TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

ZHENYU ZHENG

EXPERIMENTAL EVALUATION OF SPECTRUM SENSING ALGORITHMS FOR WIRELESS MICROPHONE SIGNAL

Master of Science Thesis

Examiner: Prof. Markku Renfors
M.Sc. Sener Dikmese
Examiner and topic approved by the Council of the Faculty of Computing and Electrical Engineering on 5th May 2014

ABSTRACT

TAMPERE UNIVERSITY OF TECHNOLOGY
Master's Degree Programme in Information Technology
Zhenyu Zheng: Experimental Evaluation of Spectrum Sensing Algorithms for Wireless Microphone Signal
Master of Science Thesis, 54 pages, 5 Appendix pages
May 2014
Major: Communications Engineering
Examiner: Prof. Markku Renfors and M.Sc. Sener Dikmese
Keywords: Cognitive radio, energy based spectrum sensing, LabView, USRP, GNU radio, noise uncertainty.

Spectrum congestion has become a critical concern in wireless communication systems due to the limited availability of frequency spectrum. Hence, efficient utilization of spectrum is one of the most important challenges in the evolution of wireless communication systems and radio devices. Cognitive radio (CR) has been introduced as an effective solution for spectrum utilization. Spectrum sensing (SS) is one of the key elements in the implementation of effective and reliable CR systems. SS algorithms are used to obtain awareness about the spectrum usage and existence of primary users in a certain spectrum band. Energy detection (ED) based SS is the most common sensing algorithm due to its low computation and implementation complexity. On the other hand, ED based SS is highly dependent on the precise knowledge of the receiver noise variance. Hence, the performance of the ED algorithm is degraded significantly, when there is uncertainty in the estimation of the noise variance.

In this thesis, the wireless microphone (WM) system using the CR concept is introduced and the sensing performance of WM signals using three different algorithms are studied. The considered algorithms are based on the ED, namely fast Fourier transform (FFT) based ED, analysis filter bank (AFB) based ED and maximum-minimum ED (Max-Min ED) are studied. Following the analytical models and scenarios of energy detector based SS algorithms, the sensing algorithms are implemented using National Instruments' (NI) Universal Software Radio Peripheral (USRP) and the NI-LabVIEW software platform, together with the necessary toolboxes. This prototype implementation provides reliable performance evaluation of these spectrum sensing approaches using real world receiver implementation and communication signals from a signal generator, as well as actual WM signals. The results of this study suggest that the performance of Max-Min ED is more robust than FFT & AFB based ED under realistic noise variance uncertainty.

PREFACE

The research leading to this thesis has been carried out at the Department of Electronics Communications Engineering, Tampere University of Technology, Finland.

First of all, I would like to express my gratitude to my supervisor Prof. Markku Renfors for giving me the chance to study on this topic and his support during my thesis study even in his busy schedule and also listening problems within the process and his invaluable guidance.

I tender my thanks to M.Sc. Sener Dikmese and M.Sc. Sudharsan Srinivasan as they always helped me with their valuable guidance and advices. I also would like to thank them for their valuable friendship.

Finally, I would like to express my heart-felt thanks to my parents and friends for their love, support and patience during my studies.

TABLE OF CONTENTS

Abstract i
Preface ii
Table of Contents iii
List of Abbreviations v
List of Symbols vii
1. Introduction 1
1.1 Motivation and Background 1
1.2 Outline of the Thesis 2
2. Cognitive Radio and Spectrum Sensing Algorithms 4
2.1 Cognitive Radio 4
2.2 Spectrum Sensing 6
2.3 Energy Detector Based Spectrum Sensing Algorithms 7
2.3.1 Analytical Model of Spectrum Sensing Algorithms 7
2.3.2 Fast Fourier Transform and Analysis Filter Bank Based Algorithms 8
2.3.3 Max-Min Energy Based Sensing Algorithm 11
3. Implementation Environment 14
3.1 Software Defined Radio 14
3.2 Universal Software Radio Peripheral 15
3.2.1 NI USRP-2932 15
3.2.2 Limitations of NI USRP 17
3.3 National Instruments LabVIEW 19
3.3.1 LabVIEW basics 19
3.3.2 Benefits of programming in LabVIEW 20
3.3.3 LabVIEW working with NI USRP 20
4. Details of Implementation 23
4.1 Transmitter 23
4.1.1 Wireless Microphone 23
4.1.2 Transmission Channel Model 25
4.2 Receiver 27
4.2.1 Receiver VI Front Panel 28
4.2.2 Receiver VI Block Diagram 32
5. Algorithm testing Results 39
5.1 Noise Variance Uncertainty 39
5.2 Testing with Indoor Channel Model 40
5.3 Testing with ITU-R Vehicular A channel model 43
5.4 Testing with Stanford University Interim 1 channel model 45
5.5 Testing with Actual Indoor Wireless Environment Channel 47
5.6 Testing with Different WM Signal Models 50
6. Conclusion 52
References 54

Appendix A: Detailed Specifications of NI-USRP 2932 58
Appendix B: LabVIEW VI Block Diagram 60
Appendix C: Matlab Code for Generating WM signal With R & S SMJ 100A Vector Signal Generator 61

LIST OF ABBREVIATIONS

This is a list of the most important and recurrently appearing abbreviations in this thesis.

ADC	Analog-to-digital converter
AFB	Analysis filter bank
AWGN	Additive white Gaussian noise
BPF	Band pass filter
CFO	Carrier frequency offset
CR	Cognitive Radio
DAC	Digital-to-analog converter
DC	Digital current
DDC	Digital down converter
DSP	Digital signal processing
DTV	Digital television
DUC	Digital up converter
FCC	Federal Communication Commissions
FFT	Fast Fourier transform
FIR	Finite impulse response
FPGA	Field programmable gate array
GUI	Graphical user interface
I	In-phase
IIR	Infinite impulse response
IF	Intermediate frequency
IFFT	Inverse Fast Fourier transform
ISM	Industrial, Scientific and Medical
ITU-R	International Telecommunication Union Radio communication Sector
I/Q	In-phase/quadrature
LNA	Low noise amplifier
LO	Local oscillator
LP	Low pass
LPF	Low pass filter
MF	Matched filter
MMSE	Minimum mean square error
NP	Neyman-Pearson
NI	National Instruments
OFDM	Orthogonal frequency division multiplexing
PC	Personal Computer
PDF	Probability density function
PSD	Power spectral density

Q	Quadrature
RF	Radio frequency
RX	Receiver
SDR	Software defined radio
SNR	Signal to noise ratio
SUI	Stanford University Interim
TX	Transmitter side
USB	Universal serial BUS
USRP	Universal software radio peripheral
WRAN	Wireless Regional Area Network
VCO	Voltage controlled oscillator

LIST OF SYMBOLS

This is a list of the principle symbols and notations used throughout the thesis

A_c	Carrier amplitude
H_0	Hypothesis 0 in Neyman-Pearson test
H_1	Hypothesis 1 in Neyman-Pearson test
$h[n]$	Channel effects
f_c	Carrier frequency
k_f	Frequency sensitivity
L_f	Length of frequency averaging window
$m(t)$	Voice signal
N	Observation length
N_{FFT}	Number of FFT points
$\mathcal{N}$	Obeying Gaussian distribution
P_D	Detection probability
P_{FA}	False alarm probability
$\Pr(\cdot\|\cdot)$	Contingent probability
$Q(\cdot)$	Gaussian Q-function
$Q^{-1}(\cdot)$	Inverse Gaussian Q-function
$s[n]$	Generated WM signal
$s(t)$	Generated FM signal
$T(y)$	Test Statistics
$w[n]$	AWGN samples
$x[n]$	Transmitted signal
$y[n]$	Observed complex signal
σ_n^2	Variance
λ	Threshold value
γ	SNR value

1. INTRODUCTION

1.1 Motivation and Background

The usage of wireless communications has increased rapidly, due to the need of mobility and convenience, in the last decade. The wireless communication technology itself has also developed rapidly in recent years in order to fulfil the demand of increasing number of users, higher data transmission speed for new applications, as well as better mobility and security during the communication. Although there are great improvements in wireless communication's technical level, there are still a few bottlenecks in the development of future wireless communication systems. Among them, the shortage of available spectrum is considered as a vital problem.

Wireless communication system transfers information through electromagnetic waveforms. Electromagnetic waveforms with different wavelength λ (frequency $f=c/\lambda$, c is the speed of light) have different propagation characteristics, and different wireless applications use different frequency range (wavelength) for communication according to their requirements [1]. The allocation of spectrum for each application is regulated by standard bodies, such as Federal Communications Commission (FCC) and International Telecommunication Union (ITU). The spectrum can be considered as a scarce resource due to the rapidly growing number of users in various wireless communication applications, especially mobile phones and wireless data communications. [2]

Due to the shortage of spectrum with feasible characteristics, a more effective way of spectrum utilization is needed [3]. As a result of this need, the concept of cognitive radio (CR) concept has been introduced [4]. The most distinct characteristic of CR systems is the ability to identify the presence or absence of unknown deterministic signals, and to determine whether a primary licensed user (PU) is active or not, based on the identification process. The corresponding secondary unlicensed user (SU) either proceeds to use the unoccupied spectrum or remains silent according to the decision regarding the availability of the spectrum. By this way, the CR system can determine the spectral holes within a frequency band, allowing the unlicensed users to access those spectral holes opportunistically. This is an effective way of spectrum utilization and a good solution for the current scarce spectrum resource problem [5], [6].

Spectrum sensing (SS) is the operation of observing spectrum holes within the interesting frequency band, and it is considered as one of the most critical operations in CR systems. There are several approaches of SS based on pre-known information of the primary users [7]. Among all those SS techniques, energy detection (ED) based spectrum sensing algorithms are by far most widely used due to their low computational

complexity and simple practical realization [7], [8]. ED based analyses are commonly formulated as a Neyman-Pearson binary hypothesis test. Each one of the two hypothesises represents a simplistic scenario: the interesting frequency band under sensing is either contains only noise or is occupied by a PU signal together with noise, both having constant power spectral density (PSD).

The main disadvantage of the regular ED algorithm is its dependence on the knowledge of the noise variance [3]. The performance of ED decreases significantly due to the effect of noise uncertainty. In order to overcome this limitation under noise uncertainty scenarios, several improvements have been developed. One of them uses the information of maximum and minimum value of the subband energies within the sensing band, and it exhibits more robust performance than the regular ED sensing algorithm. Several other improvements, such as multi-antenna sensing and cooperative sensing also perform well. However, they are impractical for several applications due to the drawback of increased hardware complexity and size. There are also many other sensing algorithms which are robust to noise uncertainty, such as eigenvalue based algorithms [9], [10], [11] and autocorrelation based algorithms [12]. Similarly, these methods have much higher computational complexity and cannot reach the performance of ED under modest noise uncertainty and PU signal-to-noise ratio (SNR) [9]- [11]. Thus our thesis focused on the performance of enhanced ED based spectrum sensing algorithms and their implementation. The aim of this thesis is to implement a real-time CR system prototype with different ED based sensing algorithms, namely FFT&AFB based ED, FFT and AFB based ED with frequency averaging window, and Max-Min based ED and test them under real world environment.

The study focuses on sensing wireless microphone (WM) signals, which is an existing CR type application where the available channels of the terrestrial TV networks are used locally for WMs. WMs are used, e.g., in lecture halls, theatres, concert halls, and in various outdoor productions. WMs may also be used within the Industrial, Scientific, Medical (ISM) band. WMs are allowed to operate in those unlicensed frequencies as a CR device according to FCC part 74 rule [2]. We are using the ISM band for transmitting WM signals to test the SS algorithms in a controllable manner. Also the sensing of actual WM signals in the TV frequency band has been tested in this work.

Our test system is divided into two parts, the transmitter and the receiver. The transmitter is implemented using a PC, MATLAB script, and Rohde & Schwarz SMJ100A Vector Signal Generator. The receiver is implemented using a PC, National Instruments (NI) Universal Software Radio Peripheral (USRP)-2932 and the NI LabVIEW software platform.

1.2 Outline of the Thesis

In the second chapter, the concept of cognitive radio and its basic properties are introduced. This chapter also includes the details of SS algorithms used in this implementa-

tion. The sensing algorithms are explained with analytical models and the analytical performance of different sensing algorithms is also briefly discussed.

The third chapter introduces background information related to the real-time implementation, which provides and explanation of the concept of software defined radio system and basic information about NI's USRP hardware devices and NI LabVIEW platform.

The fourth chapter provides the implementation details of ED based SS. It provides the detailed information about how the transmitter and receiver are implemented, with graphs and instructions.

In the fifth chapter, the real-time test results of different sensing algorithms with different channel models are discussed and shown in figures. The measurement results are compared with simulation results providing SNR versus detection probability, in order to show the detection performance with different SNR values. The effects of different channel models are also discussed in the same chapter.

The last chapter provides a brief conclusion based on the measurements results provided in Chapter 5, as well as a discussion about possible future improvements of the implementation and future research topics.

2. COGNITIVE RADIO AND SPECTRUM SENSING ALGORITHMS

The usage of wireless communication and its applications has grown rapidly in last decade. Although there are great improvements in wireless communication's technical level, the problem of shortage of available spectrum is still a vital bottleneck for future development of wireless communication technologies. The allocation of spectrum currently used can only reach limited utilization of the spectrum as it uses fixed spectrum allocation for each application. In order to make use of the unoccupied spectrum, more creative and advanced methods are needed so that better spectrum utilization can be reached. The concept of CR techniques has been introduced [4] under this demand.

SS is one of the most important operations of CR devices, it informs the system about which part of the specified frequency band are not occupied by licensed primary users. Hence, unlicensed SUs may initiate communication using some of the available spectral holes. There are various spectrum sensing methods which use different mathematical models for different scenarios.

The purpose of this chapter is to introduce the general idea of CR systems and the SS methods used in this thesis.

2.1 Cognitive Radio

In order to solve the problem of spectrum shortage and also increase data transmission rate, the concept of CR has been proposed [4]. The definition of CR given by FCC is as follows "Cognitive radio: A radio or system that senses its operational electromagnetic environment and can dynamically and autonomously adjust its radio operating parameters to modify system operation, such as maximize throughput, mitigate interference, facilitate interoperability, access secondary markets." [4]. The concept of CR is now widely recognized and there is a considerably large amount of studies and research of CR systems. There are also several applications of CR, such as IEEE 802.22 Wireless Regional Area Network (WRAN) [13].

The most distinctive difference of CR system with traditional wireless communication system is its ability of identifying the presence or absence of unknown deterministic signals, and determines whether a PU is active or not based on the spectrum sensing process. The corresponding SU can either proceed to use the unoccupied spectrum or remain silent according to the decision regarding spectrum occupancy. By this way, the CR system can determine the spectral holes within a frequency band and allowing the unlicensed users to access those spectral holes opportunistically. This is an effective

way of spectrum utilization and a good solution for the current scarce spectrum resource problem [5], [6], [14]. The aim of CR is to achieve dynamic spectrum access instead of fixed spectrum allocation and having better spectrum utilization.

In order to achieve the basic idea of spectral reuse, which gives the unlicensed secondary user permission to operate over spectrum that is allocated to licensed primary user when the specified spectrum is not fully occupied [15], there are several tasks a CR device should fulfil. These tasks and operations are usually defined as the basic cognitive cycle, which can be seen in Figure 2.1. There are three basic cognitive tasks as seen in the figure [8].

1) Radio-scene analysis: The estimation of interface temperature of the radio environment and the detection of spectrum holes.
2) Channel identification: The estimation of channel-state information and the prediction of channel capacity.
3) Transmit-power control and dynamic spectrum management.

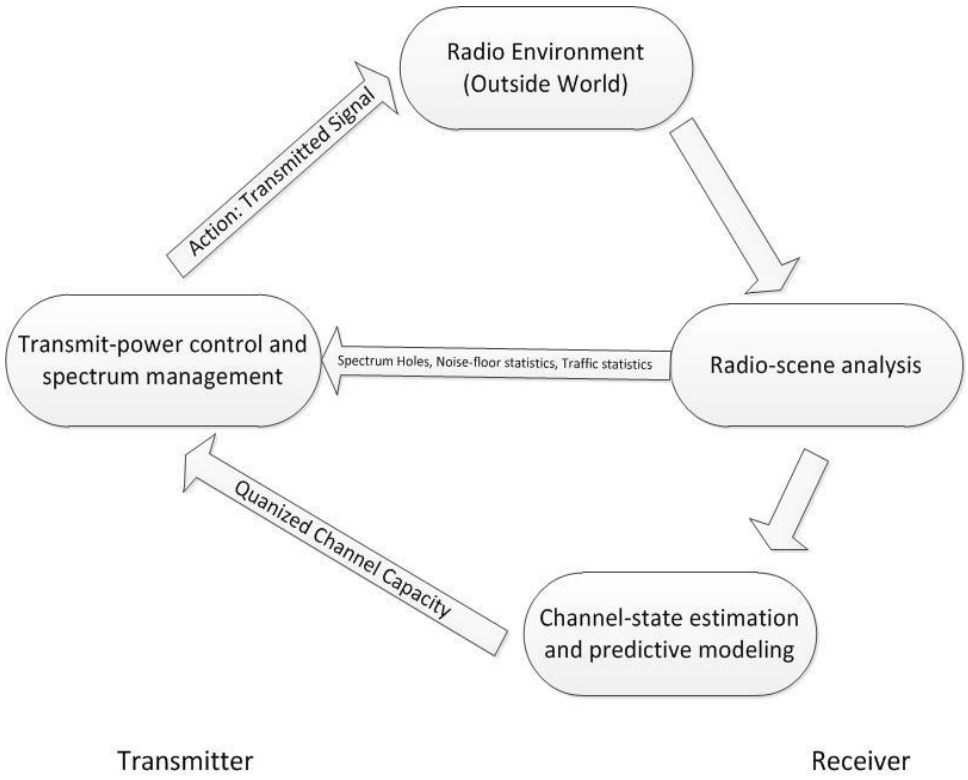


Figure 2.1 *Basic cognitive cycle [8].*

Among these three tasks, tasks 1 and 2 are carried out at the receiver side and task 3 is performed at the transmitter side. It is apparent from the cognitive cycle that within the CR system, the receiver and transmitter must work in harmony [8].

Correspondingly, there are also three basic operations of CR devices, namely SS, spectrum re-allocation and configuration of the transmission system. SS can be considered as the most important and challenging operation among these three operations, as the ability of finding spectral holes depends on spectrum sensing operation. After SS

process, the CR system determines the spectral holes which are unoccupied by licensed primary users in a certain time and geographical area and allocates these spectral holes to unlicensed secondary users. At the final step, the CR adjusts the transmission system parameters in order to achieve better performance. In this thesis, the focus is on the SS part.

2.2 Spectrum Sensing

The main duty of SS operations is to obtain awareness of the spectrum usage of licensed PUs in the specified frequency band. Unlicensed secondary users can decide whether there are spectral holes available or not based on the information provided by the SS operation. The frequency spectrum can be categorized into three different types according to the activity on sub-bands, namely white space, grey space and black space [7]. Unoccupied bands only contain noise and they are called white space. Partially occupied bands are called grey spaces and bands which are fully occupied are called black spaces. SS methods can be used to determine white and grey spaces; hence the unlicensed secondary users are able to use the specified spectrum if unoccupied.

The important task of SS operations is to prevent the primary user from being interrupted by secondary users. The PUs have absolute rights to use the assigned frequency band. The secondary users using CR technology are able to sense and occupy the spectral holes when the frequency spectrums are not occupied by primary users. It is very crucial to keep sensing for the presence of primary users and release the frequency spectrum occupied by secondary users immediately after the presence of primary user is sensed, hence the interruption can be avoided. The illustration of various aspects of the SS operations is shown in Figure 2.2.

The SS technique is still under developing, and there are several challenges and topics to be researched. Current SS techniques usually make use of time, frequency and space domains. In real-world scenario, the primary user signals are expected to be detected under very low SNR case. The mobile channel effects such as multipath fading and frequency selectivity can also introduce additional challenges.

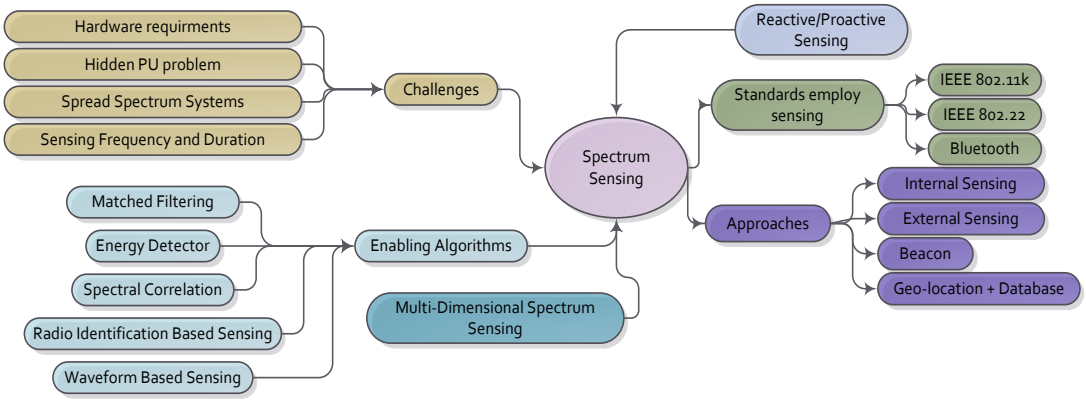


Figure 2.2 *Various aspects of spectrum sensing for CR [7].*

An effective SS method needs to have a good balance between sensing accuracy, sensing time, and the implementation complexity of the method. There are several different sensing methods with different emphasis on the balance of implementation complexity and accuracy. These sensing methods can be classified into three classes, namely blind, semi-blind and non-blind sensing methods [7]. Blind sensing methods such, as eigenvalue based and covariance based methods, require no information about PU signal or noise [9], [10], [11], [16]. Semi- blind sensing methods such as energy detector require only the information of noise variance (power) [15], [17], [18], [19], [20]. Non-blind sensing methods need prior information about the PU signal. It is obvious that different kind of sensing methods have different implementation complexity and different sensing performance.

Modified ED based spectrum sensing methods such as FFT&AFB based ED and Max-Min ED are studied and implemented using NI-LabVIEW and NI-USRP platform in this thesis.

2.3 Energy Detector Based Spectrum Sensing Algorithms

ED based SS algorithms are the most widely used sensing methods due to their low computational and implementation complexity compared to other sensing methods. Three different SS algorithms are introduced in this section.

2.3.1 Analytical Model of Spectrum Sensing Algorithms

In the analytical model, spectrum sensing is usually formulated as a binary hypothesis testing problem as follows:

$$\begin{aligned} H_0 &: y[n] = w[n] \\ H_1 &: y[n] = \overbrace{s[n] \otimes h[n]}^{x[n]} + w[n] \end{aligned} \tag{1}$$

where y[n] is the complex signal observed by the sensing receiver with $s[n]$, $h[n]$ and $w[n]$ denoting the primary user signal, the channel impulse response and the zero-mean, complex, circularly symmetric, wide-sense stationary additive white Gaussian noise (AWGN), respectively [5]. Under hypothesis H_0 the primary user is considered absent, and the received signal sample $y[n]$ contains only AGWN. On the contrary, the received signal sample $y[n]$ under hypothesis H_1 consists of the transmitted signal $x[n]$ after channel $h[n]$, together with AWGN $w[n]$. Based on this binary hypothesis test model, the test statistic for ED can be formulated as:

$$T(y) = \frac{1}{N} \sum_{n=0}^{N-1} \left| y[n] \right|^2 \tag{2}$$

where N denotes the length of the observation sequence.

The test statistic can be modelled by the Gaussian distribution [17]. According to this, the following formulation is straightforwardly deduced:

$$T(y)|_{H_0} \sim \mathcal{N}\left(\sigma_n^2, \frac{\sigma_n^4}{N}\right) \tag{3}$$

and

$$T(y)|_{H_1} \sim \mathcal{N}\left(\sigma_x^2 + \sigma_n^2, \frac{(\sigma_x^2 + \sigma_n^2)^2}{N}\right) \tag{4}$$

where σ_x^2 denotes the variance of the transmitted signal with possible fading and channel effects while transmission, σ_n^2 denotes the variance of AWGN. σ_x^2 and σ_n^2 are assumed to be statistically independent. SNR is denoted as $\gamma = \sigma_x^2 / \sigma_n^2$ .The corresponding false alarm probability P_{FA} and detection probability P_D can be expressed as:

$$P_{FA} = \Pr(T(y) > \lambda \mid H_0) = Q\left(\frac{\lambda - \sigma_n^2}{\sigma_n^2 / \sqrt{N}}\right) \tag{5}$$

and

$$P_D = \Pr(T(y) > \lambda \mid H_1) = Q\left(\frac{\lambda - \sigma_n^2(1+\gamma)}{\sigma_n^2(1+\gamma_s) / \sqrt{N}}\right), \tag{6}$$

respectively, where $Q(\cdot)$ denotes the Gaussian Q-function and λ is the predefined energy threshold. The value of λ is determined by the assumed noise variance σ_n^2, targeted false alarm probability P_{FA} and observation sequence N length as follows:

$$\lambda = \sigma_n^2\left(1 + \frac{Q^{-1}(P_{FA})}{\sqrt{N}}\right) \tag{7}$$

where $Q^{-1}(\cdot)$ denotes the inverse Gaussian Q-function.

2.3.2 Fast Fourier Transform and Analysis Filter Bank Based Algorithms

2.3.2.1 FFT and AFB Basics

In this section, the basic concept of FFT and AFB are introduced. The purpose of this section is to give the basic idea of the functionality of FFT and AFB used in the sensing algorithms considered in this thesis.

Fourier Transform is a mathematical transformation named after Joseph Fourier, it is widely used for transforming signals between time domain and frequency domain in the context of signal processing. The transformation is reversible and the transformation from frequency domain to time domain is called inverse Fourier transform [21]. Analogously, the discrete Fourier transform (DFT) used to convert a finite sequence of equally spaced samples from its time domain representation to the frequency domain. Fast Fourier transform (FFT) is an efficient algorithm to compute the DFT and its inverse is called inverse FFT (IFFT).

FFT is used to transform equally spaced samples of received signal from time domain to frequency domain in the FFT based ED sensing algorithm. The transformed samples will be used for further processes within the sensing algorithm.

In the context of signal processing, filter bank is an array of band-pass filters, it can be categorized into two types according to its function. AFB is used to separates the input signal samples into multiple sub-bands each carrying different frequency components. The other type is called synthesis filter bank (SFB), which means the reconstitution of a complete signal resulting from the filtering process. The basic block diagrams of AFB and SFB can be seen in Figure 2.3.

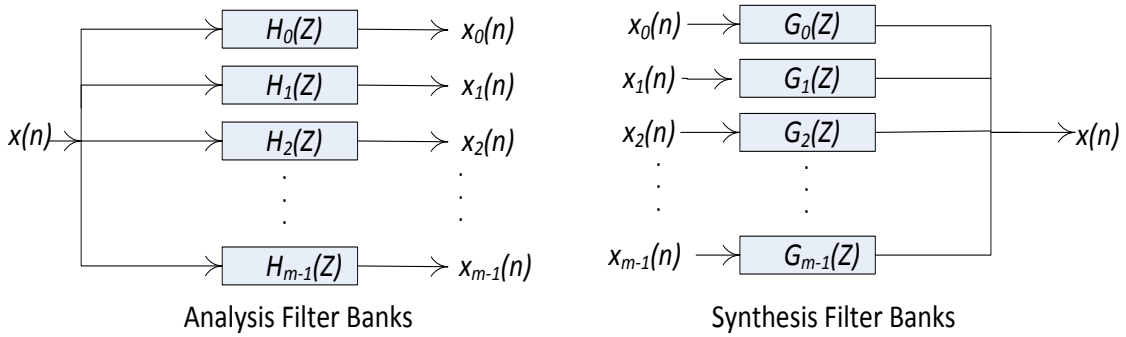


Figure 2.3 *AFB and SFB block diagrams.*

The process of AFB has the same functionality with FFT and the process of SFB has the same functionality of IFFT as seen in Figure 2.3. FFT and AFB are used for spectrum analysis purposes in the spectrum sensing algorithms. The processed data will be then used for further calculations of the sensing algorithms.

2.3.2.2 Details of the Algorithms

The basic block diagram of ED with AFB or FFT based spectrum analysis is seen in the following figure.

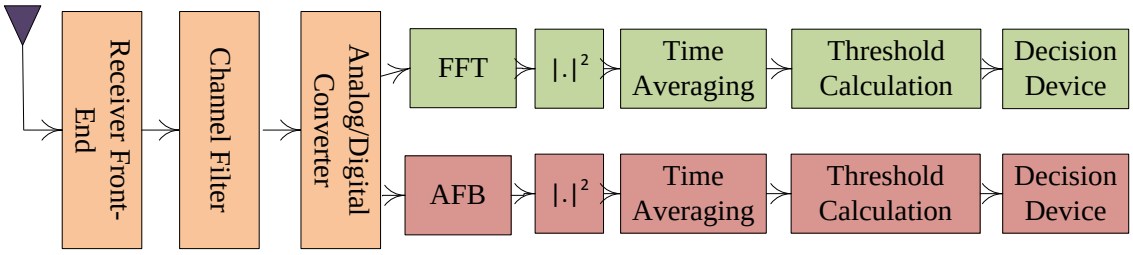


Figure 2.4 *Block diagram of energy detector with AFB &FFT based spectrum analysis.*

After the receiver front-end, channel filter and analog to digital converter (ADC), either FFT or AFB is employed to split the signal into comparatively narrow frequency bands as seen in Figure 2.4. The output data blocks can be expressed as *Y (m,k)*, where *k* is the subband index and *m* is the time index. Normally, it is assumed that the sampling rate used in each subband is equal to the ADC sampling rate divided by the number of

FFT bins or the number of subchannels used in the filter bank. In the spectrum sensing context, the subband signals can be expressed as [18]:

$$Y(m,k)=\begin{cases} W(m,k) & H_0 \\ S(m,k)H_k+W(m,k) & H_1 \end{cases} \tag{8}$$

where the transmitted PU signal, as seen in subband k of the m^{th} FFT or AFB processed block, is expressed as $S(m,k)$, and $W(m,k)$ is the corresponding channel noise sample when the signal is absent. H_0 and H_1 illustrate the absent hypothesis and present hypothesis of a PU, respectively. When there is only AWGN noise present, it is modelled as a zero-mean Gaussian random variable with variance σ_n^2 / N_{FFT}, where N_{FFT} is the number of FFT bins or subchannels, i.e., $W(m,k)=\mathcal{N}(0,\sigma_{n,k}^2)$ the subband noise variances are assumed to be equal, such that $\sigma_n^2 / N_{FFT}=\sigma_{n,k}^2$ [18]. The PU signal used in this thesis is simulated wireless microphone signals with the properties illustrated in IEEE suggestions [22], [23].

In ED based spectrum sensing algorithms, the absolute square of FFT or AFB processed data blocks are calculated to obtain the test statistics. The calculated result $\tilde{Y}_k$ is then compared with the pre-calculated threshold λ. The threshold value is calculated based on the assumed noise variance and the target false alarm probability. The detection probability P_D is the probability of the scenario that $\tilde{Y}_k$ is larger than the threshold λ, which means that the corresponding subband is considered to be occupied by a PU signal. The false alarm probability P_{FA} is the probability of the scenario that $\tilde{Y}_k$ is larger than threshold λ but there is no PU signal transmitted. The test statistics, the threshold λ, the detection probability P_D and the false alarm probability P_{FA} can be expressed as follows:

$$\tilde{Y}_k=\frac{1}{N_t}\sum_{l=1}^{N_t}|Y(l,k)|^2 \tag{9}$$

$$\lambda=\sigma_{n,k}^2\left(1+\frac{Q^{-1}(P_{FA})}{\sqrt{N_t}}\right) \tag{10}$$

$$P_D=\Pr(\tilde{Y}_k>\lambda\,|\,H_1) \tag{11}$$

$$P_{FA}=\Pr(\tilde{Y}_k>\lambda\,|\,H_0) \tag{12}$$

Here the high-rate observed sequence length is $N=N_{FFT}\cdot N_t$ samples.

The main advantage of ED based SS algorithm is its low computational and implementation complexity as seen in the above illustration. In contrast, the main limitation of ED based SS algorithm is its high dependence of the knowledge of noise variance. Noise variance uncertainty can cause apparent decrease to the sensing performance [24].

One possibility offered by the FFT or AFB based sensing method is to tune the sensing process to the expected bandwidths of the primary user signals, while keeping the

computational and implementation complexity low. This is achieved by using averaging windows both in time and frequency domains. In this thesis, algorithms with frequency domain averaging are also used for comparison. The algorithms with frequency domain averaging window are illustrated next.

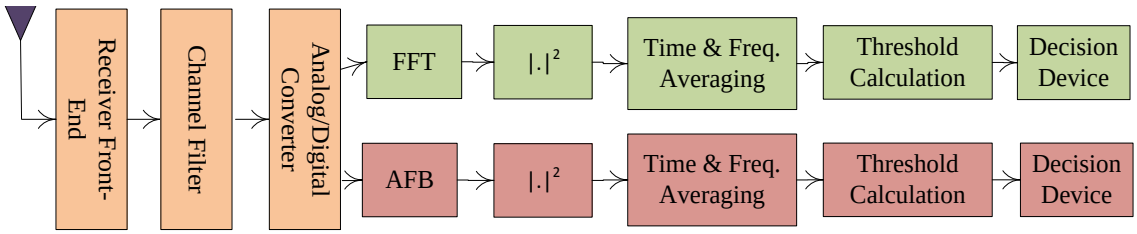


Figure 2.5 *Block diagram of energy detector with AFB & FFT based spectrum sensing with time- & frequency-domain averaging.*

The algorithm with averaging window is similar to the original sensing algorithm, the only difference is that an averaging window is employed after calculating the absolute square value of the FFT/AFB processed data blocks as seen in Figure 2.5. The other operation remains the same. For this case, the decision statistics at different frequencies can be expressed as:

$$\tilde{Y}(m,k) = \frac{1}{N_f N_t} \sum_{l=k-[N_f/2]}^{k+[N_f/2+1]} \sum_{u=1}^{N_t} |Y(u,l)|^2 \tag{13}$$

where N_f is the length of frequency averaging window (number of averaging points), $N_t = N / N_{FFT}$. The threshold λ, the false alarm probability P_{FA} and detection probability P_D can be expressed as:

$$\lambda = Q^{-1}(P_{FA})\sqrt{\sigma_{n,k}^4 / N_f N_t} + \sigma_{n,k}^2 \tag{14}$$

$$P_D = \Pr(\tilde{Y}_K > \lambda \mid H_1) = Q\left(\frac{\lambda - (\sigma_{n,k}^2 + \sigma_k^2)}{\sqrt{(\sigma_{n,k}^2 + \sigma_k^2)^2 / N_f N_t}}\right) \tag{15}$$

$$P_{FA} = \Pr(\tilde{Y}_K > \lambda \mid H_0) = Q\left(\frac{\lambda - \sigma_{n,k}^2}{\sqrt{\sigma_{n,k}^4 / N_f N_t}}\right) \tag{16}$$

It is recalled that Q(·) denotes the Gaussian Q-function, $Q^{-1}(\cdot)$ denotes the inverse Gaussian Q-function, $\sigma_{n,k}^2$ and σ_k^2 denotes the noise variance and PU signal variance respectively.

2.3.3 Max-Min Energy Based Sensing Algorithm

Another SS method that has been tested to be robust against the noise uncertainty with low SNR is referred as differential Max-Min ED [25], [16]. The decision statistics of this sensing method is based on statistics of the differential energy spectral density [25].

In this subsection, a simplified variant of subband energy statistics based spectrum sensing, referred to as Max-Min ED [16] is considered in details.

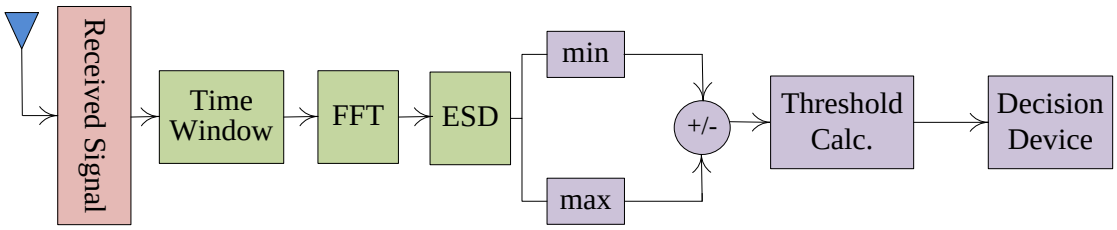


Figure 2.6 *Block diagram of Max-Min ED.*

The basic block diagram is illustrated in Figure 2.6, which can be separated into the following steps:

1. Estimating the energies across a number of subbands over the total sensing frequency band.
2. Observing the maximum and minimum value of the subband energies.
3. Comparing the processed test statistic with the selected threshold to make decisions.

The first step of the sensing process is to perform energy spectral density approximation, the process can be performed using FFT operations on rectangularly windowed sets. Same signal model is considered based on (8).

The energy detector process can be formulated using $T_k = \frac{1}{N_t}\sum_{m=1}^{N_t} |Y(m,k)|^2$, where $N_t = N / N_{FFT}$. With the aid of the central limit theorem, the following approximations can be made:

$$T_k = \begin{cases} \mathcal{N}(\sigma_{n,k}^2, \frac{2}{N_t}\sigma_{n,k}^4) & H_0 \\ \mathcal{N}(|H_k|^2\ \sigma_k^2 + \sigma_{n,k}^2, \frac{2}{N_t}(|H_k|^2\ \sigma_k^2 + \sigma_{n,k}^2)^2) & H_1 \end{cases} \tag{17}$$

The second step of the sensing process is to find the maximum and minimum values of the subband energies based on their magnitude.

The last step of the sensing algorithm is the threshold calculation and decision making. In this stage, the difference between the maximum value and minimum value is compared with the predetermined threshold to make decisions. The threshold can be calculated based on the target false alarm probability P_{FA} using the Neyman-Pearson test [25], [26], [16] with zero mean and double variance compared to basic energy detector. In this algorithm, when $D_{\max} - D_{\min} > \gamma$, the primary user signal is assumed to be present, on the contrary, when $D_{\max} - D_{\min} < \gamma$, it is assumed to be there is only noise in the interested band.

Besides the performance under noise variance uncertainty, the computational and implementation complexity is also one of the most crucial aspects of SS methods. The computational complexity has direct effects on the sensing time, the PU signal can be miss-detected if the computational complexity is too high [25]. The main complexity of the above sensing algorithm comes due to the following stages:

- N_{FFT}-point DFT introduces $O(N_{FFT}\log(N_{FFT}))$ operations.
- The evaluation of maximum and minimum energy values introduces $O(N_{FFT})$ complexity.

3. IMPLEMENTATION ENVIRONMENT

The aim of this thesis is to study the real-time sensing performance of three different ED based SS algorithms: FFT/AFB ED without frequency averaging, FFT/AFB ED with frequency averaging and Max-Min ED in a WM system working at the 2.4 GHz ISM band. In this implementation, the transmitter and receiver are working with real-time complex signals. In order to fulfil this requirement, Personal Computers (PCs), Signal Generator, NI-USRP, NI-LabVIEW and Matlab software platform [27] are used for the implementation.

The NI-USRP software defined radio (SDR) platform, paired with NI-LabVIEW software platform can be used as a powerful PC-hosted wireless communication system. NI-USRP is a software programmable radio transceiver with the ability to transmit and receive radio frequency (RF) signals across a real-time bandwidth up to 40 MHz [28]. With the assistance of NI-LabVIEW software platform's graphical programming environment and several built-in toolboxes for NI-USRP, the receiver and the transmitter can be implemented easily and work efficiently.

The main purpose of this chapter is to introduce the hardware and software used in the implementation of the SS system.

3.1 Software Defined Radio

SDR is the concept which implements radio transmitters and receivers with programmable software instead of implementing them completely using hardware [29]. This will provide the possibility of reconfiguring and adjusting the RF parameters in the software layer, so that the same radio hardware can work for different requirements.

At the transmitter side of a typical SDR system, the SDR software transforms the data into processed digital signals with the user's preferences and provides the digital signal to digital to analog converter (DAC). The DAC transform the digital signal into analog waveforms. Afterwards the analog waveform is transmitted through the antenna to the radio channel, or through cables for precise testing purposes. At the receiver side of the SDR system, the antenna acquires the received analog waveform which is then processed by the analog front end and provided to the ADC, which transforms the analog waveform into a digital signal. The digital signal then proceeds to the SDR software for further signal processing and recovering the data that has been transmitted.

The physical layer architecture of a typical current SDR system is seen in Figure 3.1. The antennas in this system can also be SDR controlled. When the antennas are SDR

controlled, the capability of the system will be increased due to its greater tunability [29].

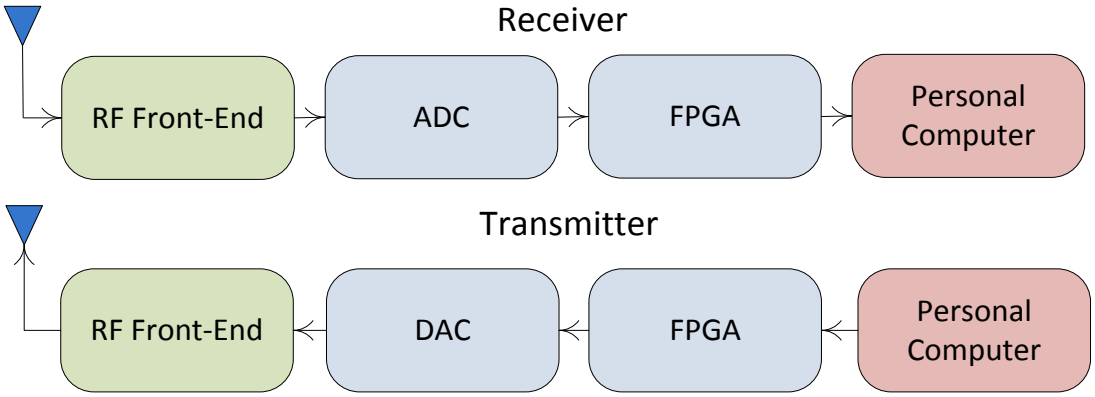


Figure 3.1 *Typical SDR system structure.*

In the above SDR system, the RF front end is in charge of get the signal from desired carrier frequency and transfers it to a specific IF or baseband at the receiver side, or shifts the baseband signal to a desired carrier frequency at the transmitter side. ADC/DAC is used to make transformations between analog and digital signal, while digital signals are used in Field Programmable Gate Array (FPGA) and PCs, analog signals are used in RF front end and antennas. FPGAs in the SDR are embedded with specific digital signal processing codes that can perform digital baseband and RF processes, such as up/down-conversion, channelization filtering and modulation. Main calculations, system monitoring and upper layer application like spectrum analysis, radio playback, etc., are performed by the PCs.

3.2 Universal Software Radio Peripheral

USRP is a widely used SDR platform which possess various abilities of SDR, such as wide range of bandwidth and great tunability. USRP connects real-time RF systems to PCs using Universal Serial Bus (USB) or Ethernet connections. USRP was outlined and designed by Matt Ettus, who is the founded Ettus Research LLC [30], a National Instruments' company. Second generation USRPs are able to work with GNU radio, NI-Labview and Simulink [31], [32] and [33]. The USRP series under National Instruments' brand is called NI-USRP and are paired with NI-Labview tool boxes, which is used in the implementation of this thesis.

3.2.1 NI USRP-2932

NI USRPs are SDR prototype platforms capable of numerous applications for education and research. The combination of NI's hardware and software offers flexibility and functionality for physical layer design, record and playback, signal intelligence, algorithm validation, and more [28] in affordable price.

NI-USRP products are categorised into two series, namely NI USRP-292x and NI USRP-293x. The NI USRP-293x series are integrated with GPS-Disciplined Clock which improves clock precision and enables global device synchronization and position [28]. Both 292x and 293x USRPs are connected to PCs using Ethernet connection working in pair with NI-Labview and both are able to perform multiple input, multiple output (MIMO) functionalities. In this thesis, the NI USRP-2932 model is used for the implementation.

Figure 3.2 *NI USRP-2932 [28]*.

NI USRP-2932 model is shown in Figure 3.2. There are two SMA signal ports RX1/TX1 (which can perform both transmission and reception) and RX2, an Ethernet connection port, a MIMO expansion port, and an external reference clock input port, a pulse per second reference input port, a power adapter port in the front panel, and a GPS antenna connection port at the back. The NI USRP-2932 model has a very wide frequency range from 400 MHz up to 4.4 GHz with an instantaneous Real-Time bandwidth of 40 MHz (with 8 bit samples) or 20 MHz (with 16 bit samples). When the USRP works as transmitter, the DAC rate is 400 MS/s for 16 bit samples with 2 channels, and the DAC Spurious Free Dynamic Range (SFDR) is 80 dB. When it works as receiver, the noise figure is around 5 to 7 dB and the ADC rate is 100 MS/s with a SFDR of 88 dB. The maximum input power at RX is 0 dBm. More detailed information about NI USRP specifications is given in Appendix A.

In terms of hardware structure of NI USRP-2932, as seen in Figure 3.3, the RF switches are used to select which port of RX/TX to use. The TX1/RX1 port can work for both transmit and receive and the RX2 port can be only used to receive. When it works as a receiver, the analog waveform is acquired from the selected RX port and fed to low noise amplifier (LNA) and drive amplifier, the amplified signal is then mixed with local oscillator (LO) in order to move the signal to baseband or IF from carrier frequency. The signals (real parts and imaginary parts) are filtered by low pass filters (LPF) with bandwidth of 20 MHz and then converted into digital signal by ADC and fed to a programmable digital down converter (DDC) to perform down-conversion, if needed, and finally the digital signal is transmitted to PC via 1 Gbit Ethernet for further processing. When working as a transmitter, the processed digital signal is first transmitted from PC to USRP hardware via Ethernet and then feed to digital up converter (DUC) for up-conversion, and then the digital signal is converted into analog signal by DAC. The analog signals are then filtered by 20 MHz LPF and mixed by local oscillator in order to transfer the signal form baseband to a specific carrier frequency and then amplified according to user's preferences by the TX amplifier. The last step is to transmit the signal through TX1 to antennas or cables (the TX/RX should be correctly selected in the setting of USRP otherwise errors will occur).

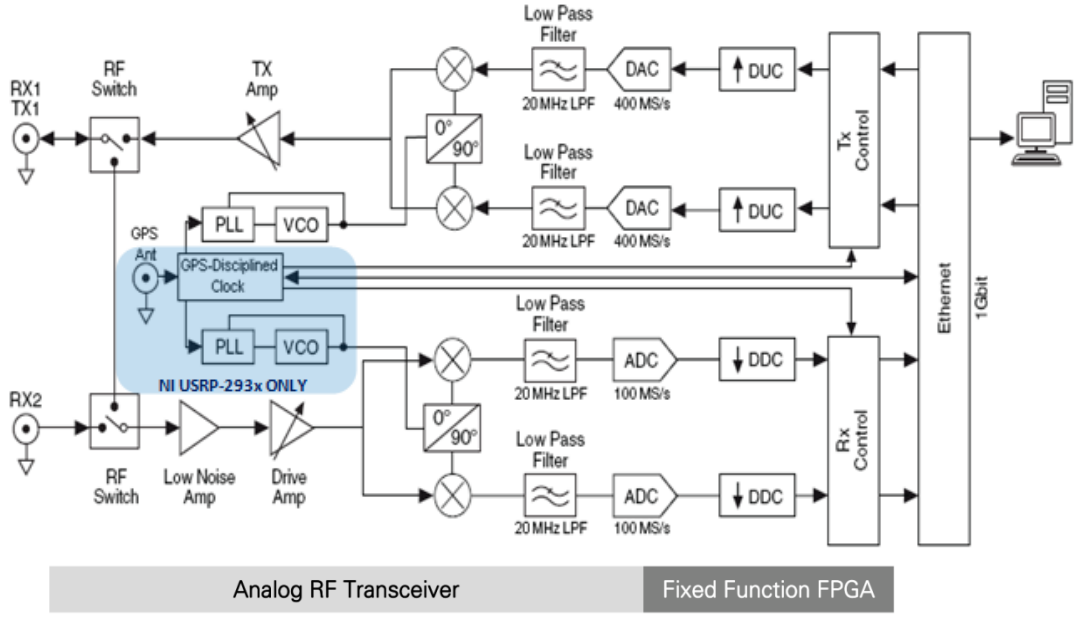


Figure 3.3 *NI USRP-2932 system block diagram [34].*

3.2.2 **Limitations of NI USRP**

Although NI USRP is a powerful and efficient model of SDR, there are some limitations in the implementation environment. Signal processing except up/down-converting and decimation are done by PC, thus a PC with powerful CPU and massive memory is required, especially when processing a wide frequency band [35].

There are two other major limitations for this model of USRP. While NI USRP-2932 is claimed to support 20 MHz bandwidth (sampling rate) real-time data acquiring, the filter effect is rather obvious when acquiring data at the maximum bandwidth (20 MHz), as seen in Figure 3.4. When comparing Figure 3.4 and Figure 3.5, it is obvious that the filter effects are more flat when the bandwidth decreased. Another major limitation is that this model of USRP working in pair with NI LabVIEW is very sensitive to LO Leakage and DC Offset, which introduced a very sharp peak at 0 Hz frequency, as seen in Figure 3.4. These two major limitations cause challenges to implement algorithms aimed for finding weak signal in wide frequency range, such as spectrum sensing. In order to overcome these limitations, some particular designs are developed, which are introduced in the next chapter.

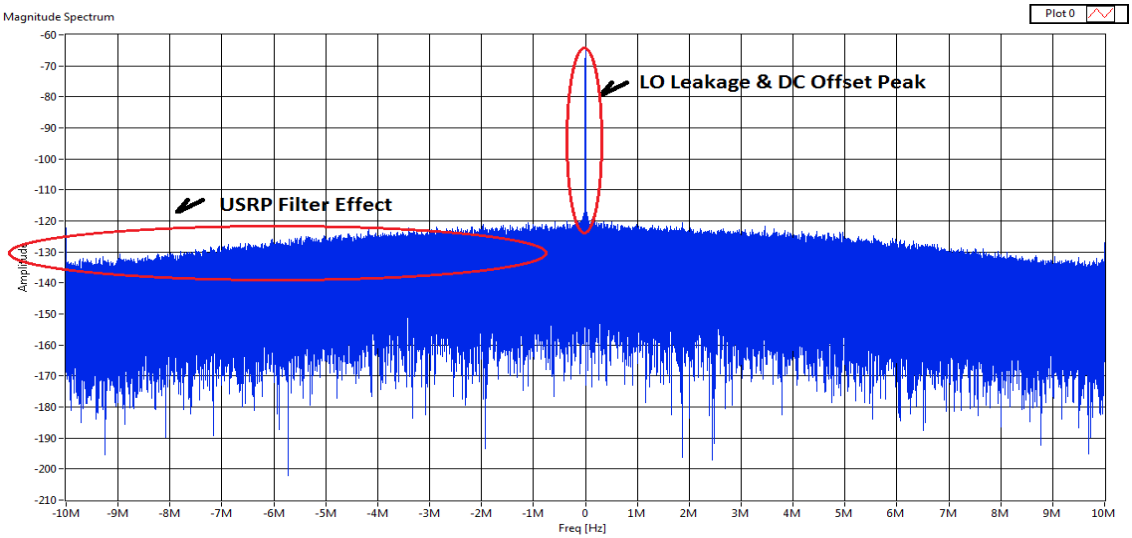


Figure 3.4 *Frequency spectrum observed by a USRP receiver with 20 MHz bandwidth when no signal is connected to the input.*

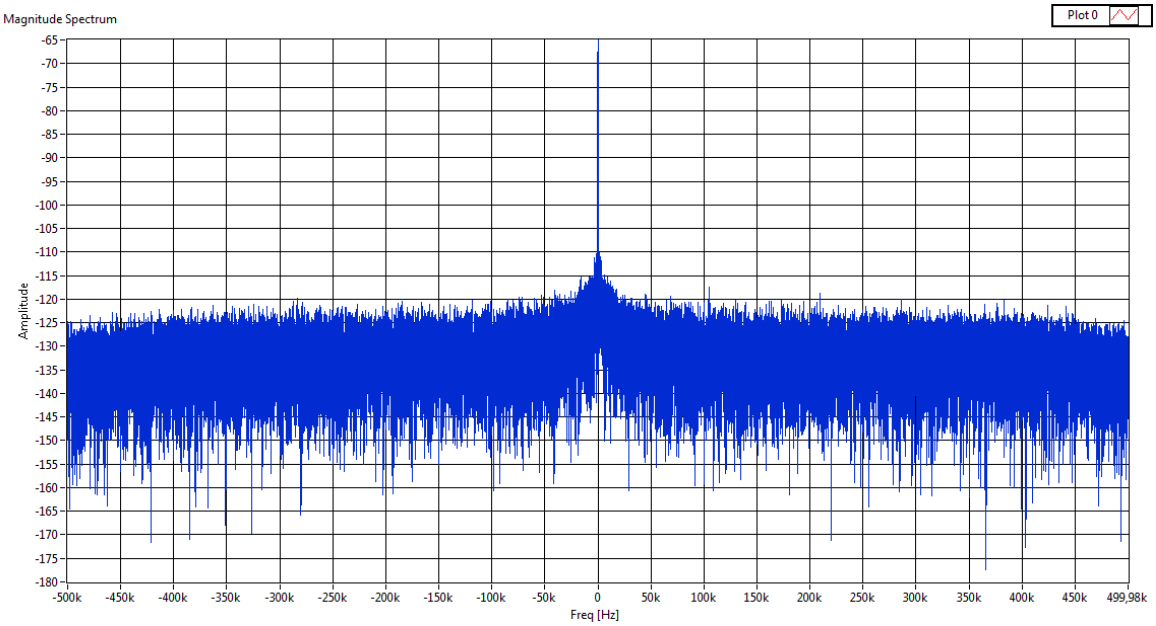


Figure 3.5 *Frequency spectrum observed by a USRP receiver with 1 MHz bandwidth when no signal is connected to the input.*

The NI USRP-2932 device has been tested to have a minimum transmission power at around -50 dBm, which is not low enough for this thesis' requirements while targeting to test SS algorithms for primary signals with very low SNR. Due to this reason, a signal generator R&S 100A has been used as transmitter instead of using a second NI USRP device.

3.3 National Instruments LabVIEW

NI LabVIEW is a graphical programming platform which is suitable for a variety control and measurement systems. It offers great integration with existing legacy software and hardware, which makes problem-solving and innovations faster and efficient [36].

3.3.1 LabVIEW basics

LabVIEW provides the concept of Virtual Instruments (VIs), every program written in LabVIEW appearears as a VI, which contains a front panel and a block diagram, as seen in Figure 3.6.

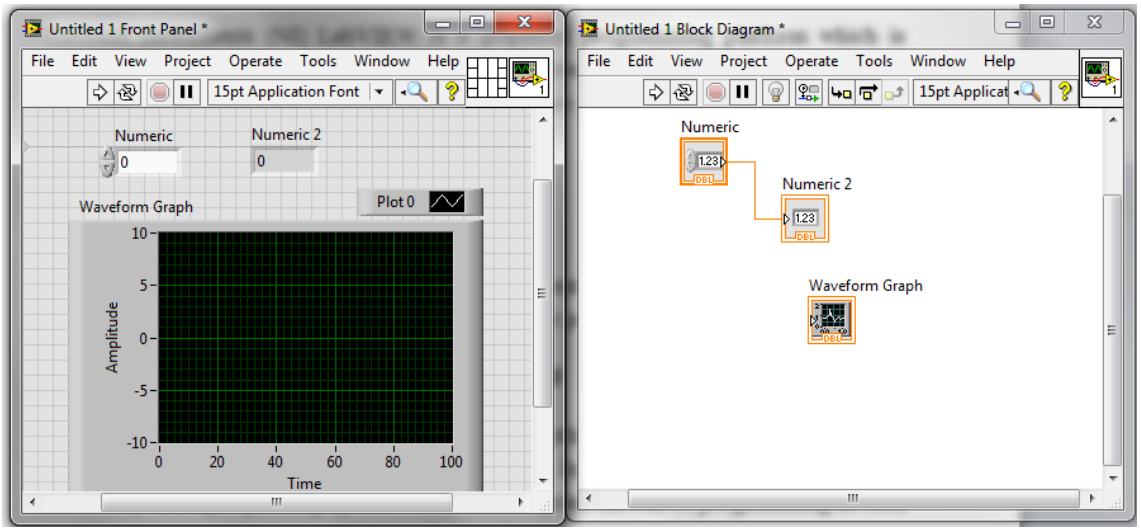


Figure 3.6 *Typical LabVIEW VI.*

The front panel of a VI has the same function as actual instrument's front panel; designers can add inputs like numeric inputs, switches, connections, etc., as well as indicators like graphs. The actual programming is done in the block diagram. When creating a user interface component in the front panel, a corresponding icon is also created in the block diagram. The design is done by connecting the icons together to have a block diagram of the specific design. There are also a great number of built-in functions and toolkits, provided by NI, which can be selected for different purposes, and they are very efficient to use while programming. More details of NI LabVIEW can be found on the National Instruments Webpage.

3.3.2 Benefits of programming in LabVIEW

One of LabVIEW's major differences from other general-purpose programming languages is its graphical, dataflow programming style. In LabVIEW, programming is performed by wiring together graphical icons on a diagram instead of programming in text, and those icons are executed according to the rules of data flow. The benefit of the graphical programing is that the users can spent more time on actual system design and focus on problem-solving, rather than spend significant time learning the specific text-based syntax of programming languages. There is a great amount of built-in functions for different purpose provided by National Instruments in different toolkits. The codes generated by the graphical programing language can run as effective as the traditional C codes [37].

LabVIEW supports smooth integration with existing software and hardware. LabVIEW contains components like Formula Node, MathScript Node, etc., which provide interfaces for C programming (Formula Node) and Matlab Script (MathScript Node). This expands the range of use of LabVIEW and saves a plenty of time for users, as there is a great number of existing C and Matlab codes for various purposes.

LabVIEW also provides a number of interactive debugging tools such as probes, highlighted execution and dataflow and immediate feedback indicating errors in the code. This makes debugging more easily and also provides monitoring methods while doing measurements and tests [37].

3.3.3 LabVIEW working with NI USRP

According to the advantages mentioned above, LabVIEW is used to control the NI USRP in our implementation. In order to enable LabVIEW to work with NI USRP, a few toolkits and drivers have to be installed, namely NI-USRP 1.2 Driver, LabVIEW Modulation Toolkit, LabVIEW MathScript RT Module, and LabVIEW Digital Filter Design Toolkit. These toolkits and drivers come with the USRP package and can also be found on National Instruments' websites. And more toolkits and drivers might be needed according to designers' particular requirements. More information about setting up the USRP and LabVIEW is given on National Instruments' webpage.

The basic control of NI USRP is very simple trough built-in functions as shown in Figure 3.7. It can be divided into three steps:

1. Configure the USRP properties such as sampling rate, carrier frequency, gain, etc., according to the designer's needs.

2. Start read (receiver) or write (transmitter) process to receive or transmit data based on the properties selected in first step. This step normally contains loops to receive or transmit continuously.

3. After completing the transmission and/or reception, the USRP connection has to be closed.

Figure 3.7 *Basic steps of controlling USRP via LabVIEW.*

There are eight built-in functions for USRP control in LabVIEW (NI-USRP driver needed), which are divided into three categories according to the steps mentioned above.

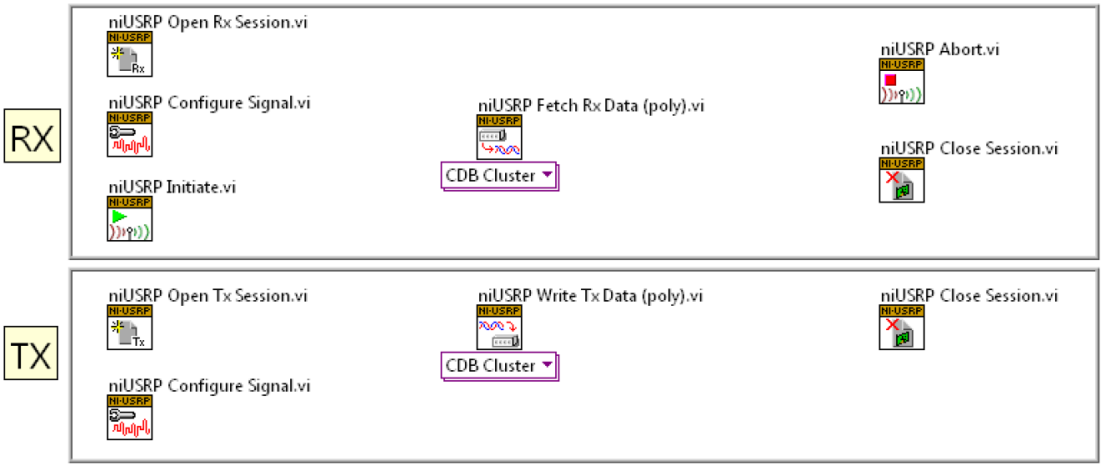


Figure 3.8 *The eight most-used NI-USRP functions [27].*

The 5 functions in the leftmost column are configure functions used to configure and initiate the USRP as seen in Figure 3.8. The two functions in the middle are used to read/write data from/to the USRP. Finally, the functions in the right most column are used to close the connections between PC and USRP

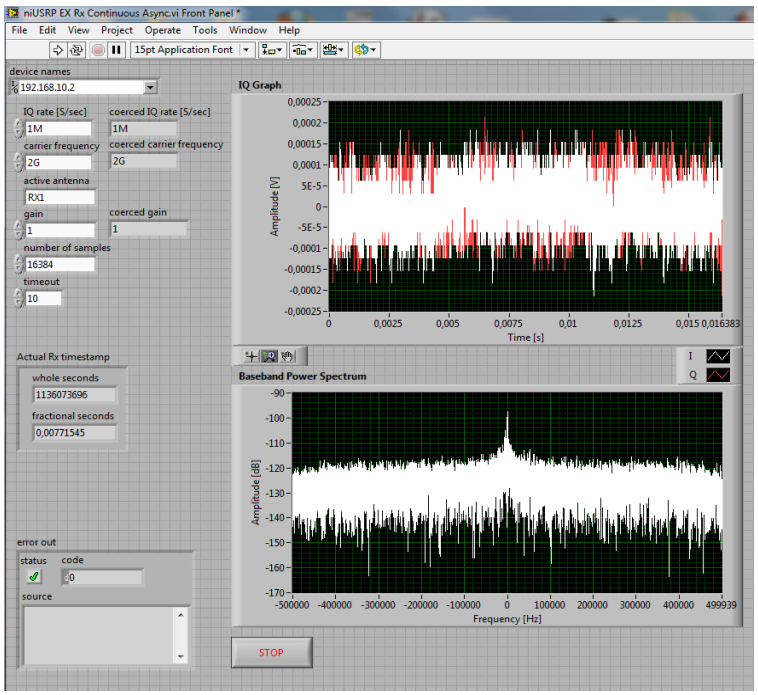


Figure 3.9 *Example front panel for USRP receiver.*

Figure 3.9 shows an example front panel of a single channel USRP receiver and Figure 3.10 shows the corresponding block diagram. As shown in Figure 3.9, the user has to select the device names, IQ sampling rate, carrier frequency, active antenna, gain and number of samples before clicking the start button. There are displays for the base-band power spectrum graph and IQ signal graph indicating the current receiving situation. There is also an error output, indicating whether there are errors or not. The selection of sampling rate and number of samples must be within the limitation of USRP and PC in order to avoid errors. The corresponding block diagram shows the actual design of the VI, such as how the components in the front panel are connected and the path and the data flow. The detailed design of the implementation is given in the next chapter.

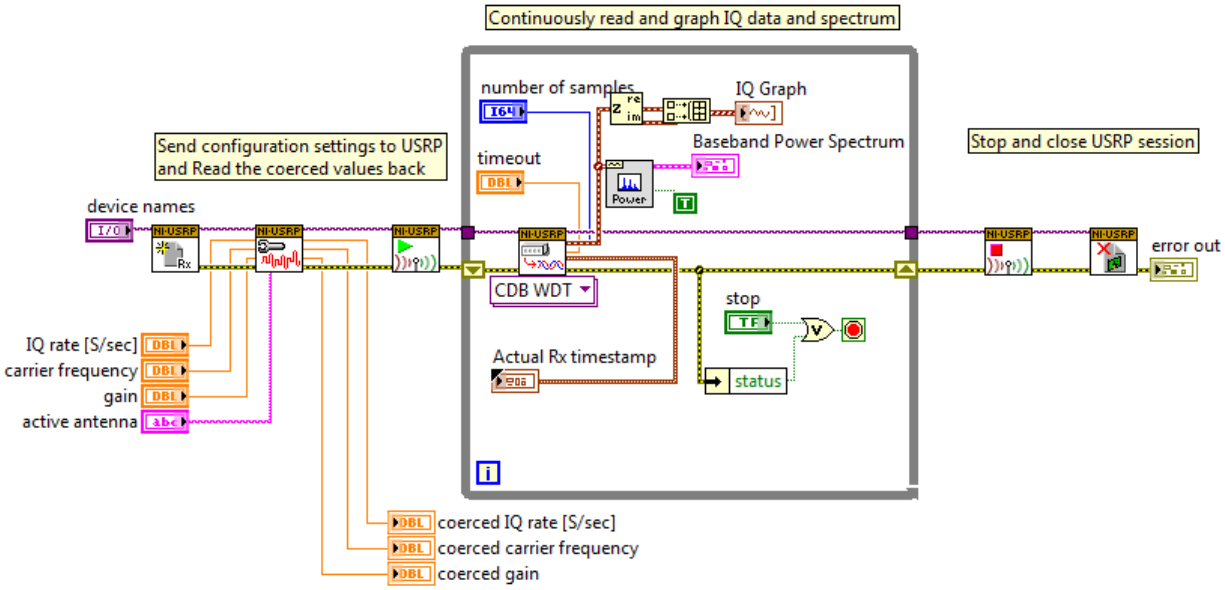


Figure 3.10 *Example block diagram for USRP receiver.*

4. DETAILS OF IMPLEMENTATION

This chapter covers the details of implementation for testing the SS algorithms used for this thesis. The system contains two parts as transmitter and receiver. As shown in Figure 4.1, the transmitter is implemented using the Rohde & Schwarz SMJ100A vector signal generator controlled by PC using MATLAB in order to control the SNR and add controllable channel effect to the system. The receiver is implemented using NI USRP-2932 and PC with LabVIEW and specific toolkits.

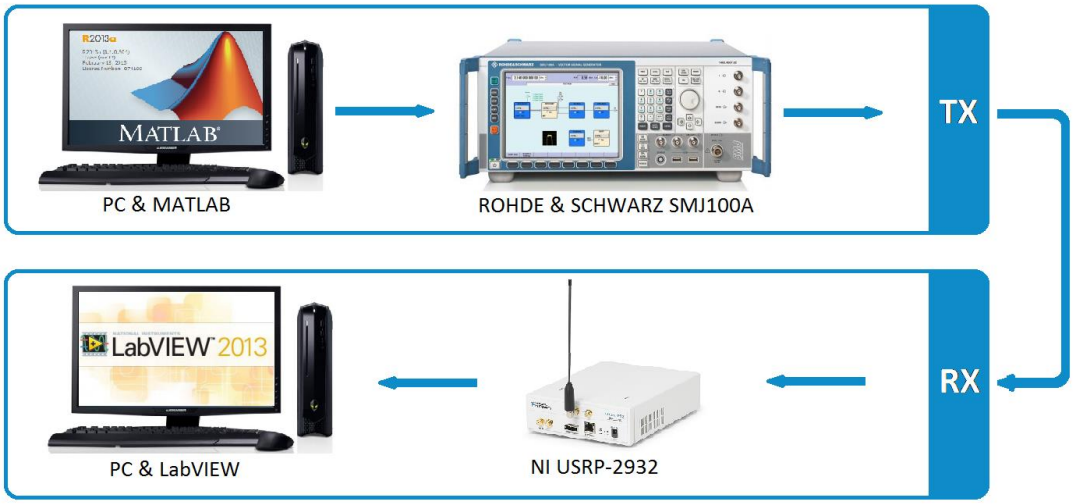


Figure 4.1 *System structure.*

4.1 Transmitter

4.1.1 Wireless Microphone

WMs are classified as licensed secondary users of the TV band. The operations of these devices are regulated by the FCC under Part 74 rules [2]. Most of the WM devices use frequency modulation (FM). The spectrum of WM signal is highly concentrated in the frequency domain. According to various signal models of WMs, the signal bandwidth is less than 200 kHz with a maximum transmission power of 50 mW [23].The spectral mask of WM is shown in Figure 4.2 which is based on European Telecommunications Standards Institute (ETSI) definitions.

Let $m(t)$ be the voice signal, then the FM modulated signal $s(t)$ can be generated according to the equation below:

$$s(t) = A_c \cos\left[2\pi f_c t + 2\pi k_f \int_0^t m(\tau) d\tau \right] \quad (18)$$

where A_c is the carrier amplitude, term f_c is the carrier frequency, and the constant k_f is the deviation factor of the modulator [38].

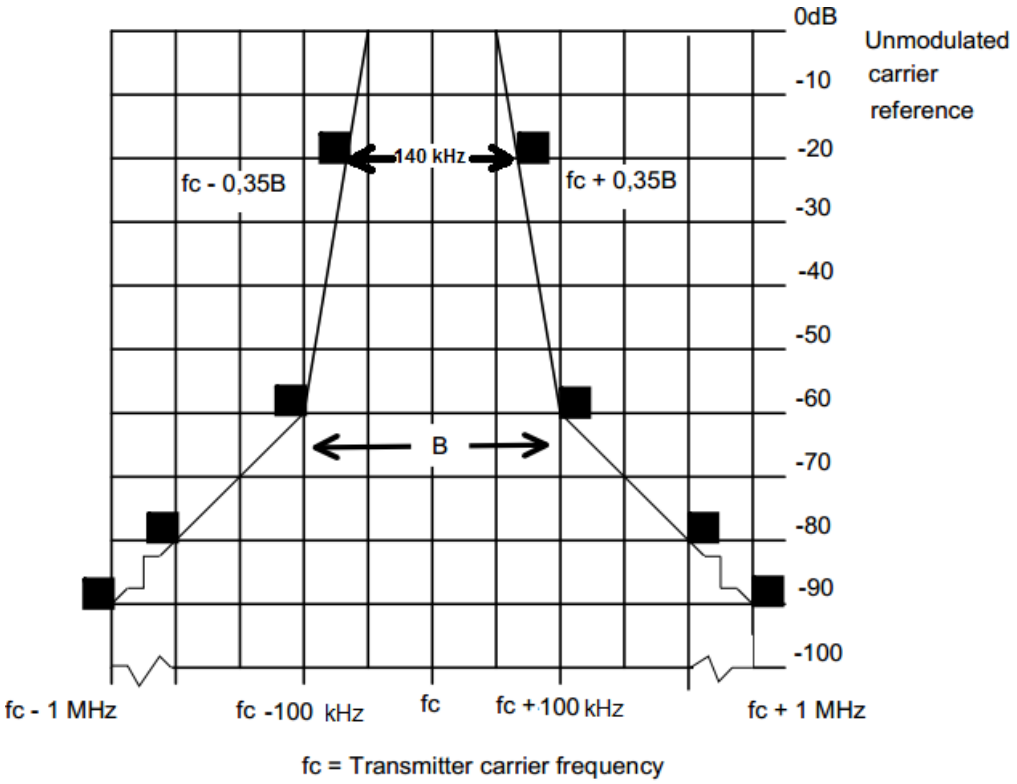


Figure 4.2 *Wireless microphone signal spectrum with ETSI spectrum mask [22].*

Three different WM signal models are suggested by IEEE 802.22 WRAN group for spectrum sensing algorithm tests [22]. They are categorized as follows:

1) Silent:
   The user is silent. In this scenario, $m(t)$ is a 32 kHz sinusoid signal and the FM deviation factor k_f is ± 5 kHz.
2) Soft speaker:
   The carrier of WM have a moderate amount of deviation. In this scenario, $m(t)$ is a 3.9 kHz sinusoid signal and the FM deviation factor k_f is ± 15 kHz.
3) Loud speaker:
   The carrier of WM have a deviation near the maximum amount. In this scenario, $m(t)$ is a 13.4 kHz sinusoid signal and the FM deviation factor k_f is ± 32.6 kHz.

In this thesis, the soft speaker WM signal model is used while it is the most general case of using WM among the three implementation models.

The WM signal is generated using the Rohde & Schwarz SMJ100A vector signal generator controlled by a PC using MATLAB. The MATLAB script is written according to the soft speaker model parameters given above.

Figure 4.3.a and Figure 4.3.b show the generated WM signal with 0 dB SNR and with -20 dB SNR, respectively. In these figures, bandwidth of noise is considered as 8 MHz which is also a bandwidth of DTV for all sensing bandwidth. The figures are captured using a Rohde & Schwarz SMJ spectrum analyser connected via cable to the signal generator.

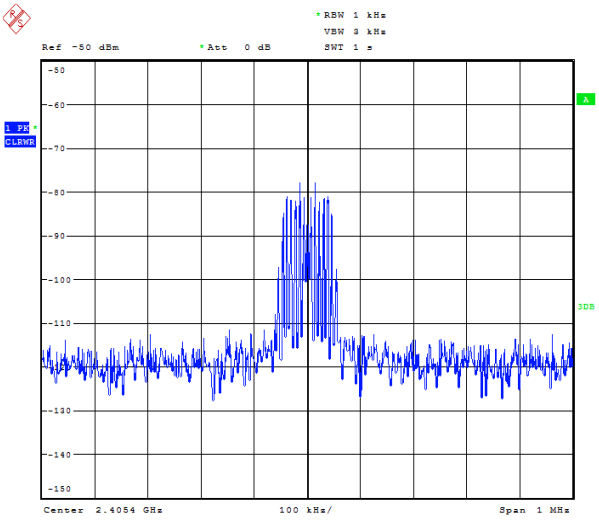


Figure 4.3.a *Generated WM Signal Spectrum (0 dB SNR).*

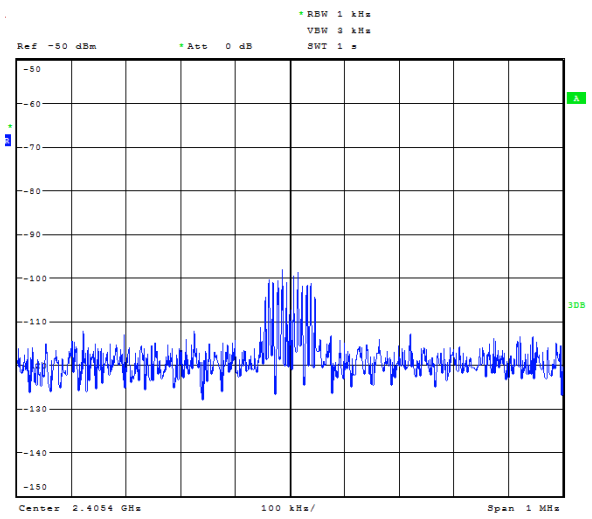


Figure 4.3.b *Generated WM Signal Spectrum (-20 dB SNR).*

It is clear that the spectral mask seen in Figure 4.2 and the practical WM signal as seen in Figure 4.3 have the same characteristic. It is also obvious from Figure 4.4 that even in very low SNR as -20 dB there are still apparent peaks of the WM signal.

4.1.2 Transmission Channel Model

The channel models refer to effects as delay, fading and path-loss caused by the transmission medium between transmitter and receiver. The characteristics of transmitted signal changes while propagating through the medium, depending on the distance and the path of transmission path's environment [39]. The mathematical model of the medium effect is called the transmission channel model.

Channel effects are combined to the original generated WM signal to make the received signal model similar to the real-world situation. In this work, three different channel models with different frequency selective characteristics are applied in order to investigate the SS performance under different selectivity effects. The used channel models are Indoor, Stanford University Interim 1 (SUI-1), and vehicular ITU-R Vehicular A channel models [39].

The Indoor channel model is an empirical channel model introduced in [40]. For realistic Indoor channel, the 16-tap channel model with 80 ns root mean square (RMS) delay spread as in [40] is used and the channel bandwidth is selected as the European terrestrial TV channel bandwidth of 8 MHz . This channel model has medium frequency

selectivity effects among all the three channel models we are considering [9]. Figure 4.4 shows the frequency spectrum of the WM signal under Indoor channel model.

The ITU-R channel model are commonly used as an empirical channel models and they are specified in ITU-R recommendation M.1225 [39]. In our study we apply the Vehicular A channel model with 6 delay taps and a maximum delay spread of about 2.5 μs. The frequency selectivity effective is the strongest among the three channel models [9]. Figure 4.5 shows the signal spectrum of WM signal under ITU-R channel model.

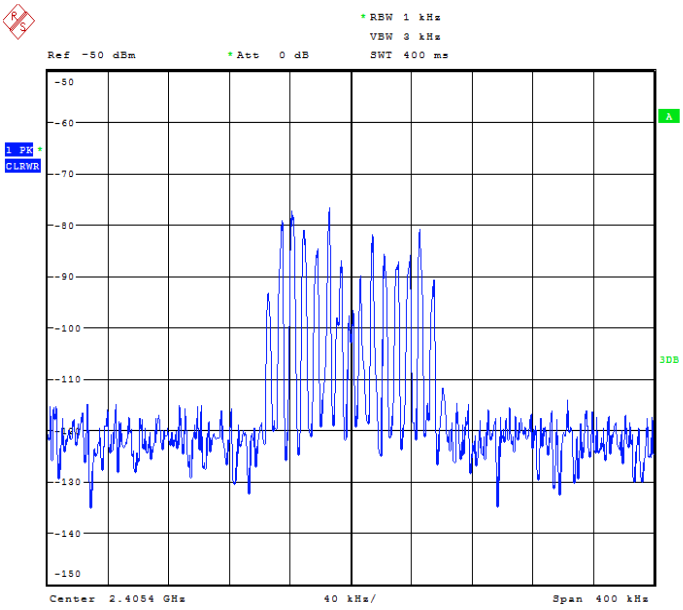


Figure 4.4 *WM signal spectrum under Indoor channel model.*

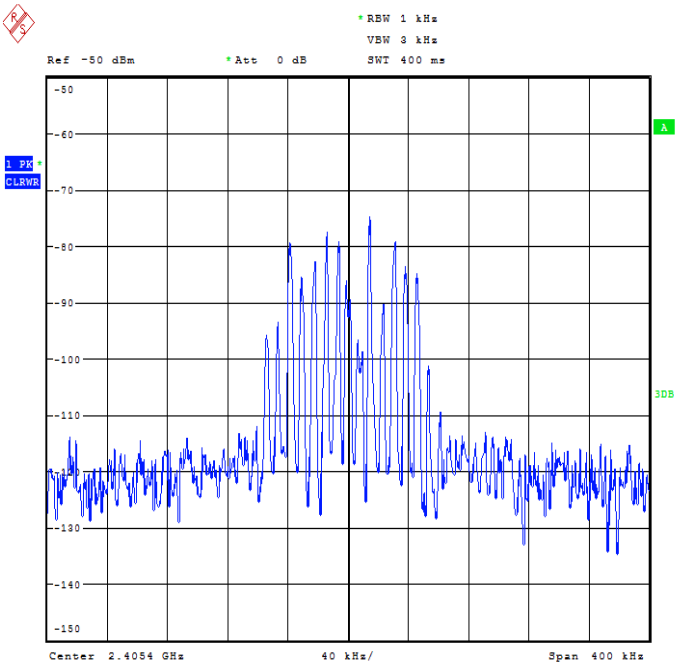


Figure 4.5 *WM signal spectrum under ITU-R channel model.*

SUI channel models consist of a set of 6 channel models with 3 different terrain types and a variety of Doppler spreads, delay spread and other channel conditions [39]. In this implementation we applied SUI-1 channel model, which has 3 Ricean fading taps and 0.9 μs delay spread, and it is the least frequency selective channel among the three channel models [9]. Figure 4.6 shows the WM signal spectrum under SUI-1 channel model.

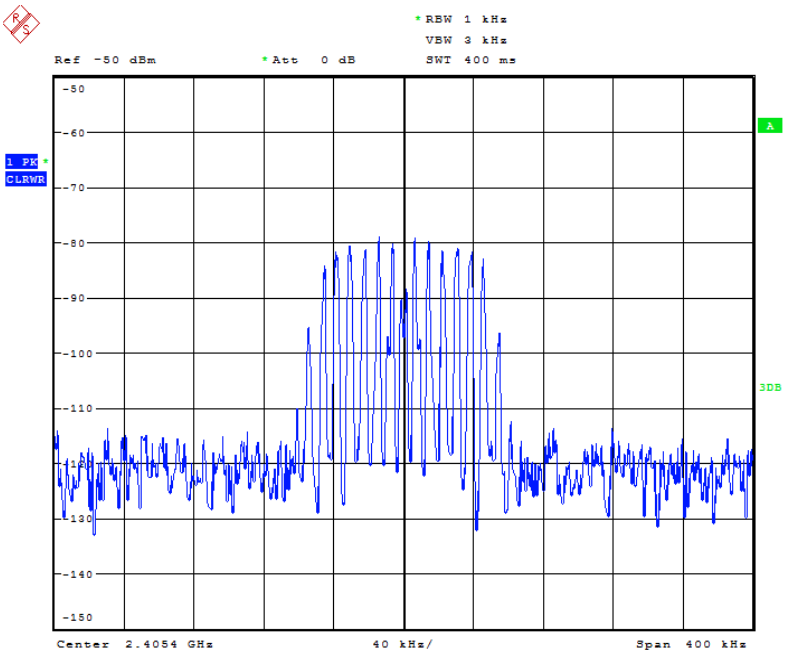


Figure 4.6 *WM signal spectrum under SUI-1 channel model.*

The channel models are generated in a MATLAB script, and the generated WM signal together with the selected channel model is transmitted to the vector signal generator to generate WM for further processing.

4.2 Receiver

The receiver part is implemented with NI USRP-2932 connected to a PC with LabVIEW via Ethernet. The programming and design are realized in LabVIEW in order to control the USRP hardware.

As introduced in Chapter 3, a LabVIEW program or VI contain two parts, the front panel and the block diagram. The front panel is the user interface, like the front panel of a real instrument, and the block diagram is the actual programming part. The details of the implementation are given in this section.

4.2.1 Receiver VI Front Panel

The front panel of the receiver VI is divided into two parts as seen in Figure 4.7. The left half is the configuration part, and the right part is used for monitoring the test states and displaying the test results.

Figure 4.7 *Receiver VI front panel.*

The left part of the front panel contains two main parts, the configuration of USRP hardware settings and the configuration of parameters for SS algorithms. As seen in Figure 4.8, the USRP hardware configuration block contains the settings such as device name, IQ sampling rate, carrier frequency, RX antenna selection and gain, and number of samples per catch.

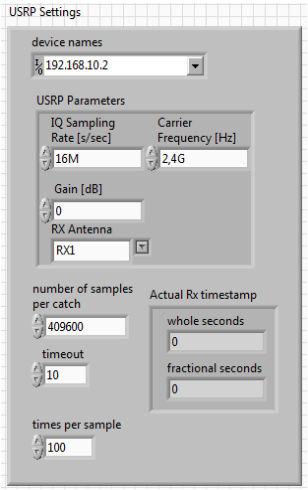


Figure 4.8 *USRP hardware configuration block.*

In this thesis, the bandwidth for spectrum sensing is 8 MHz. In order to avoid the LO leakage problem mentioned earlier, in the implementation, a total receiving bandwidth is first set to 16 MHz (f_{LO} – 8 MHz to f_{LO} + 8 MHz). Then only the 8 MHz positive frequency components (f_{LO} to f_{LO} + 8 MHz) is used. The number of samples per catch is selected with the same rule which is set to 409600 and only 204800 will be used for sensing algorithms. This process is illustrated in Figure 4.9.

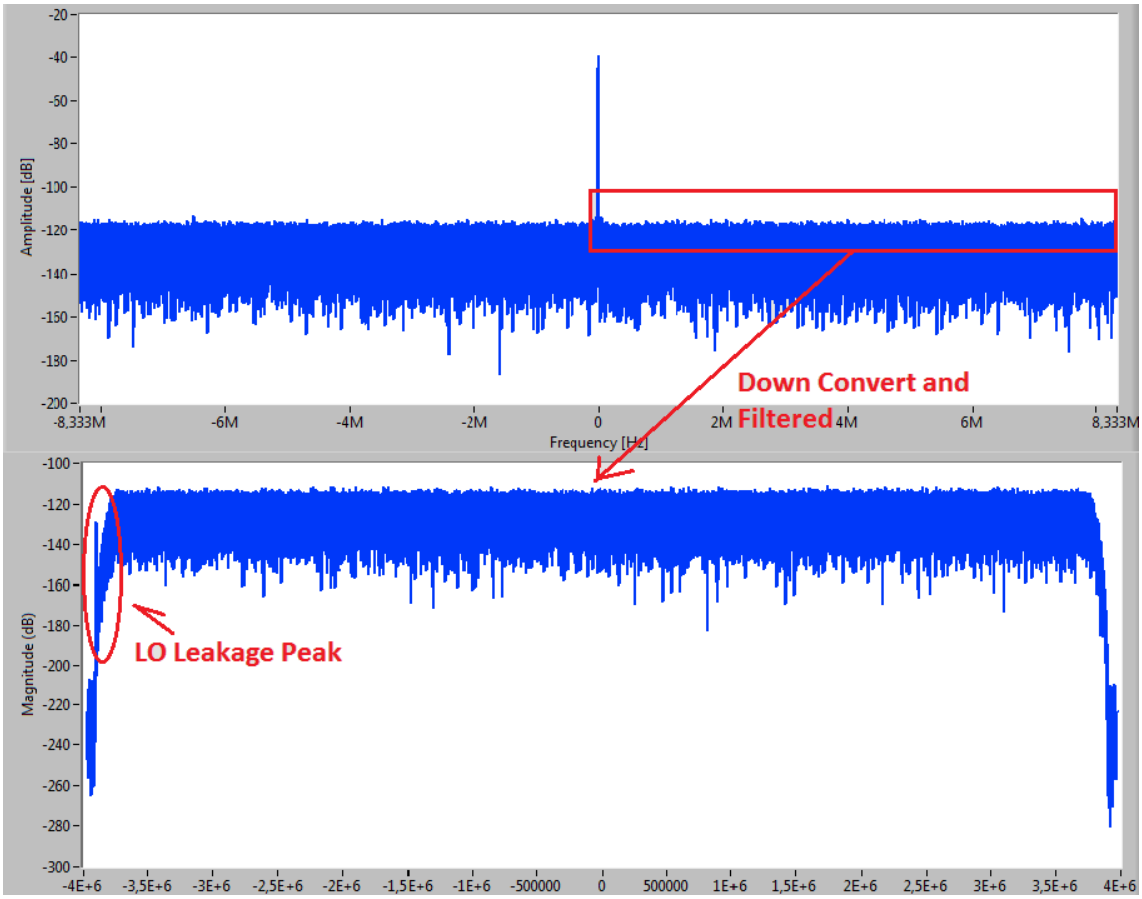


Figure 4.9 *Illustration of LO leakage removal.*

As shown in Figure 4.9, the positive (f_{LO} to f_{LO} + 8 MHz) part of the acquired frequency components are down-converted to baseband (0 LO frequency) and the LO leakage peak in the upper graph of Figure 4.7 isfiltered out by a half-band lowpass filter. It is also important to point out that these graphs are caught after the received signal (only noise is present in this illustration) passed through the whitening filter. Hence, the noise floor is observed as flat. The details of the implementation of filters will be discussed later.

The rest of USRP hardware settings are straightforward, such as the carrier frequency and the antenna gain. It is also very crucial to select the correct USRP name and RX antenna name and type to obtain correct results. Balance of precision and sensing speed has to be made for the selection of number of samples per catch.

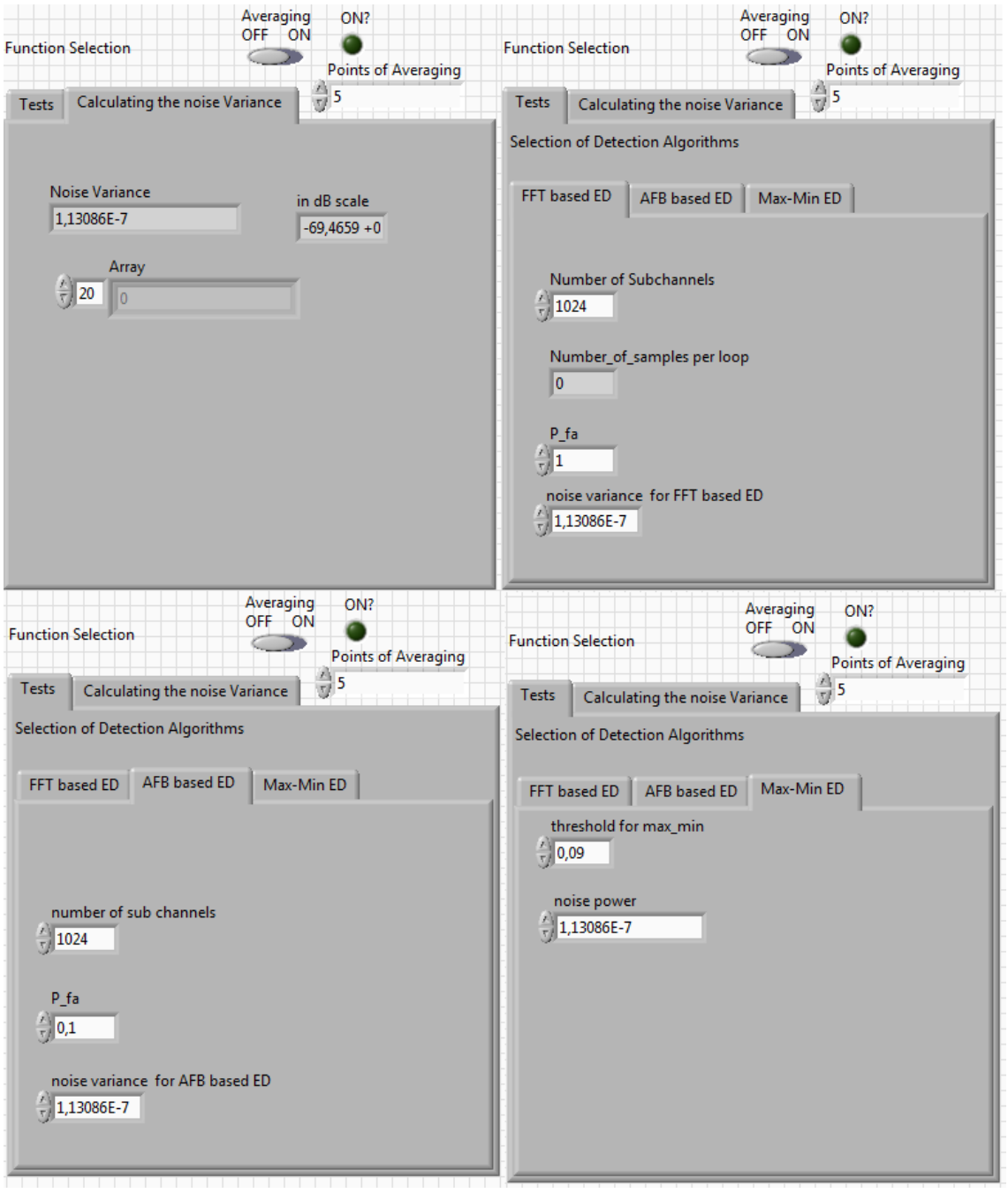


Figure 4.10 *Function selection block.*

The other part of the configuration block is the setting block of sensing algorithms, which is designed to select the process types to perform in the baseband signal processing.

There are two main categories in this block, the calculation of the noise variance and the actual spectrum sensing tests, as seen in Figure 4.10. The selection option of "Calculating the noise variance" is used for noise variance calculation. When executing the VI of this function, the TX should be turned off. Then the program will execute 50 loops and record the signal variance. When the execution is finished, the estimated average noise variance will be shown in the noise variance indicator and also in dB scale. The

number of loops is changeable from the block VI front panel, and the test results can be used as noise variance only if the TX has been turned off.

The option of "Test" executes the three sensing algorithms considered in this work. The FFT and AFB energy detection functions have similar settings, number of subchannels, false alarm probability P_{FA} and noise variance. The number of subchannels in FFT or AFB processsing is define for the sensing algorithms. The P_{FA} value and noise variance are used to calculate the thresholds for the sensing algorithms. Another important control for FFT and AFB detector is the averaging control; it is a switch button on the top of the block. If the averaging function is selected, the FFT and AFB sensing algorithms are performed using averaging window with the length that specified by "points of averaging" control. In Max-Min ED function, a threshold is manually selected instead of calculating from P_{FA} and noise variance.

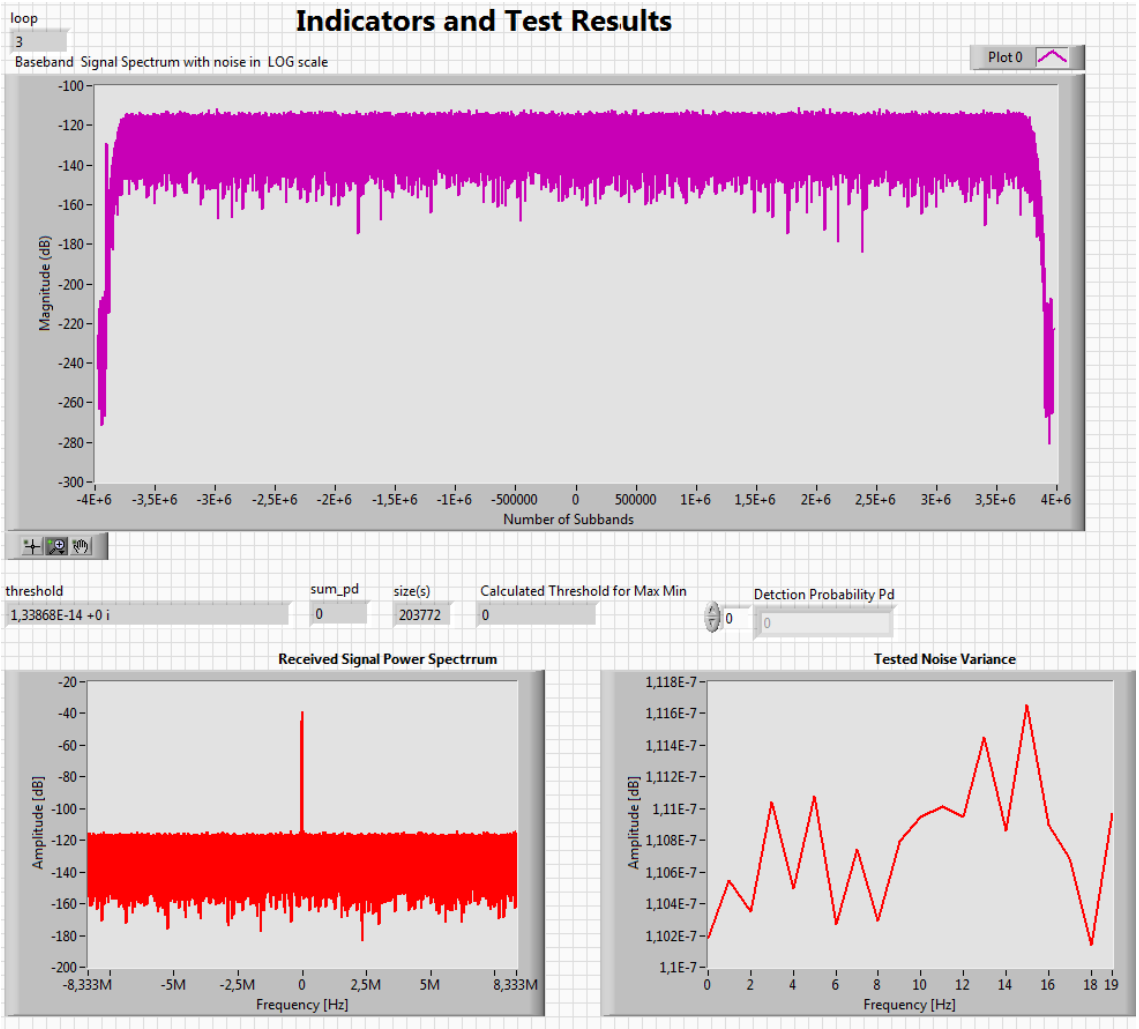


Figure 4.11 *Right half of the receiver VI front panel.*

As seen in Figure 4.11 the right half of the VI's front panel is used for monitoring and displaying the results. There are three graphs, indicating the received signal spectrum with 16 MHz bandwidth for each loop, the baseband signal spectrum with 8 MHz

bandwidth for each loop and the variation of noise variance for the noise variance calculation function. Other indicators for sample size, threshold and the detection probability P_D can also be found in this part of the front panel.

4.2.2 Receiver VI Block Diagram

The actual programming for the receiver implementation is done in the LabVIEW block diagram. The idea is to run the algorithms for a large number of loops in order to get reliable test results. As there are three different algorithms used in this thesis, a condition block should also be included. The following Figure 4.12 shows the flow chart of a single loop of this implementation. The actual LabVIEW VI block diagram is included in Appendix B due to its size.

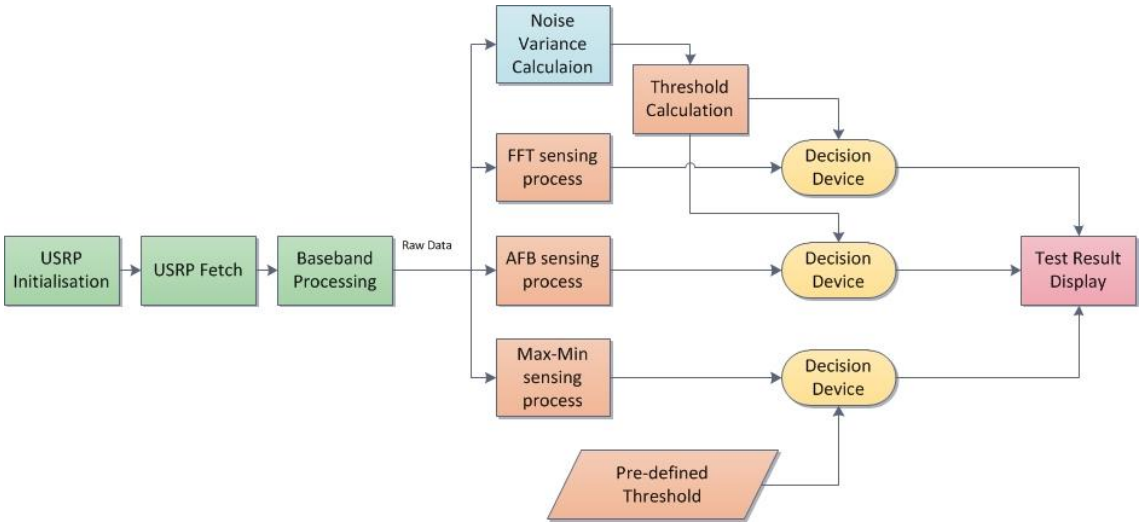


Figure 4.12 *Flow chart of a single receiver loop.*

As shown in the above Figure 4.12, the whole implementation can be divided into four main sub-blocks:

1. USRP operations and base band signal processing: The receiving parameters for USRP are selected, and baseband signal processing such as filtering and down-converting is also employed if needed.
2. Calculations of sensing algorithms and noise: Three different sensing algorithms are implemented in three different sub-blocks, the user can choose different algorithm to perform. The process of noise variance calculation can also be selected. When using FFT or AFB based ED sensing, the threshold is calculated using the estimated noise variance and target false alarm probability. When using Max-Min based ED, the threshold is pre-defined.
3. Decision making: This block is used for making decisions and depends on the data from previous blocks and the threshold.
4. Test results: The final step is to display the test results.

Sub-blocks are used to make more efficient and simple implementation in this study. Complicated functions can be grouped together, forming an icon with several inputs and outputs. This is called a sub-VI in LabVIEW, which can make the program more understandable and readable.

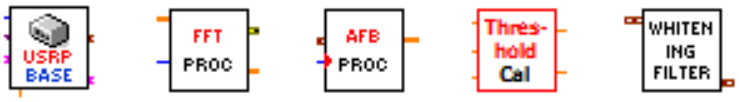


Figure 4.13 *Sub-VIs in implementation.*

In this chapter, most of the figures are screen shots of some particular part of the LabVIEW VI's block diagram to show the details due to the oversize of the actual block diagram.

4.2.2.1 USRP control and baseband signal processing block

The first step of the whole sensing implementation is to get raw data using NI USRP devices. As mentioned earlier, the control of USRP devices using LabVIEW is done by employ several built-in sub-VIs from the toolkits.

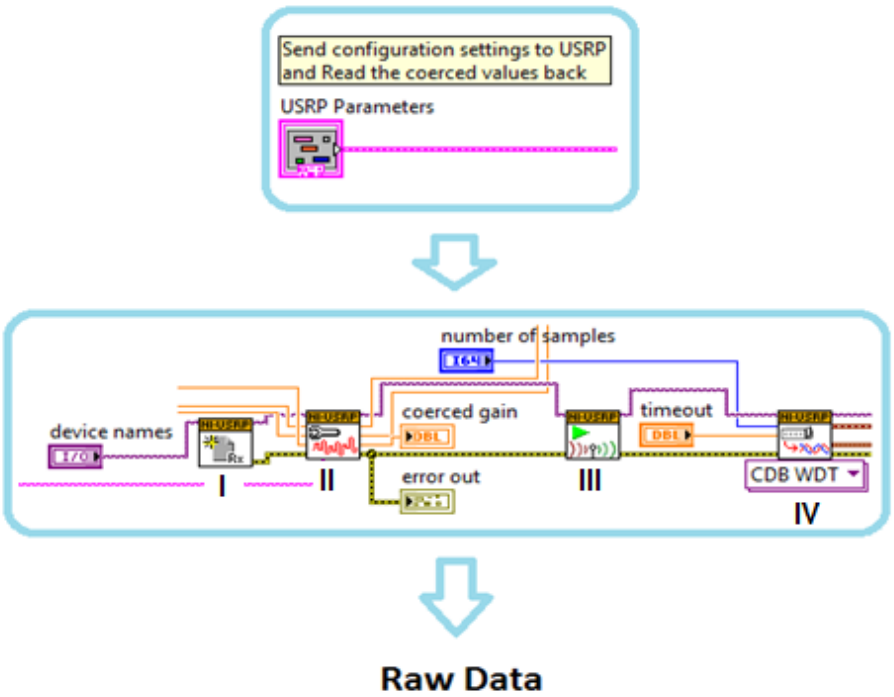


Figure 4.14 *USRP control block.*

The block diagram of USRP control block is shown in Figure 4.14. The upper part is an interface between the front panel and the block diagram. It receives the USRP parameters provided by user from the front panel and sends them to the built-in USRP control functions. The lower part shows the built-in USRP control functions and the connections in the implementation. The USRP device name is selected and sent to NI-USRP RX Open Session function (Ⅰ), this function is used to start connections between LabVIEW VI and the USRP hardware, and this function also creates a handle of the USRP device. All the USRP control functions connected with this handle operate the same USRP device. Then the user specified USRP parameters are sent to USRP Configure Signal function and USRP Initiative function (Ⅱ and Ⅲ), these two functions are used to initialize the USRP device with the user specified parameters such as carrier frequency, I/Q sampling rate, etc. The final function is the USRP RX Fetch function (Ⅳ), this function is the function actually fetching data from the receiver antenna. The received data is then transmitted as the raw data. It is obvious that the data flow in LabVIEW block diagram is from left to the right, and the functions on the left executes first, as seen in Figure 4.14.

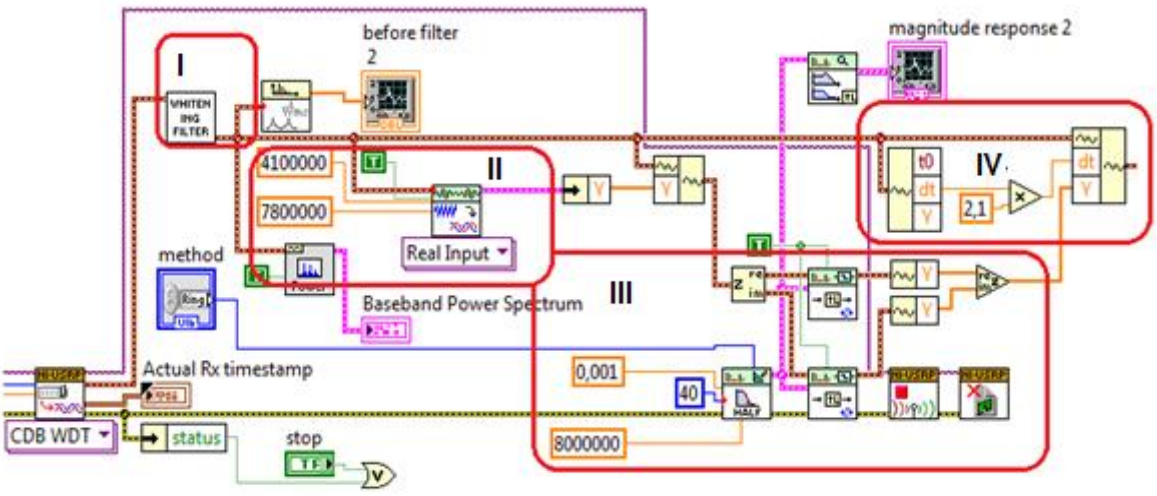


Figure 4.15 *Block diagram of baseband signal processing.*

As seen in Figure 4.15, the raw data from NI-USRP RX Fetch function is then sent to a series of baseband signal processing blocks (Ⅰ, Ⅱ, Ⅲ, Ⅳ). The first block is whitening filter, as the filter effect from the USRP devices is very significant, as mentioned before, a whitening filter (Ⅰ) is employed to make the noise floor flat. The frequency response of the designed whitening filter is shown in Figure 4.16 a. The signal is then down-converted according to the specified parameters (Ⅱ) in order to overcome the LO leakage peak problem mentioned before. The third step is half-band lowpass filter process (Ⅲ). As seen from the figure, the first function is a filter design function for half-band lowpass filter and after that there are two filter processing blocks, one for real part of the signal and another one for the imaginary part of the signal. The processed signals

are then merged again to form the processed complex signal. The final step in this part is down-sampling (Ⅳ), the signal is down sampled by two. The USRP stop and USRP close session function are seen in this figure.

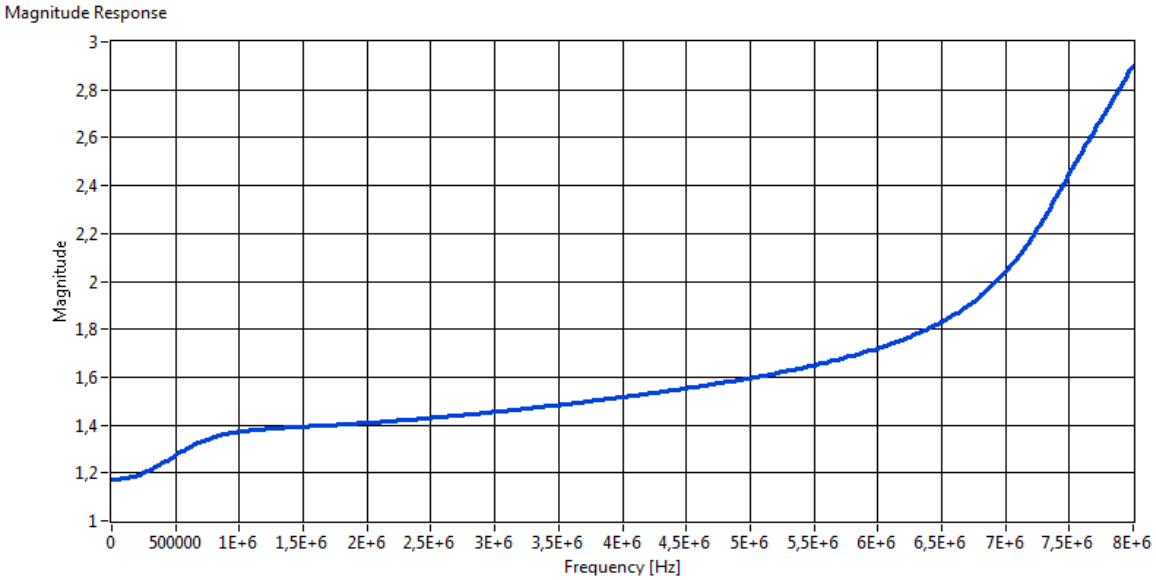


Figure 4.16.a *Frequency response of whitening filter.*

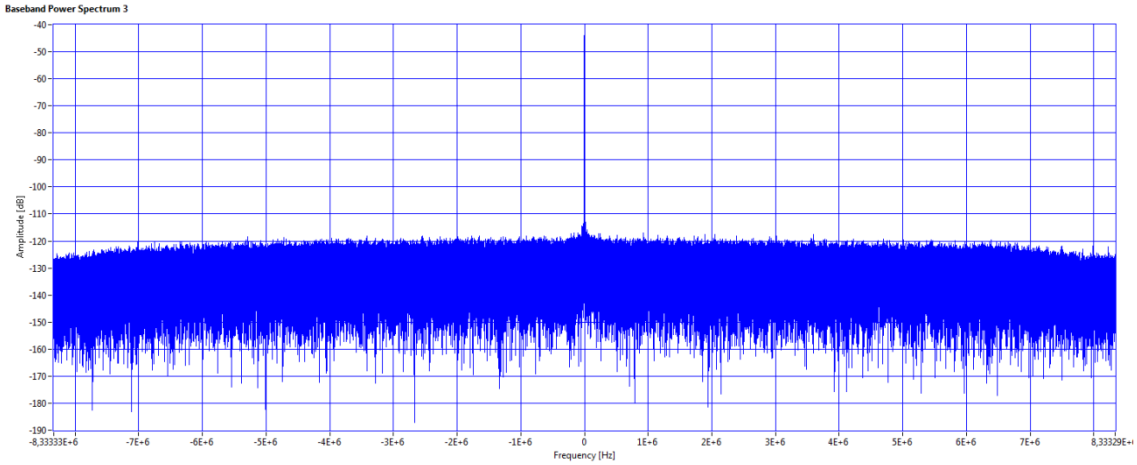


Figure 4.16.b *Signal spectrum before baseband signal processing.*

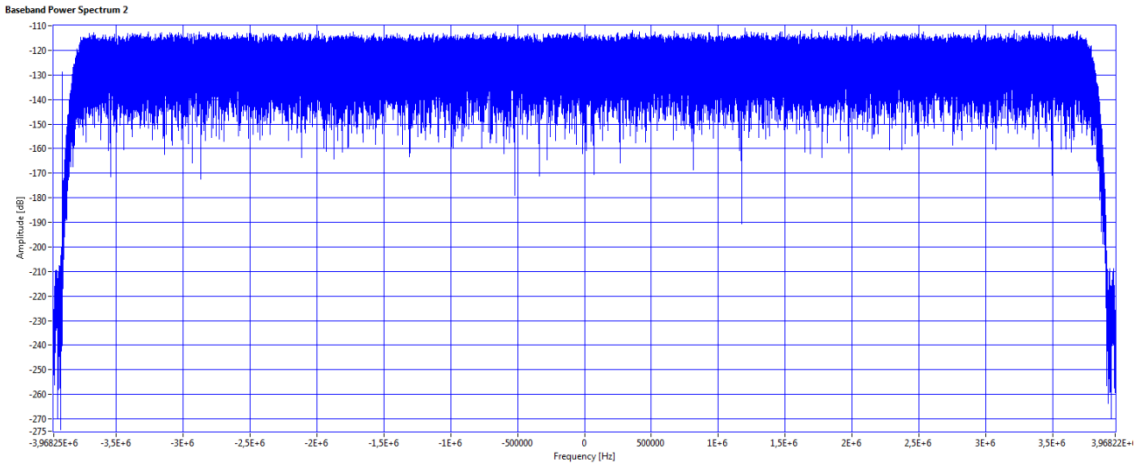


Figure 4.16.c *Signal spectrum after baseband signal processing.*

Before baseband signal processing operations, the signal spectrum is shown in Figure 4.16.b whereas Figure 4.16.c shows the signal spectrum after baseband signal processing operations. It is obvious that the signal spectrum is much flatter and the problem of LO leakage is solved.

The calculation of noise variance and threshold are much easier compared to the other processes. To calculate the noise variance, it is assumed that there are no PU signals in the specified spectrum range and the noise variance calculation function simply calculates the variance of the data from output of the previous baseband signal processing sub-VI. In order to have reliable results, the process is executed for 50 loops (changeable) and the average value is calculated for the final result. After the noise variance is estimated, the threshold for FFT and AFB based ED is calculated based on the estimated noise variance and target false alarm probability according to equation (10) and equation (14) in Chapter 2. As shown in Figure 4.17, a built-in function is used for the calculation of inverse Q-function, and a Matlab script block is used for further calculations. There are two different algorithms which can be applied without frequency averaging and with frequency averaging. The inputs of this sub-VI are the target false alarm probability P_{FA}, number of samples and estimated noise variance. The outputs are the calculated threshold without frequency averaging and with frequency averaging.

Figure 4.17 *Threshold calculation sub-VI.*

The threshold calculation function works in parallel with either FFT or AFB based ED sensing sub-VI according to the user's selection. The implementation of FFT based ED sensing algorithm is shown in Figure 4.18. As seen in the figure, it is implemented mainly using the LabVIEW built-in functions, and the scaling operation is performed using a Matlab script block. The input of this sub-VI is the data sequence provided by the previous baseband signal processing block and the output of this sub-VI is FFT sensing algorithm processed data for comparison with the threshold. As seen in equation (9) in Chapter 2, the algorithm first performs FFT operation to the data and then calculates the absolute square value of the FFT processed data, and a possible scaling operation is also needed to make the algorithm working correctly.

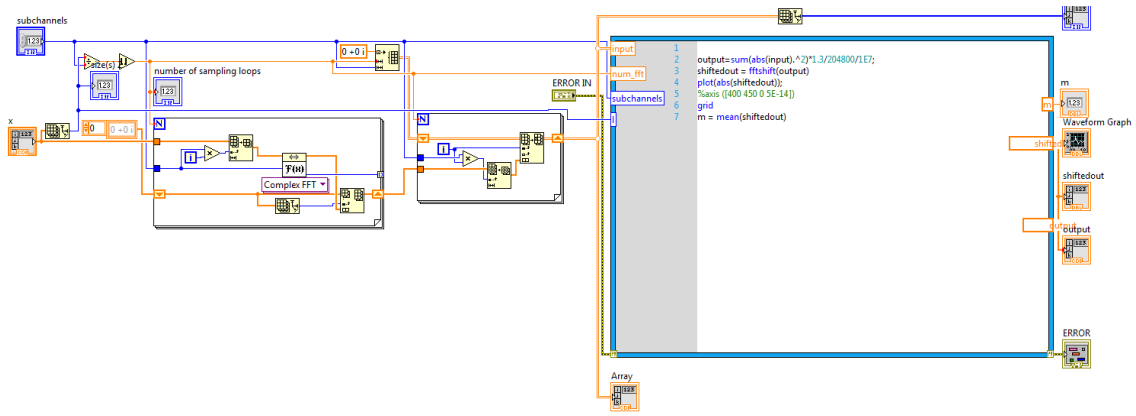


Figure 4.18 *Block diagram of FFT processing sub-VI.*

Matlab scripts are used in the implementation of AFB based ED algorithm sub-VI. This is because the AFB designed using LabVIEW's built-in filter bank designer tool cannot operate correctly with other parts of the algorithm. Therefore, the AFB is also designed using Matlab script and called as a function in the implementation; it is expected to work better in this way. The block diagram of the AFB process sub-VI is seen from Figure 4.19. The input of this sub-vi is also the data from the output of baseband signal processing block and the number of AFB points. The output of this sub-VI is AFB processed data for comparison with the calculated threshold.

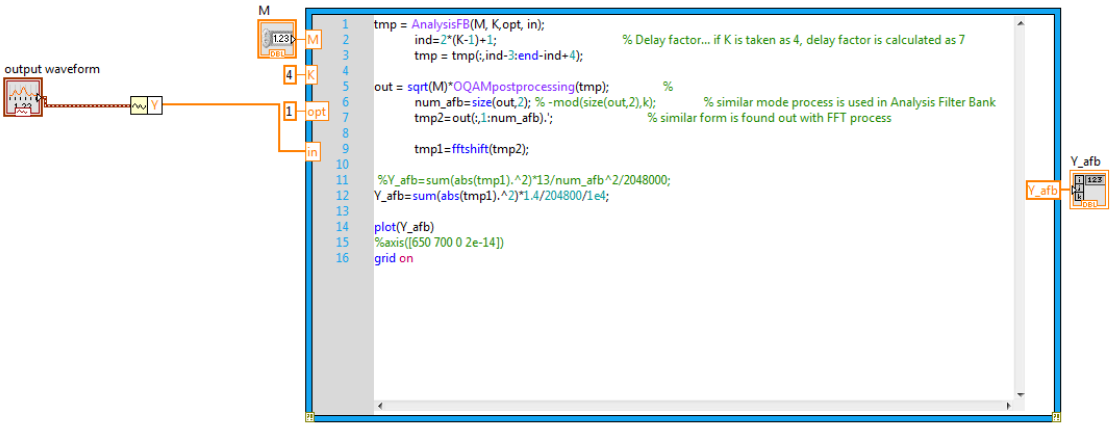


Figure 4.19 *Block diagram of AFB processing sub-VI.*

The implementation of Max-Min ED is shown in Figure 4.20. As described in Chapter 2, the first step is to perform frequency analysis using FFT and the second step is to find the maximum and minimum value of the processed data, and the final step is to calculate the difference between the maximum value and the minimum value and compare it with the threshold to decide whether there are PU signals in the specified spectrum range or not.

The final step of the whole implementation is to make comparison between the test statistics with threshold and display the results. The test statistics are compared with

thresholds according to the user's selection, and result of comparison is recorded. All the recorded test results are then combined to calculate the final result.

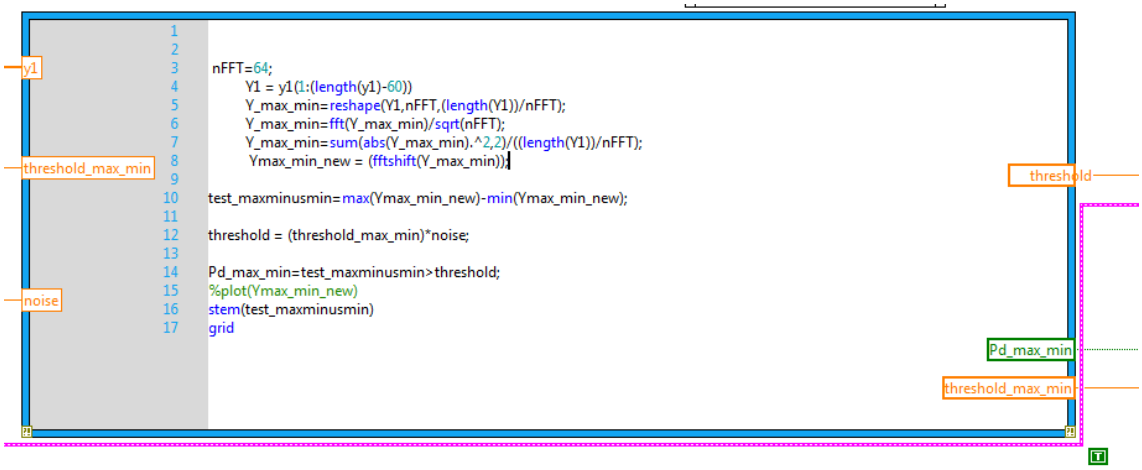


Figure 4.20 *Block diagram of Max-Min process.*

In this thesis, four to five tests are performed for each channel model and sensing algorithms, the average value of each test is then calculated to be the final result. 50 loops are applied for each SNR for each test, and the average value of those 50 test results are calculated as the result for this test. That is, over 250 test results for each SNR for each sensing algorithms under different channel models. This makes the implementation test result reliable.

5. ALGORITHM TESTING RESULTS

In this chapter, the implementation results of the WM signal in different channel models with different sensing algorithms are studied. The results will be presented according to their channel models. In order to have more reliable measurement results, each of the test results is the average value of over 250 test results under same SNR condition with 2 dB SNR step. The number of samples per catch at USRP front end is 204800 samples.

5.1 Noise Variance Uncertainty

When using ED based SS algorithms, the threshold is calculated using the information of the noise variance. Thus, the effect of noise variance uncertainty should be considered. In this implementation, the noise variance is estimated by measuring the sensing frequency band before the sensing process. The noise variance is estimated for 50 times and the averaged value of these 50 noise variance estimates is used as the value for threshold calculation.

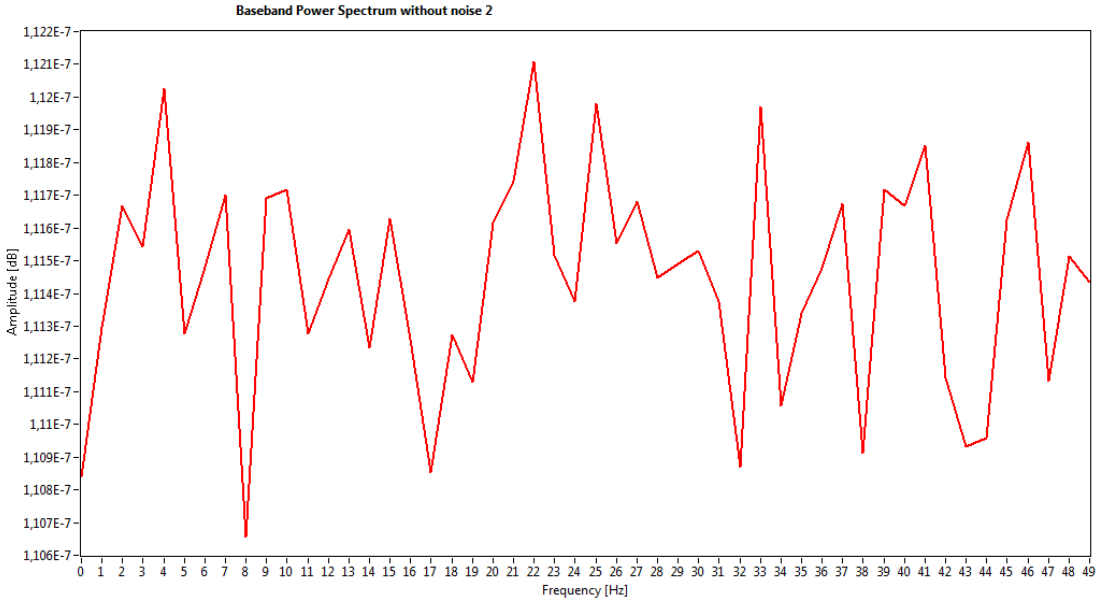


Figure 5.1 *Noise variance of 50 test loops.*

The lowest noise variance is about 1.106×10^{-7} Watts and the highest value is around1.121×10^{-7} Watts as seen in Figure 5.1. According to the concept of noise variance uncertainty mentioned in [24]: $\sigma^2_{\max} = \rho\sigma^2_n$ and $\sigma^2_{\min} = (1/\rho)\sigma^2_n$, ρ can be calculated as $\rho = \sqrt{\frac{\sigma^2_{\max}}{\sigma^2_{\min}}} = 1.007$. The noise variance uncertainty can be calculated using:

$$x = 10\log_{10}\rho = 10\log_{10}(1.007) = 0.0303\text{dB} \quad (18)$$

The above discussion provides a typical noise uncertainty value of this implementation environment. The noise variance uncertainty value might be slightly different with different USRP devices or measurement environment. For instance, the warm-up time is about 3 minutes for NI-USRP 2932 according to [34]. The noise variance uncertainty is bigger when the measurement starts right after the USRP is turned on. But the short-time noise variance uncertainty can be considered to take the value of 0.03 dB. In order to have more reliable noise variance uncertainty scenario, the measurements should be performed 3 to 5 minutes after the USRP devices are turned on.

The main interest for WM sensing is in the TV white space (TVWS) application, in the terrestrial TV frequency band. However, the following tests were carried out in the 2.4 MHz ISM frequency band, in order to avoid interferences to actual WM operation or TV reception. But the results are valid also for the TVWS application.

5.2 Testing with Indoor Channel Model

As mentioned in Chapter 4, the Indoor channel model is an empirical channel model introduced in [40]. The channel bandwidth is selected as the total interested bandwidth of 8 MHz in the terrestrial TV frequency band.

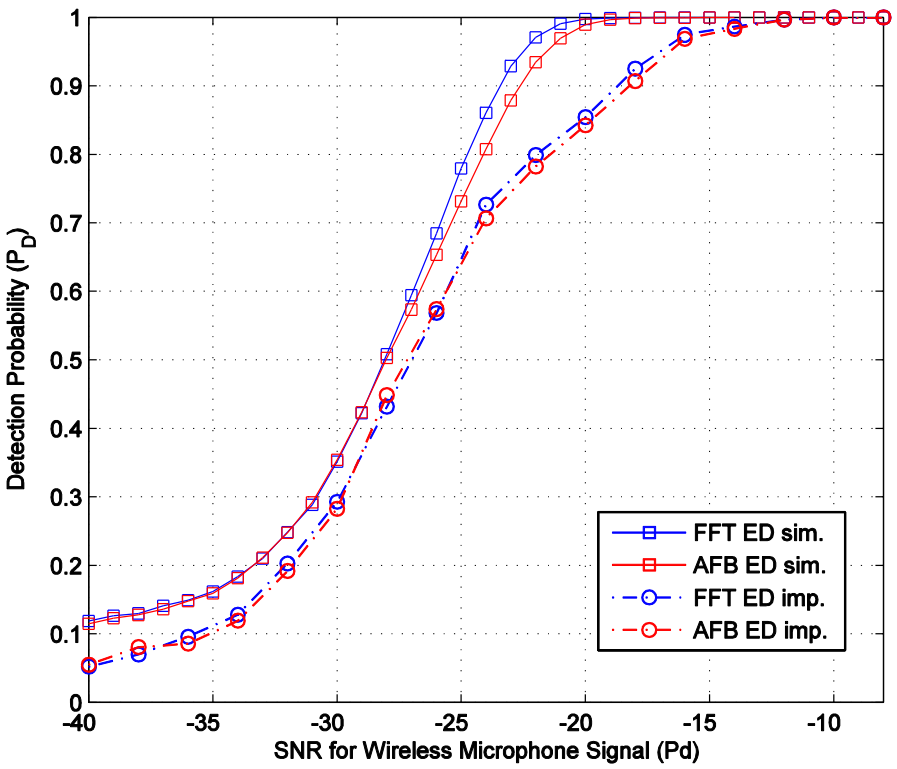


Figure 5.2 *Detection probability of wireless microphone signal using FFT & AFB based ED with* $P_{FA} = 0.1$ *and* $N_t = 200$ *for 8 MHz sensing bandwidth.*

In this implementation, the WM signal bandwidth is around 140 kHz and the sensing bandwidth is 8 MHz. The number of FFT and AFB points is 1024. The whole 8 MHz bandwidth is divided into 1024 sub-channels according to the number of FFT and AFB points, each of the 1024 sub-channel covers $8\,MHz/1024 = 7.8\,kHz$ bandwidth and thus about 18 sub-channels can cover the PU signal bandwidth. In the process of detection, 18 sub-channels are grouped as a WM channel, thus there are 56 channels in total and 56 parallel sensing process are needed to observe the whole 8 MHz band, in order to find out which channels are occupied by WM signals. In this measurement, we only observe the sensing performance of the sub-channels that contains the WM signal we transmit. Sample complexity $N_t * N_f = 1000$ and target P_{FA} is chosen as 0.1 in our study. Bandwidth of noise is considered as 8 MHz for all sensing band.

The detection probability (P_D) performance under different SNR conditions for the Indoor channel without frequency averaging (sensing only the subband at DC, i.e., the carrier frequency of the WM signal at RF) is shown in Figure 5.2. The solid line shows the simulated results for FFT (blue) and AFB (red) sensing algorithms. The dot-dash line shows the test result of implementation with FFT (blue) and AFB (red) sensing algorithms. As seen in the Figure 5.2, there is approximately 2 to 3 dB difference between the implemented test results and the simulation results. In our implementation, SNR wall is obtained as -18.55 dB, which is very well matching with noise uncertainty model as seen in Figure 5.3.

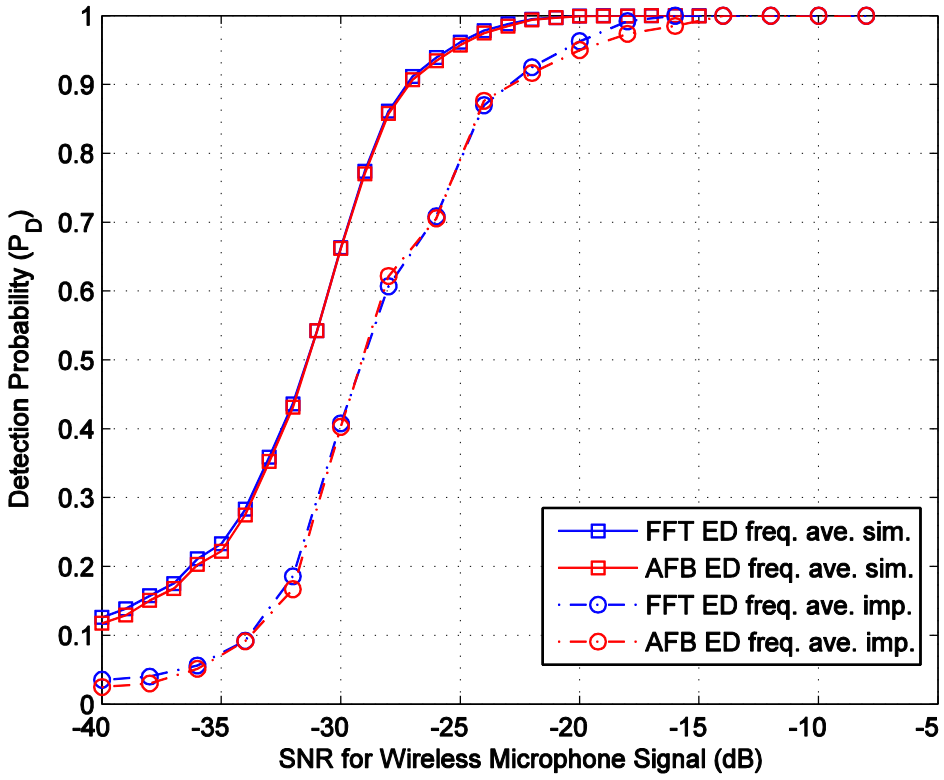


Figure 5.3 *Detection probability of wireless microphone signal using FFT & AFB based ED with* $P_{FA} = 0.1$, $N_f = 5$ *and* $N_t = 200$ *for 8 MHz sensing bandwidth.*

Figure 5.3 illustrates the simulation and the implementation results with FFT and AFB based sensing algorithms using 5-points averaging window which is determined as optimum value using simulation/implementation results. The solid line shows the simulated results for FFT (blue) and AFB (red) sensing. The dot-dash line shows the test result of implementation with FFT (blue) and AFB (red) sensing algorithms.

Figure 5.4 shows the detection performance using Max-Min ED. The blue solid curve and the red dot-dash curve show the simulation and the implementation results, respectively. As seen in Figure 5.4, both the implementation and the simulation results perform better than traditional FFT and AFB based ED sensing algorithms and the effects of noise variance uncertainty is smaller than in FFT and AFB based ED, particularly at above 0.7 P_D .

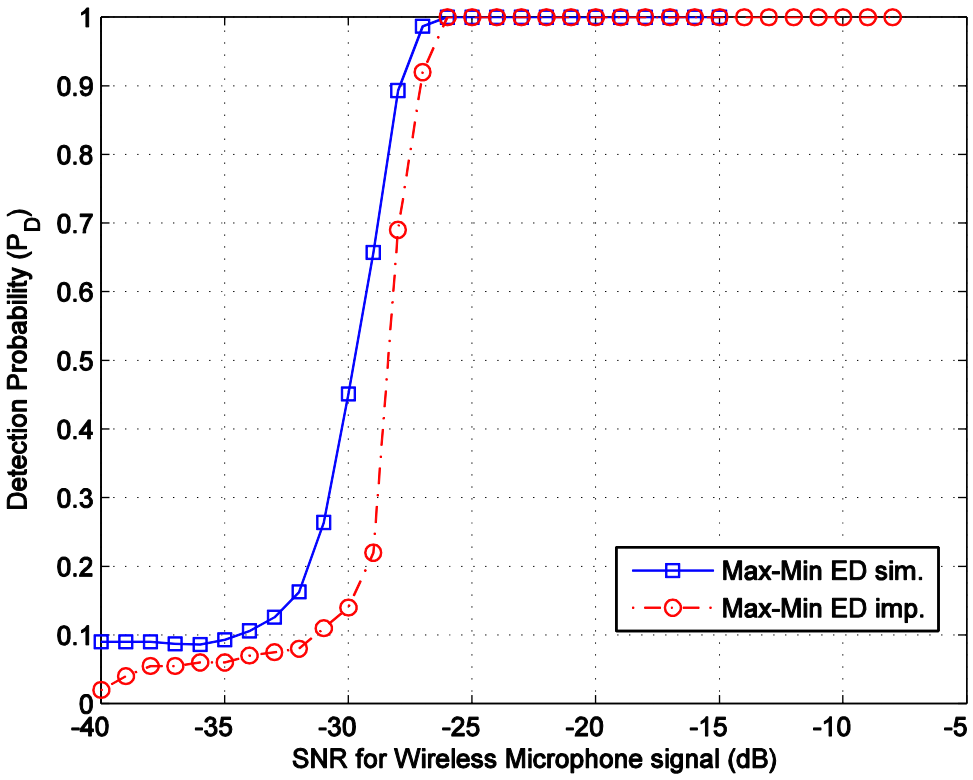


Figure 5.4 *Detection probability of wireless microphone signal using Max-Min ED with* $P_{FA} = 0.1$ *and* $N_t = 3200$ *for 8 MHz sensing bandwidth.*

5.3 Testing with ITU-R Vehicular A channel model

The second channel model used in this thesis is ITU-R Vehicular A channel model, as introduced in Chapter 4. This channel model is a commonly used set of empirical channel models specified in ITU-R recommendation M.1225 [37]. It is the most frequency selective channel model among the three channel models used in this thesis.

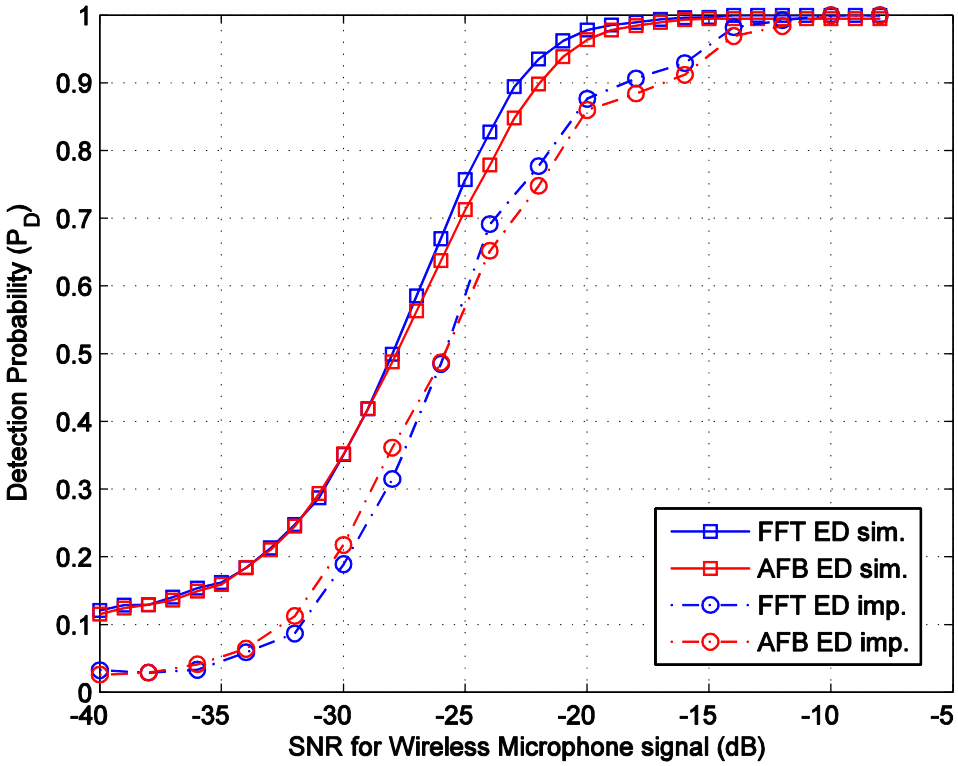


Figure 5.5 *Detection probability of wireless microphone signal using FFT & AFB based ED with* $P_{FA} = 0.1$ *and* $N_t = 200$ *for 8 MHz sensing bandwidth.*

Figure 5.5 shows the detection probability P_D under different SNR value for PU signal with basic FFT and AFB based ED sensing algorithms. The solid curves represent simulation results for FFT (blue) and AFB (red) based ED sensing algorithms and the dash-dot curves represent the measurement results of implementation of basic FFT (blue) and AFB (red) sensing algorithms.

As seen in Figure 5.5, the effects of noise variance uncertainty is still obvious and comparing with the results of Indoor channel model, the performance under ITU-R Vehicular A channel is slightly worse. This is due to the fact that the ITU-R Vehicular A channel model is more frequency selective.

Figure 5.6 shows the detection probability P_D under different SNR value for PU signal with FFT and AFB based ED sensing algorithms using 5-points frequency averaging. The solid curves represent simulation results for FFT (blue) and AFB based ED sensing algorithms and the dash-dot curves represent the implementation test results of basic FFT (blue) and AFB (red) sensing algorithms.

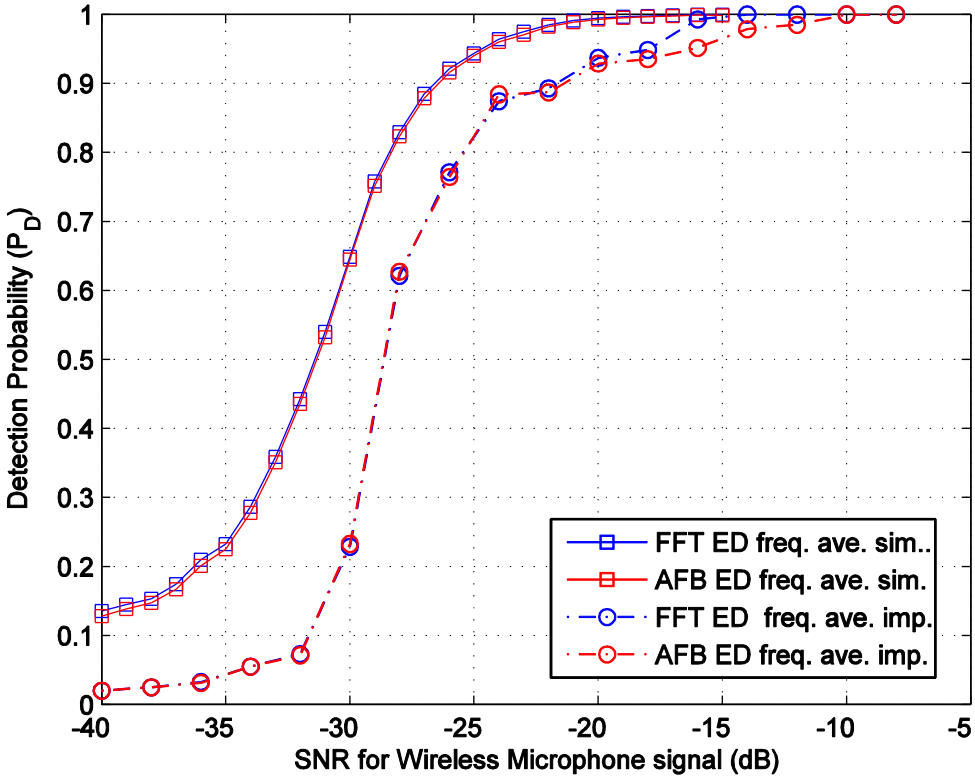


Figure 5.6 *Detection probability of wireless microphone signal using FFT & AFB based ED with* $P_{FA} = 0.1$, $N_f = 5$ *and* $N_t = 200$ *for 8 MHz sensing bandwidth.*

Compared with the test result of algorithms without frequency averaging, both the simulation result and the implementation results performed about 4 dB better using algorithms with frequency averaging. It is obviously seen the effects of noise uncertainty.

The simulation and implementation results of Max-Min based ED are shown in Figure 5.7. As illustrated in the figure, the blue solid curve is the simulation result and the dash-dot curve is the implementation test result. Comparing with the graph above, the test results with Max-Min ED algorithm performs better than FFT/AFB based ED. And the effect of noise uncertainty is less obvious in Max-Min based ED.

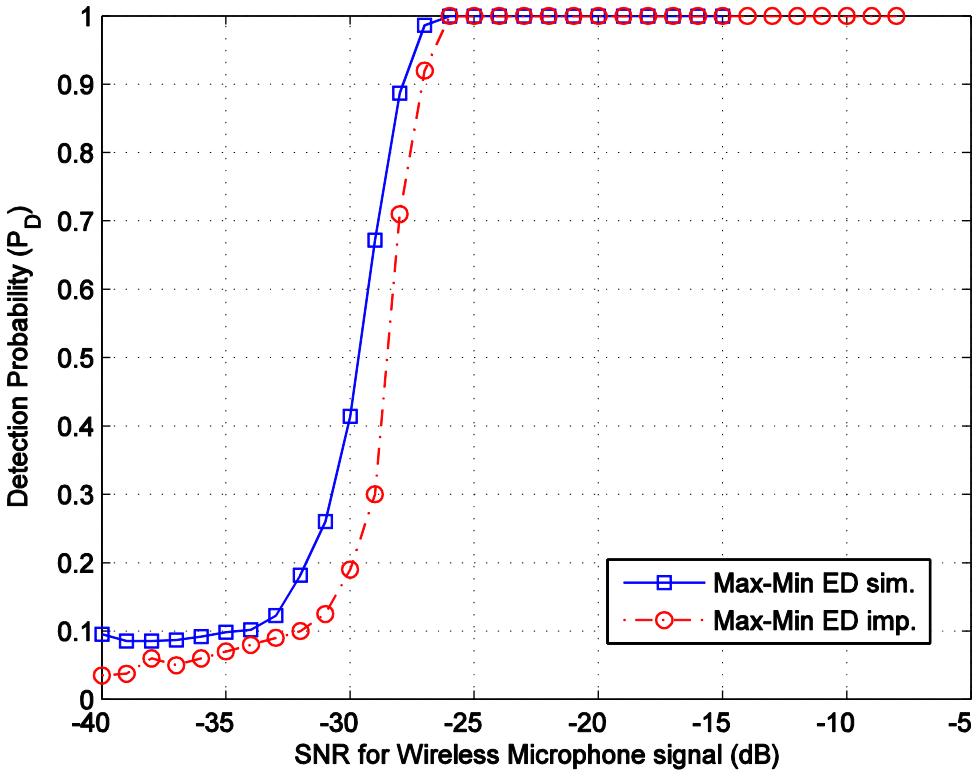


Figure 5.7 *Detection probability of wireless microphone signal using Max-Min ED with* $P_{FA} = 0.1$ *and* $N_t = 3200$ *for 8 MHz sensing bandwidth.*

5.4 Testing with Stanford University Interim 1 channel model

In this test, we applied the SUI-1 channel model which has 3 Ricean fading taps and 0.9 μs delay spread. This is the least frequency selective channel among the three channel models [41].

The simulation and the implementation results of FFT and AFB based ED sensing algorithms are shown Figure 5.8. The solid curves represent simulation results for FFT (blue) and AFB (red) based ED sensing algorithms and the dash-dot curves represent the implementation results of FFT (blue) and AFB (red) sensing algorithms.

As shown in Figure 5.8, there is about 3 to 4 dB difference in the detection performance due to the effect of noise uncertainty effect. Comparing with the test results under channel models introduced earlier, the test result under SUI-1 channel model performs slightly better because of that this channel model is the least frequency selective one among the channel models used in this thesis.

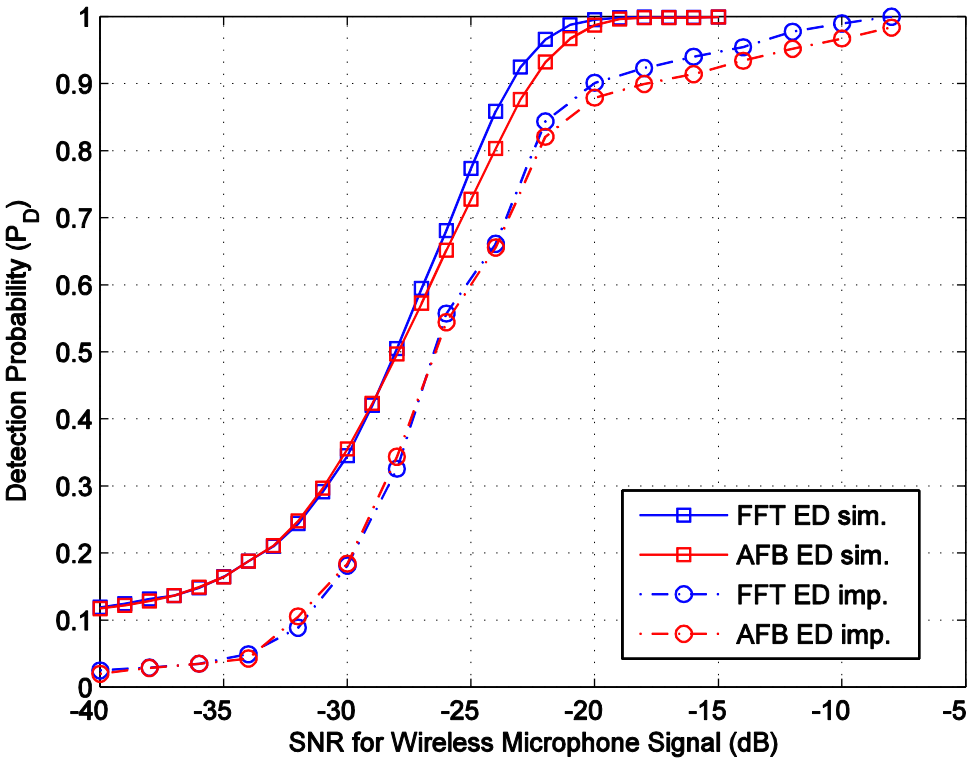


Figure 5.8 *Detection probability of wireless microphone signal using FFT & AFB based ED with* $P_{FA} = 0.1$ *and* $N_t = 200$ *for 8 MHz sensing bandwidth.*

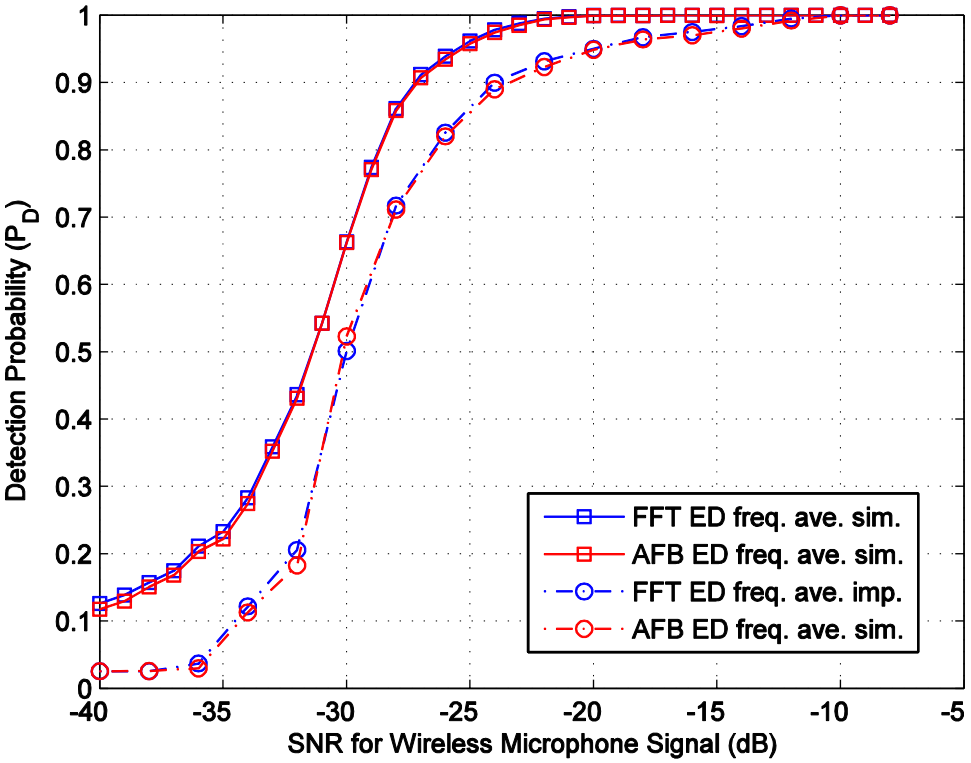


Figure 5.9 *Detection probability of wireless microphone signal using FFT & AFB based ED with* $P_{FA} = 0.1$, $N_f = 5$ *and* $N_t = 200$ *for 8 MHz sensing bandwidth.*

Figure 5.9 shows the simulation and implementation results for FFT and AFB based ED using 5-points frequency averaging. Comparing with the test results without frequency averaging, the frequency averaging also improved the detection performance in this channel model and the effect of noise uncertainty is also obvious.

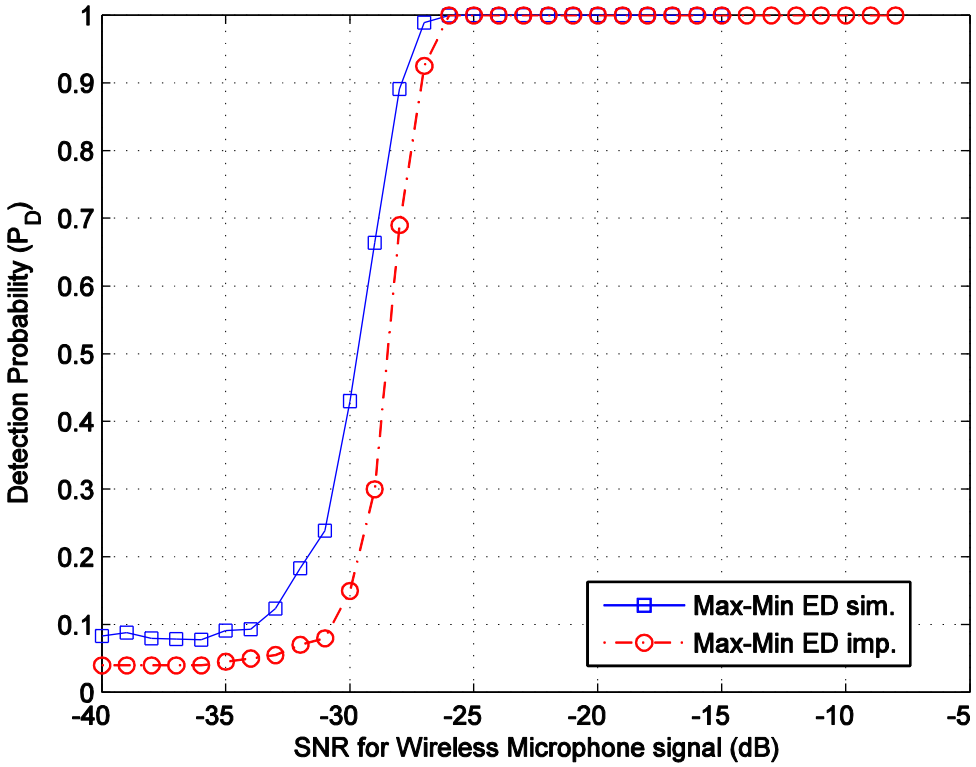


Figure 5.10 *Detection probability of wireless microphone signal using Max-Min ED with* $P_{FA} = 0.1$ *and* $N_t = 3200$ *for 8 MHz sensing bandwidth.*

Figure 5.10 shows the detection performance of Max-Min ED under SUI-1 channel model. The blue solid curve is the simulation results whereas the dash-dot curve is the implementation results. Similar to other channel models, the test results with Max-Min ED algorithm performs better than FFT/AFB based ED. The effect of noise uncertainty is less obvious in Max-Min ED.

5.5 Testing with Actual Indoor Wireless Environment Channel

Besides testing with the above mentioned three commonly used empirical channel models, a group of measurements using the real indoor environment has also been done in this thesis. The measurements are done in our RF laboratory using wireless connections between the transmitter and the receiver. The transmission channel is a line-of-sight (LOS) wireless channel with typical indoor obstacles such as walls, chairs and desks. The distance between the transmission antenna and the receiving antenna is 4.2 meters

in our case. The WM signal is transmitted from the signal generator without any channel model. The received PU signal level is obtained by measuring it at a high SNR case and using a fixed attenuator during the actual spectrum sensing test. The rest of the measurement setup is the same as used in the empirical channel model tests.

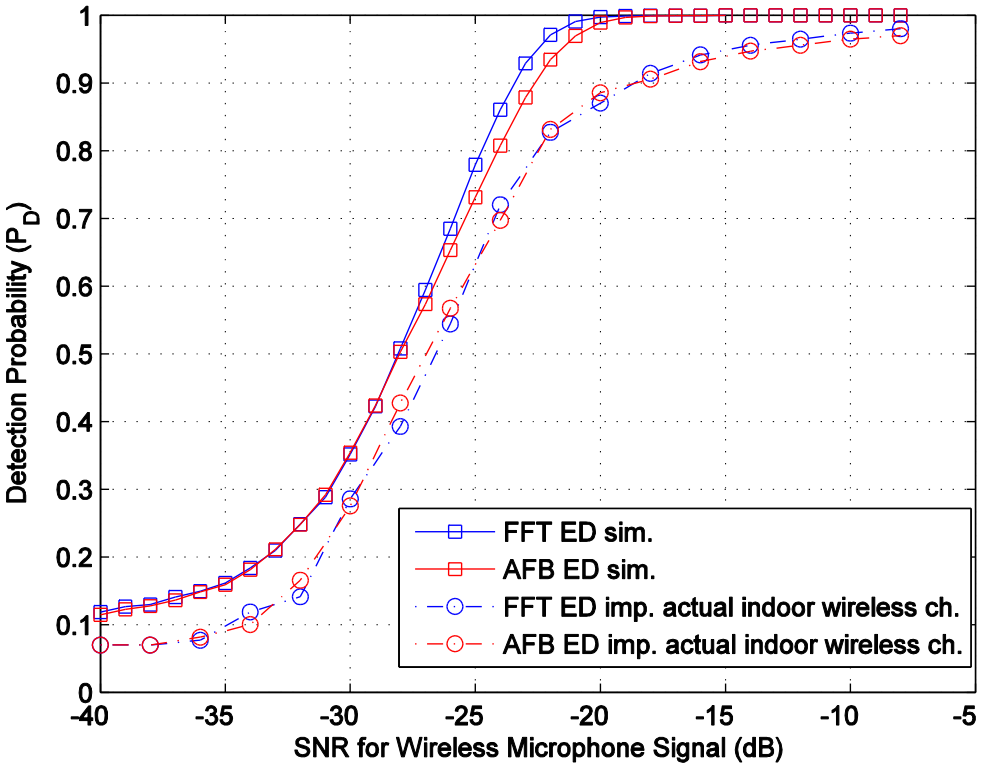


Figure 5.11 *Detection probability of wireless microphone signal using FFT & AFB based ED with* $P_{FA} = 0.1$ *and* $N_t = 200$ *for 8 MHz sensing bandwidth.*

Figure 5.11 shows the simulation results (solid curves) of basic FFT (blue) and AFB (red) under empirical Indoor channel model. The dash-dot curves represent the implementation result under actual wireless indoor channel. The performance under wireless indoor channel is quite similar to other empirical channel models, as shown in the previous figures. This is because the spectrum sensing process is rather insensitive the frequency selectivity exhibited by the typical indoor channels. .

Figure 5.12 shows the simulation result (solid curves) of basic FFT (blue) and AFB (red) under empirical Indoor channel model using 5-points frequency averaging. The dot curves represent the measurement result under wireless Indoor channel using 5-points frequency averaging. Comparing with the test results without frequency averaging, the performance of algorithms with frequency averaging is better. Comparing with the test results of other empirical channel models, the results are similar, too.

It is also good to point out that comparing the test results of indoor wireless channel and other empirical channel models; it can be seen that the detection probability P_D increases slower after 0.97 P_D. This is expected to be due to the noise uncertainty effects.

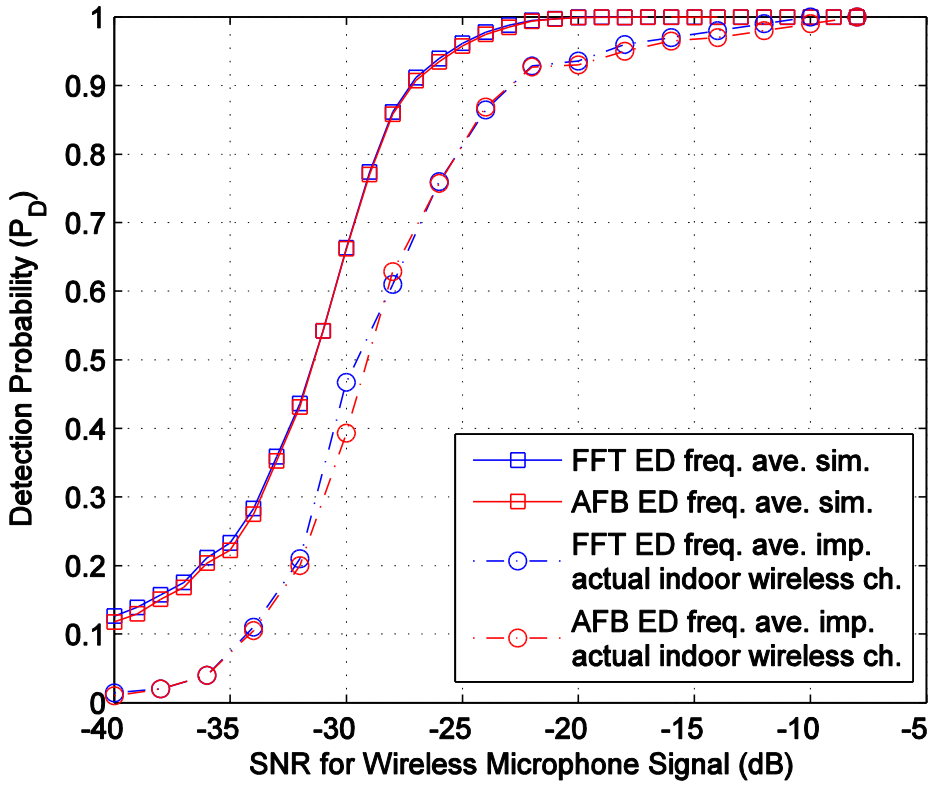


Figure 5.12 *Detection probability of wireless microphone signal using FFT & AFB based ED with* $P_{FA} = 0.1$, $N_r = 5$ *and* $N_t = 200$ *for 8 MHz sensing bandwidth.*

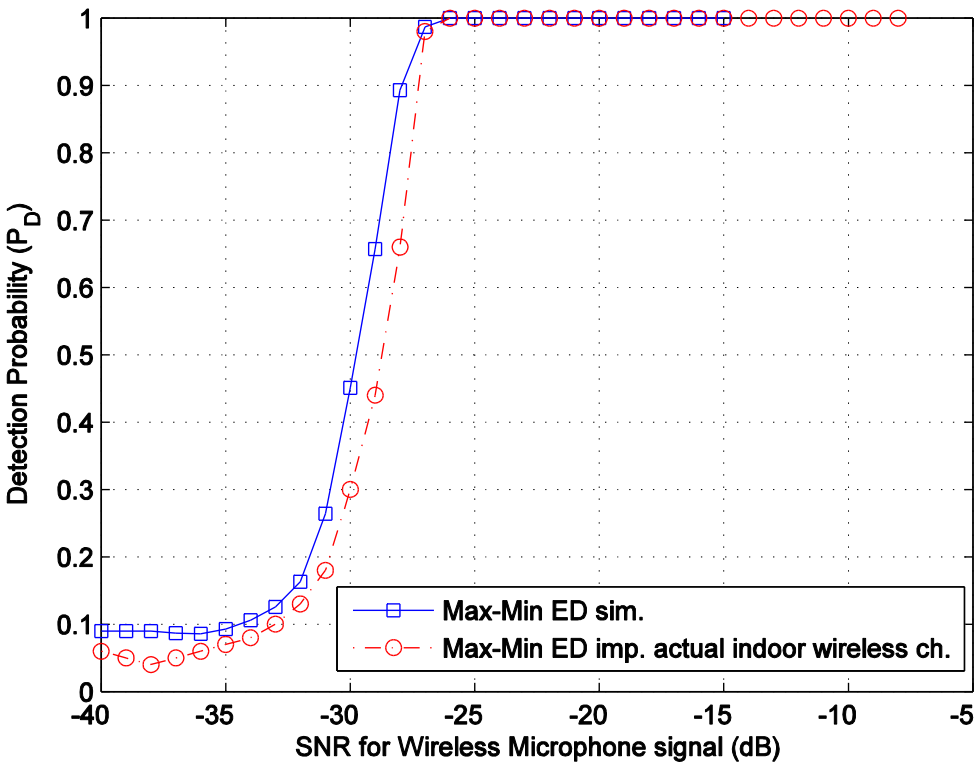


Figure 5.13 *Detection probability of wireless microphone signal using Max-Min ED with* $P_{FA} = 0.1$ *and* $N_t = 3200$ *for 8 MHz sensing bandwidth.*

Figure 5.13 shows the simulation results under empirical Indoor channel model (solid blue curve) and the implementation results under Indoor wireless channel (dash-dot red curve). The measurement results of implementation is also quite similar comparing to the simulation results and also the results of other empirical channel models

5.6 Testing with Different WM Signal Models

As mentioned in Chapter 4, there are three different WM models provided in [22], and the signal model of soft speak is used in above measurement due to its similarity of a normally operating wireless microphone. It is important to consider different WM signal models in terms of SS performance.

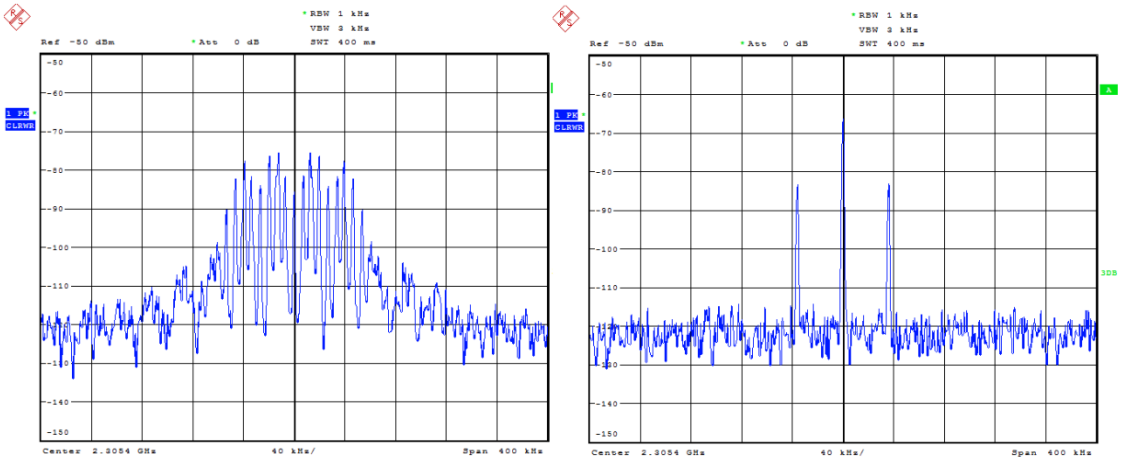


Figure 5.14 *Loud speak WM model (left) and silent WM model (right).*

Figure 5.14 shows the spectra of the other two WM models, the loud speak model in the left and the silent model in the right. As seen Figure 5.14, the loud speak model has about the same amount of peaks compared to soft speak model but more separated, and the silent model only have 5 major peaks.

The simulation results of different WM signal models using FFT/AFB based spectrum sensing algorithm is shown in Figure 5.15. As shown in the figure, the detection performance is greatly degraded for silent WM signal as there are only 3 to 5 major peaks in this model and they can be attenuated due to the channel effects.

Figure 5.16 shows the sensing performance of the three different WM signal models using Max-Min ED. Both the simulation and the implementation results are considered is this figure. On the contrary to the result of FFT/AFB based sensing algorithms, the detection performances using Max-Min ED are quite similar for three different WM signal models and the measurements of implementation are also quite similar with the simulation results, which means better robustness to the noise variance uncertainty.

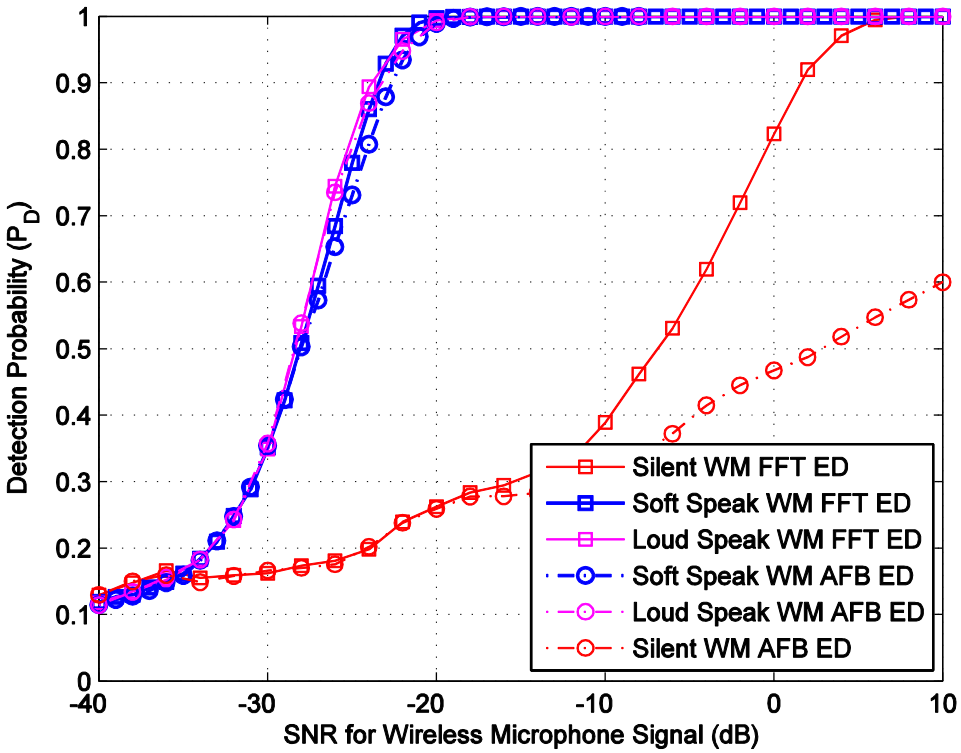


Figure 5.15 *Simulated FFT & AFB based ED sensing performance for different WM models.*

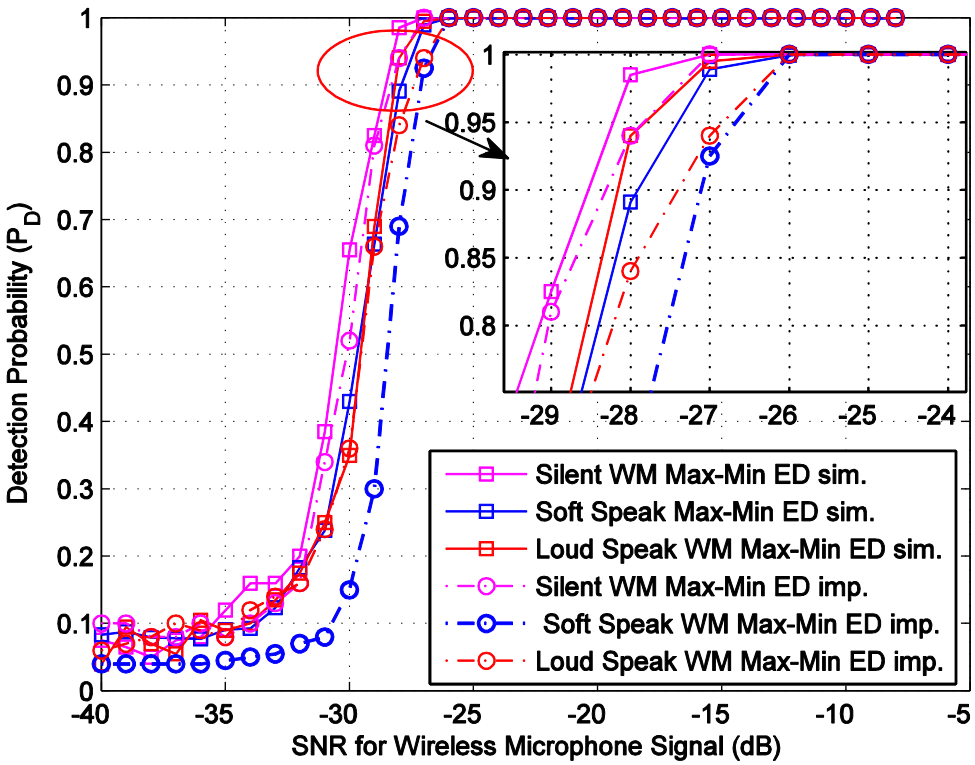


Figure 5.16 *Max-Min ED sensing performances for different WM models.*

6. CONCLUSION

The ultimate aim of this thesis was to implement different kind of ED based SS algorithms using NI USRP hardware devices and NI-LabVIEW platform under different transmission channel models. The concept of CR was briefly introduced in the beginning of this thesis, and then the basics FFT & AFB based ED spectrum sensing algorithms with/without frequency averaging and Max-Min ED were provided. Thereafter, the basic information about hardware and software used in the implementation environment were introduced. This included properties and basics of NI-USRP devices, NI-LabVIEW properties and how it can work in pair with NI-USRP devices. Then the details of implementation of SS algorithms using NI-USRP and NI-LabVIEW were explained. Finally, the implementation results were presented together with Matlab simulation result for the purpose of comparison and evaluating the effect of noise variance uncertainty.

As discussed in Chapter 2, the FFT and AFB based ED SS algorithms are sensitive to noise variance uncertainty due to the usage of noise variance in threshold calculations, while the Max-Min ED sensing algorithms performs more robustly. In this thesis, WM signals were generated using PC controlled vector signal generator and received using NI-USRP devices via cable (empirical channel models) or wireless channel (wireless Indoor channel) with different SNR levels to investigate the performance of each sensing algorithm.

According to the implementation test results presented in Chapter 5, the noise variance uncertainty has a considerable effect on the sensing performance of FFT and AFB based energy detector SS algorithms. Optimized selection of the sensing bandwidth is critical for the performance of these algorithms. On the contrary, Max-Min ED is shown to have more robust performance under noise variance uncertainty scenario. It is also important to point out that in the test of the three different WM signal models, Max-Min ED is proved to have better performance no matter how narrow the PU signal spectrum is. On the contrary, FFT/AFB based ED performances are greatly degraded when the PU signal spectrum gets narrower.

In this implementation, both the FFT/AFB based and Max-Min ED have low computational and implementation complexity, but with good sensing performance. As the tested primary user signal is a WM signal and the sensing spectrum range is 50 to 60 times larger than the PU signal bandwidth, the FFT/AFB sensing algorithms are operated in a subband-wise structure. This means that the whole 8 MHz interested band is divided into 56 WM channels.The WM signal can be prersent at any of those channels. The FFT/AFB detector detects the 56 sub-channel in parallel in order to find whether

any of those sub-channels were occupied or not. This also increases the false alarm probability P_{FA} of the whole system. On the other hand, the Max-Min based ED can sensing the whole 8 MHz at one time with better performance.

The real-time SS implementation prototype provided by this thesis gives a very useful platform for future research on CR and SS. The solution of removing local oscillator leakage problem and USRP RF front-end filter effect also provide useful information on NI-USRP device properties and implementation guidelines.

The further study topics will include finding a practical method to test and validate the implemented sensing system with real WM using wireless channels. During this thesis work, some preliminary tests with actual WM signals in the TV frequency band were carried out. However, it turned out to be quite difficult to perform quantitative sensing performance measurements with actual WMs. Another possible further study could be deriving optimum configuration for sensing of the whole 8 MHz TV channel using the Max-Min ED algorithm. This includes optimum number of subbands and threshold setting for reaching the target P_{FA} for the whole sensing process.

REFERENCES

[1] A. Goldsmith, Wireless Communications, Cambridge University Press, 2005.

[2] Federal Communications Commission, "Part 74 Experimental radio, auxiliary, special broadcast and other program distribution services", [Online]. Avaliable: http://www.gpo.gov/fdsys/pkg/CFR-2009-title47-vol4/pdf/CFR-2009-title47-vol4-part74.pdf, accessed 05 June 2014.

[3] Y.C. Liang, A. T. Hoang Y. Zeng and R. Zhang, "A Review on Spectrum Sensing for Cognitive Radio: Challenges and Solutions," *EURASIP Jor. on Ad. in Sig. Proc,* pp. 1-15, January 2010.

[4] J. Mitola III and G. Q. Maguire, Jr., "Cognitive radio: making software radios more personal," *IEEE Personal Communications Magazine, vol. 6, nr. 4,* p. pp. 13–18, Aug. 1999.

[5] H. Urkowitz, "Energy detection of unknown deterministic signals," *Proc. IEEE, vol. 55, no. 4,* pp. 523-531, Apr 1967.

[6] H. Poor, "An Introduction to Signal Detection and Estimation", *2nd ed. Berlin, Germany，Springer-Verlag,* 1994.

[7] T. Yucek and H. Arslan, "A Survey of Spectrum Sensing Algorithms for Cognitive Radio Applications," *IEEE Communications surveys & tutorials, vol. 11, no. 1, first quater,* pp. 116-129, 2009.

[8] S. Haykin, "Cognitive radio: Brain-empowered wireless communications", *IEEE J. Select. Areas Commun. vol. 23, no. 2,* pp. 201-220, Feb. 2005.

[9] S. Dikmese and M. Renfors, "Performance analysis of eigenvalue based spectrum sensing under frequency selective channels", *in Proc. CROWNCOM 2012, Stocholm, Sweden,* pp. 356-361, 2012.

[10] S. Dikmese, A. Gokceoglu, M. Valkama and M. Renfors, "Reduced Complexity Spectrum Sensing Based on Maximum Eigenvalue and Energy", in *Proc. ISWCS 2013, Ilmenau, Germany,* pp. 1-5, 2013.

[11] S. Dikmese, J. L. Wong, A. Gokceoglu, E. Guzzon, M. Valkama and M. Renfors, "Reducing computational complexity of eigenvalue based spectrum sensing for cognitive radio", *in Proc. CROWNCOM 2013, Washington DC, USA,* pp. 61-67, 2013.

[12] Y. Zeng and Y. C. Liang, "Spectrum sensing algorithms for cognitive radio based on statistical covariance", *IEEE Trans. Veh. Technol. vol. 58, no. 4,* pp. 1804-1815,

May 2009.

[13] 802.22 Working Group, "IEEE 802.22 D1: draft standard for wireless regional area networks", *EURASIP Journal on Advances in Signal Processing 13,* p. March, 2008.

[14] S.Dikmese, S.Srinivasan, M. Shaat, F. Bader and M.Renfors, "Spectrum sensing and resource allocation for multicarrier cognitive radio systems under interference and power constraints", *EURASIP Journal on Advances in Signal Processing 2014, 2014:68.*

[15] S. Dikmese, P. C. Sofotasios, T. Ihalainen, M. Renfors and M. Valkama " Efficient Energy Detection Methods for Spectrum Sensing under Non-Flat Spectral Characteristics," accepted with minor revision for *IEEE Journal on Selected Areas in Communications (JSAC),* Nov. 2014.

[16] S. Dikmese, P. C. Sofotasios, M. Renfors, M. Valkama , "Reduced Complexity Energy and Eigenvalue Based Spectrum Sensing under Frequency-Selective Spectral Characteristics". *Submitted to IEEE Transaction on Signal Processing 2014.*

[17] S. Dikmese, S. Srinivasan and M. Renfors, "FFT and Filter Bank Based Spectrum Sensing and Spectrum Utilization for Cogntive Radios" *in Proc. ISCCSP 2012,Rome, Italy pp. 1-5, 2012.*

[18] S. Dikmese, M. Renfors and H. Dincer, "FFT and Filter Bank Based Spectrum Sensing for WLAN Signals", *in Proc. 20th European Conference on Circuit Theory and Design (ECCTD), Linkoping, Sweden,* pp. 781-784, 2011.

[19] S. Dikmese, M. Renfors, "Optimized FFT and filter bank based spectrum sensing for Bluetooth signal," *2012 IEEE Wireless Communications and Networking Conference, (WCNC2012), Paris, France,* pp. 803-807, 2012.

[20] S. Dikmese, T. Ihalainen and M. Renfros, "Analysis and Optimization of Energy Detection for Non-Flat Spectral Characteristics", Chapter 3 in *Cognitive Communication and Cooperative HetNet Coexistence*, Springer International Publishing, 2014, pp. 47-69.

[21] G. D. Knott, "Fourier Transforms", Civilized Software Inc., 2014.

[22] C. Clanton, M. Kenkel and Y. Tang, "Wireless Microphone Signal Simulation Method", *IEEE 802.22-07/0124r0,* 2007.

[23] E. Reihl, "Wireless Microphone Characteristics", *IEEE 802.22-06/0070r0,* 2006.

[24] R.Tandra and A.Sahai, "SNR walls for signal detection," *IEEE J.Selected Topics Signal Processing, vol. 2,* pp. 4-17, February 2008.

[25] S. Dikmese, P. C. Sofotasios, M. Renfors and M. Valkama, "Maximum-Minimum Energy Based Spectrum Sensing under Frequency Selectivity for Cognitive Radios," *in Proc. CROWNCOM 2014, Oulu, Finland,* pp. 347-352, 2014.

[26] J. Neyman and E. Pearson, "On the problem of the most efficient test for statistical

hypotheses," *Philosophical Trans. of the Royal Society of London, vol. 231, no.4,* pp. 289-337, Feb 1933.

[27] National Instruments, "A Hands-on Introduction to Software Defined Radio with the NI USRP and NI LabVIEW,". [Online]. Available: http://www.ni.com/white-paper/14518/en/, accessed 05 June 2014.

[28] National Instruments Co., "National Instruments Corporation Website.," National Instruments Corporation, [Online]. Available: http://sine.ni.com/ds/app/doc/p/id/ds-355/ , 05 June 2014.

[29] K. Madani, M. Dilinger, and N. Alonistioti, Software Deffined Radio: Acrchitectures, Systems and Functions, West Sussex, England: Wiley&Sons, 2003.

[30] M. Ettus, " Ettus Research Products and Roadmap," GNU Radio, 08 2012. [Online]. Available: http://gnuradio.org/redmine/projects/gnuradio/wiki/GRConf 2011, accessed 05 June 2014..

[31] " GNU Radio Community," GNU Radio , 2014. [Online]. Available: http://www.gnuradio.org, accessed 05 June 2014.

[32] Labview. [Online]. Available: http://www.ni.com/labview .

[33] "Matlab Simulink," [Online]. Available: http://www.mathworks.se/products/simulink, accessed 05 June 2014.

[34] National Instruments Corporation, "NI USRP-292x/293x Datasheet," 2014. [Online]. Available: http://sine.ni.com/ds/app/doc/p/id/ds-355, accessed 05 June 2014.

[35] J. Gilmore, "HDTV and the USRP," gnu.org, 2012. [Online]. Available: http://lists.gnu.org/, accessed 05 June 2014.

[36] National Instruments, "LabVIEW System Design Software," National Instruments, [Online]. Available: http://www.ni.com/labview/, accessed 05 June 2014.

[37] National Instruments, "Benefits of Programming Graphically in NI LabVIEW," National Instruments, [Online]. Available: http://www.ni.com/white-paper/14556/en/, accessed 05 June 2014.

[38] S. Haykin, Communication Systems, John wiley & Sons, Inc., 2001.

[39] R. Jain, "Channel Models: A Tutorial," WiMAX Forum AATG, February 2007.

[40] T.Yucek, "Channel, Spectrum, and Waveform Awareness in OFDM Based Cognitive Radio Systems," Ph.D. thesis, Tampa, 2007.

[42] Y. Zeng and Y. C. Liang, "Eigenvalue-based spectrum sensing algorithms for cognitive radio," *IEEE Trans. Commun., vol.57 no. 6,* pp. 1784-1793, June 2009.

[43] D. J. Thomson S. Haykin and J. H. Reed, "Spectrum Sensing for Cognitive Radio," *Proceedings of the IEEE, vol. 97, no. 5,* May 2009.

[44] H. Urkowitz, "Energy detection of unknown deterministic signals," *Proc. IEEE, vol. 55, no. 4,* pp. 523-531, Apr 1967.

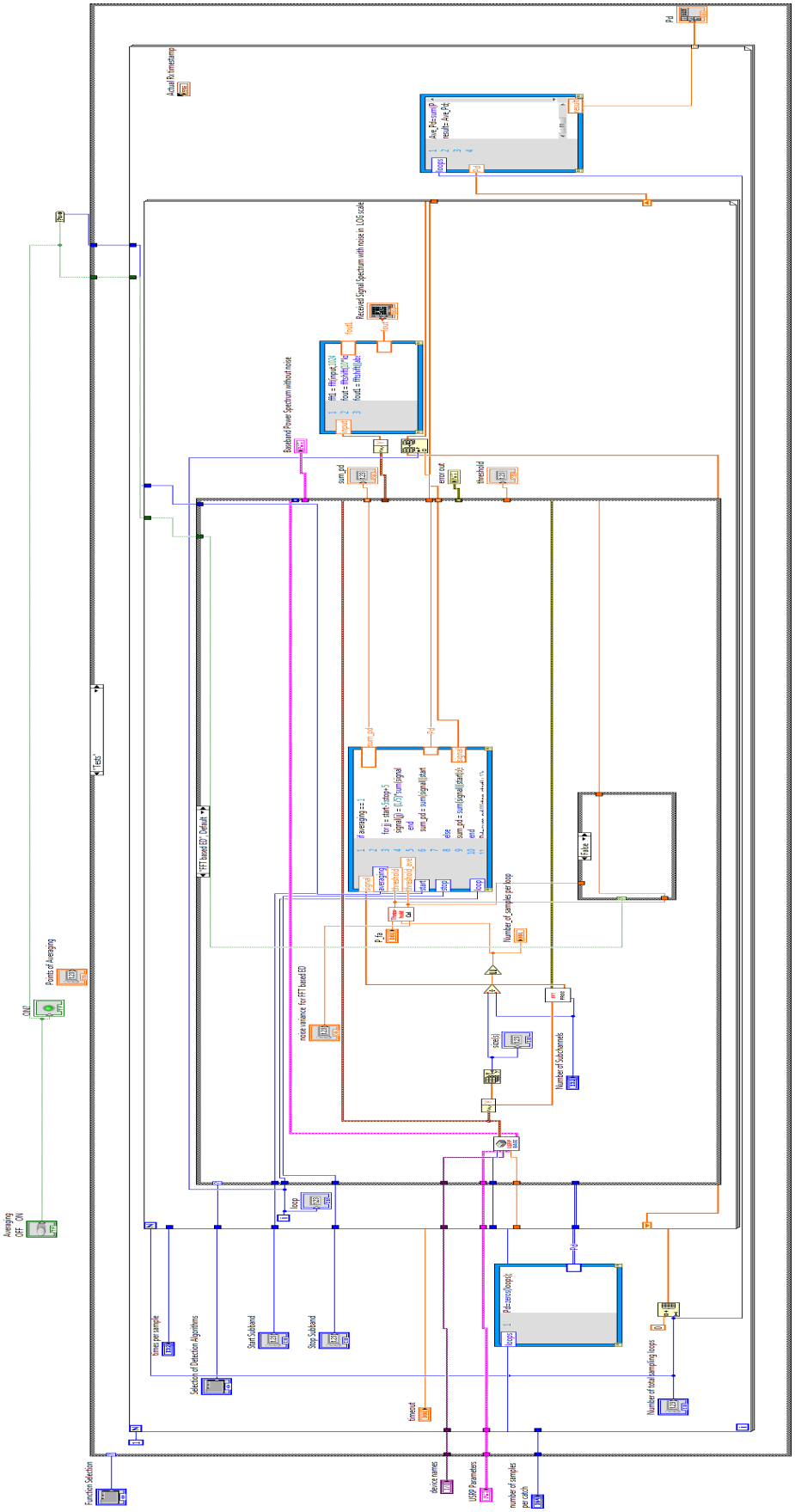
[45] R. Tandra and A. Sahai, "SNR Walls for Signal Detection," *IEEE Journal of selected topic in signal processing, vol. 2, no. 1,* pp. 4-17, Feburary 2008.

Appendix A: Detailed Specifications of NI-USRP 2932

Characteristics	NI USRP-2920	NI USRP-2921	NI USRP-2922	NI USRP-2930	NI USRP-2932
Transmitter					
Frequency Range	50MHz-2.2GHz	2.4GHz-2.5GHz 4.9GHz-5.9GHz	400MHz-4.4GHz	50MHz-2.2GHz	400MHz-4.4GHz
SW Adjustable TX Frequency Step	< 1KHz				
MAX Output Power	15 dBm - 20 dBm				
TX Output Power Gain Range	0 dB - 31 dB				
SW Adjustable Output Power Step Size	1dB				
Instantaneous Real-Time Bandwidth	20MHz (16bit samples) 40MHz (8bit-samples)				
DAC (Digital to Analog Conversion)	2 channels, 400MS/s, 16 bit				
DAC SFDR (Spurious Free Dynamic Range)	80 dB				
Receiver					
Software Adjustable RX Frequency Step	< 1KHz				
Max Input Power (Pin)	0 dBm				
Noise Figure	5 to 7 dB				
Instantaneous Real-Time Bandwidth	20MHz (16bit samples) 40MHz (8bit-samples)				
ADC (Analog to Digital Conversion)	2 channels, 100MS/s, 14 bit				
ADC SFDR (Spurious Free Dynamic Range)	88 dB				

Shared Characteristics (Apply all NI-29xx devices)			
Connections		**Physical Specifications**	
TX1 RX1, TX2 RX2, RX2 Ports	SMA	Enclosure Dimensions	6.25" Wide x 1.9" Tall x 8.35" Deep
Ethernet Connection	1 Gigabit Ethernet	Weight	2.63 lbs
Power Adapter	6VDC, 3A	Operating Temperature	23° C ±5° , Room Temperature
Ref Clock (10-Mhz external reference input)	SMA, 10 MHz		
PPS Input (Pulse Per Second reference input)	SMA, 3-5V TTL Compatible		
MIMO Expansion Port	High-Speed SerDes protocol		

Appendix B: LabVIEW VI Block Diagram



Appendix C: Matlab Code for Generating WM signal With R & S SMJ 100A Vector Signal Generator

```
clc,clear all,close all,
% device selection
DeviceContainer         =   createInstruments();
FSG                     =   DeviceContainer{1};
SMJ100A                 =   DeviceContainer{2};
%%

guard_subcs=15;
params.bw=20e6;          % 5-MHz.
channel_flag=1;
channel_vehic=0;
params.channel_model='INDOOR';    %channel model selection
%params.channel_model='ITUR-A';
%params.channel_model='SUI1';
params.rmsds=80e-9;
params.fft_size=256;
params.cyclic_prefix=64;
params.pdp=generate_pdp(params);

%% Signal Generation

noisepower = 1.12e-7; %Estimated Noise power
noisepowerdb = 10*log10(noisepower);%Estimated Noise power in dB scale
SNRdB = 0;           % target SNR value
lossdB = 6;          % Estimated Path loss

SNR=10.^((SNRdB+noisepowerdb + lossdB )./20);

kf=5e3;            % FM deviation factor is ±5 kHz in SILENT mode
Fm=32e3;           % m(t) is a 32 kHz sinusoid signal in SILENT mode

Fc=100e3;
Fs=100e3;
Ts=1/Fs;         % sampling time

%%% vocal signal %%%

t = 0 : Ts :2*(1-Ts);                    % interval
mt = cos(2*pi*Fm*t);                        % m(t)

%%% FM signals %%%
attenuation=1;

y = cos(2*pi*(Fc).*t+2*pi.*((kf/Fm).*mt));
y=attenuation.*(y/sqrt(sum((abs(y).^2))./length(y)));

%generating channel effects

if channel_flag
        if channel_vehic ==1
            ch=veha8(1,1);
            y=conv(y,ch);
```

```
            y_ch=y(ind1:ind2);
            y_ch=y_ch/sqrt(mean(abs(y_ch).^2));
        else
            ch=generate_cir(params);
            y_ch=filter(ch,1,y);
            y_ch=y_ch/sqrt(mean(abs(y_ch).^2));
        end
        %%%%%%%%%%%%%%%%%%%%%%%%
    else
        y_ch=y;1,
end
 signal = SNR * y_ch; % attenuating the generated signal according to
SNR

signal                  =    signal/max(abs(signal));

%% Signal Generator parameters
CenterFrequency         =    2.3054e9; %% Center Frequency
comm                    =    'QAM';
copy                    =    'TTY';
path                    =    'c:\';
name                    =    'QAM16.wv';
play                    =    1;
plot_figure             =    0;
Fs                      =    100e3;
PowerLevel              =    0%-30;

markerlists.one   = [[0 0];[0 0];[0 0];[0 0];[0 0]];
markerlists.two   = [[0 0];[0 0];[0 0];[0 0];[0 0]];
markerlists.three = [[0 0];[0 0];[0 0];[0 0];[0 0]];
markerlists.four  = [[0 0];[0 0];[0 0];[0 0];[0 0]];

PowerLevel              =    SNRdB+noisepowerdb +lossdB %-30;
Reset(SMJ100A);
GenerateWaveform-
ForSMJ(SMJ100A,signal,CenterFrequency,PowerLevel,Fs,comm,copy,path,...
                      name,play,plot_figure,markerlists);

setModulatorStatus(SMJ100A,'ON');
setOutputState(SMJ100A,'ON');
setFrequency(SMJ100A,CenterFrequency);
setPowerLevel(SMJ100A, PowerLevel, 'dBm');
%setImpairments(SMJ100A,gain_imb,phase_imb,Ileakage,Qleakage,'OFF');
setImpairments(SMJ100A, 1, 1, 1, 1, 'OFF')

%% Spectrum Analyzer parameters

RBW                     =    1e3;
Span                    =    1e6;
RefLevel                =    -50;

Reset(FSG);
setFrequency(FSG,1,CenterFrequency);
setSpan(FSG,1,Span);
setRefLevel(FSG,1,RefLevel);
setResBW(FSG,1,RBW);
setTraceMode(FSG,1,1,'WRIT');
dummy(FSG,'DET POS',0);
dummy(FSG,'INP:ATT 0 DB',0);
```