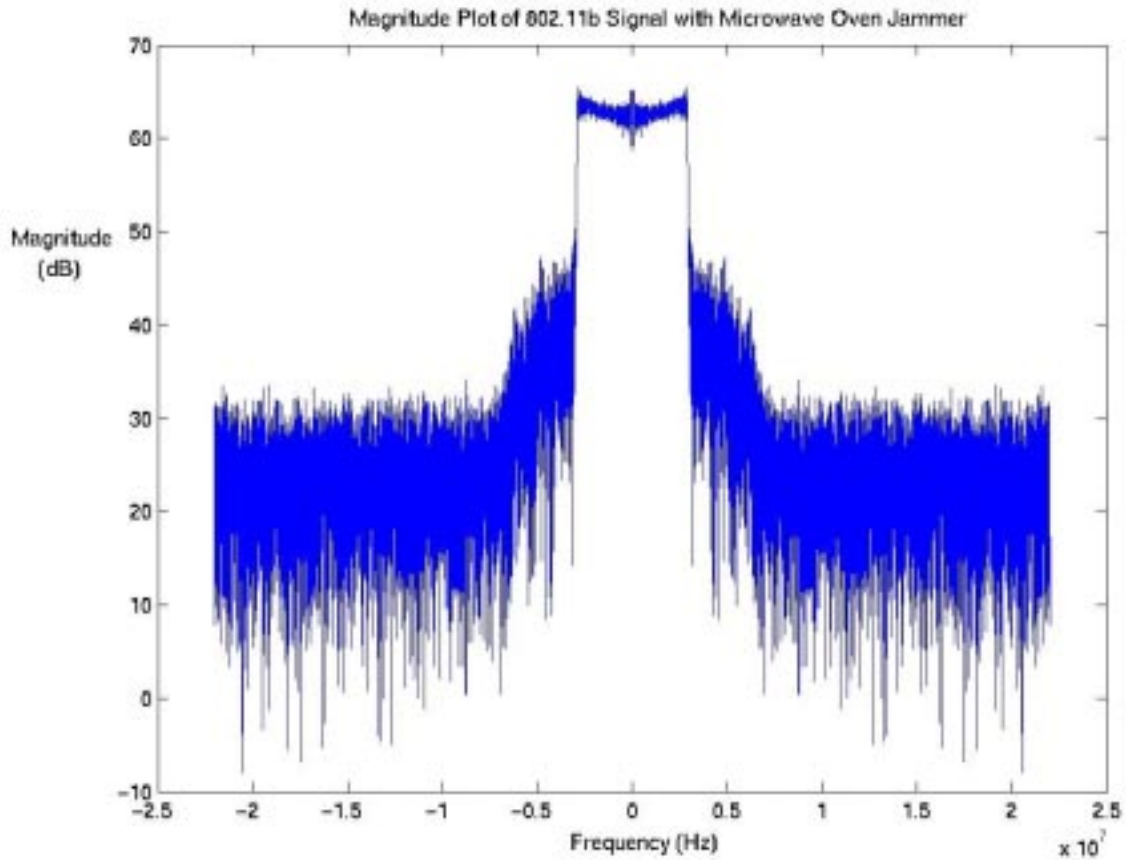The received signal was then processed by the filter. Figure 5.9 shows the magnitude plot of a particular STFT sequence of the signal, windowed at the very end of the packet. The frequency of the jammer is clearly stationary at approximately 3 MHz over the windowed portion.
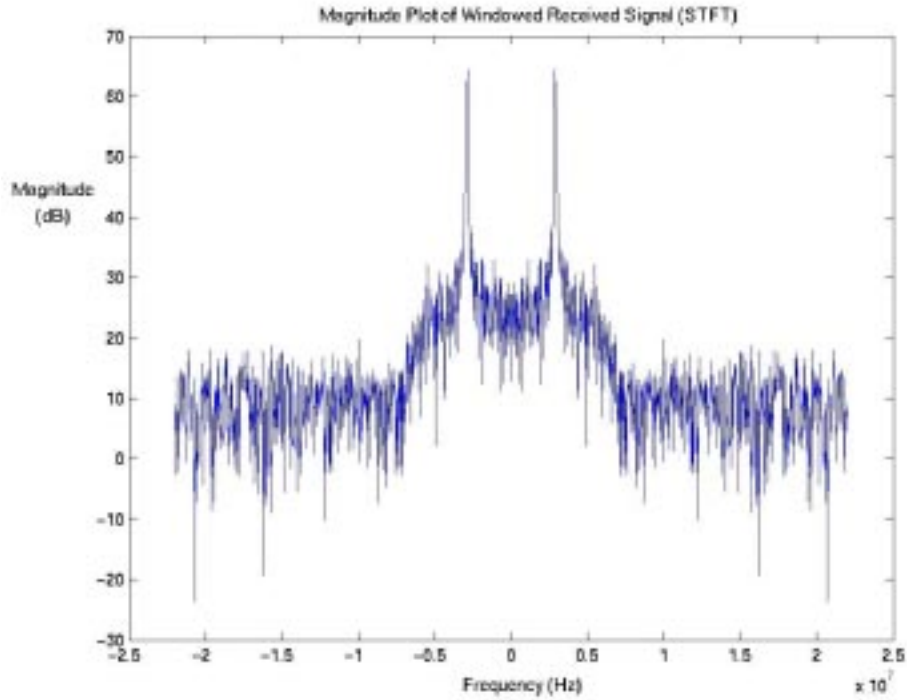
Figure 5.9. Magnitude plot of the last windowed portion of the signal with the interference. Note how the jammer is localized into one frequency (as seen in the sharp spikes) by the STFT operation.

A threshold of 44.93 dB was computed for the STFT shown above. The frequency bin exciser was notified about the threshold and set all bins exceeding the value to zero. An extra 3 frequency bins were set to zero from the left and rightmost bins that exceeded the threshold in order to include any frequency bins occupied by the jammer that possibly did not meet the value. Figure 5.10 illustrates the resulting magnitude plot after the
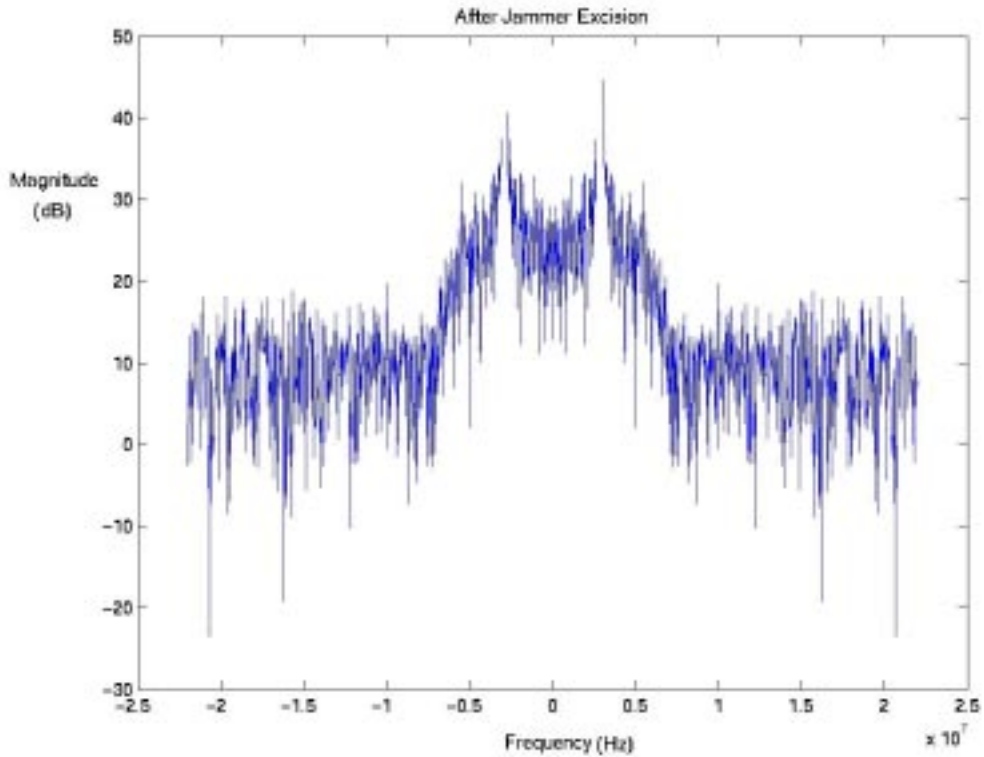
Figure 5.10. Magnitude plot of the resulting windowed portion after jammer excision. Some frequency bins occupied by the jammer are still left intact, although the strongest portion is notched out.

jammer frequency bins were set to zero. Although the peaks of the spikes were notched out, it can also be observed that there were remnants of the jammer that did not meet the threshold; this is a limitation of the filter that will be discussed in Chapter 8.

An IFFT operation was performed on the sequence and an overlap-adder block added the output accordingly to the earlier filter outputs. The magnitude plot of the filter's entire output, shown in figure 5.11, clearly shows the absence of the microwave oven jammer; the interference observed in figure 5.8 was removed.
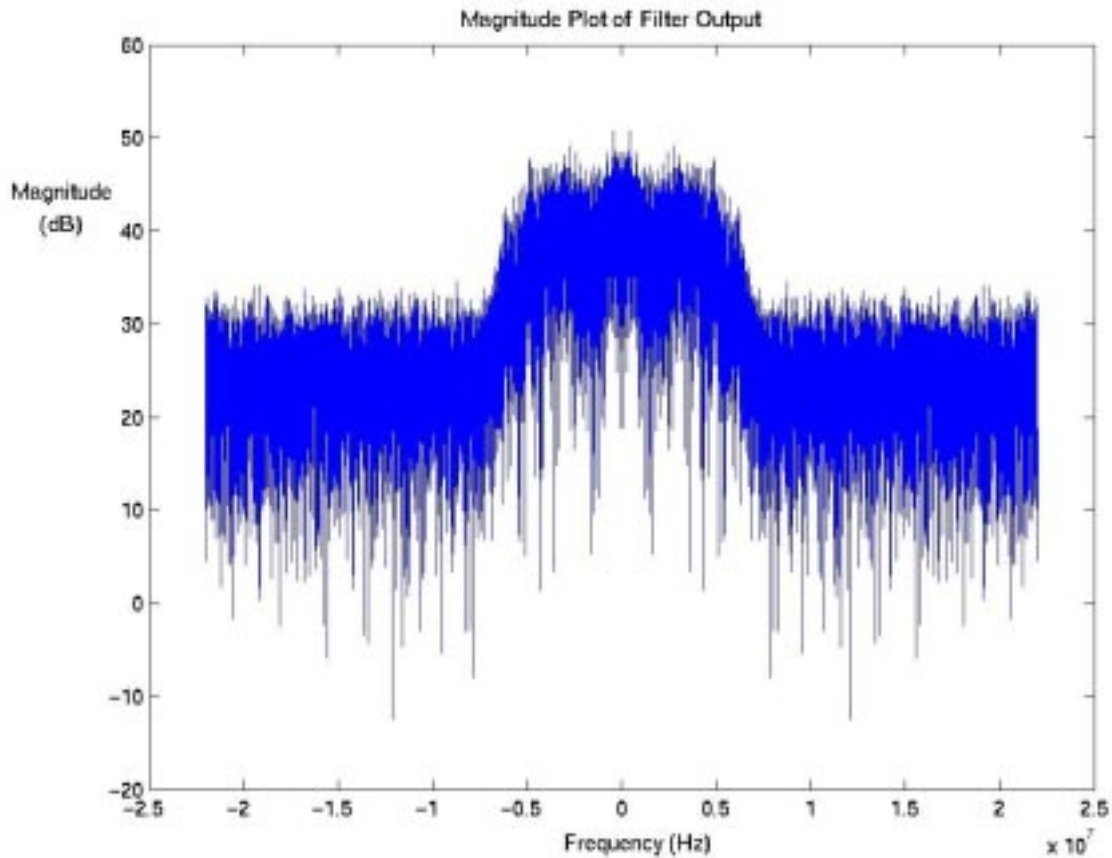
Figure 5.11. Magnitude plot of the 802.11b signal with the microwave oven interference ('blob' in the middle) and white Gaussian Noise.

Finally, the nearly identical spectrograms of the actual jammer and the filtered jammer, shown in figure 5.12, confirm that the filter successfully tracked the microwave oven jammer as it swept in frequency. On a side note, thin, light vertical lines can be observed for fixed time intervals of approximately 20 µs in Figure 5.12b, indicating the presence of small discontinuities in the filtered waveform for brief periods of time. These discontinuities arise from the fact that there are different thresholds between adjacent windowed portions, which result in a different scaling of the filtered jammer for each window.
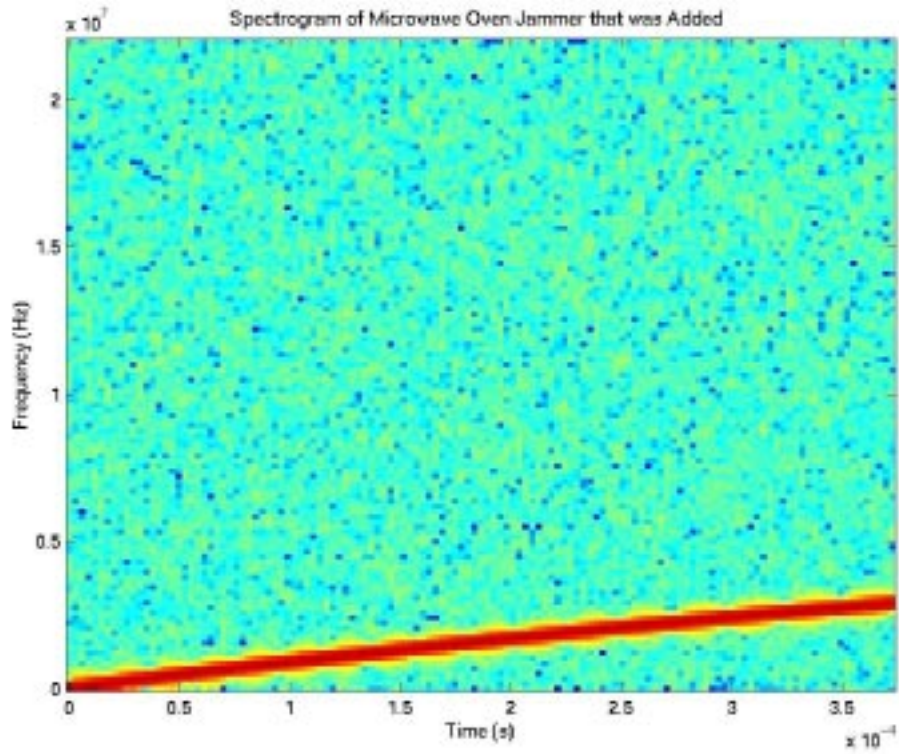
(a)
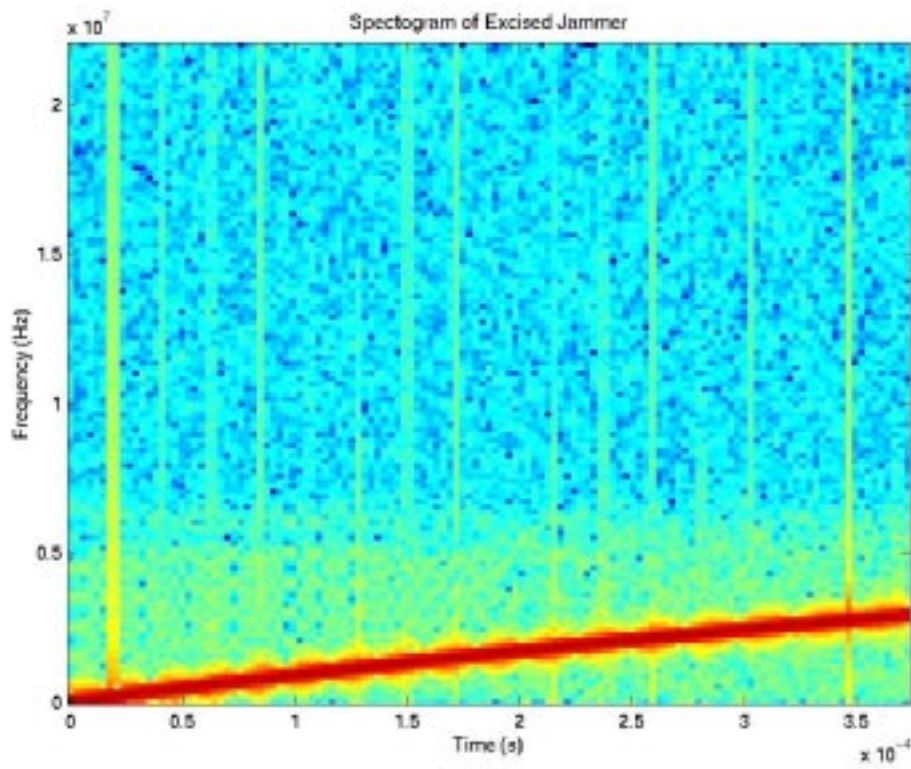
Figure 5.12. (a) Spectrogram of the adder jammer. (b) Spectrogram of the jammer that was removed by the filter. Clearly, the interference was tracked and removed by the filter.

5.5. Summary

The transform domain filter presented in this chapter exploits frequency-domain differences between the 802.11b signal and the microwave oven interference; the SQRRC filter used for pulse-shaping has a flat frequency response over its 3-dB bandwidth, while the microwave oven jammer can be viewed over short time intervals to have a sharp peak at a fixed frequency. A Hamming window was chosen for computing the STFT, since it provided a reasonable tradeoff between its main lobe width and side lobe attenuation, and was also suitable for the proposed thresholding algorithm. The thresholding algorithm derived a threshold over the FFT bins in the 3-dB bandwidth of the signal, based on the computed mean and standard deviation.

Chapter 6: Least-Mean-Square (LMS) Interpolation Filtering

6.1. Normalized-LMS Interpolation Filter

A Least-Mean-Square (LMS) Interpolation filter is an adaptive, finite impulse response (FIR) filter that has the ability to track and interpolate a signal in the presence of background noise. As illustrated in Figure 6.1, the filter produces an estimate $\hat{x}[n]$ of the current input sample $x[n]$ using a sum of N weighted values of past and future samples of the input signal. The weights placed on the samples are the time-varying filter taps, which are iteratively updated at each time instant using a mean-squared error (MSE) cost function in order to improve future estimates and thereby reduce the error.

Although there are various methods for updating the filter taps, the filter explored in this chapter implements the Windrow-Hoff Normalized-LMS (NLMS) algorithm. The algorithm is a recursive approximation method that updates the taps in the
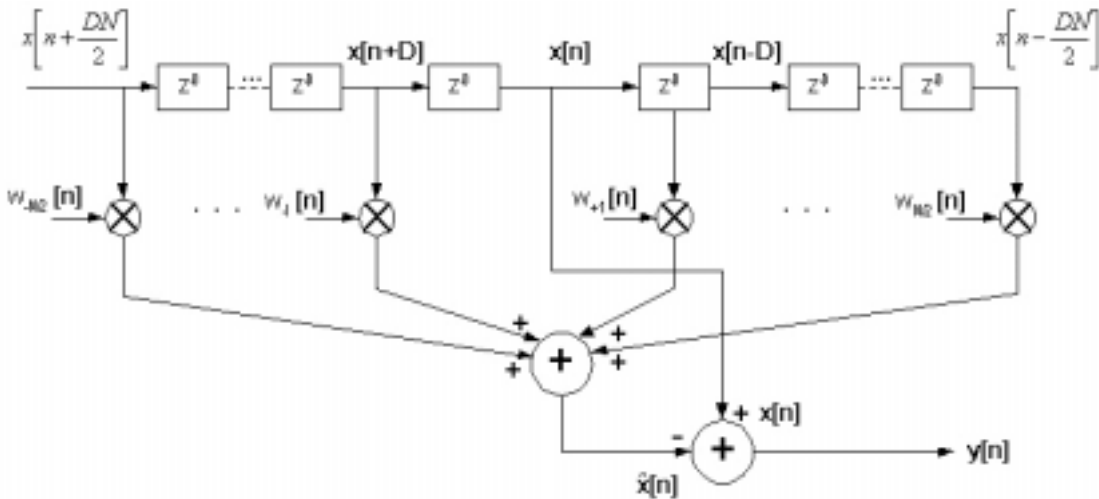
Figure 6.1. A basic diagram of the LMS Interpolation filter. The filter taps are updated each time an output sample is generated (mechanism not shown above).

direction of the *instantaneous* gradient of the MSE between the filter estimate and the input; the algorithm converges to the optimal set of taps that will minimize the average MSE.

The filter can be applied to the microwave oven interference problem by configuring it such that it reconstructs the jammer and subtracts it out of the received signal. However, if the filter were to estimate only the jammer, it must be configured to treat the 802.11b signal as background noise. Consequently, the filter output will be the estimation error $y[n] = x[n] - \hat{x}[n]$, composed of only the 802.11b signal and white noise in the received signal.

The filter has three parameters, namely the number of taps *N*, the time-varying tap updating-coefficient $\mu[n]$, and an additional parameter *D*, which is a time delay between taps. The parameter *D*, discussed in more detail in section 6.3.2, was a modification introduced to the normal LMS filter in order for it to effectively track the jammer added to an 802.11b signal; a typical LMS filter has *D*=1.

6.2.   Integration with Receiver

Similar to the placement of the TDF filter presented in Chapter 5, the LMS filter was integrated between the Analog-to-Digital Converters (ADCs) and the SQRRC matched filters at the receiver. Figure 6.2 shows the configuration.
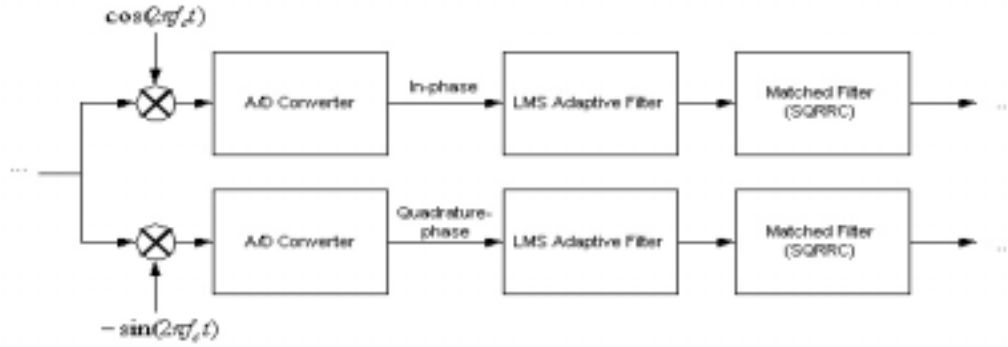
Figure 6.2. The LMS adaptive filters are placed between the ADCs and the matched filters.

6.3. Principles of Operation

The received signal $r[n]$, passed as input to the filter, is comprised of the 802.11b

signal $s[n]$, the microwave oven jammer $j[n]$, and additive white Gaussian noise

(AWGN) $\zeta[n]$. The input can be expressed as:

$$r[n] = s[n] + j[n] + \zeta[n] \qquad (6.1)$$

The filter produces an output $y[n] = r[n] - \hat{r}[n]$, where $\hat{r}[n]$ is the estimate of the

input and is simply the convolution sum of the filter taps with the input. Consider the

basic LMS filter with N taps and $D = 1$. Assuming that N is even for simplicity, the

estimate $\hat{r}[n]$ can be expressed as:

$$\hat{r}[n] = \sum_{l=1}^{\frac{N}{2}} w_l r[n-l] + w_{-l} r[n+l] \qquad (6.2)$$

In order for the filter to find the optimum values for the taps, it minimizes the mean-squared error (MSE) between the estimate of the jammer and the received signal, which can be written as:

$$MSE = E\{\epsilon^2[n]\} = E\{(r[n] - \hat{r}[n])^2\} \quad (6.3)$$

$$= E\left\{\left(r[n] - \sum_{l=1}^{\frac{N}{2}} w_l r[n-l] + w_{-l} r[n+l]\right)^2\right\} \quad (6.4)$$

Minimizing the MSE with respect to each of the filter taps in $\{w_l\}$, its partial derivative is taken with respect to each tap and set to zero:

$$\frac{\partial E\{N\}}{\partial w_i}\bigg|_{1 \leq |i| \leq \frac{N}{2}} = E\left\{2\left(r[n] - \sum_{l=1}^{\frac{N}{2}} w_l r[n-l] + w_{-l} r[n+l]\right)r[n-i]\right\} = 0 \quad (6.5)$$

Rearranging the terms:

$$E\{r[n]r[n-i]\}\bigg|_{1 \leq |i| \leq \frac{N}{2}} = \sum_{l=1}^{\frac{N}{2}} w_l E\{r[n-l]r[n-i]\} + w_{-l} E\{r[n+l]r[n-i]\} \quad (6.6)$$

However, the expectation terms above are autocorrelation values of $r[n]$ noted as $\rho_{rr}[i]$, and so the equation above can be written as:

$$\rho_{rr}[i] = \sum_{l=1}^{\frac{N}{2}} w_l \rho_{rr}[i-l] + w_{-l} \rho_{rr}[i+l] \qquad (6.7)$$
$$\scriptstyle 1 \le |i| \le \frac{N}{2}$$

The set of equations defined in equation 6.7 is also known as the set of Yule-Walker normal equations for $r[n]$. The set of filter taps $\{w_{opt}\}$ that satisfies the equations is optimal for linearly predicting $r[n]$ in the mean-squared error sense.

In the context of this problem, however, an estimate of the *jammer* and not the entire *received* signal is needed. By analyzing the autocorrelation function of $r[n]$ and the various properties of the signals that compose it, it becomes apparent that the filter can be modified to interpolate the jammer instead of the received signal.

Recalling equation 6.1, the autocorrelation function of $r[n]$ can be expressed as:

$$\rho_{rr}[i] = E\{r[n]r[n-i]\} = E\{(s[n] + j[n] + \zeta[n])(s[n-i] + j[n-i] + \zeta[n-i])\} \qquad (6.8)$$

for $\left\{ i : 1 \le |i| \le \dfrac{N}{2}, i \in N, \right\}$. However, $\zeta[n]$ is AWGN, which means it is a zero-mean,

i.i.d. process, and has a unit-sample autocorrelation function, as defined in equation 6.9:

$$\rho_{\zeta\zeta}[i] = E\{\zeta[n]\zeta[n-i]\} = \begin{cases} \sigma_\zeta^2, & i = 0 \\ 0, & otherwise \end{cases} \qquad (6.9)$$

where σ_ζ^2 is the variance of the noise. Given this property and the fact that $\zeta[n]$, $s[n]$ and $j[n]$ are mutually independent, all the terms in equation 6.8 with $\zeta[*]$ go to zero, leaving:

$$\rho_{rr}[i] = \rho_{ss}[i] + \rho_{sj}[i] + \rho_{js}[i] + \rho_{jj}[i] \qquad (6.10)$$

Furthermore, an 802.11b signal in baseband is zero-mean ($E\{s[n]\} = 0$), for all four data rates. This property allows the cross-correlation terms of $s[n]$ and $j[n]$ to drop out, resulting in:

$$\rho_{rr}[i] = \rho_{ss}[i] + \rho_{jj}[i] \qquad (6.11)$$

If the $\rho_{ss}[i]$ term drops out, $\rho_{rr}[i] = \rho_{jj}[i]$ and the linear equations in equation 6.7 reduce to a matrix of normal equations for interpolating the jammer. Similar to the case of the noise term $\zeta[n]$, $\rho_{ss}[i]$ has to be an impulse in order for it to fall out of the equation.

The autocorrelation function of the transmitted signal $s[n]$, however, is not an impulse. The signal $s[n]$ is the output obtained from passing the Barker/CCK symbols through the pulse-shaping filter. Since the symbol streams have zero-mean, unit variance and noise-like autocorrelative properties as discussed in section 2.6.1, we can assume that their autocorrelation functions are of the same form as equation 6.9. With a pulse-shaping filter impulse-response $h[n]$ and input autocorrelation function $\rho_{xx}[i]$, the autocorrelation of $s[n]$ can be solved for as:

$$\rho_{ss}[i] = h[i] * h[-i] * \rho_{xx}[i] \qquad (6.12)$$

$$\rho_{ss}[i] = \left(\sum_{k=-\infty}^{+\infty} h[k]h[i+k] \right) * \delta[i] = \rho_{hh}[i] \qquad (6.13)$$

$$\rho_{ss}[i] = \rho_{hh}[i] \qquad (6.14)$$

In order to illustrate this effect, figure 6.3 shows the autocorrelation function of the output of the pulse-shaping filter from a Barker code input, where the output is essentially the resulting waveform from modulating one bit. The function is a slowly-decaying pulse that reaches a value of zero only at about a lag of 50. As a result, the non-zero autocorrelation values within this range contribute to the interpolation of the jammer, producing an erroneous estimate.
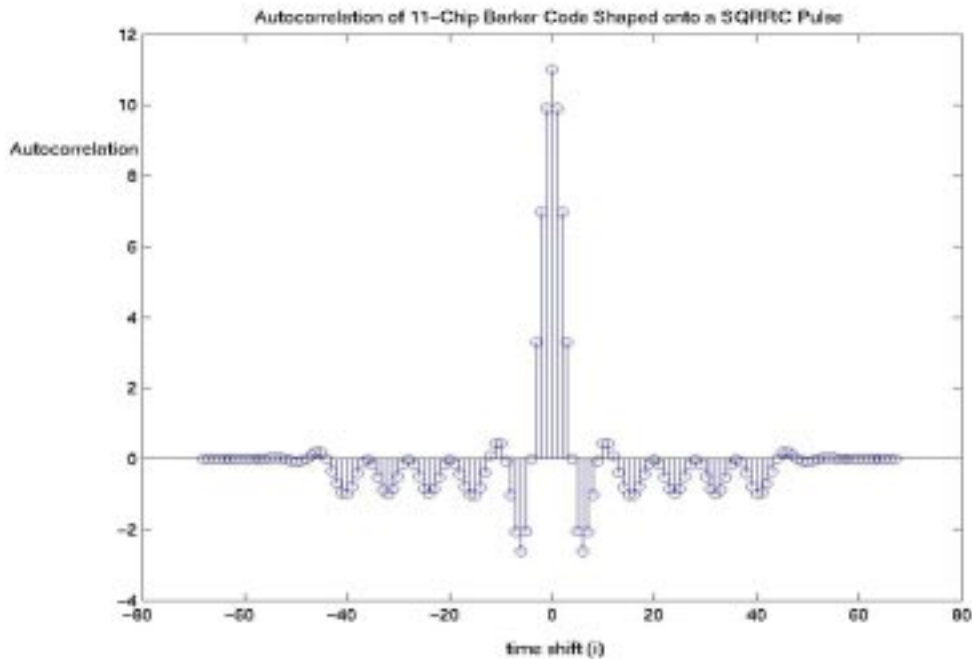

Figure 6.3. The autocorrelation function of the output of the SQRRC filter with the Barker code as input.

As it turns out, a simple modification can be made to the filter to get around this problem: introducing the additional delay parameter *D* mentioned in section 6.1 transforms the autocorrelation function of $s[n]$ such that it looks like a unit-sample. The explanation for *D* will be skipped for now and will be discussed in more detail in section 6.4a. With this problem resolved, equation 6.11 simplifies down to:

$$\rho_{rr}[i] = \rho_{jj}[i] \qquad (6.15)$$

The filter now settles at the set of taps $\left\{w_l^{opt}\right\}$ for interpolating the microwave oven jammer. The filter produces an estimate of the microwave oven jammer $\hat{j}[n]$ and subtracts it out to produce the interference-mitigated output $\tilde{r}[n]$.

$$\tilde{r}[n] \approx s[n] + \zeta[n] \qquad (6.16)$$

6.3.1. Updating Algorithm

The Windrow-Hoff Normalized-LMS (NLMS) algorithm implemented by the filter is a recursive approximation method that minimizes the *instantaneous* mean-squared error (MSE) between the received signal $r[n]$ and the interpolated signal $\hat{j}[n]$. The algorithm is memoryless; the current values of the filter taps incorporate all past states of the filter. Since the tap values vary with time, each tap w_i at time *n* will be hereby noted as $w_i[n]$. The algorithm uses a *gradient* or *steepest-descent* method, wherein the

taps are updated along the direction of the gradient of the MSE. For this method, the recursive estimator for the taps of the form:

$$w[n+1] = w[n] + \mu[n]V[n] \quad (6.17)$$

where $\mu[n]$ is the updating coefficient and $V[n]$ is the corresponding gradient metric at time n. In the Windrow-Hoff algorithm, an instantaneous estimate of the *negative* gradient of the MSE cost function is used as the update parameter: $V[n] = -\nabla \hat{E}\{e^2[n]\}$. All filter taps are initialized to zero ($w[0] = 0$).

Recalling the definition of the MSE cost function from section 6.2:

$$MSE = E\{e^2[n]\} = E\{(r[n] - \hat{j}[n])^2\} \quad (6.18)$$

The negative gradient can be expressed as:

$$-\nabla E\{e^2[n]\} = -\frac{\partial E\{e^2[n]\}}{\partial w_i} = -E\{2(r[n] - \hat{j}[n])[n-i]\} \quad 1 \le |i| \le \frac{N}{2} \quad (6.19)$$

An instantaneous estimate of the negative gradient uses the current gradient value produced by the filter:

$$-\nabla \hat{E}\{e^2[n]\} = -2(r[n] - \hat{j}[n])r[n-i] = -2y[n]r[n-i] \quad (6.20)$$

94

$$-\nabla \hat{E}\{e^2[n]\} = -2y[n]r[n-i] \qquad (6.21)$$

Substituting this value in for $V[n]$ in Equation 6.17, the formula for updating the filter taps is defined as:

$$w_i[n+1] = w_i[n] - 2\mu[n]y[n]r[n-i], \quad 1 \le |i| \le \frac{N}{2} \qquad (6.22)$$

The parameter $\mu[n]$ is the time-varying coefficient that affects the speed of convergence and stability of the filter. In order for the filter to converge to the optimum solution for equation 6.15, $\mu[n]$ has to be within the following range:

$$0 < \mu[n] < \frac{2}{\lambda_{max}(R)} \qquad (6.23)$$

where R is the autocorrelation matrix of $r[n]$ derived in equation 6.7 and $\lambda_{max}(R)$ is the greatest eigenvalue of R. If the value for $\mu[n]$ is optimized within this range such that the cost function $V_N[n]$ at time n is minimized, it is found to be [21]:

$$\mu[n] = \frac{1}{\left(\displaystyle\sum_{i=1}^{\frac{N}{2}} r^2[n-i] + r^2[n+i]\right)} \qquad (6.24)$$

The parameter $\mu[n]$ is normalized with the energy of the signal inside the filter, hence the name "Normalized-LMS (NLMS)" algorithm. A minor complication can arise when the energy of $r[n]$ is close or equal to zero: $\mu[n]$ can blow up. In order to resolve this, the equation can be slightly modified to:

$$\mu[n] = \frac{\alpha}{\beta + \left(\displaystyle\sum_{i=1}^{\frac{N}{2}} r^2[n-i] + r^2[n+i] \right)} \qquad (6.25)$$

where $\alpha \in (0,2)$ and $0 \le \beta$. The constant α is a relaxation factor, while β is introduced to prevent $\mu[n]$ from blowing up when the energy of the signal is very small. The value of 0.1 for both α and β was adapted from a similar implementation of the algorithm in [21].

Substituting equation 6.25 in Equation 6.24, the formula for updating the LMS filter taps can be written as:

$$w_i[n+1] = w_i[n] - 2\frac{\alpha}{\beta + \left(\displaystyle\sum_{i=1}^{\frac{N}{2}} r^2[n-i] + r^2[n+i] \right)} y[n]r[n-i], \quad 1 \le |i| \le \frac{N}{2} \qquad (6.26)$$

6.3.2. Filter Parameters

The efficacy of the filter depends on its two main parameters: the number of filter taps N and the sample delay D.

a.) Sample Delay, D – As mentioned in section 6.2, introducing a delay D between the filter taps resolves the issue with the autocorrelation of the SQRRC filter output. It was shown in equation 6.14 that the autocorrelation function of the filter output from a white-noise input with zero-mean and unit variance (i.e. the inputs coming from the spreader and CCK channel encoder) is just equal to $\rho_{hh}[i]$, the deterministic autocorrelation function of its impulse response (see Figure 6.3). Since $\rho_{hh}[i]$ is a slowly decaying spike with a considerable width, it will have non-zero values within the range of i. Thus, the term $\rho_{ss}[i]$ in equation 6.11 does not cancel out and the filter produces an inaccurate jammer estimate.

However, introducing a sample delay D allows the filter to process a *downsampled* version of the autocorrelation function of $r[n]$, such that $\rho_{ss}[i]$ can be approximated as an impulse. An LMS filter with a distance of D instants between each tap has a convolution sum that can be rewritten as:

$$\hat{j}[n] = \sum_{l=1}^{\frac{N}{2}} w_l r[n - Dl] + w_{-l} r[n + Dl] \qquad (6.27)$$

Proceeding in the same manner as in section 7.2, minimizing the MSE leads to the Yule-Walker normal equations:

$$\rho_{rr}[i] = \sum_{l=1}^{\frac{N}{2}} w_l \rho_{rr}[D(i-l)] + w_{-l} \rho_{rr}[D(i+l)], \quad 1 \le |i| \le \frac{N}{2} \qquad (6.28)$$

The updating algorithm converges to the set of taps that satisfies equation 6.28. Consequently, a downsampled version of $\rho_{ss}[i]$ will have low values for the range of i and can thus be dropped from equation 6.11, yielding $\rho_{rr}[i] \approx \rho_{jj}[i]$. Figure 6.4 compares the autocorrelation functions of the SQRRC filter and its downsampled version.

On the other hand, the jammer's downsampled autocorrelation function is still a sinusoid, so the filter is able to estimate the jammer. Although microwave oven interference is a frequency-sweeping sinusoid, its frequency can be approximated to be stationary within the relevant time frame spanned by the filter's impulse response. The autocorrelation function of a sinusoid $v[n] = \sin[w_o n]$ at a fixed frequency of w_o can be written as:

$$\rho_{vv}[m] = \sum_{n=-\infty}^{\infty} \sin[w_o n] \sin[w_o (n+m)] = \frac{\cos[w_o m]}{2} \quad (6.29)$$

Downsampling the autocorrelation function of $v[n]$ by D produces

$\rho_{vv}[Dm] = \frac{\cos[Dw_o m]}{2}$, which is clearly another sinusoid at a higher frequency of D.
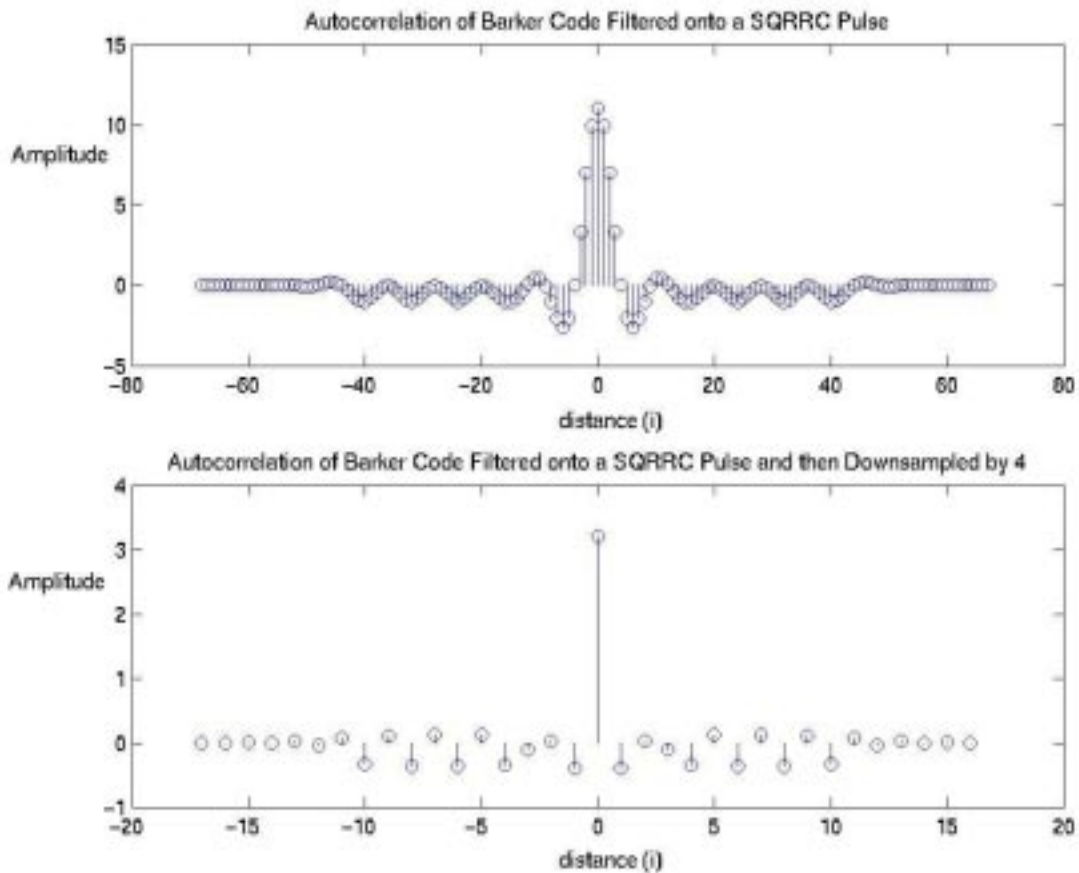
Figure 6.4. The autocorrelation function of a Barker code shaped onto a SQRRC Pulse is plotted on top. The autocorrelation of the same output downsampled by 4 is graphed on the bottom. It looks more like an impulse.

b.) Number of Taps, N – The number of taps determines the filter's ability to reconstruct the jammer. A large set of filter taps allows the filter to use more information from the signal in producing an estimate of the jammer. However, the computational costs increase with number of taps, and too many taps may actually provide a less accurate estimate, especially if the signal has dynamic characteristics (e.g. changing frequencies). At the other extreme, too few filter taps does not capture enough information about the jammer for an accurate reconstruction.

The filter parameters used for the simulations were set to $N = 32$ and $D = 4$. The sample delay D was set to 4 since the pulse-shaping filter upsamples the symbol stream by a factor of 4 before shaping each symbol onto a copy of the impulse response. Thus, given that adjacent symbols of the Barker Code are uncorrelated, signal values 4 samples apart can be expected to likewise have low correlation.

Unfortunately, previous research on determining the number of filter taps based on input characteristics was scarce; a suitable value for N was chosen through conducting simple performance tests on a range of possible values. Its value was chosen such that it provided a good trade-off between interference tracking and computational costs.

6.4. Convergence Time

Since the filter starts with a zero impulse response and adapts itself according to the input in time, its error level has to be reduced to within a certain threshold before packet fields are processed by the receiver. In particular, the filter has to settle before the receiver demodulates the SFD field of the preamble, which provides information on the format of the packet and is the first processed field. In other words, its *convergence time* has to be less than the duration of the SYNC field of the packet preamble. The strictest time constraint is given by the short packet, which has a SYNC field that is 56 bits long, lasting approximately 56 µs since it is modulated at 1 Mbps.

In order to get an approximation of the convergence time of the filter, a single-tone sinusoid and a frequency-sweeping sinusoid were passed as inputs. As can be seen from figure 6.5a, the filter converged and canceled out the sinusoid within 20 µs, well
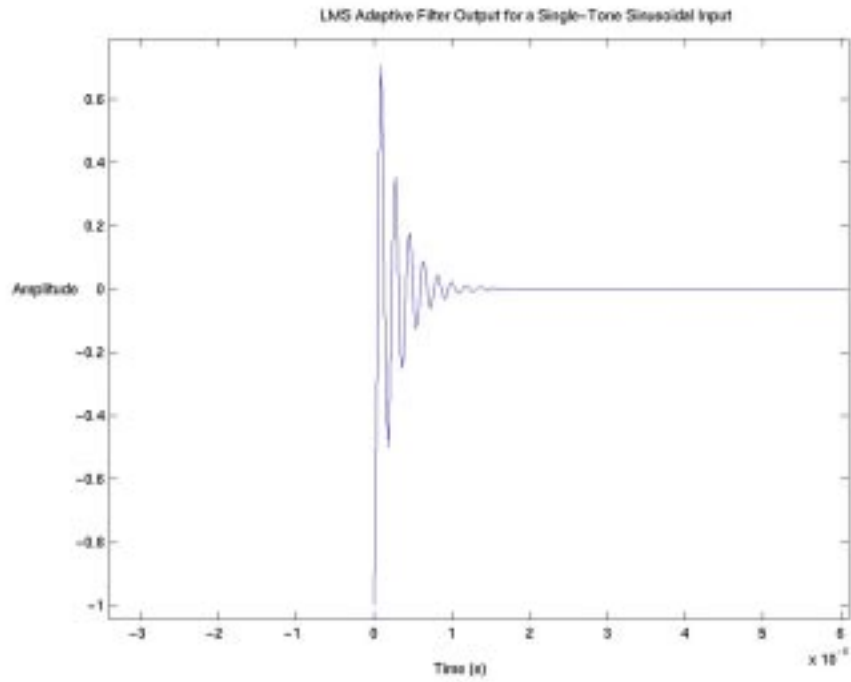
(a)
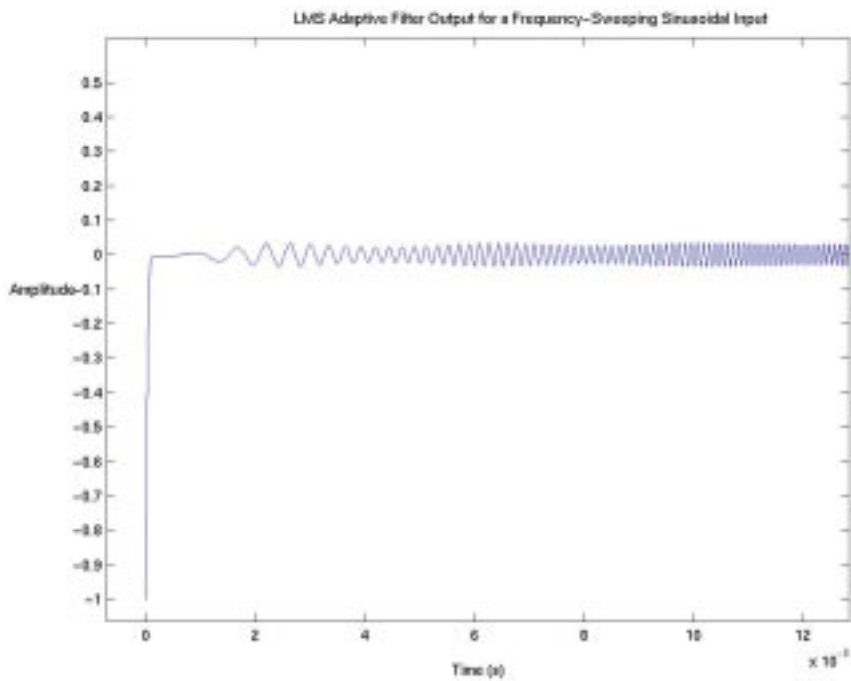
(b)

Figure 6.5. (a) Plot of the filter output for a 5.5MHz sinusoid. (b) Plot of the filter for a quadratically frequency-sweeping sinusoid from 0 to 5.5MHz. The output in this case is the error between the estimate of the sinusoid and the input itself, which converges to zero within 20 µs in both cases.

101

before the 56 μs time constraint. Probabilistically speaking, this is a more likely scenario, since the microwave oven interference is at a fixed-frequency of 5.5 MHz for approximately 5.1 ms out of the 7.5 ms emission time (see Table 4.4).

For the frequency-sweeping case, figure 6.5b shows that the filter seemed to have settled within the first 20 μs, although a dynamic sinusoidal error is observed while the input is also sweeping in frequency. This effect can be attributed to the filter's constant adaptation to the changing input characteristics. However, it can be noted that the error amplitude is less than 0.05, an order of magnitude smaller than the input amplitude, which was set at 1.

Although these test results reflect idealistic scenarios without background noise, they do give an indication that the filter can be expected to converge within the necessary time limit since the filter ignores the noise.

6.5.   Interference Mitigation Example

To illustrate the operation of the LMS Filter, it was simulated on a packet with microwave oven interference and noise. Similar to the example in Chapter 6, the packet used the Long PLCP format and was transmitted at 11 Mbps, with a payload of 2048 randomly generated bits. The signal-to-jammer ratio (SJR) was set to –21 dB, giving the jammer superior power over the signal, while the signal-to-noise ratio (SNR) was fixed at 8.25 dB.

The following plots in this section show the signal at the in-phase; similar results can be expected for the quadrature-phase. Figure 6.6 shows the magnitude plot of the 802.11b signal with white noise before encountering microwave oven interference.
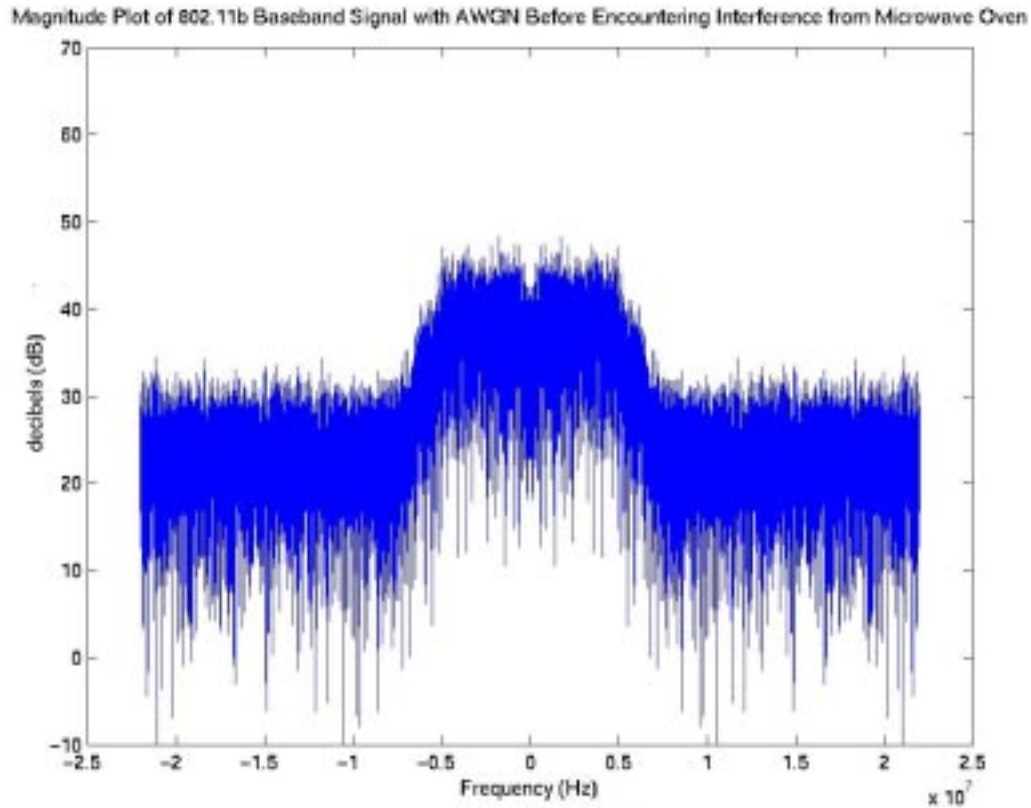
Figure 6.6. Magnitude plot of the 802.11b packet with white noise at an SNR of 8.25 dB. This particular SNR is used for the BER simulations in Chapter 9 since the receiver is supposed to give a 10^{-5} BER at this SNR.

The microwave oven interference was added to the transmitted packet at the beginning of its frequency-sweeping phase. Figure 6.7 shows the magnitude plot of the resulting signal, which shows a significant amount of interference up to 3 MHz, as indicated by the white 'blob' in the middle. The interference can be seen to rise above the signal by approximately 21 dB, consistent with the SJR setting used for the simulation.

The received packet was then processed by the LMS filters to remove the microwave oven jammer. The magnitude of the filter output is plotted in figure 6.8. It appears that the jammer was successfully removed, although some distortions in the output can also be observed.

Figure 6.7. Magnitude plot of the 802.11b packet with noise after encountering the microwave oven jammer at an SJR of –21 dB. The jammer is seen as the 'blob' in the middle rising to 65 dB.
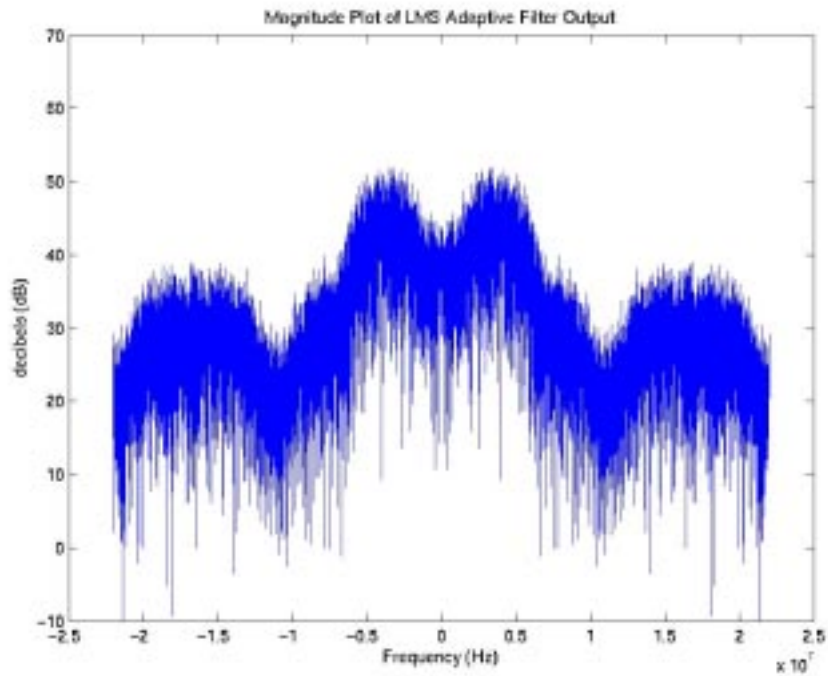
Figure 6.8. Magnitude plot of the filter output. Although it appears that the jammer has been removed, the signal also seems relatively distorted compared to the interference-free plot in Figure 6.6.
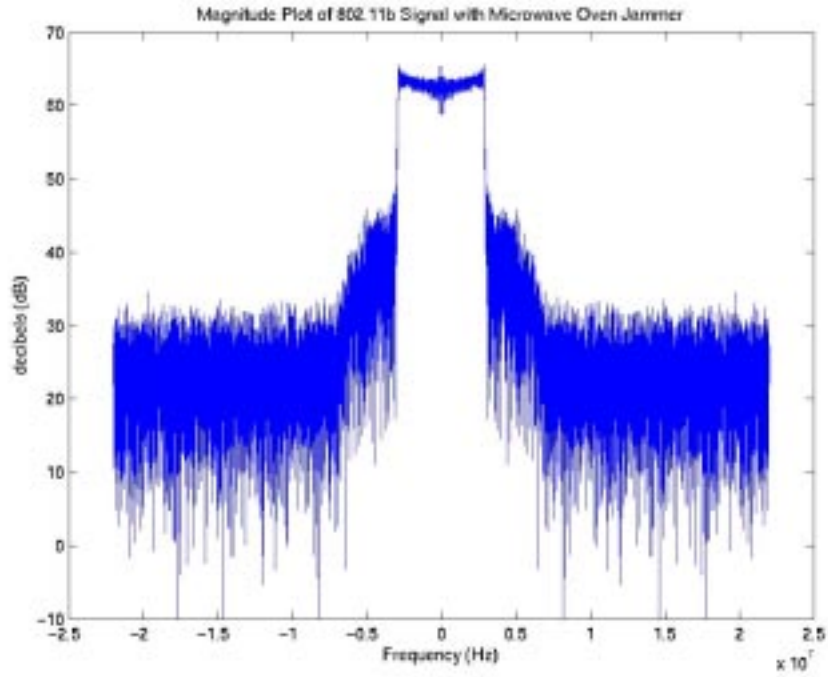
Distortions can be observed at certain frequencies, such as the depression around

0 Hz as well as the outer lobes from 1.75-2.25 MHz, which were not observed in the

magnitude plot of the original signal in figure 6.6. A closer inspection of the jammer

estimate that was subtracted out, shown in figure 6.9, also shows the presence of a main

lobe similar to that of the 802.11b signal, aside from the white 'blob' rising to 65 dB that

represents the jammer. This lobe suggests that a portion of the 802.11b signal might have

also been interpolated.
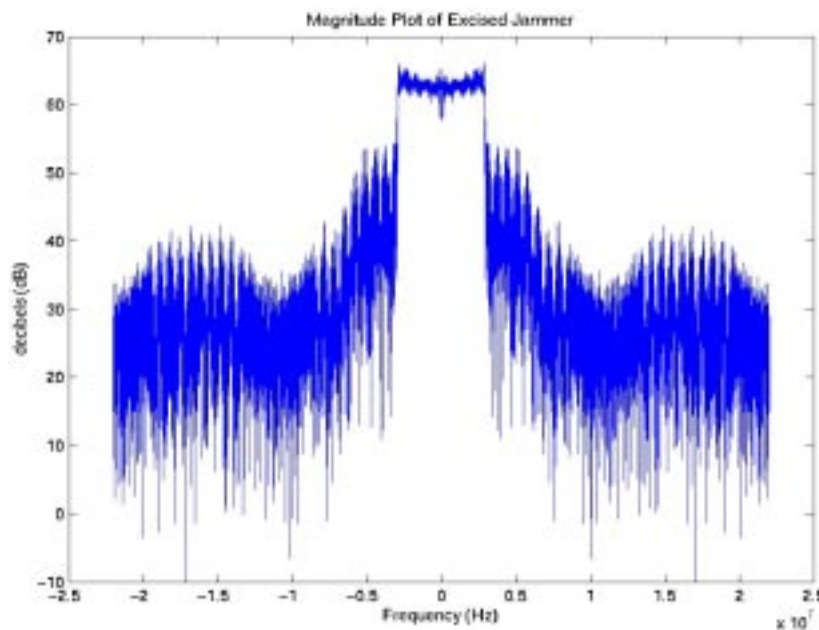
Figure 6.9. Magnitude plot of the jammer estimate that was subtracted by the LMS filter from the received signal. Although it can be seen that the jammer was removed, the presence of the side lobes indicate that part of the signal was interpolated.

The spectrogram of the subtracted jammer estimate is consistent with these

results, as seen in its superposition with the spectrogram of the added jammer in figure

6.10. Darker colored portions within the 3-dB bandwidth of the 802.11b signal indicate

that it was also interpolated, albeit to a lesser degree.
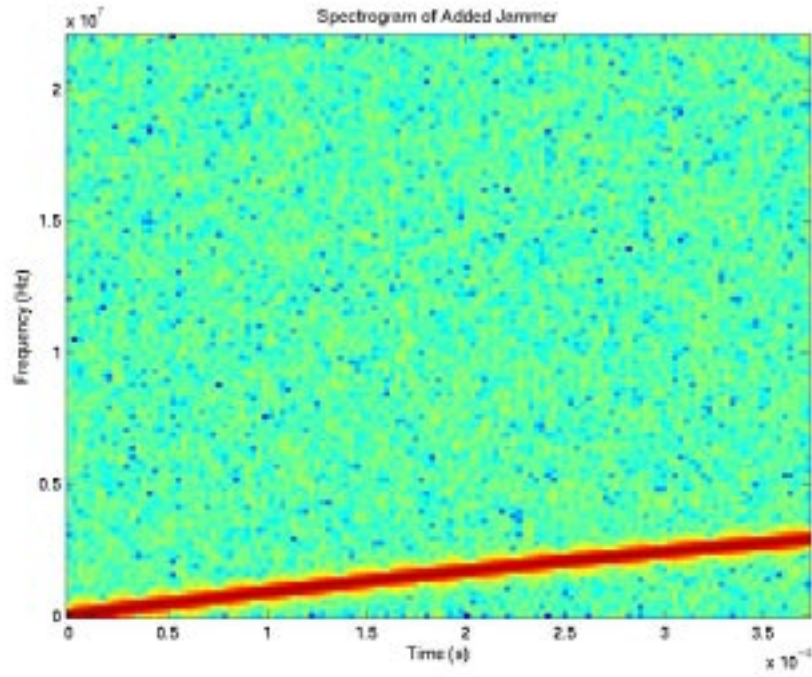
(a)
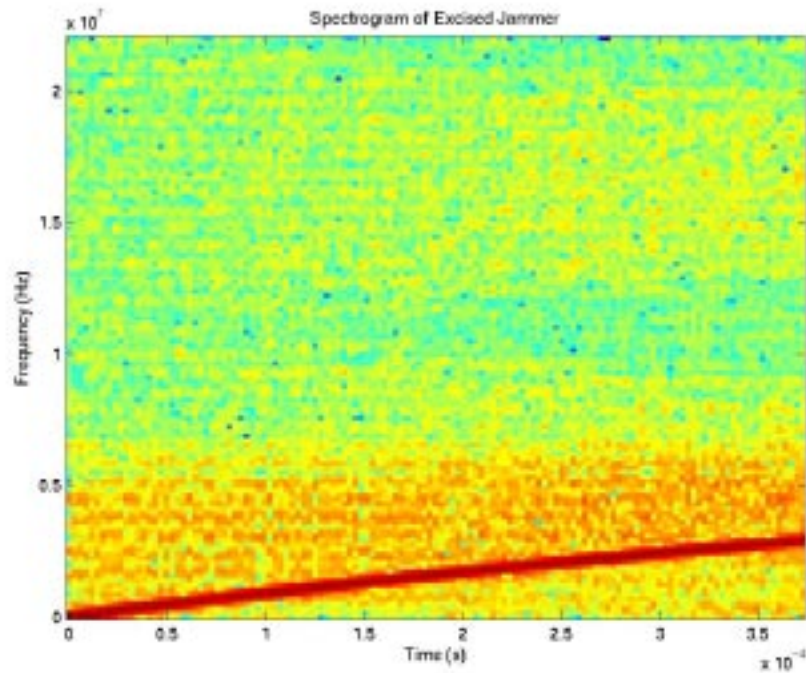
(b)

Figure 6.10. (a) Spectrogram of the jammer that was added. The frequency can be seen to parabolically sweep within the time range. (b) Spectrogram of the jammer estimate that was subtracted. It can be seen that the jammer was successfully tracked, although signal content was also subtracted out as indicated by the light red areas surrounding the jammer frequency track.

106

Although these distortions were observed, caution must be exercised, however, in concluding that the filter is ineffective since a direct relation of the distortions to its effects on the receiver's bit error rates (BER) cannot be drawn based on the observations alone. As will be seen in Chapter 9, the BER performances of the LMS filters show that they are effective in certain situations and actually provide some improvement to the bit error rates.

On another note, the parabolic curve in figure 6.10b shows that most of the estimate's energy was along the frequency of the jammer, suggesting that the filter successfully tracked the interference. Furthermore, the plot shows that the filter converged well before the 56 µs time constraint, since the red curve indicating the jammer's frequency is seen immediately.

6.6.   Summary

The Least Mean Square (LMS) adaptive filter iteratively minimizes the mean-square error between the input and its estimate by continuously updating its filter taps using the Windrow-Hoff algorithm. In order for the jammer to be interpolated from the 802.11b signal, the parameter D was introduced into the filter, which effectively allowed it to view a downsampled autocorrelation function of the input, such that the signal is seen as noise while the jammer maintains its sinusoidal autocorrelation function. Although the filter converges to the jammer before the 56 µs time limit, part of the signal is subtracted along with the estimate of the jammer in this particular implementation.

Chapter 7: Adaptive Lattice IIR Notch Filtering

7.1. Introduction

Adaptive notch filters (ANF) are particularly suitable for filtering out microwave oven interference due to their ability to track and filter out sinusoids with changing frequencies in the presence of background noise. The notch filter explored in this chapter implements a lattice structure and consequently has an infinite impulse response (IIR). This lattice structure was chosen for its tracking ability which is highly resistant to background noise, as opposed to the alternative *direct form* implementation. Figure 7.1 shows a block diagram of the filter [22].
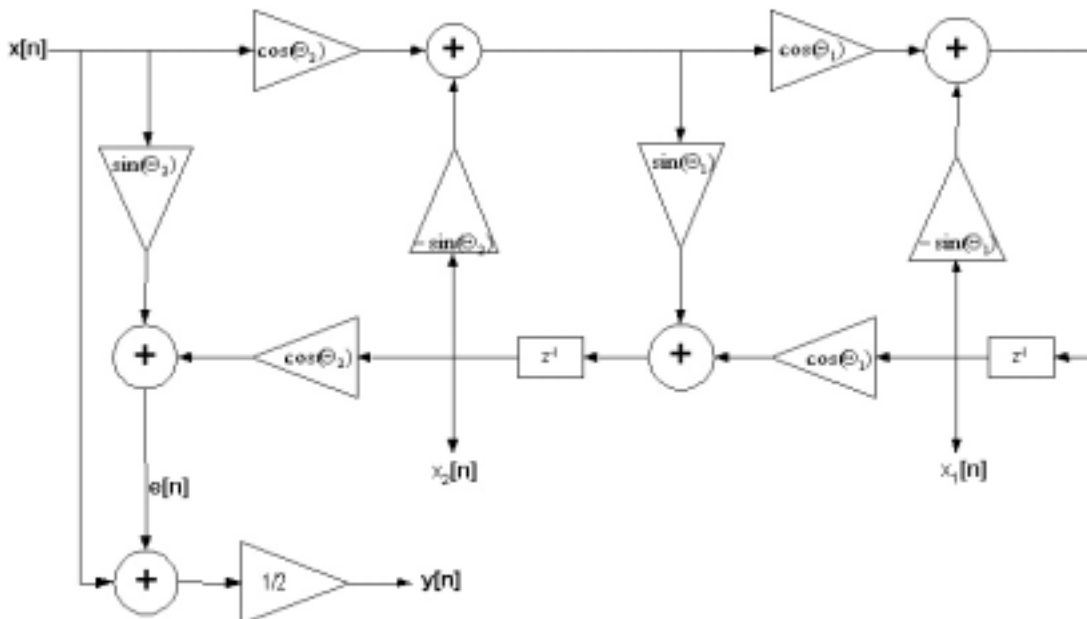
Figure 7.1. The Adaptive IIR Lattice Notch Filter structure, with input x[n] and output y[n]. The regressor signals $x_1[n]$ and $x_2[n]$ are state variables of the filter.

The filter is defined by two parameters, Θ_1 and Θ_2, which define the filter's notch frequency ω_o, and 3-dB bandwidth, B, respectively. In order for the notch frequency ω_o to converge to the input sinusoid's frequency, Θ_1 is updated at every time instant using an appropriate updating algorithm.
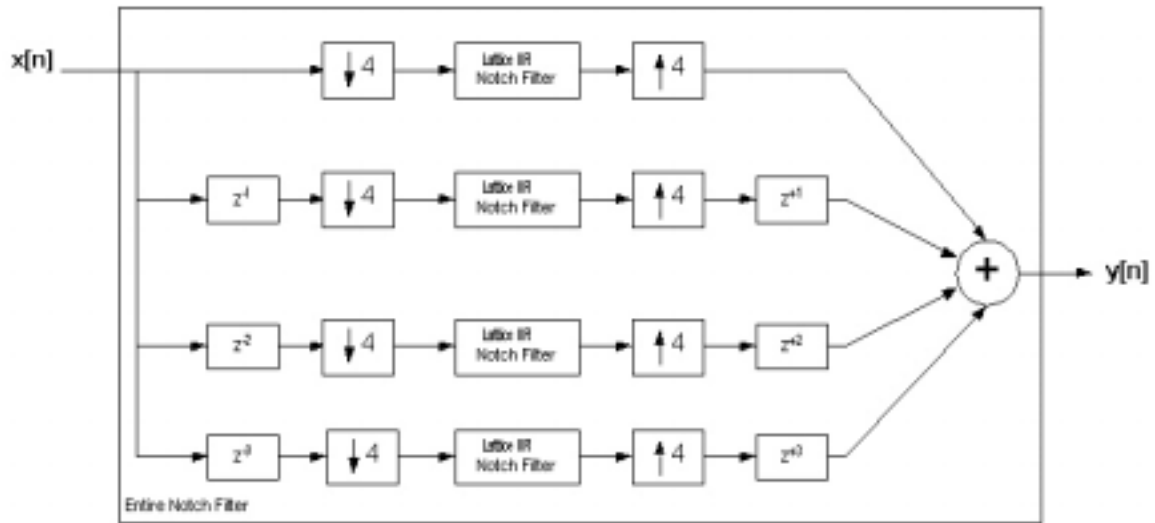
Figure 7.2. A polyphase decomposition is performed on the input signal for the microwave oven interference. Each polyphase signal is filtered with the adaptive notch filter.

Similar to the problem with the LMS filter in chapter 6, in order for the filter to effectively track and notch out the interference in the presence an 802.11b signal and noise, it needs to treat the 802.11b signal as background noise. In order to accomplish this, a polyphase decomposition needs to be performed on the input. This reduces the correlation between 802.11b signal values, making it appear noise-like, while preserving the properties of the microwave oven jammer. Each polyphase signal is passed through the filter and combined with the other outputs. Figure 7.2 illustrates the entire filtering process.

7.2. Integration with Receiver

The adaptive notch filter is integrated with the receiver at the same positions as the TDF and LMS Adaptive Filters: right in between the Analog-to-Digital Converters (ADCs) and the SQRRC Matched Filters in the receiver. Figure 7.3 shows the configuration.
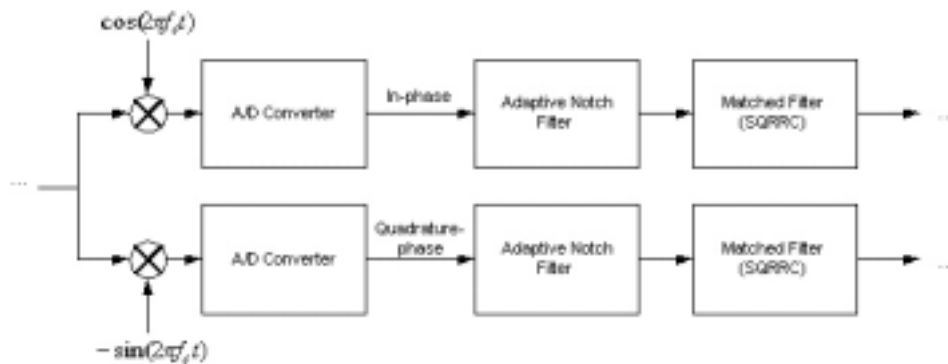
Figure 7.3. The Adaptive Notch Filter is placed in between the ADC and the Matched Filter in each receiver chain.

7.3. Principles of Operation

An analysis on the operation of the filter can be broken down into two parts: the polyphase decomposition, where the signal is divided into several downsampled components and the actual filtering operation performed by the lattice notch filter.

7.3.1. Polyphase Decomposition

The lattice IIR notch filter is designed to take in a sinusoidal input embedded in white noise. That is, it effectively filters inputs of the form:

$$x[n] = p_1 \sin[\omega_1 n + \phi] + \zeta[n] \qquad (7.1)$$

where $\zeta[n]$ is white noise. The microwave oven jammer $j[n]$ can be written in the form

$j[n] = p_1 \sin(\omega_1 n + \phi)$. Although the jammer frequency and amplitude vary with time,

they will be referred to hereon as ω_1 and p_1 for simplicity.

As previously defined, however, the received signal $r[n]$ can be expressed as

$r[n] = s[n] + j[n] + \zeta[n]$; it is a combination of the sinusoidal jammer $j[n]$, white

noise $\zeta[n]$, and the 802.11b signal $s[n]$. Because the signal $s[n]$ is not white, but rather

has an autocorrelation function resembling that of the SQRRC filter, it affects the ability

of the notch filter to determine and track the jammer frequency. In order to transform the

input into the form in equation 7.1 that the notch filter can work with, a polyphase

decomposition is performed on $r[n]$.

A 4-level decomposition was chosen for the signal for the same reason the

parameter D was set to 4 for the LMS filter (see section 6.5). Before modulation, the

symbols coming out of the CCK encoder and spreader are clocked at 11 MHz. The

symbol streams are then upsampled by 4 when pulse-shaped onto the SQRRC waveform

to produce a 44 MHz waveform. Since the Barker/CCK codes have low correlation

between adjacent values, waveform values 4 time samples apart consequently have low-

correlation.
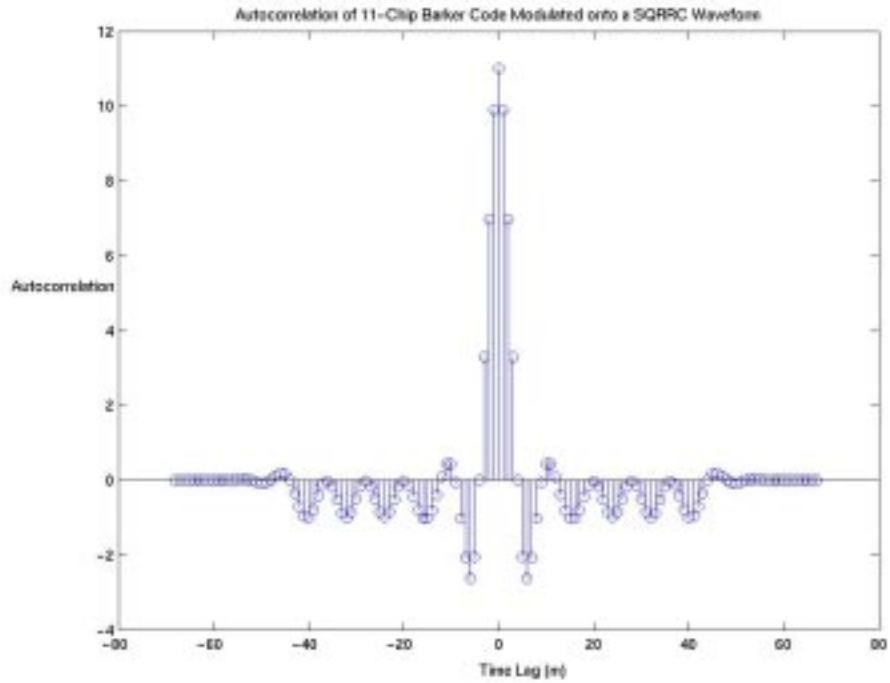
(a)
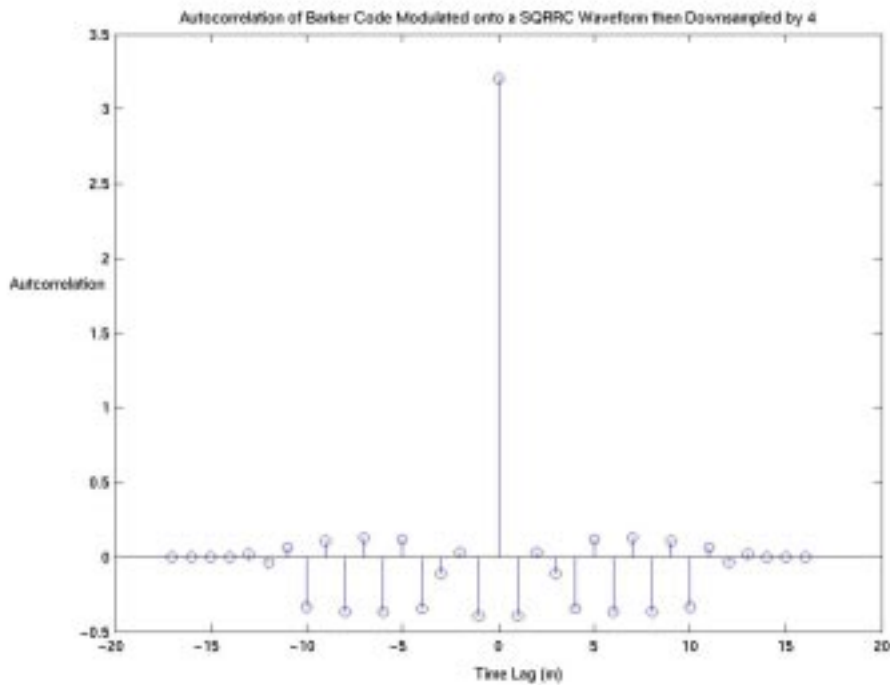
(b)

Figure 7.4. (a) Autocorrelation function of the Barker Code passed through the SQRRC Filter. (b) The autocorrelation function of the signal in (a) downsampled by 4 can be approximated as an impulse.

112

Let $s[n]$ be a transmitted waveform resulting from the modulation of one bit.

Figure 7.4a shows the autocorrelation function of $s[n]$ before decomposition, while

figure 7.4b shows the resulting function of each of its polyphase signals. The latter can

roughly be approximated as an impulse, indicating that the polyphase signals have been

whitened to a degree. On the other hand, performing the decomposition on $j[n]$ and $\zeta[n]$

still gives a sinusoid and white noise, respectively.

After performing the decomposition, $r[n]$ can be expressed as the sum of its

polyphase components $r_i[n]$ for $0 \leq i \leq 3$.

$$r[n] = \sum_{i=0}^{3} r[4n+i] = \sum_{i=0}^{3} r_i[n] \qquad (7.2)$$

Each signal $r_i[n]$ can be expressed as:

$$r_i[n] = p_1 \sin(\omega_1(4n+i)+\phi) + s[4n+i] + \zeta[4n+i] \qquad (7.3)$$

The downsampled background noise can be rewritten as a new noise term,

$\zeta_{1i}[n] \equiv \zeta[4n+i]$, while each polyphase signal $s[4n+i]$ can be approximated as a

second noise term $\zeta_{2i}[n]$.

$$\zeta_{2i}[n] \approx s[4n+i] \qquad (7.4)$$

Rewriting each $r_i[n]$ in terms of the new definitions results in:

$$r_i[n] = p_1 \sin(4\omega_1 n + \phi_i{}') + \zeta_{1i}[n] + \zeta_{2i}[n] \qquad (7.5)$$

where a new phase angle $\phi_i{}' \equiv i\omega_1 + \phi$ is defined. It is now clear that each $r_i[n]$ resulting

from the decomposition is of the form in equation 7.1, although with two noise terms

instead of one.[18] Filtering $r_i[n]$ with the lattice filter removes the polyphase jammer

signal, resulting in the output $y_i[n]$:

$$y_i[n] \approx \zeta_{1i}[n] + \zeta_{2i}[n] \qquad (7.6)$$

Adding up all the $y_i[n]$ signals yields the jammer free output:

$$y[n] = \sum_{i=0}^{3} y_i[n] = \sum_{i=0}^{3} s[4n+i] + \zeta_1[4n+i] = s[n] + \zeta[n] \qquad (7.7)$$

7.3.2. Lattice IIR Notch Filter Operation

The lattice notch filter in Figure 8.1a has a transfer function that can be written as

$H(z) = \dfrac{1}{2}[1 + V(z)]$, where $V(z)$ is a second-order, causal all-pass filter with the

following system function [22]:

---

[18] Combining them into a single noise term is a trivial operation.

$$V(z) = \frac{\sin\Theta_2 + \sin\Theta_1(1+\sin\Theta_2)z + z^2}{1 + \sin\Theta_1(1+\sin\Theta_2)z + \sin\Theta_2 z^2} \qquad (7.8)$$

Substituting equation 7.8 for the definition of $H(z)$ results in:

$$H(z) = \frac{1+\sin\Theta_2}{2} \frac{1+2\sin\Theta_1 z + z^2}{1+\sin\Theta_1(1+\sin\Theta_2)z + \sin\Theta_2 z^2} \qquad (7.9)$$

When $H(z)$ is evaluated at its notch frequency ω_o, the frequency response is characterized by:

$$H(e^{jw_o}) = 0 \text{ and } \phi(\omega_o) = \pi \qquad (7.10)$$

Using the latter result, the notch frequency ω_o and bandwidth B are found to be related to Θ_1 and Θ_2 by the following equations [22]:

$$\omega_o = \Theta_1 + \frac{\pi}{2} \qquad (7.11)$$

$$\sin\Theta_2 = \frac{1 - \tan(\frac{B}{2})}{1 + \tan(\frac{B}{2})} \qquad (7.12)$$

115

7.3.3. Updating Algorithm

In the actual filter implementation, the parameter Θ_2 was fixed at the value

0.455π to give the filter a bandwidth of $B = \dfrac{\pi}{5}$, while Θ_1 was initially set to $\dfrac{\pi}{2}$. These

values were adapted from the filter implementation in [22]. The parameter Θ_1 is

iteratively updated in order for ω_o to converge to and track ω_1 as it sweeps. The

recursive algorithm uses the equation:

$$\Theta_1[n+1] = \Theta_1[n] - \mu[n]y[n]x_1[n] \qquad (7.13)$$

where $\mu[n]$ is a time-varying scaling parameter, $y[n]$ the filter output, and $x_1[n]$ a

regressor signal that can be picked off from the node marked in figure 7.1. The equivalent

frequency response $G(z)$ of the filter portion that produces $x_1[n]$ from the input $x[n]$ can

be expressed as:

$$G(z) = \frac{z\cos\Theta_1\cos\Theta_2}{1 + \sin\Theta_1(1 + \sin\Theta_2)z + \sin\Theta_2 z^2} \qquad (7.14)$$

In order to see that equation 7.13 leads to convergence, it can first be noted that

the behavior of $\Theta_1[n]$ over time closely follows the behavior of the following differential

equation:

$$\frac{d\Theta_1}{dt} = -E\{y[n]x_1[n]\} = -\frac{1}{2\pi} \int_{-\pi}^{\pi} S_{jj}(e^{j\Omega_1}) H(e^{j\Omega_1}) G(e^{-j\Omega_1}) d\Omega \qquad (7.15)$$

where $S_{jj}(e^{j\omega_1})$ is the power spectral density (PSD) of the polyphase signal $j[4n+i]$.

Assuming that $r[n]$ is divided into polyphase signals in the form of $x[n]$ in equation 7.1, evaluating the integral in $\frac{d\Theta_1}{dt}$ results in:

$$\frac{d\Theta_1}{dt} = -p_1^2 \frac{\cos\Theta_1 \cos\Theta_2}{\left|F(e^{j\Omega_1})\right|^2} (\cos\omega_1 + \sin\Theta_1) \qquad (7.16)$$

where $F(z) = 1 + \sin\Theta_1(1 + \sin\Theta_2)z + \sin\Theta_2 z^2$.

As shown in [22], $\frac{d\Theta_1}{dt}$ in equation 7.16 is globally asymptotically stable to a stationary point where $\sin\Theta_1 = \cos\omega_1$ as $t \to \infty$. This can also be restated as:

$$\lim_{t\to\infty} \Theta_1 = \omega_1 + \frac{\pi}{2} \qquad (7.17)$$

Combining this with the result in equation 7.11, it can be shown that the notch frequency converges to the jammer frequency:

$$\lim_{t\to\infty} \omega_o = \omega_1 \qquad (7.18)$$

7.3.4. Scaling Parameter

The scaling parameter $\mu[n]$ affects the speed of convergence of the filter. In order to achieve faster convergence, $\mu[n]$ was defined as:

$$\mu[n] = \frac{\alpha}{\sum\limits_{i=1}^{N} r_i[n-i]} \qquad (7.19)$$

where the values of $\alpha = 0.01$, $N = 10$ were used. The denominator of $\mu[n]$ is the total energy of the past N samples; the scaling parameter $\mu[n]$ acts as a normalization factor in the updating algorithm in equation 7.13, similar to the definition of $\mu[n]$ in section 6.3.1. High energy samples in the filter result in a small $\mu[n]$ that stabilizes fluctuations in the notch frequency and vice versa.

7.4. Convergence Time

Similar to the LMS Filter discussed in Chapter 6, the adaptive notch filter processes its input in real-time and must converge to the jammer frequency before the 56 µs time constraint set by the duration of the SYNC field of the short PLCP packet.

The same sinusoidal input tests were carried out on the notch filter as those for the LMS filter. The first test consisted of a frequency-sweeping input, sweeping between 0 and 3 MHz over its duration. The results are shown in figure 7.5: a plot of the filter's notch frequency parameter over time is shown in figure 7.5a, while the spectrogram of the actual input is displayed in figure 7.5b. It can be seen that the notch frequency
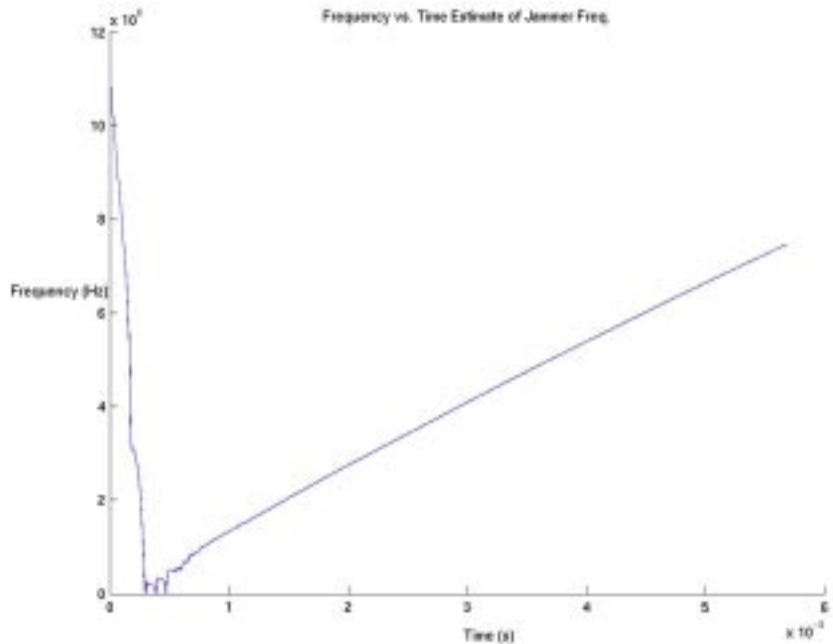
(a)
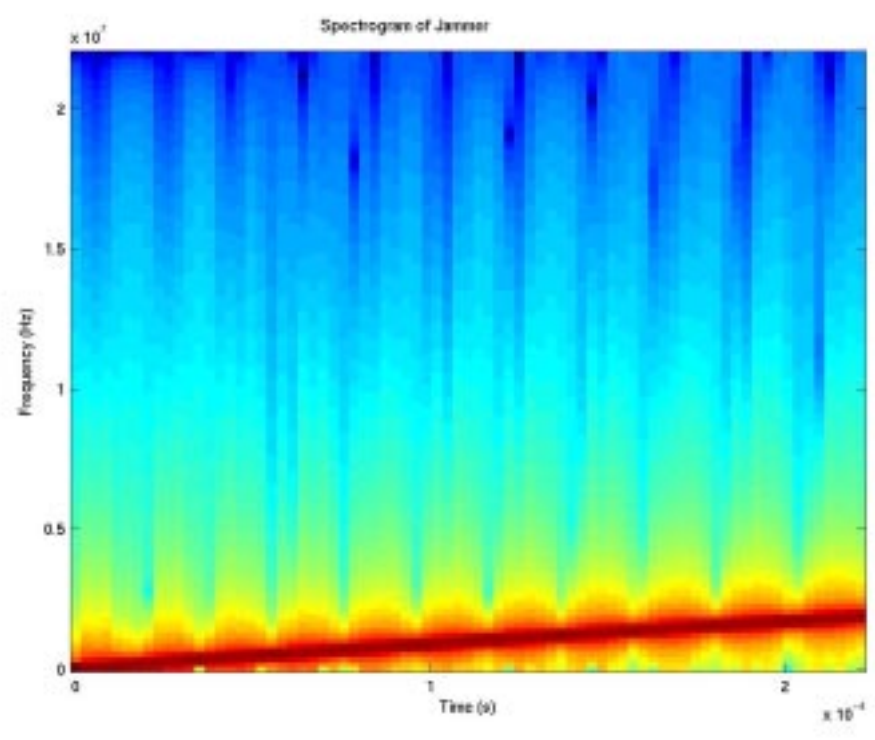
(b)

Figure 7.5. (a) The filter notch frequency vs. time. (b) The sinusoid frequency vs. time. It can be seen that the filter converges to the frequency of the input sinusoid within 20 μs and effectively tracks it in time.

119
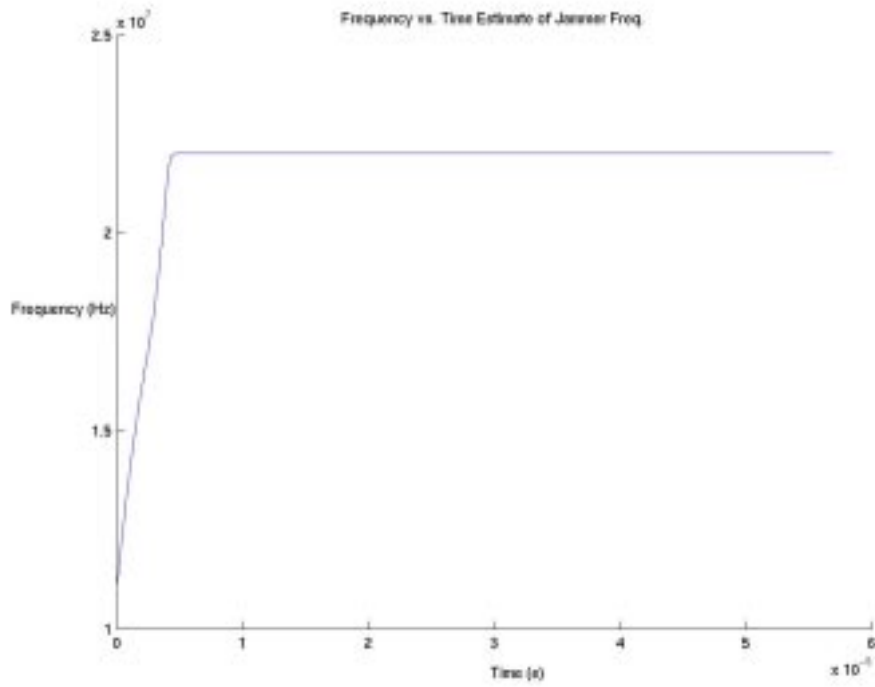
(a)
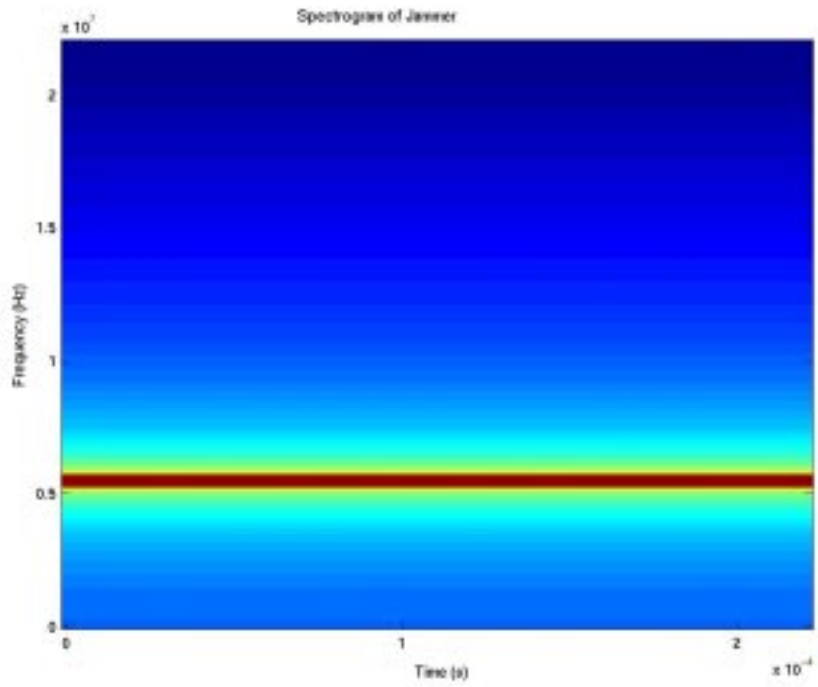
(b)

Figure 7.6. (a) The filter's notch frequency vs. time. (b) The sinusoid frequency vs. time. Similar to figure 7.5, the filter can be seen to converge to the input sinusoid's frequency within 20 μs.

converges to the input frequency after approximately 20 μs, well before the length of the SYNC field.

The second convergence test was on a single-tone sinusoid input at 5.5 MHz. Figure 7.6a shows the filter's notch frequency over time while figure 7.6b shows the spectrogram of the input. Just as in the first case, the notch frequency can be seen to quickly rise and settle at the jammer frequency in approximately 20 μs.

Similarly from these results, the filter can be expected to converge to the jammer's frequency before the 56 μs time constraint.

7.5.   Interference Mitigation Example

The adaptive notch filter was simulated on a packet with a microwave oven jammer beginning its frequency-sweeping phase. The packet was modulated at 11 Mbps, used a Long PLCP format and had a payload of 1024 randomly generated bits. The signal-to-jammer (SJR) ratio was set to –21 dB, giving the jammer superior energy over the 802.11b signal, while the signal-to-noise (SNR) was set at 8.25 dB. Figure 7.7a plots the magnitude of the signal with AWGN before encountering the microwave oven jammer.

The magnitude response of the signal with the microwave oven interference is shown in figure 7.7b. Again, the interference is represented by the white 'blob' that rises by 21 dB over the signal, consistent with the SJR value that was picked for the simulation.

(a)
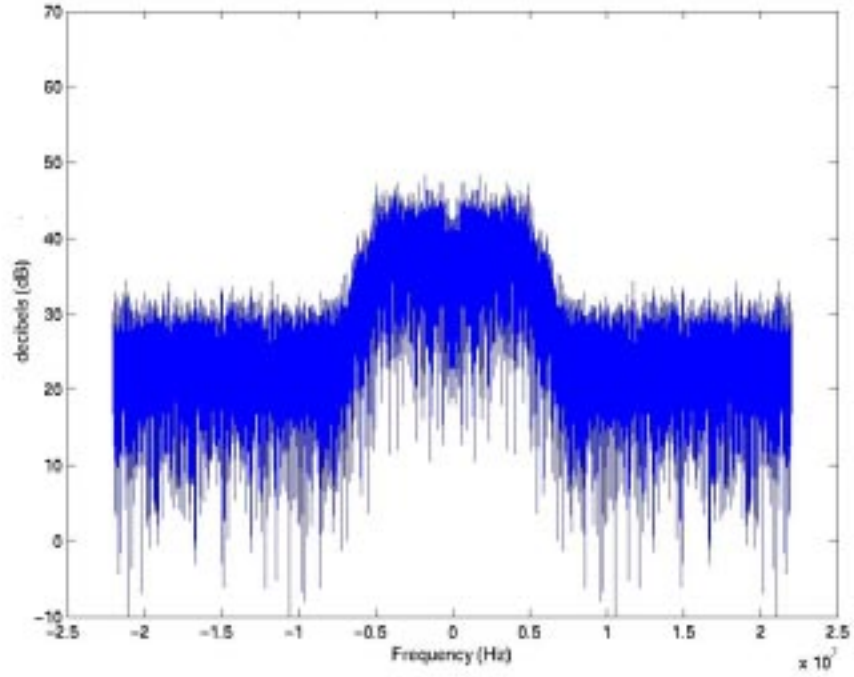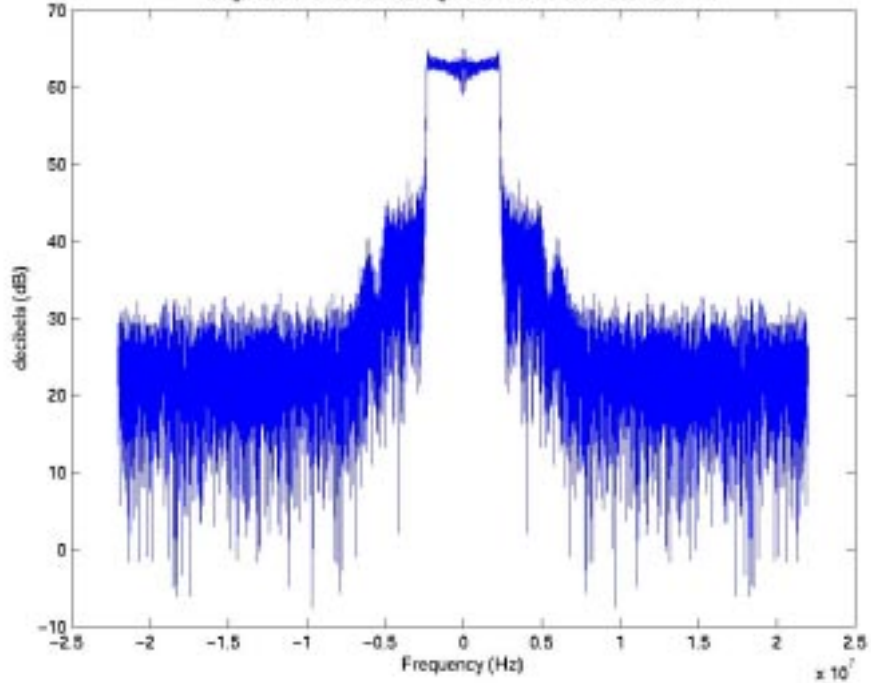
(b)

Figure 7.7. (a) The 802.11b signal with Gaussian noise. (b) The signal with the addition of microwave oven interference, seen as the white 'blob' towering over the signal at 65 dB.

The magnitude plot of the filter output is shown in figure 7.8. Aside from a sharp spike that stands out at 0 Hz. the rest of the magnitude plot resembles that of the original 802.11b signal, indicating that the filter successfully removed the jammer at those frequencies.
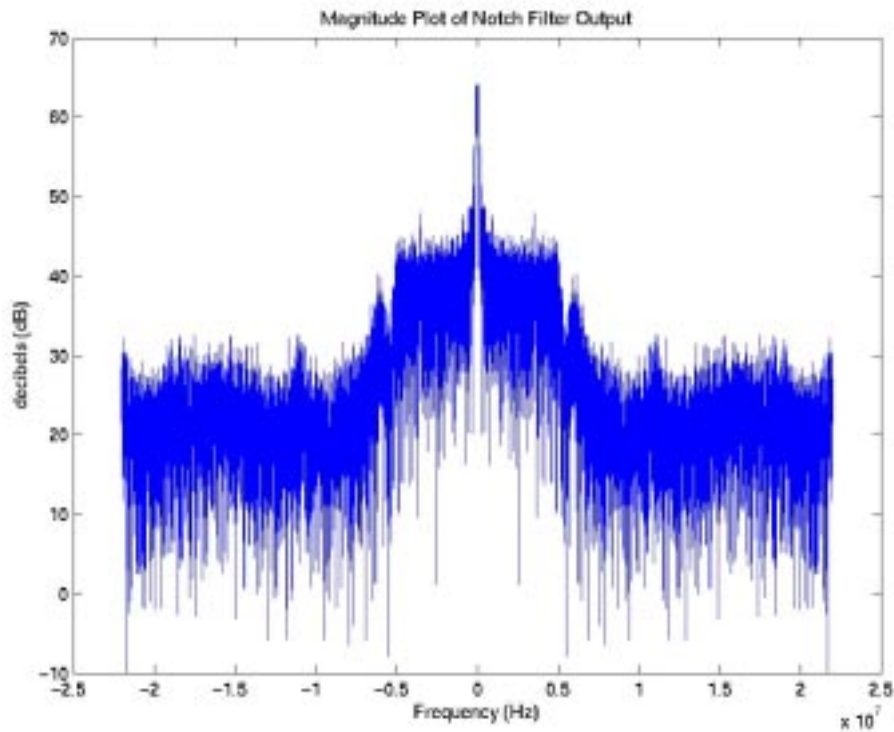
Figure 7.8. The magnitude plot of the filter output shows that the filter successfully removed the jammer with the exception of the small spike left at 0 Hz. The spike was left since it occurred at the time during which the filter was still converging to the frequency of the jammer.

The spike seen above represents the portion of the jammer at the beginning of the packet, when it began to sweep in frequency starting at 0 Hz. This portion was not excised since the filter was still in the process of converging to the jammer frequency. This is verified by the plot of the notch frequency over time, shown in figure 7.9. The notch frequency starts at 22 MHz and only converges to the jammer frequency after

123

10 µs; the jammer is not removed before that time. In any case, this is not a problem at all

since the time constraint is 56 µs.
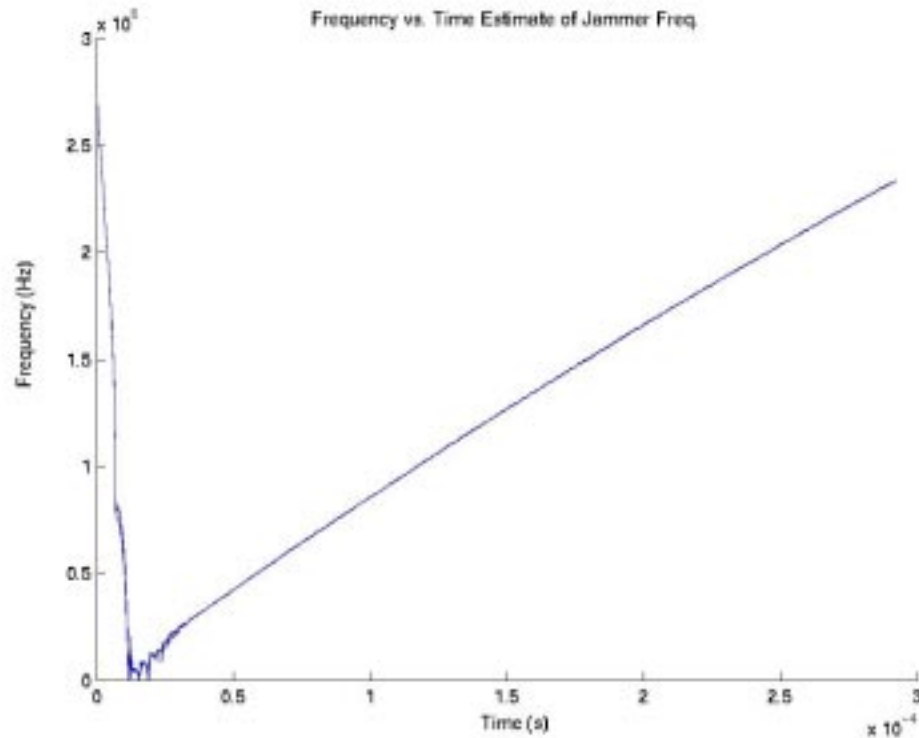
Figure 7.9. The notch frequency of the filter is plotted versus time. The filter converges to the input frequency within 20 µs and tracks it successfully thereafter.

In order to confirm that the notch frequency does track the jammer frequency, the

spectrogram of the added jammer is plotted in figure 7.10. Comparing the two plots, it

can be concluded that the filter successfully notches out the microwave oven jammer

even as it sweeps in frequency. The filter notch frequency converges to the frequency of

the jammer within 20 µs and successfully tracks it thereafter.

124
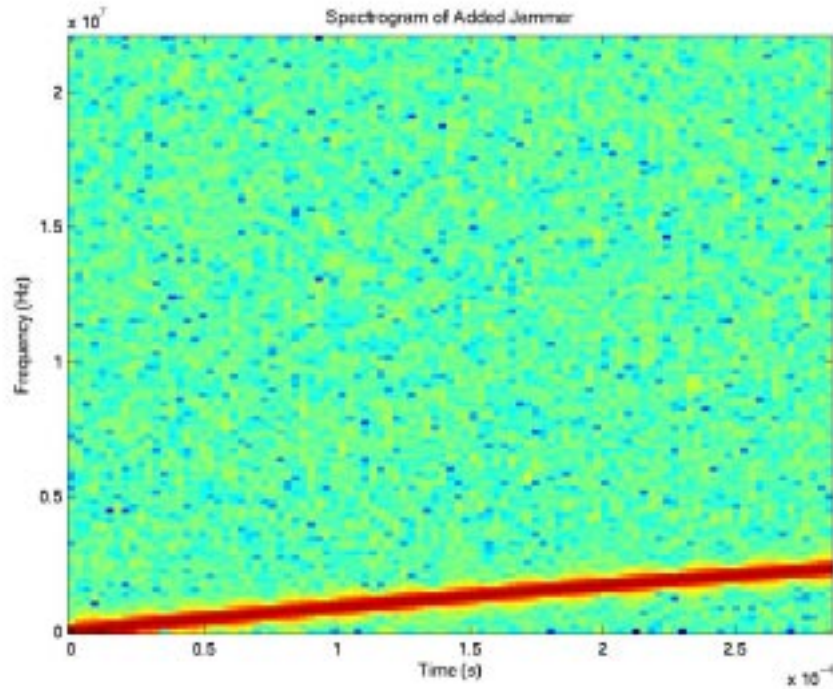
Figure 7.10 The spectrogram of the microwave oven interference that was encountered by the packet.

7.6. Summary

The adaptive notch filter used has a lattice structure and operates on sinusoidal inputs in the presence of background noise. In order to work on an 802.11b signal, a polyphase decomposition is performed on the received signal to reduce the correlation between signal values and make it appear as noise to the notch filter. Each polyphase signal is processed separately and the filtered outputs are combined accordingly. The filter was observed to successfully converge to the input frequency and track its movement in time.

Chapter 8: Simulation Results & Analyses

8.1. Introduction

The filters were integrated with the receiver model and tested for their efficacy in mitigating microwave oven interference. In order to provide a basis for comparison for their performance, control simulations on the receiver without any interference mitigation were first run for two types of inputs: packets with AWGN, and packets with both the microwave oven jammer and AWGN. The order of the simulations presented in this chapter is as follows:

1) Receiver Performance with AWGN – these simulations characterized the performance of the receiver as a function of the relative strength of the signal to the AWGN encountered in the channel. The relative strength is measured by the Signal-to-Noise Ratio (SNR), which is defined as:

$$SNR = 10\log_{10}\left(\frac{signal\_energy}{noise\_energy}\right) = 10\log_{10}\left(\frac{\sum_{n=-\infty}^{\infty} s^2[n]}{\sum_{n=-\infty}^{\infty} \zeta^2[n]}\right) \qquad (8.1)$$

2) Receiver Performance with Microwave Oven Interference and AWGN – these simulations characterized the actual performance degradation brought about by the microwave oven interference and provided a basis of comparison for the interference mitigation techniques.

126

3) Performance of the Receiver integrated with Interference Mitigating Filters – these were the core simulations of the thesis, which characterized the performance of the filters as a function of the relative strength of the microwave oven jammer encountered in the channel. The relative strength of the signal to the jammer is defined as the Signal-to-Jammer Ratio (SJR) and is defined as:

$$SJR = 10\log_{10}\left(\frac{signal\_energy}{jammer\_energy}\right) = 10\log_{10}\left(\frac{\sum_{n=-\infty}^{\infty} s^2[n]}{\sum_{n=-\infty}^{\infty} j^2[n]}\right) \quad (8.2)$$

Several performance measures were used to characterize the receiver for the simulations:

a.) Bit Error Rate (BER) – Each transmitted packet has a payload known to both the transmitter and receiver. Assuming that the CRC check for a packet is valid, the receiver continues on to demodulate the payload bits. The reproduced payload can be compared to the original and the number of bit errors can be counted. The BER is defined as:

$$BER = \frac{total\_bit\_errors}{total\_payload\_bits} \quad (8.3)$$

127

Consequently, dropped packets are not counted in the BER. The BER is determined for each particular SJR value; a BER curve is produced for each simulation.

b.) Frame Error Rate (FER) – The frame error rate is defined as the percentage of packets dropped at the receiver out of the total number of transmitted packets. The FER is defined as:

$$FER = 100\left(\frac{total\_dropped\_packets}{total\_transmitted\_packets} \right) \qquad (8.4)$$

A third measure was introduced for the filter simulations:

c.) Signal-to-Jammer Ratio (SJR) Improvement – this is defined as the average improvement of the signal power over the microwave oven jammer as a result of interference mitigation. The SJR improvement is defined as:

$$SJR\_improvement = 10\log_{10}\left(\frac{\sum_{n=-\infty}^{\infty} s^2[n]}{\sum_{n=-\infty}^{\infty} (y[n]-s[n])^2} \right) - Original\_SJR \qquad (8.5)$$

where $y[n]$ is the output of the filter and $(y[n]-s[n])$ is the noise remaining at the output.

The simulations covered all four data rates. Each packet used a long PLCP format and had payload of 1024 bytes, where each bit was randomly generated. A simulation consisted of generating packets, transmitting, adding interference and/or noise, and demodulating until a total of 10^4 bit errors were found. This gives each measured BER a margin of error of 1% [23]. When a simulation terminated, the corresponding BER, FER, and SJR improvement measures were computed and recorded. Furthermore, a limit on the number of transmitted packets was set for high SJR values or other cases where the BER was expected to be low or zero, since the simulations could run indefinitely.[19]

8.2. Receiver Performance with AWGN

The first step towards characterizing the receiver involved testing its performance in the presence of AWGN over a range of SNR values. The simulations swept the SNR from –10 dB up to 5 dB since this range provided a full excursion of the FER from 100% to 0%, and produced the relevant range of BER values. The results are shown in figure 8.1: Figure 8.1a displays plots of the BER vs. SNR curves, while figure 8.1b shows a plot of the FER vs. SNR curve resulting from the tests. There is only one FER curve, which is an average of the FER curves for the four data rates, since the packets for all the data rates use the long PLCP format and should produce the same FER values.

From simple inspection, the BER curves clearly follow a *waterfall* model, where the values plateau at their maximum for low SNR values and drop fast at high SNR values. As seen in the results, a lower data rate will always have a curve to the left of a higher data rate; it generally has a lower BER for any SNR value. This is expected, since

---

[19] The limit was determined on a case-to-case basis, depending on previous simulation results. The new limits may have increased the margin of error, which is a limitation of the simulations.
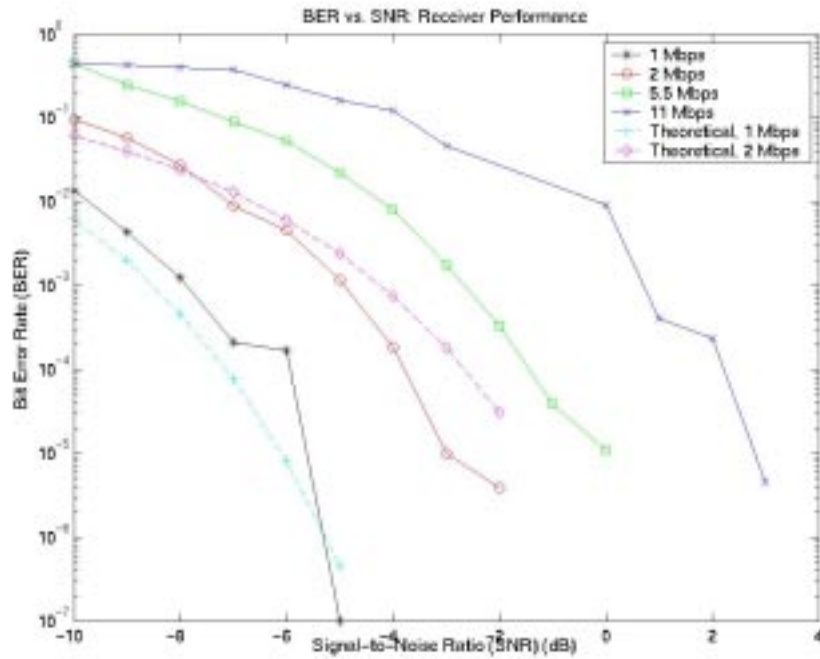
(a)
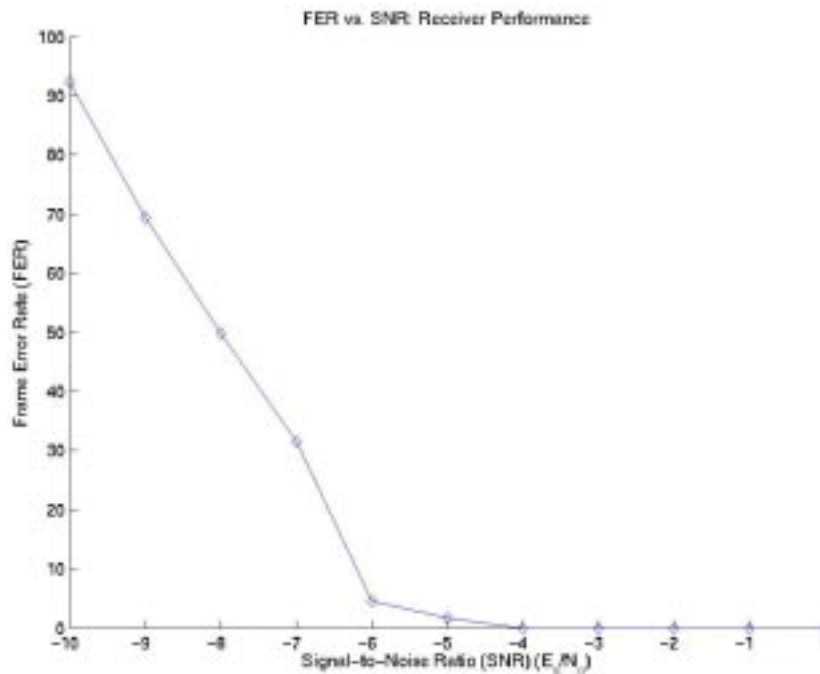
(b)

Figure 8.1. (a) The BER vs. SNR curves for each of the four data rates. (b) FER vs. SNR curves. The BER curves show a waterfall model, which are typical for the case of an AWGN channel. The theoretical BER curves for 1 and 2 Mbps are superimposed. The FER linearly rises with a decrease in the input SJR.

130

a lower data rate offers higher processing gain and thus more resistance to noise and interference (see section 2.6).

Based on the results, the receiver performs closely to the theoretical BER performance, based on the superimposed theoretical curves for 1 and 2 Mbps. In order to achieve a BER of 10^{-5}, which is a typical desired level of performance, SNR values of –6, -3, 0, and 3 dB are needed for 1, 2, 5.5 and 11 Mbps, respectively.[20] The FER curve on the other hand linearly sweeps up from 0 to 100 from –4 dB to –10 dB.

8.3.   Receiver Performance with Microwave Oven Interference

After characterizing the basic performance of the receiver, microwave oven interference was added to the transmitted packets. Each packet encountered the jammer at a randomly chosen point in its active phase. As expected, the interference caused degradation in performance, which manifested itself clearly as higher FER and BER values. For each simulation, the SNR was fixed at the minimum required value for the receiver to have a BER of 10^{-5} for that particular data rate. To get a full excursion of the BER and FER curves, the SJR was swept from –30 dB to 10 dB. Figure 8.2a shows the resulting bit error rates for the four data rates due to the interference.

Similar to the BER curves in figure 8.1, the results show a waterfall model. Again, BER curves for a lower data rate is to the left of a BER curve of a higher data rate, indicating that it will typically have less number of bit errors for a particular signal-to-jammer ratio. This property can again be attributed to the fact that lower data rates offer larger processing gains than higher data rates.

---

[20] The receiver model meets the specifications set in Mobilian's Receiver HLD document, which were 0, 3, 5, and 8.25 dB SNR (E_s/N_o) for a BER of 10^{-5}, for the data rates 1, 2, 5.5, and 11 Mbps respectively.
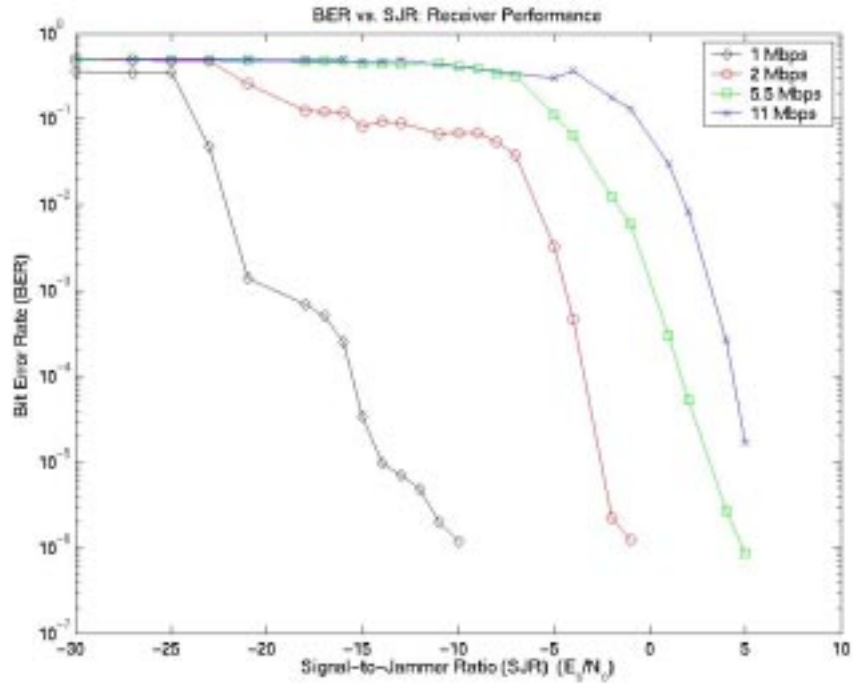
(a)
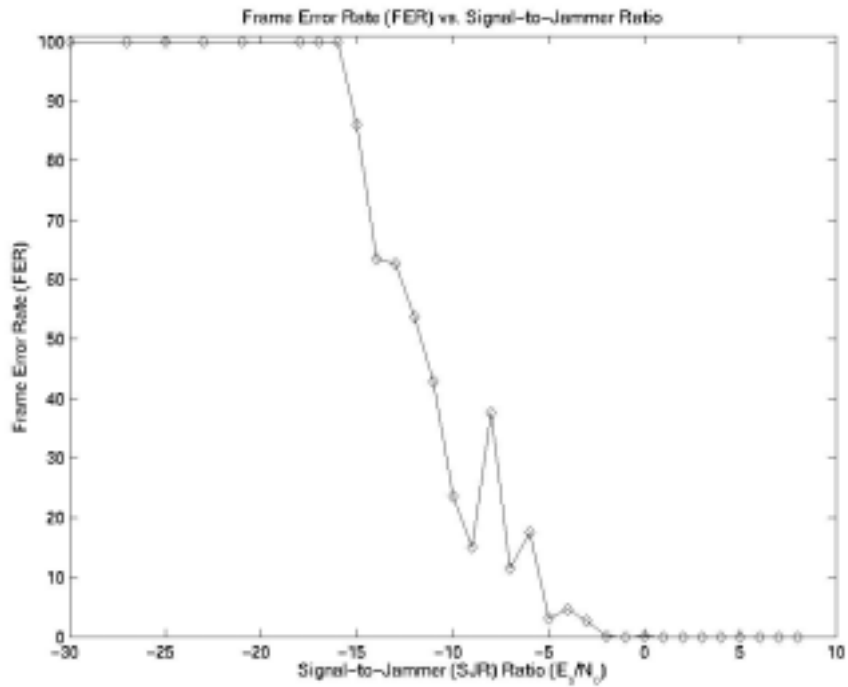
(b)

Figure 8.2. (a) The BER curves for the interference. (b) The FER curve for the interference. A similar waterfall model can be observed for the BER curves. The FER curve has some deviations (between an SJR of -10 and 5 dB), but can be generally observed to behave linearly as well.

132

Furthermore, a sharp contrast can be observed: the BER for 5.5 and 11 Mbps settle at the peak value for SJR values of -4 dB and -3 dB, respectively, while SJR values of -23 dB and -25 dB are needed to bring the BER to the same level for 2 and 1 Mbps, respectively. On the other hand, it can also be concluded that the microwave oven interference does not cause any visible degradation in the performance of the receiver for SJR values of -10, -1, 5 and 6 dB or higher for the data rates of 1, 2, 5.5 and 11 Mbps, respectively. That is, for the same SNR value, adding the jammer at those SJRs does not change the bit error rate.

The FER can be observed to hit 100% at an SJR of -16 dB. However, the BER for 1 Mbps is only at 2.5×10^{-4}, indicating that packets were most likely not dropped due to bit errors in the header, but rather because the receiver was not able to lock on to the proper timing synchronization. As a consequence, the SFD field of each packet could not be identified.

8.4. Interference Mitigation Filters - Simulation Results

For the final set of simulations, the three filters were integrated with the receiver and tested with the same set of SNR and SJR values as in section 8.2. As mentioned in section 8.1, the improvement in the SJR at the output of each filter was also measured as an additional performance criterion.. The results are presented in figures 8.3 thru 8.6 in the following pages, and are ordered by increasing data rate, starting at 1 Mbps. For each data rate, the BER curves for the filters are shown on top, followed by the measured SJR-improvement curves on the bottom.
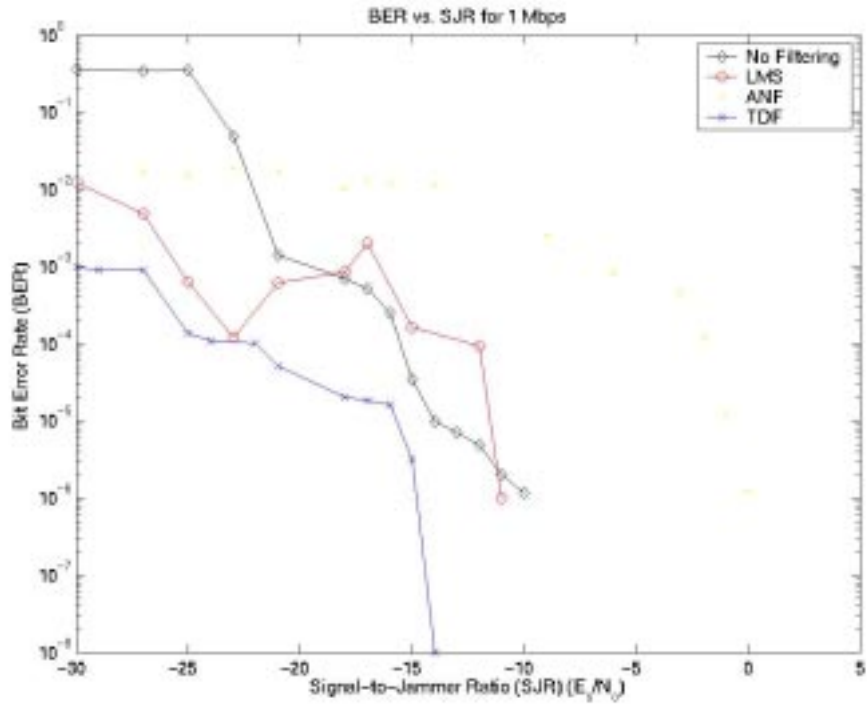
(a)
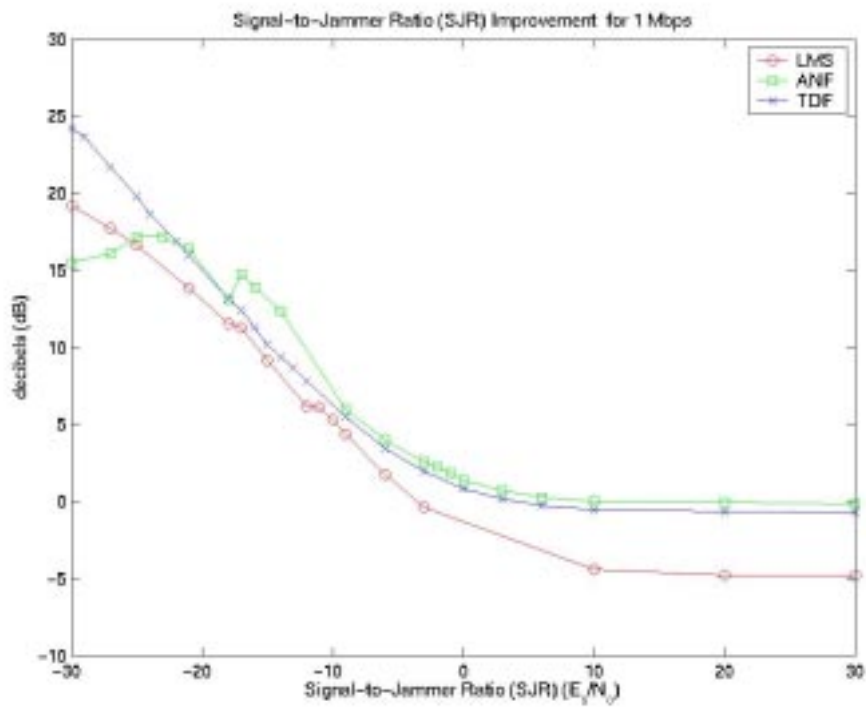
(b)

Figure 8.3. The corresponding BER curves (a) and SJR improvement curves (b) for 1 Mbps

134
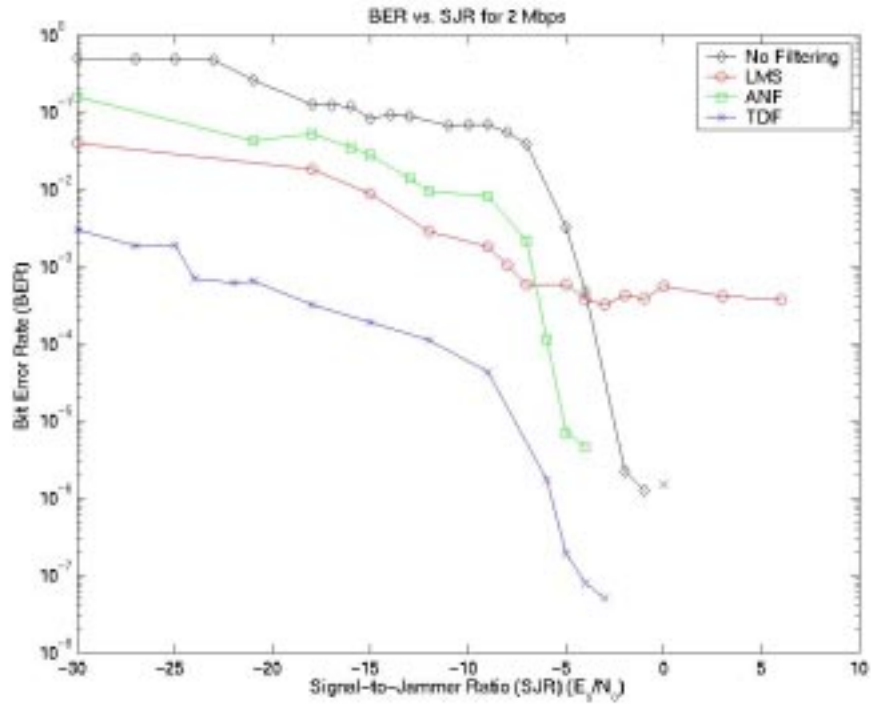
(a)
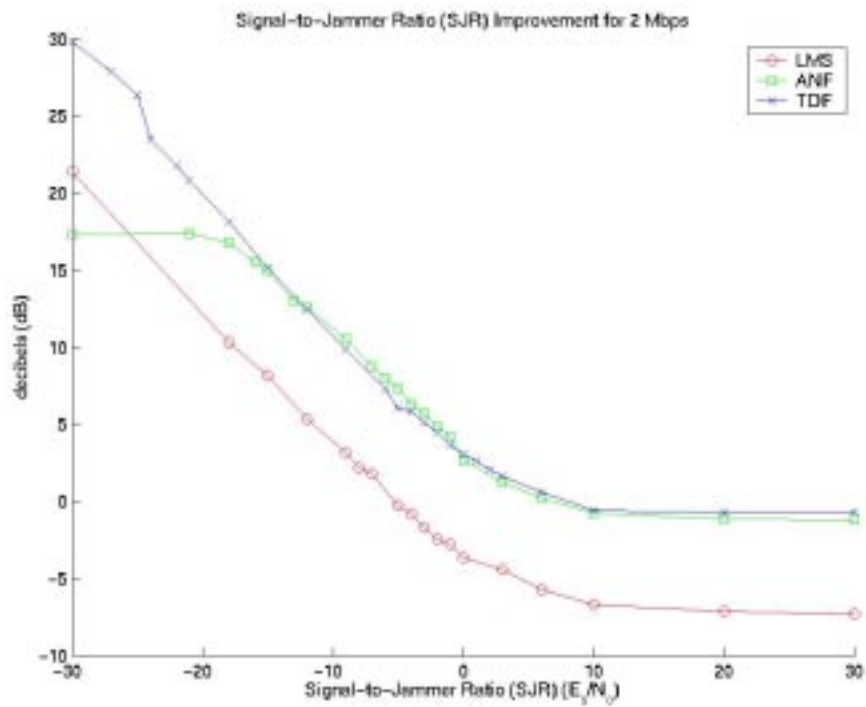
(b)

Figure 8.4. The corresponding BER curves (a) and SJR improvement curves (b) for 2 Mbps

135
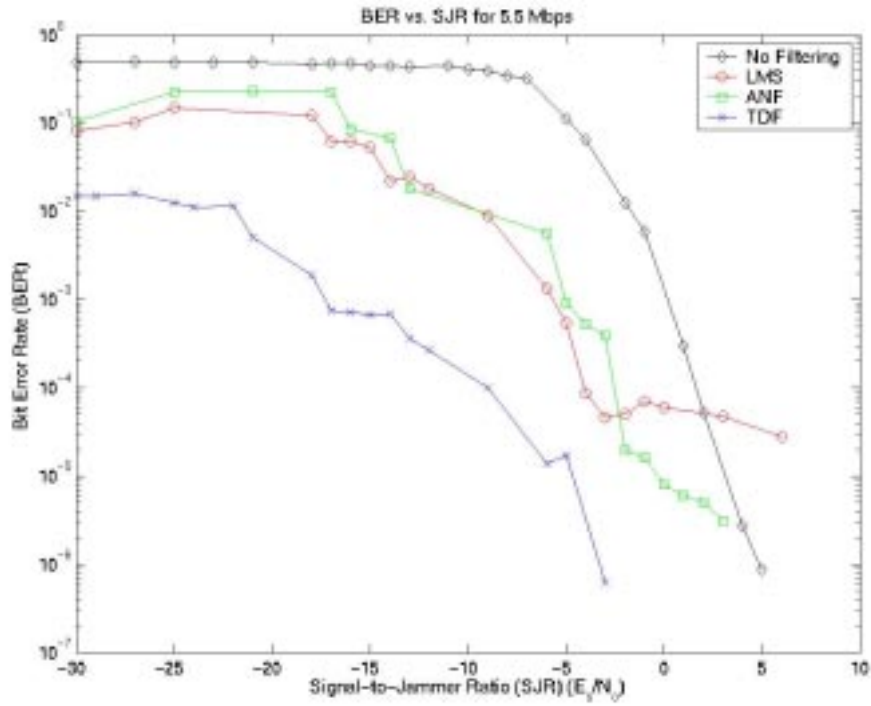
(a)
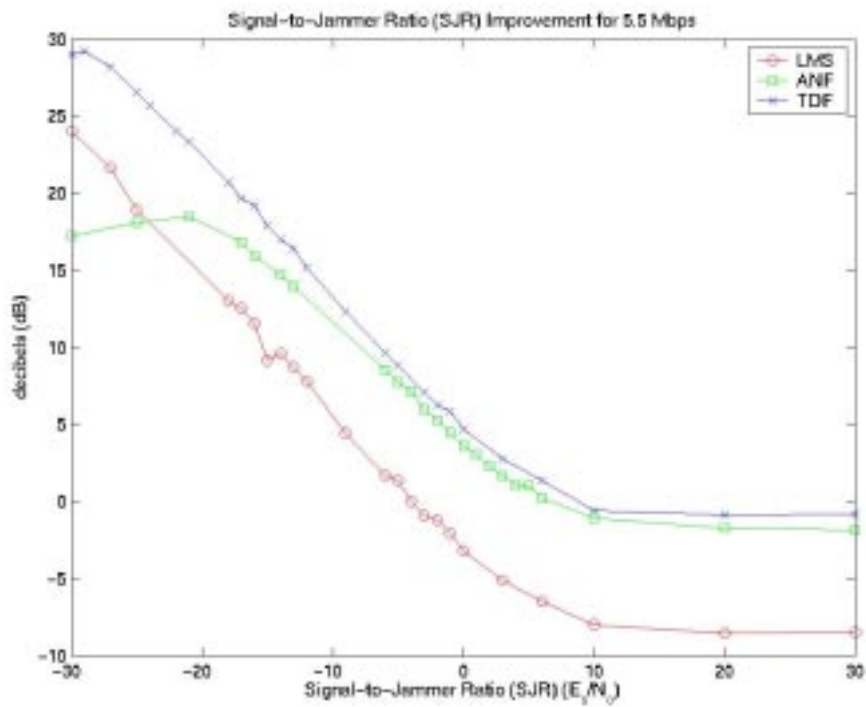
(b)

Figure 8.5. The corresponding BER curves and SJR improvement curves for 5.5 Mbps
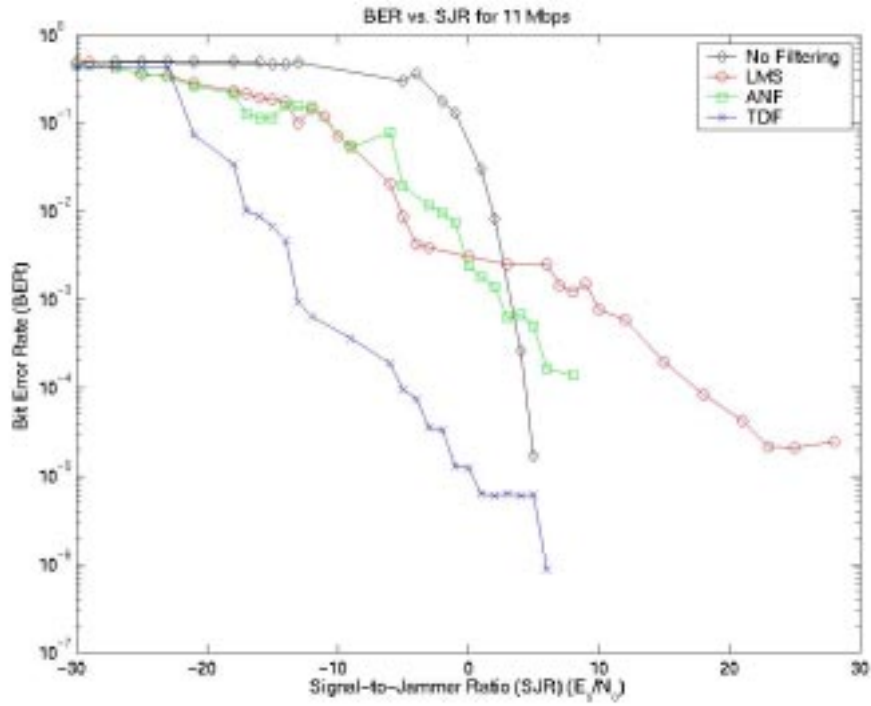
(a)
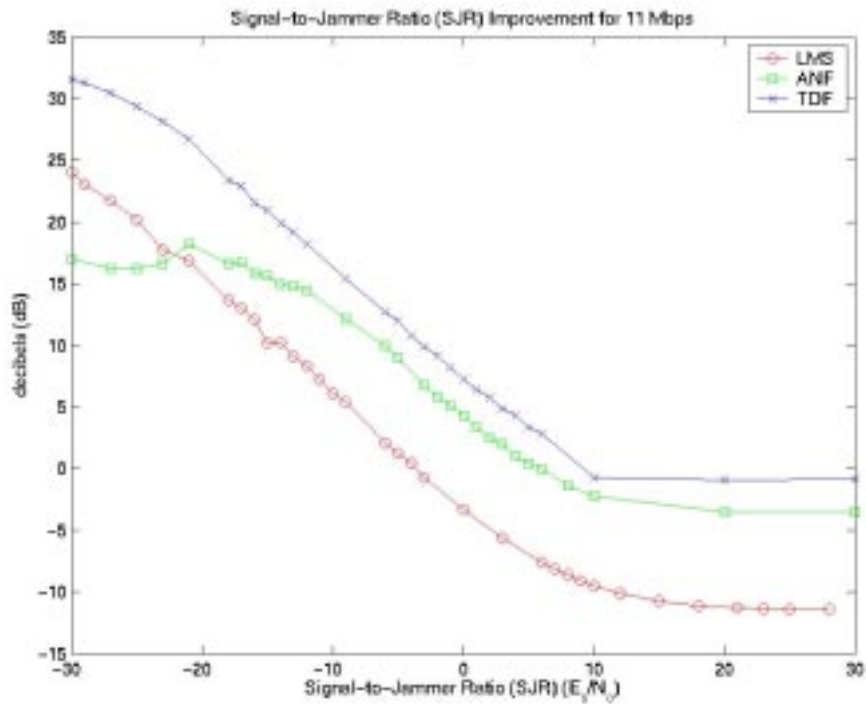
(b)

Figure 8.6. The corresponding BER curves (a) and SJR improvement curves (b) for 11 Mbps

137
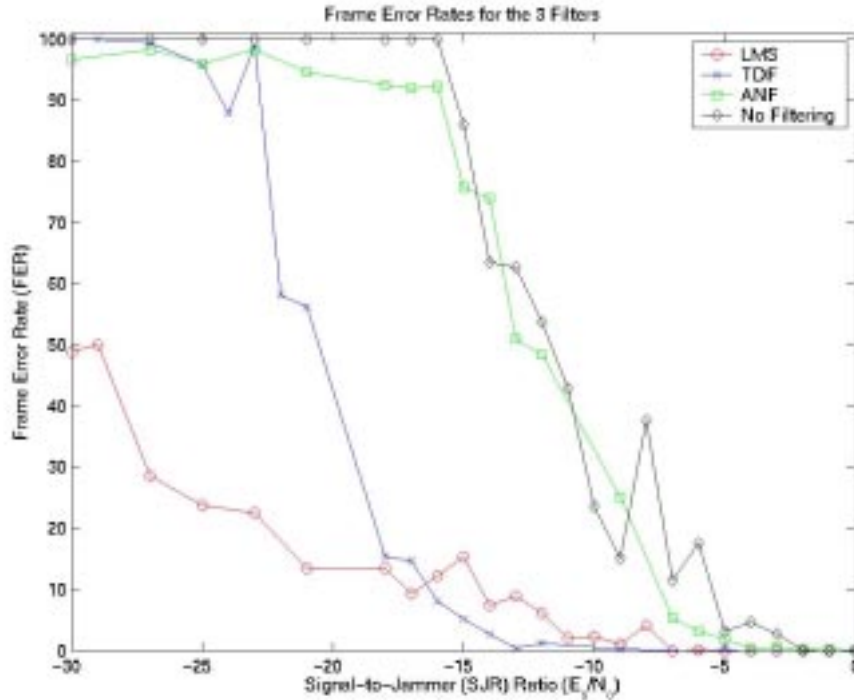
Figure 8.7. The FER curves for each of the filters as well as the FER curve without filtering.

Figure 8.7 shows the FER curves for each filter; there is only one curve per filter since the long PLCP format was used for all the data rates. The leftmost curve is for the LMS filter, which has an FER of only 50% at an input SJR of -30 dB. On the other end, the adaptive notch filter's FER curve is approximately in line with the unmitigated FER curve.

8.5. Analysis of Results

In analyzing the performance of each of the filters for all the data rates, the improvements in the BER, FER and signal-to-jammer ratio were measured. The BER and FER improvements are actually measured by the difference in the SJR needed between the unmitigated and filtered case in order to achieve a particular BER/FER level, as

138

opposed to the ratio/difference of the BER/FER for a fixed SJR level. This improvement is defined as the "gain", since the filter allows this increase in the jammer strength in order to achieve the same BER/FER level. The following are the analyses of each filter's performance.

8.5.1. Transform Domain Filter (TDF) Performance

The TDF was observed to have BER curves completely to the left of the no-filtering curves for all the data rates, indicating that it provides an improvement in the bit error rate and effectively filters out the interference. Table 8.1 lists several measured SJR requirements and corresponding gains for several benchmark BER levels.

Gains in BER with TDF Filtering				
Data Rate (Mbps)	Desired BER Level	SJR Requirement (dB)		BER Gain (dB)
		With Filtering	Without	
1	10^{-2}	n/a	-22	n/a
	10^{-3}	-27	-19	8
	10^{-4}	-22	-16	6
	10^{-5}	-16	-14	2
2	10^{-2}	n/a	-5.25	n/a
	10^{-3}	-24.5	-3.5	21
	10^{-4}	-10.5	-2.5	8.5
	10^{-5}	-7.5	-2.5	5
5.5	10^{-2}	-21.5	-2	19.5
	10^{-3}	-17.5	0	17.5
	10^{-4}	-9	2	11
	10^{-5}	-4.5	3	7.5
11	10^{-2}	-17	2	19
	10^{-3}	-12.5	3	15.5
	10^{-4}	-5	4.5	9.5
	10^{-5}	0.5	5.5	5

Table 8.1. The improvements in the SJR requirement for the TDF filter for several benchmark BER levels.

Based on the results, all gains are positive and are observed to increase with the desired BER levels. Note that there is an "n/a" entry for a BER of 10^{-2} at 1 and 2 Mbps, since the filter never reaches this level for these data rates. At the weakest SJR test value of -30 dB, the BER with no interference mitigation was at the peak value of 0.50 for all the data rates. Although the TDF also had a BER of 0.50 for at 11 Mbps for this SJR, it only peaked at 10^{-3}, $3x10^{-3}$, $1.5x10^{-2}$ for 1, 2, and 5.5 Mbps, respectively.

Consequently, the TDF filter provides gains for all FER levels. Table 8.2 lists the improvements for several benchmark FER levels.

Gains in FER with TDF Filtering			
FER Level (%)	SJR Requirement (dB)		FER Gain (dB)
	With Filtering	Without	
0	-13	-2	11
10	-16	-5	11
20	-18.5	-9.5	9
30	-19	-10.5	8.5
40	-20	-11	9
50	-20.5	-12	8.5
100	-25	-16	9

Table 8.2. The improvements in the SJR requirement for the TDF filter for several benchmark FER levels.

From the results listed in the table above, the filter provides a consistent gain between 8.5 and 11 dB in the FER level. The filter reaches a zero FER level at -13 dB, compared to the no-filtering case where it reaches zero only for -2 dB.

These results are explained more clearly by the plots of the measured improvements in the actual signal-to-jammer ratio, or the difference between the average SJR of the output and the SJR of the input. Looking at the SJR-improvement plots, they

140

can be observed to follow a general pattern: they are linear and downward sloping for low SJRs up to a certain point, after which they flatten out.

SJR Improvement Measured at Output					
Data Rate (Mbps)	Linear Region				Improvement at High SJRs (dB)
	|Slope|	Lower Limit (dB)	Upper Limit (dB)	Improvement at 0 dB (dB)	
1	0.77	-30	0	1.09	-0.7
2	0.77	-30	0	2.82	-0.71
5.5	0.8	-30	10	4.83	-0.84
11	0.81	-30	10	7.29	-0.81

Table 8.3. Characterizations of the improvement in the SJR at the output.

The linear regions for all the data rates have the same (absolute value) slope of approximately 0.8 dB.[21] That is, the filter removes 0.8 dB of the jammer for every decibel that the jammer energy increases over the signal energy, within the linear range. This is actually a desirable feature of the filter, since it indicates that as the jammer increases strength over the signal (i.e. SJR gets lower), the filter likewise excises more of the jammer.

For high SJR cases where the jammer is relatively weak, the improvement flattens and filtering actually results in a slight degradation in the SJR between -0.7 and -0.8 dB. This is consistent with the result obtained in section 5.2.5, where a degradation of -0.76 dB is observed when the input to the filter does not have any jammer.

One notable relation is that for a particular input SJR to the filter, the SJR improvement measured at the output closely follows the gain for the resulting BER level. This is because the jammer's strength at the output of the filter is reduced by the value of

---

[21] Although the slope is negative, its absolute value is referred to, since this directly relates the increase in jammer excision per unit increase in the relative jammer strength to the signal.

the SJR improvement, thus the bit error rate attained is equivalently the BER for the SJR at the output without any interference mitigation. However, the filter could have introduced its own distortions into the signal, and this partly accounts for the discrepancies. This relation actually applies to all three filters.

Overall, the filter performs extremely well in practice, since it provides significant BER, FER and SJR improvements for all SJR input levels, and also does not cause any deterioration in performance in the absence of the microwave oven jammer.

8.5.2. Least Mean Square (LMS) Adaptive Filter Performance

The LMS filter has BER curves that cross the no-filtering curves at high SJR levels, indicating that it actually causes degradation in performance for those cases. At high SJR values for 2 and 5.5 Mbps, the BER was observed to stay at the same level instead of going down to zero; this behavior will be explained in the discussion of the SJR-improvement plots later.

For low SJR values, however, the BER curves are to the left of the no-filtering curves, showing that the filter provides an improvement in performance for strong jammer cases. Table 8.4 lists the SJR requirements and gains of the filter.

The "n/a" entries in the table indicate that the filter did not produce those BER levels for the range of SJR values that were tested. As can be observed from the BER curve plots in Figures 9.4-9.6, the filter did not reach a BER of 10^{-5} for 2, 5.5 and 11 Mbps, and furthermore did not attain a BER of 10^{-4} for 2 Mbps.

In general, positive gains were observed for target BER levels of 10^{-2} and 10^{-3}, indicating that the filter is only practical for applications that can tolerate high bit error rates (e.g. voice transmission).

Gains in BER with LMS Filtering				
Data Rate (Mbps)	Desired BER Level	SJR Requirement (dB)		BER Gain (dB)
		With Filtering	Without	
1	10^{-2}	-30	-22	8
	10^{-3}	-25.5	-19	6.5
	10^{-4}	-12	-16	-4
	10^{-5}	-11.5	-14	-2.5
2	10^{-2}	-15	-5.25	9.75
	10^{-3}	-7.5	-3.5	4
	10^{-4}	n/a	-2.5	n/a
	10^{-5}	n/a	-2.5	n/a
5.5	10^{-2}	-9	-2	7
	10^{-3}	-6	0	6
	10^{-4}	-4	2	6
	10^{-5}	n/a	3	n/a
11	10^{-2}	-5	2	7
	10^{-3}	7	3	-4
	10^{-4}	14	4.5	-9.5
	10^{-5}	n/a	5.5	n/a

Table 8.4. Improvements in the SJR requirement for the LMS filter for several benchmark BER levels.

The gains in the FER, however, are much more significant. Table 8.5 lists the gains for several benchmark levels. The FER gains that the filter provides go as high as 17 dB, and furthermore the filter's peak FER was only 50%, found for the worst case SJR of -30 dB. Observing the FER plots in figure 8.7, the LMS filter curve is to the left of all the curves, indicating that it provides the best FER gains out of all the filters.

Gains in FER with LMS Filtering			
FER Level (%)	SJR Requirement (dB)		FER Gain (dB)
	With Filtering	Without	
0	-11	-2	9
10	-14	-5	9
20	-23	-9.5	13.5
30	-27	-10.5	16.5
40	-28	-11	17
50	-29	-12	17
100	n/a	-16	n/a

Table 8.5. The improvements in the SJR requirement for the LMS filter for several benchmark FER levels.

Again, these results can be directly tied to the amount of SJR improvement that was measured at the filter's output. As can be observed in the plots, the filter has a linear region of SJR improvement for low SJR values, while it flattens out at high SJR values, similar to the TDF filter. Table 8.6 lists the properties of the SJR improvement curves for each of the data rates.

SJR Improvement Measured at Output					
Data Rate (Mbps)	Linear Region				Improvement at High SJRs (dB)
	|Slope|	Lower Limit (dB)	Upper Limit (dB)	Improvement at 0 dB (dB)	
1	0.72	-30	0	-2	-4.8
2	0.83	-30	0	-3.32	-7.27
5.5	0.9	-30	10	-3.32	-8.55
11	0.91	-30	8	-3.32	-11.4

Table 8.6. Characterizations of the improvement in the SJR at the output

Looking at the plots in figures 8.3-8.6, the filter's SJR improvement curves are shifted down below that of the other filters', showing it provides the least improvement in the SJR out of all the filters because of the negative offset. Furthermore, for high SJR values, the filter introduces severe degradation, by as much as -11.4 dB for 11 Mbps; this

is a good estimate of the decrease in SJR that the filter provides in the case where there is no microwave oven jammer in the input. This severe degradation explains why the BER curves stay at the same level for high SJRs instead of going down to zero.

Overall, the filter was found to improve the performance of the receiver only for cases where the jammer was much stronger than the signal. For high SJR values however, it seems that the filter interpolates the signal since the jammer is weak and subtracts it out, causing degradation. The filter may be feasible for applications where high bit error rates are tolerated and low frame error rates are needed.

8.5.3. Adaptive Notch Filter (ANF) Performance

Except for11 Mbps, the BER curves for the adaptive notch filter stay completely to the left of the no-filtering curves, indicating that it gives an improvement in the performance of the receiver. In the case of 11 Mbps, the curves cross at an SJR of approximately 5 dB. Table 8.7 lists the SJR requirements and gains of the filter.

The entries for 1 Mbps are listed as "n/a" since the filter was not observed to attain those BER levels; the simulation results only showed zero bit error rates for the filter even at low signal-to-jammer ratios.[22]

In general, the filter provides modest BER gains for all data rates. Positive gains can be observed in all the cases except for the 10^{-4} and 10^{-5} rates in 11 Mbps. The highest improvements are observed for BER levels around 10^{-2} and are within the range of 5-7 dB.

---

[22] The simulations were limited to 24 hours on MIT machines, insufficient to produce more accurate BER values.

Gains in BER with ANF Filtering				
Data Rate (Mbps)	Desired BER Level	SJR Requirement (dB)		BER Gain (dB)
		With Filtering	Without	
1	10^{-2}	n/a	-22	-22
	10^{-3}	n/a	-19	-19
	10^{-4}	n/a	-16	-16
	10^{-5}	n/a	-14	-14
2	10^{-2}	-12	-5.25	6.75
	10^{-3}	-6.5	-3.5	3
	10^{-4}	-5.5	-2.5	3
	10^{-5}	-5	-2.5	2.5
5.5	10^{-2}	-9	-2	7
	10^{-3}	-5.25	0	5.25
	10^{-4}	-2.5	2	4.5
	10^{-5}	-0.5	3	3.5
11	10^{-2}	-2.5	2	4.5
	10^{-3}	2.5	3	0.5
	10^{-4}	10	4.5	-5.5
	10^{-5}	10	5.5	n/a

Table 8.7. Improvements in the SJR requirement for the ANF for several benchmark BER levels.

The gains for several benchmark FER levels are listed in table 8.8. No significant improvement in the FER can be observed, and this can be confirmed by observing the FER curve for the filter in figure 8.7 and noticing that it is right on top of the no-filtering curve. Furthermore, the filter is the first to reach an FER greater than 90%, which happens at an input SJR of -15 dB similar to the FER curve without any interference mitigation.

These phenomena are clearly explained by the SJR-improvement plots, which have their characteristics listed in table 8.9 below. Although there is also a linear region in each SJR-improvement plot, the plots show that the improvement plateaus at 18 dB when the SJR goes approximately below -16 dB for all the data rates.

Gains in FER with ANF Filtering			
FER Level (%)	SJR Requirement (dB)		FER Gain (dB)
	With Filtering	Without	
0	-5	-2	3
10	-7.5	-5	2.5
20	-8	-9.5	-1.5
30	-9.5	-10.5	-1
40	-11	-11	0
50	-12.5	-12	0.5
100	-25	-16	9

Table 8.8. Improvements in the SJR requirement for the ANF for several benchmark FER levels.

This explains why the FER and BER quickly rise to their peak levels past an input SJR of -18 dB. Also, referring to table 8.9, the filter is observed to provide slight SJR degradation at high input SJRs, and the 3-dB degradation at 11 Mbps accounts for the crossing of those BER curves.

SJR Improvement Measured at Output					
Data Rate (Mbps)	Linear Region				Improvement at High SJRs (dB)
	|Slope|	Lower Limit (dB)	Upper Limit (dB)	Improvement at 0 dB (dB)	
1	0.77	-16	0	1.39	-0.14
2	0.77	-18	0	2.82	-1.2
5.5	0.77	-17	10	3.89	-1.86
11	0.7	-16	8	4.6	-3

Table 8.9. Improvements in the SJR requirement for the ANF for several benchmark FER levels.

Overall, the filter improves the BER performance of the receiver in the presence of microwave oven interference, while no significant improvement in the FER is observed. Furthermore, its use is not recommended for strong jammer cases where the SJR values are low, since it loses its efficacy in that operating region.

8.6. Complexity Analyses

Along with the performance of each filter, an important factor in considering its feasibility is the number of computations it performs per output sample, since this directly relates to the filter's power consumption and processing delay. In this context, a computation encompasses a variety of mathematical operations, ranging from the basic arithmetic expressions such as +,-,*,/, comparisons (>,<,=), to more complex operations such as log(*), sqrt(*).[23] The following is a breakdown on the approximate number of computations each filter performs to produce one output sample.

a.) Transform Domain Filter – Let l_1 and l_2 be the number of operations for sqrt(*) and log(*) respectively. Given that the sliding Hamming window is k samples long, the number of computations for a windowed sequence is:

$klog_2k$ - Number of FFT computations for a window of length k

$(3+l_1)k$ - Magnitude computation defined as: $|a + jb| = \sqrt{a^2 + b^2}$

$(1+l_2)k$ - Conversion to decibels: $20log_{10}(x)$

k - Averaging, adding up all k samples and dividing by k.

$2k + 1$ - Standard deviation computation $\sigma_x = \sqrt{\sigma_x^2} = \sqrt{\frac{1}{k+1}\sum_{i=0}^{k}(x[i] - \mu_x)^2}$

3 - Compute threshold $l_x = \mu_x + \alpha\sigma_x$

$2k$ - linearly scanning, comparing at each step against the threshold, and setting the frequency bins to zero.

---

[23] Log(*) and sqrt(*) computations are performed using Taylor Series expansions, which take a constant number of operations for an input.

$klog_2k$ - Number of Inverse FFT computations for a window of length k

$2klog_2k + (9+ l_1 +l_2)k + 4$ - Total number of computations for k samples.

Thus, the average number of operations per output sample is just $2log_2k + (9+ l_1 +l_2)k + 4/k \approx 2log_2k + (9+ l_1 +l_2)$. Given that $k = 1838$, and bounding l_1 and l_2 by a constant c, then there are approximately $20 + 2c$ operations per output sample.

b.) Least-Mean Square (LMS) Adaptive Filter – The breakdown on the computations of the filter with $N = 32$ taps for one output sample are as follows:

$(2N -1)$ – Convolution computation (N multiplications, $N -1$ additions)

1 - Subtraction of convolution output from current input sample.

4 - Updating the scaling parameter _ (implementation specific).

$2N$ - Updating the filter taps

$4N + 4$ - Total number of operations per output sample.

c.) Lattice IIR Notch Filter – based on the schematic diagram of the filter in Figure 8.1, there are a total of 20 operations (multiplications and additions) for one output sample. The polyphase decomposition was considered as an indexing problem for the received sequence that does not require any mathematical operations.

Based on the respective computational costs of each of the filters, the lattice IIR notch filter is the most efficient, requiring only 20 computations per sample. The transform domain and LMS filters can be potentially costly, depending on the values for l_1 and l_2 for the former and N for the latter.

8.7. Comparative Analysis

In comparing the efficacies of the filters for mitigating microwave oven interference, their respective improvements in the BER and FER were considered, as well as their computational costs and feasibility. Based on the results and analyses, the TDF filter is the most effective in reducing the BER out of all the filters, and provides considerable gains for FER levels as well. Furthermore, almost no signal degradation is introduced in the absence of the jammer. The computational costs may get expensive, however, due to threshold computations. If power and complexity are not issues, then the TDF is an excellent filter for the interference.

The adaptive notch filter comes in next, providing smaller improvements in the BER than the TDF and modest FER gains. Although the filter does not worsen the performance of the receiver for the most part, it has a low tolerance for strong jammers. Furthermore, a small reduction in the SJR can be observed for high SJR scenarios, although this still kept BER values to low levels. The filter has a fixed, low number of computations per output sample, which gives it an advantage if processing times and power consumption are issues.

The LMS Interpolation filter comes in last in BER performance, since it only reaches BER levels on the order of 10^{-4} and 10^{-5} for very high SJRs. Furthermore,

deterioration in the performance of the receiver is observed for high SJR values, since the filter seems to interpolate the signal instead. The number of computations of the filter is directly proportional to the number of taps, and may get costly. If high BER levels can be tolerated as long as FER levels are kept to a minimum, then the filter might be a good candidate.

8.8.   Summary

The simulations without interference mitigation were used to characterize the receiver and provide a basis of comparison for the filter simulations. The filter simulation results, on the other hand, showed that the three filters offer some improvement in the bit error rate and frame error rate for all the data rates. Overall, the TDF filter was observed to provide the best improvement in the BER, while the LMS filter kept the FER to a maximum of 50%. Meanwhile, the adaptive notch filter (ANF) requires a low number of computations while providing modest performance gains.

Chapter 9: Conclusion

9.1. Conclusion

The research work produced several important results. First off, it was confirmed that microwave oven interference degrades the data throughput in an 802.11b communications link and can make communication impossible in some cases. The microwave oven interference was characterized and found to have periodic, frequency-sweeping sinusoidal behavior; a model was developed based on these measurements. A basic 802.11b network model was also constructed and integrated with the interference model.

Simulations characterized the degradation in the BER and FER performance of an 802.11b receiver in the presence of microwave oven interference, and were consistent with the results of the actual experiments.

Finally, it was shown that various filtering techniques can be employed to mitigate the microwave oven interference. The filters were all adaptive in a general sense, since they had to cancel out interference that had dynamic characteristics. The most effective filter was found to be the Transform Domain Filter, which filters out the jammer in the frequency domain. The TDF filter provided the largest gains in the BER, and likewise provided significant improvements for the FER.

The adaptive notch filter also provided modest improvements in performance but had limited range since it was found to be ineffective at low SJRs. The filter operated based on an iterative algorithm for converging and tracking a sinusoid input's frequency.

Finally, the LMS filter provided improvement mostly for strong jammer scenarios, and worsened performance for high SJR cases since it interpolated the signal instead of the jammer. The filter implements a recursive estimation algorithm for interpolating and subtracting the jammer.

9.2. Future Research

Several aspects of the thesis can be focused on in future research. Given more time and computational resources, more accurate modeling of the wireless channel environment can be done, where channel filter responses mimic multipath environments and introduce their effects on the receiver. It might be possible that the mitigation filters will perform differently in this environment. More microwave oven subjects can be tested in order to ensure that the interference models are precise and ensured to be consistent with the throughput degradation measurements.

The thesis only simulated the *center dwelling frequency* of the interference at the edge of the 3-dB bandwidth of the 802.11b signal, since the simulations were patterned after the set of measurements made in the experiments. For future research work, different overlapping patterns between the microwave oven interference and the signal should be examined, since each scenario may result in a different set of BER and FER curves. For example, the *center dwelling frequency* of the interference could be placed in different regions in and out of the 3-dB bandwidth of the signal.

Further work on the filter parameters for the TDF, LMS, and ANF filters can also be carried out. A more accurate thresholding algorithm can be developed for the TDF filter, where the optimal BER can be achieved by notching the exact number of frequency

bins needed. More work is needed to determine the appropriate sample delay and number

of filter taps for the LMS filter, since the set chosen for this thesis showed that there was

possible room for improvement. Similar research can be carried out on the adaptive notch

filter to figure out if there is a more appropriate level for the polyphase decomposition

can be determined. Other improvements to the filter can be made, such as saving

computational costs by sharing the notch frequency parameter across the polyphase

filters, since these will have the same value at any time instant. Also, longer simulations

need to be run on the LMS and ANF filters for 1 Mbps, since these simulations took a

considerable amount of time and did not provide enough information about their

performance.

Lastly, the problem of mitigating microwave oven interference can be extended to

other growing wireless standards, such as the 802.11a and 802.11g standards, both of

which operate at 5 and 2.4 GHz, respectively, and employ Orthogonal Frequency

Division Multiplexing (OFDM) for modulation. For example, the transform domain filter

(TDF) might have some use for this type of modulation scheme since OFDM

demodulates the received signal through its Fast Fourier Transform (FFT).

Appendix A: 802.11b Data Scrambling and Channel-Encoding Schemes

A.1. Data Scrambling and Descrambling

After the packet bits are generated, they are passed through a data scrambler before being modulated. At the receiver, the packets are descrambled is performed to reproduce the packet bits. The transfer function polynomial $G(z) = z^{-7} + z^{-4} + 1$ is used for both scrambling and descrambling. The current output bit of the scrambler is the *xor* sum of the 7th and 4th earlier *output* bits and 1, while for the descrambler *input* bits are used instead of the output for computing the sum. An initializing value referred to as a 'seed' is used as the values in the polynomial. The seed pattern is [1 1 0 1 1 0 0] for the long preamble, while the reverse bit pattern is used for the short preamble.
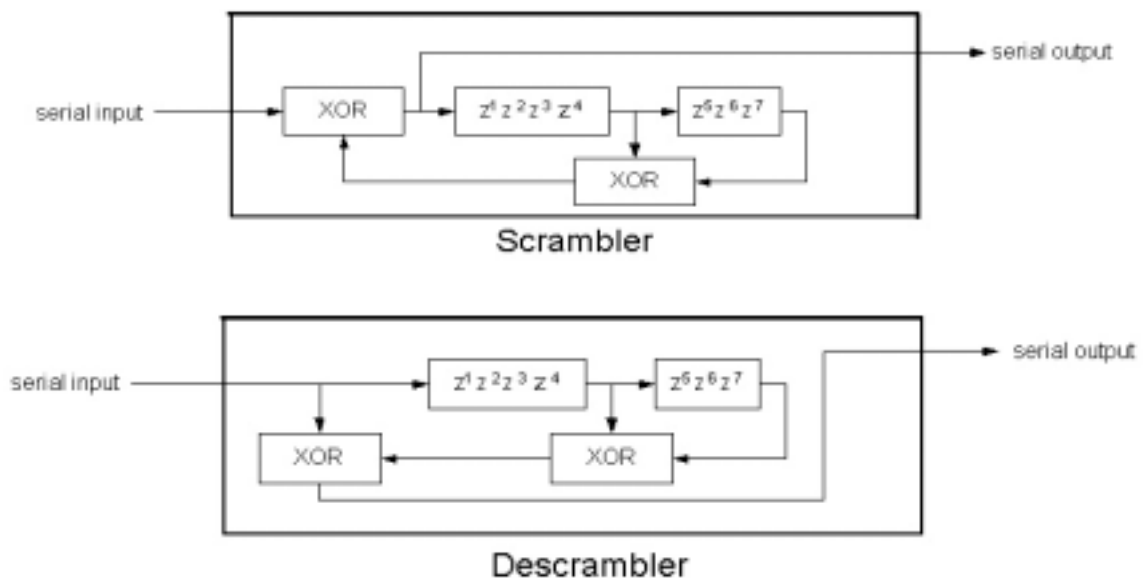
Figure A.1. The data scrambler and descrambler use the same polynomial for scrambling. Source: [4].

The feed-through configuration descrambler is self-synchronizing; no prior knowledge of the transmitter initialization of the scrambler is needed for processing at the receiver.

A.2. Data Rates and Modulation Techniques

After the packet bits are scrambled, they are passed through a channel-encoder, which maps them to complex-valued symbols in a symbol constellation. These symbols are later shaped onto root-raised cosine pulses to create the waveform for transmission. Different modulation techniques provide different data rates for transmission. The data rate depends on the number of bits that are encoded into one output symbol and on the rate at which the bits are coming into the modulator. As mentioned in section 2.5, DBPSK is used to produce a data rate of 1 Mbps, while DQPSK gives 2 Mbps. For the higher data rates of 5.5 and 11 Mbps, CCK encoding is used.

A.2.1. Differential Binary Phase Shift Keying (DBPSK)

The DBPSK scheme works by mapping bits onto phase changes within adjacent output symbols. The encoder takes in one bit at a time from the incoming bit stream. An input of 0 corresponds to no change in the phase of the previous symbol; the output is exactly the same as the previous symbol. An input of 1, on the other hand, corresponds to a 180 degree rotated copy of the previous symbol. The table below summarizes this mapping.

Bit Input	Phase change (Radians)
0	0
1	π

]Table A.1. The DBPSK encoding table

A reference symbol is defined for the first symbol that comes out of the encoder. In the implementation of the modulator, the reference is complex-valued, wherein its in-phase (real) and quadrature-phase (imaginary) parts are equal to each other. As a result, the sequence of in-phase and quadrature-phase values for any sequence of output symbols will be exactly the same.

In the 802.11b standard, the bits coming into the modulator are clocked at 1 MHz. Since there is one bit encoded per symbol and the symbol rate is set at 1 MSymbols/s, then 1 megabits are sent every second using this modulation technique.

A.2.2. Differential Quadrature Phase Shift Keying (DQPSK)

DQPSK modulation works in the same fashion as DBPSK, but encodes two bits at a time to produce one output symbol. The four possible two-bit inputs correspond to four possible phase changes of the previous symbol, set in 90-degree increments. The table below describes this mapping.

Dibit Input	Phase change (Radians)
00	0
01	π/2
11	π
10	3π/2 (-π/2)

Table A.2. The DQPSK encoding table

Similarly to DBPSK, a reference symbol is needed for the first output symbol. Since DQPSK is used to encode only the header and/or payload, the reference symbol used will be the last symbol encoded in the preceding section of the packet.

Using this encoding scheme, twice as many bits are encoded into one output symbol than in DBPSK, resulting in a data rate of 2 Mbps.

A.2.3. Complementary Code Keying (CCK)

Complementary Code Keying (CCK) modulation, as its name implies, uses complementary codes for modulation. Each code, consisting of a sequence of values or 'chips', is the equivalent of one output symbol. The codes used in both data rates are 8 complex-chips long and use the formula below.

$$cck\_symbol = \left[e^{j(\phi_1+\phi_2+\phi_3+\phi_4)}, e^{j(\phi_1+\phi_3+\phi_4)}, e^{j(\phi_1+\phi_2+\phi_4)}, -e^{j(\phi_1+\phi_4)}, e^{j(\phi_1+\phi_2+\phi_3)}, e^{j(\phi_1+\phi_3)}, -e^{j(\phi_1+\phi_3)}, e^{j(\phi_1)} \right]$$

$$(A.1)$$

The input bits at the encoder determine the parameters φ_1, φ_2, φ_3, and φ_4, which in turn defined the output symbol. The two data rates below differ in the number of bits that encode these parameters, which result in the different rates for transmission.

A.2.3.1. CCK Encoding for 5.5 Mbps

For this data rate, 4 bits at a time are encoded into one codeword. To aid the description of the encoding, every four bits will be labeled sequentially as d0, d1, d2, and d3, where d0 comes first in time. The first dibit pattern, bits d0 and d1, encode φ_1 using

159

the same DQPSK encoding in table A.2. However, an extra 180-degree rotation is given

to φ_1 if the current symbol being encoded is odd-numbered. Furthermore, the first symbol

to be encoded is numbered '0', which is even. The rest of the parameters are determined

using the following equation:

$$\phi_2 = d2 \times \pi \, , \ \phi_3 = 0, \text{and } \phi_4 = d3 \times \pi \qquad \text{(A.2)}$$

The symbol rate for CCK modulation is set at 1.375 MSymbols/s to produce the

same instantaneous bandwidth as the spread signals in the lower data rates. With 4 bits

encoded into one symbol, a data rate of 5.5 Mbps is achieved.

A.2.3.2. CCK Encoding for 11 Mbps

For the 11 Mbps data rate, encoding is done using 8 bits at a time. Using the same

labeling procedure for the bits in the preceding section, the first two bits, d0 and d1, are

encoded in the same manner used in the 5.5 Mbps data rate. The remaining bits are

quadrature phase shift keying (QPSK) encoded, two bits at a time, starting with d2 and

d3.

Although the same mapping in table A.2 is used, in this case, dibit patterns

determine phase values that are used to compute the codeword, and not directly used to

create phase changes from the previous output symbol, which is why the encoding is

referred to as QPSK and not DQPSK.

Since twice as many bits are encoded at a time compared to the 5.5 Mbps data rate, twice the number of bits is transmitted over the same time interval, resulting in a data rate of 11 Mbps.

Appendix B: Signal-to-Noise Ratio (SNR) Conversion & Theoretical Bit Error Rates/Probabilities of Error

B.1. Signal-to-Noise Ratio Units

Various units for the signal-to-noise ratio are used in analyzing bit error rates and frame error rates. This appendix relates the metric used for the simulations, SNR as defined in equation 8.1, to other commonly used units such as $\frac{E_s}{N_o}$ and $\frac{E_b}{N_o}$, which are the symbol-to-noise energy and bit-to-noise energy ratios, respectively. Note that the SNR defined in equation 8.1 is in decibels (dB), while the values for $\frac{E_s}{N_o}$ and $\frac{E_b}{N_o}$ are in the normal (linear) scale. Thus, if we call SNR_{Linear} the corresponding value of the SNR

in the normal scale: $SNR_{linear} = \dfrac{\sum_{n=-\infty}^{\infty} s^2[n]}{\sum_{n=-\infty}^{\infty} \zeta^2[n]}$

By definition, we can relate the two as:

$$SNR = 10\log_{10} SNR_{linear} \quad \Leftrightarrow \quad SNR_{linear} = 10^{SNR/10} \qquad \text{(B.1)}$$

From hereon, SNR_{Linear} will be used in place of the original SNR, since these two can easily be related by B.1. The relations of the three different units mentioned above can be derived from the following equations:

$$\frac{E_b}{N_o} = \left(\frac{\#bits}{symbol}\right)\frac{E_s}{N_o} \qquad (\text{B.2})$$

$$\frac{E_s}{N_o} = SNR_{linear}\left(\frac{T_{symbol}}{T_{samp}}\right) \qquad (\text{B.3})$$

where for the model implementation discussed in Chapter 4, $T_{symbol} = 11$ MHz while

$T_{samp} = 44$ MHz.

B.2. Theoretical Bit Error Rates/Probabilities of Error

The probability of obtaining a bit error is equivalent to the bit error rate (BER).

Each modulation scheme used in the 802.11b standard has corresponding bit error

probabilities, which yield the theoretical BER curves that were superimposed with the

simulation results in figure 8.1. The following formulas, obtained from [24], show the

theoretical probabilities of error for each of the 802.11b data rates in an AWGN channel

as functions of $\dfrac{E_b}{N_o}$:

a. DBPSK Modulation $- P_e = \dfrac{1}{2}\exp\left(\dfrac{-E_b}{N_o}\right) \qquad (\text{B.4})$

b. DQPSK Modulation – $P_e = Q_1(a,b) - \dfrac{1}{2}I_o(ab)\exp\left(-\dfrac{1}{2}(a^2 + b^2)\right)$, where

$Q_1(a,b)$ is the Marcum Q function, $I_o(ab)$ is the modified Bessel function,

$$a = \sqrt{\frac{2E_b}{N_o}\left(1 - \sqrt{\frac{1}{2}}\right)} \text{ and } b = \sqrt{\frac{2E_b}{N_o}\left(1 + \sqrt{\frac{1}{2}}\right)} \qquad (B.5)$$

c. CCK Encoding – Let M be the number of bits modulated per 8-chips in each CCK symbol. For 5.5 Mbps, M = 4 and for 11 Mbps, M = 8. The probability of error is:

$$P_e = 1 - \int\limits_{-X}^{\infty}\left(\frac{1}{\sqrt{2\pi}}\int\limits_{-v-X}^{v+X}\exp\left(\frac{-y^2}{2}\right)dy\right)^{\frac{M}{2}-1}\exp\left(\frac{-v^2}{2}\right)dv, \text{ where } X = \sqrt{\frac{2E_b}{N_o}} \quad (B.6)$$

Bibliography:

1.) Staff, AllNet Devices. "Public WLANs Growing Rapidly in U.S.". AllNetDevices. [Online] Source: http://www.allnetdevices.com/wireless/nes/2002/02/07/public_wlans.html. February 7, 2002.

2.) Walton, M. "Wireless 'Cloud' May Offer Silver Lining". CNN. [Online] Source: http://www.cnn.com/2002/TECH/science/07/31/coolsc.wireless.cloud/index.html. July 31, 2002.

3.) Staff, In-Stat Research. "WLAN Chips to Embark on Incredible Journey". In-Stat Research. [Online] Source: http://www.instat.com. April 3, 2002.

4.) IEEE 802.11b Working Group. "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band". http://www.ieee.org. September 16, 1999.

5.) Pearson, Bob. "Complementary Code Keying Made Simple". Intersil Corporation. [Online] Source: http://www.intersil.com/data/an/an9/an9850/an9850.pdf. November, 2001.

6.) Meel, J. "Spread Spectrum". De Nayer Instituut & Sirius Communications Publication. [Online] Source: http://www.sssmag.com/pdf/Ss_jme_denayer_intro_print.pdf. October 6, 1999.

7.) Federal Communications Commission (FCC).Part 15- Radio Frequency Devices, Code of Federal Regulations. Title 47, Volume 1. April, 2002.

8.) Gallagher, R. "6.450 Lecture Notes, Lecture 21: Input/Output Models for Wireless Channels". MIT 6.450 Lectures Notes. [Online] Source: http://web.mit.edu/6.450/www/Lectures/lecture21.pdf.

9.) Federal Communications Commission (FCC). Section 1030.10- Microwave Ovens, Code of Federal Regulations. Title 21, Volume 8. April, 2002.

10.) Gallawa, J.C. "How Does a Microwave Oven Work?: Basic Theory of Operation". Excerpt from The Complete Microwave Oven Service Handbook. [Online] Source: http://www.gallawa.com/microtech/how_work.html. Copyright 2001.

11.) Gallawa, J.C. "The Magnetron Tube Structure and Operation". Excerpt from The Complete Microwave Oven Service Handbook. [Online] Source: http://www.gallawa.com/microtech/magnetron.html. Copyright, 2001.

12.) US Department of Health (DOH). "Microwave Oven Radiation Safety". [Online] Source: http://www.doh.wa.gov/ehp/rp/rp-oven.htm.   March 8, 2000.

13.) Kim, K.S. "Certificate of Compliance, Model No: R-419CK". LG Electronics Inc. Quality & Reliability Center. [Online] Source: http://www.fcc.org/fccid/ . November 4, 1999.

14.) Kamerman, A. & Erkocevic, N. "Microwave Oven Interference on Wireless LANs Operating in the 2.4 GHz ISM Band". HF Journal, [Online] Source: http://www.sitel.org/hfdir.htm. February, 2000.

15.) Unawong, S., Miyamoto, S. & Morinaga, N. "Effects of Microwave Oven Interference on the Performance of ISM-Band DS/SS System". IEEE. 1998.

16.) Kanemoto, H., Miyamoto, S. & Morinaga, N. "A Study of Modeling of Microwave Oven Interference and Optimum Reception". IEEE. 1998.

17.) Kumpumaki, T. , Isohookana, M. & Juntti, J. "Narrow-band Interference Rejection Using Transform Domain Signal Processing in a Hybrid DS/FH Spread-Spectrum System". IEEE. 1997.

18.) Horne, J. & Vasudevan, S. "Modeling and Mitigation of Interference in the 2.4 GHz ISM Band". March/April 1997. Wireless Communications Conference, 1996. March/April 1997.

19.) Oppenheim, A. & Schafer, R. Discrete-Time Signal Processing, 2nd Edition. p. 471. Prentice Hall, February 15, 1999.

20.) Glisic, S., Mammela, A., Veli-Pekka, K. & Pajkovic, M. "Rejection of Frequency-Sweeping Signal in DS Spread Spectrum Systems Using Complex Adaptive Filters". IEEE Transactions on Communications, Vol. 43, No.1. January, 1995.

21.) Glentis, G., Berberidis, K. & Theodoridis, S. "Efficient Least Squares Adaptive Algorithms for FIR Transversal Filtering". IEEE Signal Processing Magazine, July 1999.

22.) Regalia, P. Adaptive IIR Filtering in Signal Processing and Control. pp. 554-599. Dekker, Marcel Inc. 1994.

23.) Smith, S. "Sample Size Needed for Specified Margin of Error". [Online] Source: http://www.cate.org/sms99/writ99/sizesam.htm

24.) Fainberg, M. & Goodman, D. "Maximizing Performance of a Wireless LAN in the Presence of Bluetooth". The 3rd IEEE Workshop on Wireless Local Area Networks. [Online] Source: http://www.wlan01.wpi.edu/proceedings/wlan05d.pdf. September, 2001.

25.) Medley, M., Saulnier, G. & Das, P. "Narrow-Band Interference Excision in Spread Spectrum Systems Using Lapped Transforms". IEEE Transactions on Communications, Vol. 45, No. 11. November, 1997.

26.) Geier, J. "802.11 MAC Layer Defined". 802.11b Tutorials. Source: http://www.80211-planet.com. June 4, 2002.

27.) Shima, J.M. "Narrowband Jammer Suppression Using an Adaptive Filter and Temporal Root Tracking". Texas Instruments Publication.

28.) Moeneclaey, M., Van Bladel, M. & Sari, H. "Sensitivity of Multiple-Access Techniques to Narrow-band Interference". IEEE Transactions on Communications, Vol. 49, No. 3. March, 2001.

29.) Wang, C. & Wu, K. "A New Narrowband Interference Suppression Scheme for Spread-Spectrum CDMA Communications". IEEE Transactions on Signal Processing, Vol. 49, No. 11. November, 2001.

30.) Wang, C., Amin, M. & Lindsey, A. "Open Loop Adaptive Filtering for Interference Excision in Spread Spectrum Systems". IEEE. 1998.

31.) Saarnisaari, H. "Delay Estimation in the Presence of a Tone Jammer: Two Simple Estimators". IEEE. 1999.

32.) Thomas, J. & Geraniotis, E. "Joint Iterative MMSE Multiuser Detection and Narrowband Jammer Suppression in Coded DS-CDMA Channels". IEEE. 1999.

33.) Kaleh, G.K. "A Frequency Diversity Spread Spectrum System for Communication in the Presence of In-Band Interference". IEEE. 1995.

34.)   Ouyang, X. & Amin, M. "Short-Time Fourier Transform Receiver for Nonstationary Interference Excision in Direct Sequence Spread Spectrum Communications". IEEE Transactions on Signal Processing, Vol. 49, No. 4. April, 2001.

35.)   Ranheim, A. "Narrowband Interference Rejection in Direct-Sequence Spread-Spectrum System Using Time-Frequency Decomposition". IEE Proc.-Communications, Vol. 142, No. 6. December, 1995.

36.)   Akan, A., Yalcin, M. & Chaparro, L. "An Iterative for Instantaneous Frequency Estimation". IEEE. 2001.

37.)   Buzzi, S., Lops, M. & Tulino, A. "Time-Varying Narrow-Band Interference Rejection in Asynchronous Multiuser DS/CDMA Systems over Frequency-Selective Fading Channels". IEEE Transactions on Communications, Vol. 47, No. 10. October, 1999.

38.)   Benesty, J. & Duhamel, P. "A Fast Exact Least Mean Square Adaptive Algorithm". IEEE Transactions on Signal Processing, Vol. 49, No. 12. December, 1992.

39.)   Abramowitz, J. "Wireless LANs- Poised for Untethered Growth". Wireless LAN Association, [Online] Source: http://www.wlana.org. 2001.

40.)   Carlemalm, C., Poor, H.V. & Logothetis, A. "Suppression of Multiple Narrowband Interferers in a Spread-Spectrum Communication System". IEEE Journal on Selected Areas in Communications, Vol. 18, No.9. August, 2000.

41.)   Capozza, P., Holland, B., Hopkinson, T. & Landrau, R. "A Single-Chip Narrow-Band Frequency-domain Excisor for a Global Positioning System (GPS) Receiver". IEEE Journal of Solid-State Circuits, Vol. 35, No. 3. March, 2000.

42.)   Kwok, H. & Jones, D. "Improved Instantaneous Frequency Estimation Using an Adaptive Short-Time Fourier Transform". IEEE Transactions on Signal Processing, Vol. 48, No. 10. October, 2000.

43.)   Rusch, L.A. "Indoor Wireless Communications: Capacity and Coexistence of the Unlicensed Bands". Intel Architecture Group, Intel Corporation.

44.)   Vasudeva, S., Horne, J. & Varanasi, M. "Reliable Wireless Telephony using the 2.4 GHz ISM Band : Issues and Solutions".

45.)   Chang, W., Lee., Y., Ko, C. & Chen, C. "A Novel Prediction Tool for Indoor Wireless LAN Under the Microwave Oven Interference". Department of EE, Tamkang University, Taiwan.

46.)   Staff, Intersil Corporation. "Effect of Microwave Oven Interference on IEEE 802.11 WLAN Reliability". IEEE P802.11-98/240. May, 1998.

47.)   Uchino, M. Tagiri, O. & Shinozuka, T. "Real-time Measurement of Noise Statistics". IEEE Transactions on Electromagnetic Compatibility, Vol. 43, No. 4. November, 2001.

48.)   Yamane, K, Shinozuka, T. & Ohnuma, K. "Pseudo Noise Generator with Arbitrary APD, PDD and PSD". IEEE. 2000.

49.)   Unawong, S., Miyamoto, S. & Morinaga, N. "Application of Chip-Interleaving and Blanker for DS-SS System Under Microwave Oven Interference Environment". Electronics Letters, Vol. 35, No. 9. April 29, 1999.

50.)   Kanemoto, H., Miyamoto, S. & Morinaga, N. "Statistical Model of Microwave Oven Interference and Optimum Reception". IEEE 1998.

51.)   Miyamoto, S., Yamanaka, Y., Shinozuka, T. & Morinaga, N. "A Study of the Effect of Microwave Oven Interference on the Performance of Digital Radio Communication Systems". Electronics and Communications in Japan, Part 1. Vol. 80, No. 12, 1997.

52.)   Miyamoto, S. & Morinaga, N. "Performance of Radio Communication Systems Under Microwave Oven Interference Environment". IEEE. 1997.

53.)   Gawthrop, P., Sanders, F., Nebbia, K. & Sell, J. "NTIA Report 94-303-1: Radio Spectrum Measurements of Individual Microwave Ovens, Volume 1". National Telecommunications and Information Administration. March, 1994.

54.)   Butterweck, H.J. "Basic Theory of the LMS Adaptive Filter: Recent Developments". Department of EE, Eindhoven University of Technology.

55.)   Amin., M. Wang, C. & Lindsey, A. "Optimum Interference Excision in Spread Spectrum Communications Using Open-Loop Adaptive Filters". IEEE Transactions on Signal Processing, Vol. 47, No. 7. July, 1999.

56.)   Menon, Q. "Interference and Noise Removal from DS-Spread Spectrum Receiver Using a Fast Multiresolution Transform". Hamdard Institute of Information Technology, Pakistan.

57.)   Hsu, F. & Giordano, A. "Digital Whitening Techniques for Improving Spread Spectrum Communications Performance in the Presence of Narrowband Jamming and Interference". IEEE Transactions on Communications, Vol. COM-26, No.2. February, 1978.

58.)   Rusch, L. & Poor, V. "Narrowband Interference Suppression in CDMA Spread Spectrum Communications". IEEE Transactions on Communications, Vol. 42 2/3/4. February, 1994.

59.)   Li, L. & Milstein, L. "Rejection of Narrow-Band Interference in PN Spread-Spectrum Systems Using Transversal Filters". IEEE Transactions on Communications, Vol. COM-30, No. 5. May, 1982.

60.)   Saulnier, G., Das, P. & Milstein, L. "Suppression of Narrow-Band Interference in a PN Spread-Spectrum Receiver Using a CTD-Based Adaptive Filter". IEEE Transactions on Communications, Vol. COM-32, No. 11. November, 1984.

61.)   Iltis, R. & Milstein, L. "An Approximate Statistical Analysis of the Widrow LMS Algorithm with Application to Narrow-Band Interference Rejection". IEEE Transactions on Communications, Vol. COM-33, No. 2. February, 1985.

62.)   Reichman, A. & Scholtz, R. "Adaptive Spread-Spectrum Systems Using Least-Squares Lattice Algorithms". IEEE Journal on Selected Areas in Communications, Vol. SAC-3, No. 5. September, 1985.

63.)   Saulnier, G., Das, P. & Milstein, L. "An Adaptive Digital Suppression Filter for Direct-Sequence Spread-Spectrum Communications". IEEE Journal on Selected Areas in Communications, Vol. SAC-3, No. 5. September 1985.

64.)   Iltis, R. & Milstein, L. "Performance Analysis of Narrow-Band Interference Rejection Techniques in DS Spread-Spectrum Systems". IEEE Transactions on Communications, Vol. COM-32, No. 11. November, 1984.

65.)   Masry, E. "Closed-Form Analytical Results for the Rejection of Narrow-Band Interference in PN Spread-Spectrum Systems – Part I: Linear Prediction

Filters". IEEE Transactions on Communications, Vol. COM-32, No. 8. August, 1984.

66.) Masry, E. "Closed-Form Analytical Results for the Rejection of Narrow-Band Interference in PN Spread-Spectrum Systems – Part II: Linear Interpolation Filters". IEEE Transactions on Communications, Vol. COM-33, No. 1. January, 1985.

67.) Glisic, S. & Pajkovic, M. "Rejection of an FH Signal in a DS Spread-Spectrum System Using Complex Adaptive Filters". IEEE Transactions on Communications, Vol. 43, No. 5. May, 1995.

68.) Li, L. & Milstein, L. "Rejection of Pulsed CW Interference in PN Spread-Spectrum Systems Using Complex Adaptive Filters". IEEE Transactions on Communications, Vol. COM-31, No. 1. January, 1983.

69.) Wang, K. & Yao, Y. "New Nonlinear Algorithms for Narrow-Band Interference Suppression in CDMA Spread-Spectrum Systems". IEEE Journal on Selected Areas in Communications, Vol. 17, No. 12. December, 1999.

70.) Wang, Y. & Milstein, L. "Rejection of Multiple Narrow-Band Interference in Both BPSK and QPSK DS Spread-Spectrum Systems". IEEE Transactions on Communications, Vol. 36, No. 2. February, 1988.

71.) Benesty, J. & Duhamel, P. "A Fast Exact Least Mean Square Adaptive Algorithm". IEEE Transactions on Signal Processing, Vol. 40, No. 12. December 1992.

72.) Li, C., Hu, G. & Liu, M. "Narrow-Band Interference Excision in Spread-Spectrum Systems Using Self-Orthogonalizing Transform-Domain Adaptive Filters". IEEE Journal on Selected Areas in Communications, Vol. 18, No. 3. March 2000.

73.) Regalia, P. "An Improved Lattice-Based Adaptive IIR Notch Filter". IEEE Transactions on Signal Processing, Vol. 39, No. 9. September, 1991.

74.) Ketchum, J. & Proakis, J. "Adaptive Algorithms for Estimating and Suppressing Narrow-Band Interference in PN Spread-Spectrum Systems". IEEE Transactions on Communications, Vol. COM-30, No. 5. May, 1982.

75.) DeBrunner, V. "An Adaptive Notch Filter Used for Sinusoidal and Transient Modeling of Speech Signals". IEEE 1999.

76.) Clarke, D.W. "On the Design of Adaptive Notch Filters". International Journal of Adaptive Control and Signal Processing, Vol. 15, pp. 715-744. March, 2001.

77.) Liu, Y., Laakso, T. & Diniz, P. "A Complex Adaptive IIR Notch Filter Algorithm with Optimal Convergence Factor". Helsinkin University of Technology, Signal Processing Laboratory, Finland.

78.) Bouvier, M. Jr. "The Rejection of Large CW Interferers in Spread Spectrum Systems". IEEE Transactions on Communications, Vol. COM-26, No.2. February, 1978.

79.) US FDA Center for Devices and Radiological Health. "Microwave Oven Radiation". [Online] Source: http://www.fda.gov/cdrh/consumer/microwave.html.

80.) University of Politecnica de Valencia. "Microwave Oven Voltage-Doubler Circuit: Theory of Operation". [Online] Source: http://www.upv.es/gcm/docente/doubler.html