Sudan University of Science and Technology

College of Graduate Studies

DETECTING AND CLASSIFYING LOW PROBABILITY OF INTERCEPT PULSE RADARS SIGNALS

اكتشاف و تصنيف اشارات الرادارات النبضية ذات الاحتمالية الاقل للالتقاط

A Thesis Submitted in Partial Fulfillment of the requirements for the degree of M.Sc.in electronic engineering (communications engineering)

Prepared by:

Ahmed Nidal Ahmed NourEldeen

Supervised by:

Dr. FathElrahman Ismael Khalifa

January 2017

بسم الله الرحمن الرحيم

قال الله تعالي:

وَوَرِثَ سُلَيْمَانُ دَاوُودَ ۖ وَقَالَ يَا أَيُّهَا النَّاسُ عُلِّمْنَا مَنطِقَ الطَّيْرِ وَأُوتِينَا مِن كُلِّ شَيْءٍ ۖ إِنَّ هَٰذَا لَهُوَ الْفَضْلُ الْمُبِينُ

صدق الله العظيم

سورة النمل الآية 16

To My Family.....

ACKNOWLEDGEMENT

I thank god (**ALLAH**) for giving me the endurance and perseverance to complete this work.

I am truly indebted and thankful to my supervisor **Dr. FathElrahman Ismael Khalifa** for his suggestions, criticism and guidance throughout the thesis work.

I could not complete this work without the continuous support of my **family;** I would like to give special thanks to my **family.**

I am very thankful to my friend: **ALI Omer Mohamed Aldow**for his constructive comments and supporting in pursuing this work.

ABSTRACT

Digital intercept receivers are currently moving away from Fourier-based analysis and towards classical time-frequency analysis techniques, such as the Wigner-Ville distribution, Choi-Williams distribution, spectrogram, for the purpose of analysing Low Probability of Intercept (LPI) radar signals.The LPI signals are difficult to detect, intercept and recognize, because they have many combined features implemented that help the LPI radar prevent its detection by modern intercept receivers.The instantaneous frequency measurement (IFM) receiver is normally a subunit used in electronic warfare receivers. Its function is to measure the frequency of pulse and continuous wave signals over a very wide bandwidth and power dynamic range.The band of interest can range from 500 MHz to 40 GHz. This is an extremely wide bandwidth. These receivers have a very high resolution (typically 1 MHz) over this very wide bandwidth and the error of these receivers is typically a few megahertz.In this work we use IFM receiver long with prony's analyses and multiple delay lines to intercept LPI signals. We find that our scheme achieved good results in order to intercept LPI signals even when it appear simultaneously at the receiver and number of simultaneously received signals are function of number of delay lines.

المستخلص

نتمو اجهذة الاستقبال الرقمية في الاونة الاخيرة مبتعده عن من استخدام تحويل فورير وفي اتجاه الطرق الكلاسيكية لتحليل تردد و زمن الاشارات مثل توزيع وينجرفيل, توزيع جوي ويليامز, اسبكتوغرام لغرض تحليل اشارات الرادارات المعقدة. تطبق الرادارات ذات القابلية الاقل للالتقاط العديد من الطرق التي تساعدها علي اخفاء اشاراتها من المستقبلات الحديثة. يستخدم مستقبل التردد اللحظي بشكل واسع في منظومات الحرب الاكترونية. حيث يقوم بقياس تردد الاشارات النبضية و المستمرة في نطاق ترددي و ديناميكي واسع جدا يمتد من 500 ميقاهيرتز ال 40 قيقاهيرتز وهو نطاق واسع جدا. ولهذه المستقبلات دقة عالية جدا (1ميقاهيرتز)ونسبة خطأ في حدود بعض الميقاهيرتز. في هذا البحث استخدمنا مستقبل التردد اللحظي و تحليل بروني مع استخدام عدد من خطوط تاخير الاشارة لحل مشكلة الاشارات ذات القابلية الاقل للالتقاط. وجدنا ان الطريقة اعلاه اعطت نتائج جيدة في استقبال الاشارات المعقده حتي لو وصلت هذه الاشارات بصورة متزامنة حيث وجدنا ان عدد الاشارات المتزامنة المستقبلة داله في عدد خطوط التاخير المستخدمة في المستقبل.

Table of Contents

List of Figures

List of Abbreviations

ADC	Analogue-to-Digital Converter
AOA	Angle Of Arrival
ASP	Antenna Scan Period
AST	Antenna Scan Type
CVR	Crystal Video Receiver
CW	Continuous Wave
DAC	Digital-to-Analogue Converter
DLD	Delay Line Discriminator
DPM	Data-Processing Module
DRFM	Digital Radio Frequency Memory
DSP	Digital Signal Processing
ECCM	Electronic Counter-Countermeasures
ECM	Electronic Counter Measures
ELINT	ELectronic INTelligence
ESM	Electronic Support Measures

EW	Electronic Warfare
FD	Frequency Discriminator
FFT	Fast Fourier Transform
FPGA	Field-programmable Gate Array
IF	Intermediate Frequency
PD	Phase Discriminator
POI	Probability of Intercept
PRF	Pulse Repetition Frequency
PRI	Pulse Repetition Interval
PW	Pulse Width
RADAR	RADIO DETECTION AND RANGING
RF	Radio Frequency
RWR	Radar Warning Receiver
SNR	Signal-to-Noise Ratio
SSD	Simultaneous Signal Detection
TOA	Time Of Arrival
VCO	Voltage Controlled Oscillator

List of symbol

ρ	signal amplitude
φ	phase delay
β	phase constant
λ	Wavelength
L	Length
v	Velocity
f	Frequency
k	Constant
f_{BW}	Bandwidth
$f_{BW_{MAX}}$	maximum unambiguous bandwidth
τ	delay time
V	Amplitude
M_1	mixing product
M_2	mixing product
A	Amplitude
ω	angular frequency

j	imaginary unit
t	Time
P	Power
exp	Exponential
a_1	Constant
a_2	Constant
z_i	Variable

List of Table

Chapter One

Introduction

1.1 Preface

The word RADAR is an abbreviation for "RAdio Detection And Ranging" .In general radar system use modulation waveforms and directive antenna to transmit electromagnetic energy into specific volume in space to search for targets. The range, or distance, to the target is found from the time it takes for the radiated energy to travel to the target and back. The targets are then processed by the radar receiver to extract target information such as velocity, angular position and other target identifying characteristics [1].

The principle of radar employs the fact that some materials reflect electromagnetic radiation (“ER”). Examples of objects that reflect ER are most metallic items, the Earth’s ionosphere, ionized “trails” behind meteors, satellites, re-entering space debris, the Moon, the surface of the Earth, migratory birds, etc… Many materials, e.g. air, wood, plastic, most glass, etc., may not reflect electromagnetic radiation. Whether a target reflects a radar signal can depend on its size, the material it is made of, and the frequency of the impinging electromagnetic radiation, etc...[2].

Most radar, such as surveillance and target tracking radars have to contend with very capable and advanced threats on today’s battlefields. These threats range from Anti-Radiation Missiles (ARMs), Radar Warning Receivers (RWRs), Electronic-warfare Support (ES) interception capabilities, and Electronic Attack (EA) systems. All of these are designed to contribute to the degradation of radar performance by jamming, evasion, or destruction [3].

To survive these countermeasures and accomplish their missions, radars have to hide their emissions from hostile receivers. For this purpose, and to mask their presence, radars use power management, wide operational bandwidth, frequency agility, antenna side lobe reduction, and advanced scan patterns (modulations). These types of radars are called Low Probability of Intercept (LPI) radars and they use techniques “to see and not to be seen” by modern and capable intercept receivers [4].

Modern electronic intercept systems must perform the tasks of detection, classification, identification and exploitation in a complex environment of high noise, interference and multiple signals. Some waveforms are intentionally designed to make the detection process nearly impossible. Such signals are referred to as (LPI) waveforms [5].

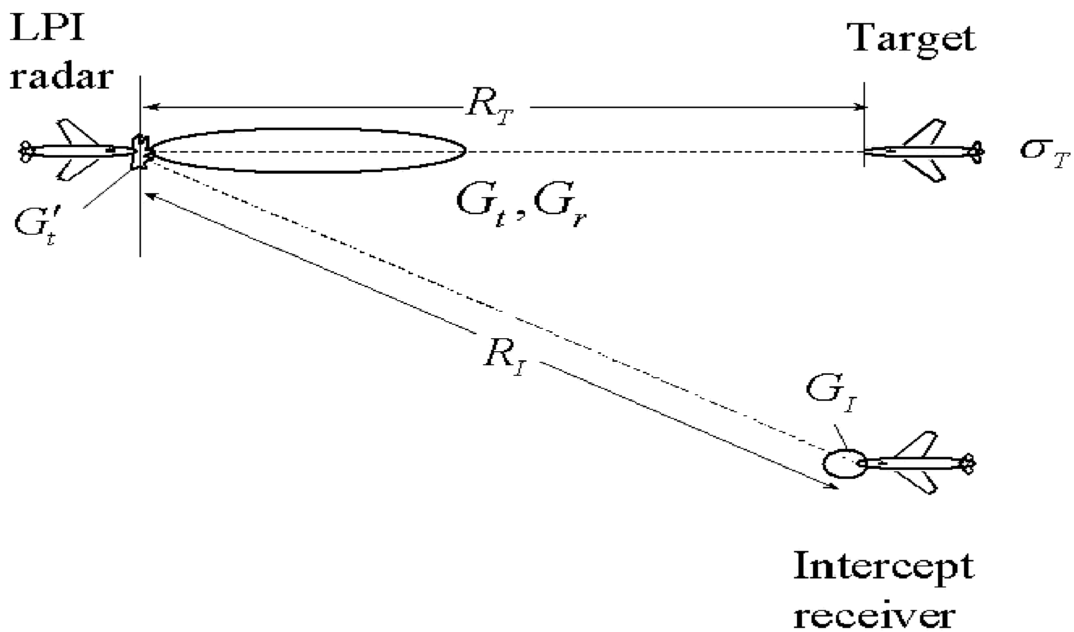

Figure 1.1: LPI radar and intercept receiver configuration**[5]**.

Parameters such as carrier frequency, modulation type, data rate and time or angle-of-arrival are just a few of the fundamental features that distinguish one signal from another, The sorting and cataloging of signals leads to the process of identification. The task of classification requires sorting into groups having similar characteristics. Each of these initial processes: detection, classification identification and exploitation require advanced signal processing techniques. A combination of FPGA (pre-processing) and DSP processor (post processing) is being used to extract all the parameters of LPI radar. The complete information of a pulse is embedded in the form of a Pulse Descriptor Word which is further processed to display the parameters of an emitter on ESM display.

1.2 Problem Statement

The LPI signals are difficult to detect, intercept and recognize, because they have many combined features implemented that help the LPI radar prevent its detection by modern intercept receivers.

1.3 Proposed Solution

In order todetect Low Probability of Intercept pulse Radar, an electronic warfare scheme will be developed and tested via MATLAB simulation program. According to the detection process the intercepted signal will be analyzed and classification takes place.

1.4 Objectives

The objective of this research is to design a signal processing scheme capable of detecting LPI radar signals using electronic ware fare receiver.

To achieve this mathematical model, a detection process will be developed and correlation parameter will be calculated, this task will be accomplished in the digital signal processing stage.

1.5 Research Scope

The scope of this thesis is electronic warfare scope we concentrate on investigating methods and means to counter LPI pulse radar threats integrated into a modern platforms and weapons and focus on the related techniques, strategies, and technology.

The interception of LPI signal was incoherent technique and we don’t have a control on the signal that needs extensive processing in order to extracting the radar parameter from signals.

1.6 Methodology

Articles, books, periodicals, thesis and IEEE documents related to the subject will be collected and thoroughly examined. Then parameters of LPI signals will be calculated and profile of signal will present as a discrete time signal this is the detection parte as digital signal processing.

The classification part will be achieved by applying correlation through the detected profiles. That needs extensive digital signal processing at the ES receiver.

LPI radars use the basic idea of the spread radiation in the time domain and frequency domain (that called broad signal) to achieve a power spectrum density below the noise level in interception receiver input. Detecting LPI signals need high signal processing gain. The gain of signal processing is usually obtained in digital signal processing.

1.7 Thesis Outlines:

Chapter 1: Present an Introduction to the LPI radar, LPI signal and Electronic Warfare concept.

Chapter 2: Briefly describes the EW techniques, characteristics and waveforms used in this thesis work.Important parameter of radar signals and radar working mechanisms such as PDW, scan and PRI types are discussed. A brief discussion of the related works and introduction of basic and digital IFM receiver is the last part of this chapter.

Chapter 3: Describe detection methods of LPI radars. For this purpose ES receivers and signal processing algorithms are examined in detail. Uses of prony's method and delay line are described in this chapter.

Chapter 4: Analyzes of prony's method along with using of single and multiple delay lines are described with one, two, three and four simultaneous signals are tested. .

Chapter 5: Summarize the conclusions and recommendations of thesis.

Chapter Two

Literature Review

2.1 Theoretical Background

This chapter presents about all concepts and parameter definition needed to understand mathematical model and simulation work of the thesis. Also summarize the previous works that done by the researcher's in the same scope.

2.1.1 Electronic Warfare (EW)

Electronic Warfare (EW) is a military action involving the use of electromagnetic energy to determine, exploit, reduce or prevent hostile use of the electromagnetic spectrum and to maintain friendly use of spectrum. One possible task of an EW system is to sort and classify received pulses from a dense environment of hostile RADARs so that the pulses can be processed; this process is known as pulse deinterleaving. The deinterleavig process needs to be completed with very low latency to support timely decision making required in modern EW environments.

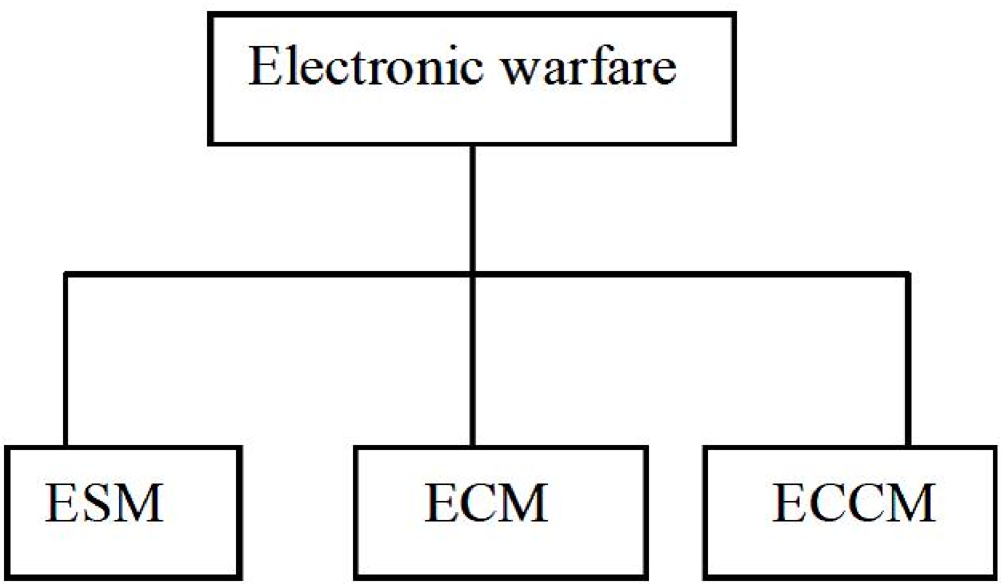

Figure 2.1: Classical Electronic Warfare

EW consists of three main principal elements, which are also illustrated in Figure 2.1:

- Electronic Support Measures (ESM): Gathering and immediate analysis of electronic emission of weapon systems to determine a proper and immediate reaction. Also called Electronic Support (ES)
- Electronic Counter Measures (ECM): Development and application of equipment and tactics to deny enemy use of electromagnetically controlled weapons. Also called Electronic Attack (EA)
- Electronic Counter-Countermeasures (ECCM): Actions necessary to ensure use of the electromagnetic spectrum by friendly forces. Also called Electronic Protection (EP)

The goal of deinterleaving is to classify RADAR signals by their unique characteristics and use this data to [6]:

- Identify enemy RADARs operating in the environment,
- Determine their location or direction,
- Inform friendly forces about their threats, Display this information to the operator.

Figure 2.2 show a simplified block scheme of a typical EW system, in the Figure the top two plots represent periodic pulse trains emitted from two individual RADARS. The center plot shows how the interleaved signals will appear at the EW receiver. The bottom plots represent the successful deinterleaving of the received signal, which in a perfect scenario, should identically match the top plots.

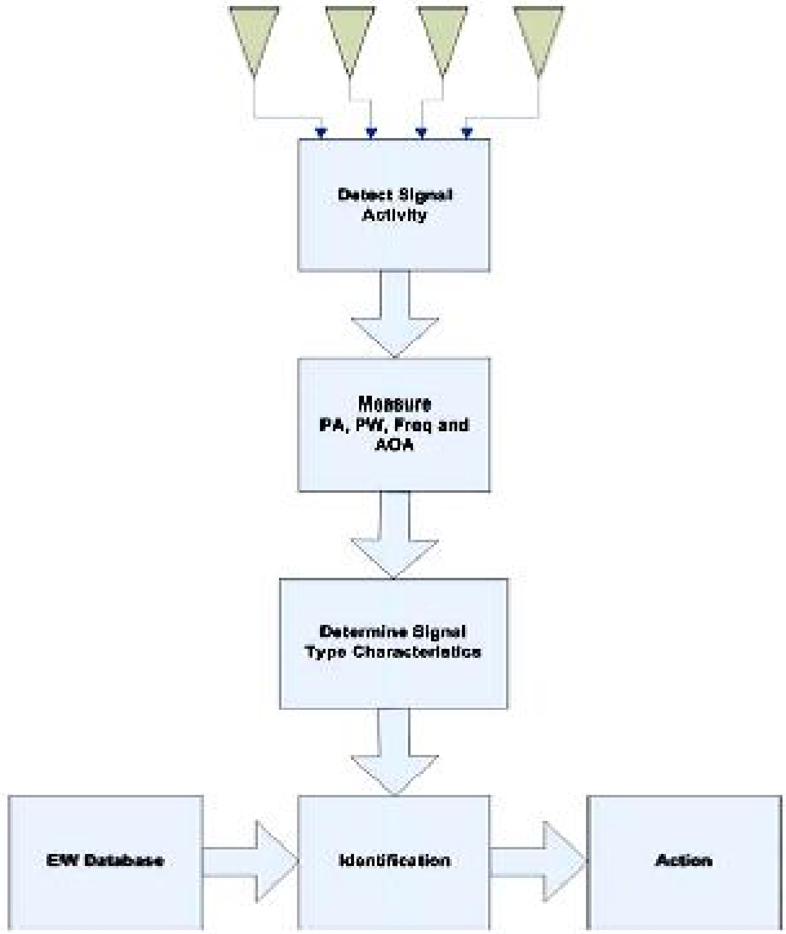

Figure 2.2: ESM block diagram

2.1.2 Pulse Descriptor Word (PDW)

ESM receivers measure pulse characteristics as each pulse is received. For every pulse the measured parameters are packed into a structure called a Pulse Descriptor Word (PDW) and passed along for emitter identification processing. PDWs often contain information on Pulse Amplitude (PA), carrier Frequency (CF), Pulse Width (PW), Time Of Arrival (TOA), and Angle Of Arrival (AOA).

Although it cannot be measured directly on an interleaved pulse train, another important characteristic useful in correlating a pulse to a particular emitter is Pulse Repetition Interval (PRI) or its reciprocal, Pulse Repetition Frequency (PRF). Figure 2.3 illustrates some of these parameters.

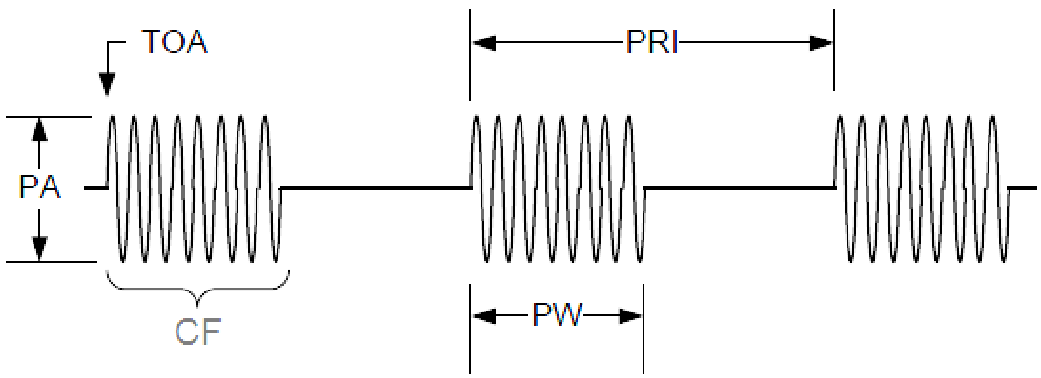

Figure 2.3 : RF Pulse Parameters

2.1.2.1 Angle of Arrival (AOA)

Angle of arrival is a measurement which determines the direction of the transmitter with respect to the receiver. Angle of arrival is generally emphasized as the best sorting parameter for clustering process. The reason for this notification is that it cannot be varied rapidly by the hostile RADARs from pulse to pulse. It is a fact that even airborne RADARs cannot change their location in a few milliseconds of the PRI time, so the AOA measurement by an intercept receiver on the RADAR is relatively stable [7].

2.1.2.2 Carrier Frequency (CF)

In many of pulse parameter of clustering applications, it is emphasized that the carrier frequency is the next most important parameter after AOA [8]. Carrier frequency is the sinusoidal frequency of a particular pulse. The major advantage of frequency parameter is highlighted if it is noticed that

RADARs physically near to each other cannot operate on the same frequency.

2.1.2.3 Time of Arrival and Pulse Repetition Interval (TOA &PRI)

Time of arrival is the system timestamp corresponding to the start of a particular pulse. TOA is an important measurement required for determining the PRI, or the time period between successive transmitted pulses from particular RADAR. PRI determines the maximum unambiguous range of the RADAR [8].

2.1.2.4 Pulse Width (PW)

Pulse width is the duration of time for which a pulse is being transmitted. Most Radar has fairly low duty-cycles in order to keep the average output power significantly low when compared with its peak output power [8].

PW is accepted as a less effective clustering parameter because many types of RADARs are similar in this respect and also PW varies with pulse amplitude. Multipath situations may also cause variations in measured PW values. PW parameter cannot be used for separating same type of RADARs whose pulses are interleaved [6].

2.1.2.5 Pulse Amplitude (PA)

Pulse amplitude is simply the strength of the received signal as seen by the receiver. The PA can be heavily influenced by the signal environment and the transmitter / receiver geometry.

The pulse amplitude is commonly defined as the peak value of the received RADAR pulse. It is generally used along with TOA for deriving the scan pattern of the RADAR and not used for clustering. First of all, PA is not a reliable parameter for clustering because of its variability within a pulse train due to antenna scanning. However, PA has a major advantage since it is not changed too much from pulse-to-pulse [6].

Table 2.1 : parameters of radar signal[6]

	Frequency (MHz)	**Angle of Arrival (Degree)**	**Pulse width (µs)**	**PulseRepetition Interval (µs)**	**Max Pulse Amplitude (dBm)**	**Pulse Count**
1	4000.7	50.0	3.0	170.0	-22.7	1742
2	4500.0	30.0	7.0	215.0	-19.4	1518
3	8400.4	20.0	11.0	220.2	-29.7	516
4	4500.0	50.0	3.0	300.0	-24.7	7107
5	3000.0	45.0	7.0	200.0	-31.6	97
6	3000.0	45.0	7.0	450.0	-30.4	60
7	3000.0	45.0	7.0	749.9	-29.5	50
8	3000.0	45.0	7.0	979.9	-31.3	43
9	3000.0	35.0	15.0	800.0	-19.1	6456
10	3000.2	65.1	7.0	1136.8	-35.2	62
11	4999.8	50.0	7.0	1425.0	-19.8	513
12	5500.2	50.1	7.0	1425.0	-19.9	513
13	6000.2	50.0	7.0	1425.0	-19.6	516

According to the parameters value like in table 2.1 the classification take place. The grater RF signed to smaller antenna that used for naval platforms the smallest RF signed to grater antenna used for ground radars. In this thesis we classifying according to RF because of the reasons mentioned above.

2.1.3 Types of RADAR PRIs

The military technology is growing with a high speed. As a result, many different types of RADAR have been developed and are still being developed. ESM systems have to be improved in order to catch up with the development of RADARs. This work is focused on three types of RADAR PRI modes, which are well known in EW: Stable PRI mode, and Stagger PRI mode. Jittered PRIs are not taken as a separate PRI mode, because this type of PRI sequence can be detected by the same methods of Stable PRI mode detection. These PRI modes are explained in figure 2.5[9].

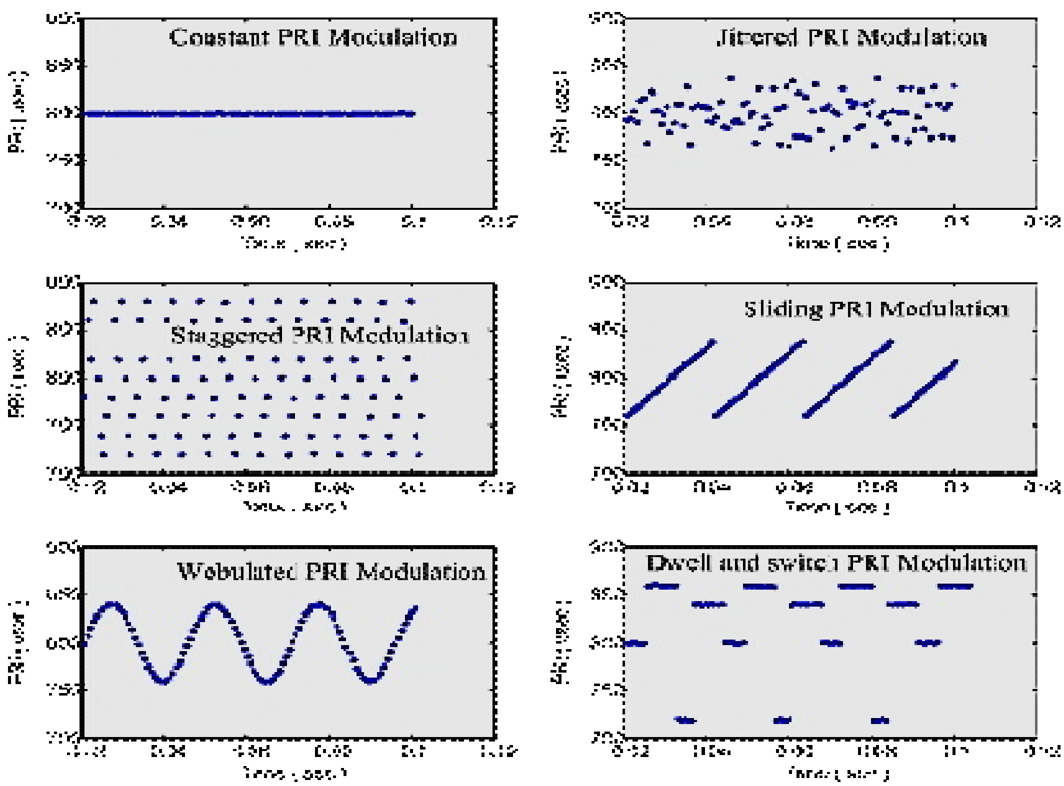

Figure 2.4:PRI types**[9]**

2.1.4 Scanning

Scanning is "a programmed motion given to the major lobe of an antenna for the purpose of searching a larger angular region than can be covered with a single direction of the beam, or for measuring the angular location of a target. There are two basic ways of classifying scanning methods[10]:

- From the viewpoint of principles of beam steering (another term for scanning), the methods are described as mechanical, electromechanical, or electronic.
- From the viewpoint of the type of beam motion introduced to scan a volume, the methods are described as Circular scan, helical scan, Raster scan, Palmer scan, Conical scan, Spiralscan and so forth. Some of those scan types are shown in the Figures below.

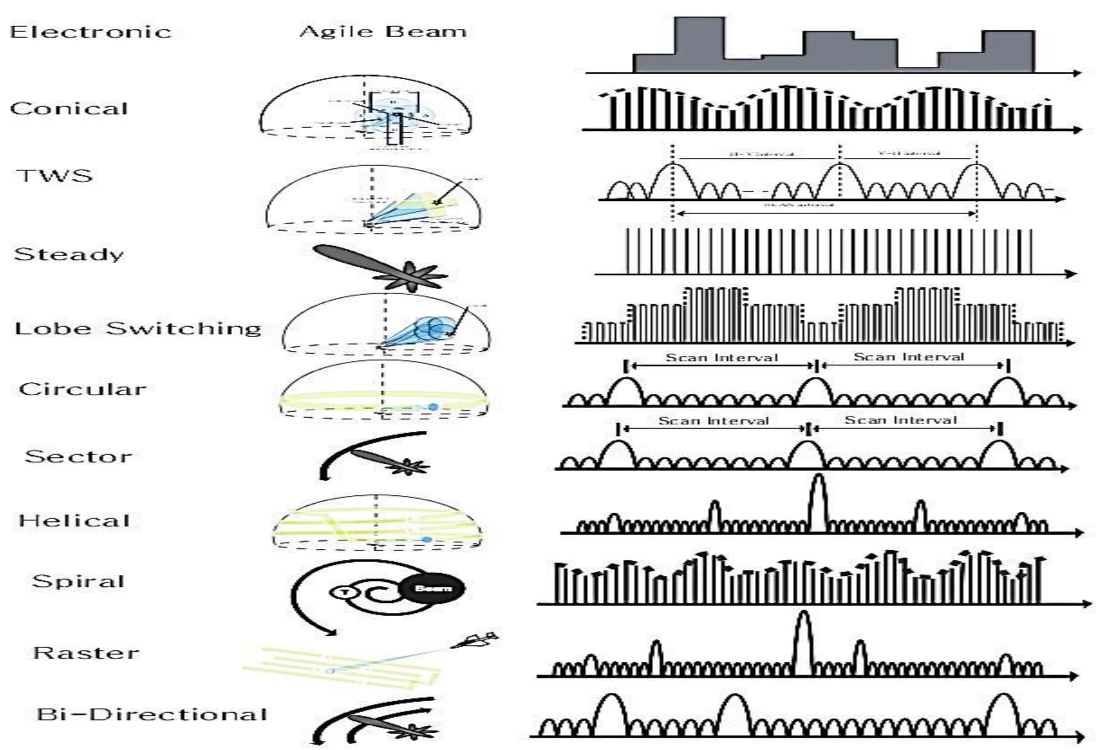

Figure 2.5:The Scan Type [10]

2.2 Related works

In [11], the author have useddeferent mechanisms to circumvent this problem, they show how to use multiple spatially distributed sensors to detect and localize an emitter whose waveform is completely unknown. They present the generalized likelihood ratio detector which optimally combines the multiple sensor information for improved detection. Additionally, as part of the detector the Maximum likelihood Estimators MLE for target location is available, leading to improved localization.

The authors in [12] were able to deal with LPI signal by the mean of Higher Order Spectral Analysis (HOSA) techniques enabling them to extract much more information from the same intercept and hence facilitating detection. Their paper reports the results of HOSA techniques (Bi-spectrum, Bi-coherence and Tri-spectrum) applied to LPI Radar signals. Bi-phase Barker coded signals of different lengths, P1, P2, P3 and P4 Poly-phase coded signals and Frank signal are analyzed using HOSA techniques to produce 2-D signatures of these signals which serve as reference for computing correlation coefficient with respect to the similar plots obtained for an unknown received signal. Their system identifies the type of the signal by the maximum value of correlation coefficient obtained. The results obtained clearly indicated the promising capability of this technique to identify the type of LPI signal with SNRs as low as –3 dB and even lower.

Authors in [6] and [13] were able to presents the state of the art single board Digital Receiver solution for intercepting and analyzing complex radarsignals. Also the pre and post processing methodologies werediscussed from both the algorithmic as well as hardware point of view.

2.3 Basic IFM Receiver Principles

IFM receivers will be covered in more detail in the following sections.

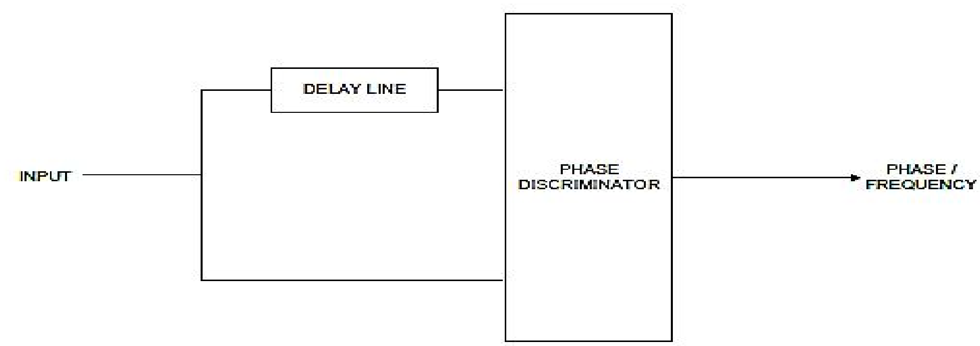

Figure 2.6 : Basic IFM receiver [14]

The basic principle of frequency measurement in an IFM receiver is comparing the direct and the delayed versions of a signal using a phase discriminator (PD). The phase difference is directly proportional to frequency. Assuming no phase modulation, this method approximates the true instantaneous frequency as the delay approaches zero, hence the name IFM [14-18].

2.4 Basic IFM Receiver Components

Figure 2.7 shows the components of a basic IFM receiver [4].

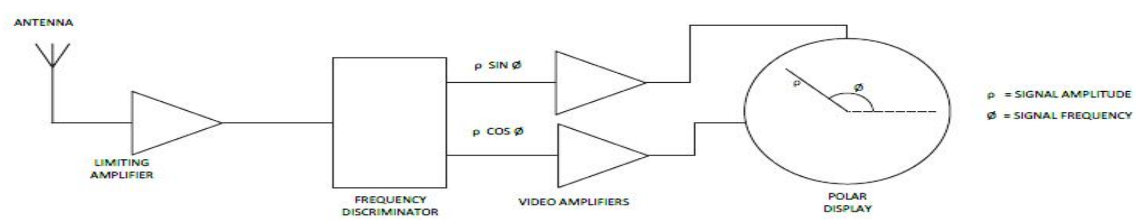

Figure 2.7 : Basic IFM receiver components [4]

2.4.1 Limiting Amplifiers

Limiting amplifiers are the answer to both dynamic range and sensitivity problems. Low power input signals can then be considerably amplified, whereas high power signals drive the amplifier to limiting so that one's FDsare not overdriven. The requirements placed on these limiting amplifiers are harsh, making them a very expensive module in an IFM receiver.

2.4.2 Frequency Discriminator

The function of the FD is to accept an RF input signal and produce two video signals, $\rho \sin \varphi$ and $\rho \cos \varphi$, ρ is a measure of the signal amplitude and φ is directly proportional to frequency.

The FD block can be broken up into a power splitter, delay line and PD, as shown in Figure 2.8. In this form it is often called a delay line discriminator (DLD). It makes use of the well-established principle of measuring frequency in terms of phase delay when a signal is propagated down a transmission line of known length. Almost all IFM receivers reported in this literature make use of DLDs and were pioneered at Mullard Research Laboratories in England approximately 60 years ago [17]. The next subsections will discuss the DLD in more detail.

2.4.2.1 Basic Principles of the Delay Line Frequency Discriminator

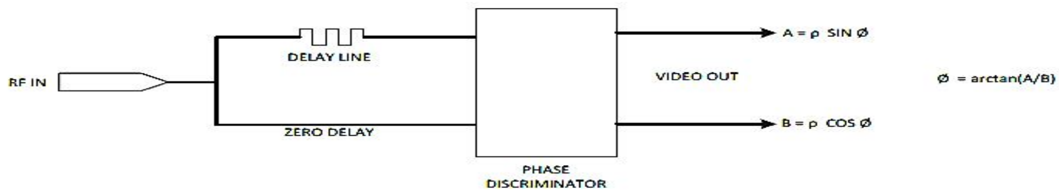

Figure 2.8 : Delay Line Discriminator [17]

If a delay line of length L and phase constant β is used, the phase delay φ is given by

$$\varphi = \beta L \tag{2.1}$$

If the signal wavelength is λ, then

$$\beta = \frac{2\pi}{\lambda} \tag{2.2}$$

If v is the velocity of propagation and f is the signal frequency, then

$$\lambda = \frac{v}{f} \tag{2.3}$$

Substituting (2.3) into (2.2)

$$\beta = \frac{2\pi}{v} f \tag{2.4}$$

Substituting (2.4) into (2.1)

$$\varphi = \frac{2\pi}{\lambda} f L \tag{2.5}$$

Alternatively

$$\varphi = k f \tag{2.6}$$

where k is a constant

$$k = \frac{2\pi}{v} L \tag{2.7}$$

Hence phase delay is directly proportional to signal frequency and can be used as a direct measure of frequency.

If $f_1$and f_2 are the edge frequencies of the DLD's pass band, and $f_2 > f_1$, then from (2.6)

$$\varphi_1 = kf_1 \tag{2.8}$$

$$\varphi_2 = kf_2 \tag{2.9}$$

so that

$$\varphi_2 - \varphi_1 = k(f_2 - f_1) \tag{2.10}$$

The requirement for no ambiguity is

$$\varphi_2 - \varphi_1 < 2\pi \tag{2.11}$$

If f_{BW} is the bandwidth of the DLD. i.e. $f_{BW} = f_2 - f_1$, then from (2.10)

$$f_{BW} = \frac{\varphi_2 - \varphi_1}{k} \tag{2.12}$$

The maximum unambiguous bandwidth is

$$f_{BW_{MAX}} = \frac{2\pi}{k} \tag{2.13}$$

Substituting k for from (2.7) one gets

$$f_{BW_{MAX}} = \frac{v}{L} \tag{2.14}$$

Equation (2.14) shows that the shorter the delay line length, the larger the unambiguous bandwidth. The frequency accuracy of the DLD then depends on the accuracy of the PD. If phase inaccuracies could be reduced without limit there would be no conflict between frequency accuracy and unambiguous bandwidth. In practice, however, an accuracy of even two degrees would be very difficult and expensive to realize. From (2.5), an indication of the frequency accuracy is given by

$$\frac{d\varphi}{df} = \frac{2\pi}{v}L \tag{2.15}$$

Equation (2.15) shows that the longer the delay line, the more accurate the frequency measurement. This is in direct conflict with unambiguous bandwidth requirements and for a receiver of the type a trade-off would have to be chosen.

Work on building a broadband "wide-open" or non-tuning FD began at Mullard Research Laboratories in 1954. First results were not very successful, with the relationship between frequency and any other derived parameter being too non-linear and unrepeatable for accurate measurement.

The break-through and birth of modern delay lines FDs came in 1957 when the identity $\cos^2\left(\frac{\varphi}{2}\right) - \sin^2\left(\frac{\varphi}{2}\right) = \cos\varphi$was used to measure the phase delay φdown a delay line of known length, this being directly related to frequency. A second such operation employing an additional delay of $\frac{\pi}{2}$radians at all frequencies provided the orthogonal $\sin\varphi$component.

Figure 2.9 shows the basic components of a DLD. The delay line causes a delay $\tau$and the delayed signal is multiplied with a direct signal and also with a signal shifted a fixed $\frac{\pi}{2}$radians to give the sine and cosine outputs after low-pass filtering, as shown.

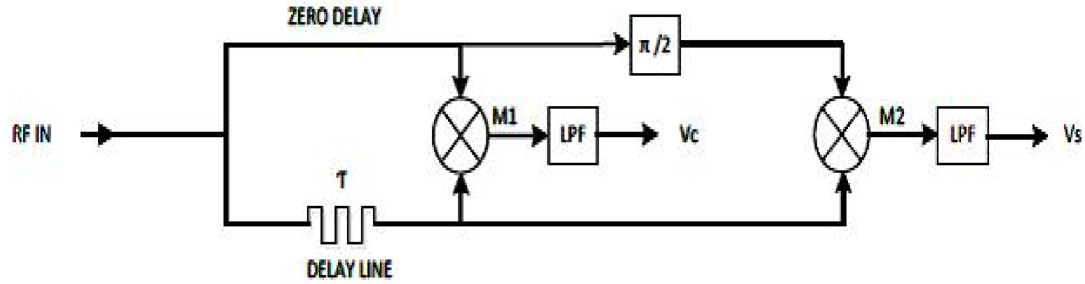

Figure 2.9 : Basic frequency discriminator [17]

Consider a fixed frequency, constant amplitude, sinusoidal input signal $V\cos(2\pi ft)$where f is the signal frequency and V its amplitude. The output of the delay line is then $V\cos(2\pi ft-\varphi)$ where $= 2\pi f\tau$.

The mixing product $M_1$is given by

$$M_1 = V\cos(2\pi ft) \times V\cos(2\pi ft-\varphi) \tag{2.16}$$

By using the product and sum trigonometric formulas,M_1 is also

$$M_1 = \frac{V^2}{2}\cos(\varphi) + \frac{V^2}{2}\cos(4\pi ft-\varphi) \tag{2.17}$$

Low-pass filtering rejects the sum frequency so that the output V_C is given by

$$V_C = \frac{V^2}{2}\cos(\varphi) \tag{2.18}$$

The inputs to the second mixer are $V\cos\left(2\pi ft-\frac{\pi}{2}\right)$ and$V\cos(2\pi ft-\varphi)$.

The mixing product $M_2$is then

$$M_2 = V\cos\left(2\pi ft-\frac{\pi}{2}\right) \times V\cos(2\pi ft-\varphi) \tag{2.19}$$

or

$$M_1 = \frac{V^2}{2}\cos\left(-\frac{\pi}{2}+\varphi\right) + \frac{V^2}{2}\cos\left(4\pi ft-\frac{\pi}{2}-\varphi\right) \tag{2.20}$$

Low-pass filtering and the use of identity $\cos\left(-\frac{\pi}{2}+\varphi\right) = \sin(\varphi)$ gives V_s as

$$V_s = \frac{V^2}{2}\sin(\varphi) \tag{2.21}$$

2.4.3 Polar Display

If the outputs from the FD $\rho \sin \varphi$ and $\cos \varphi$, are applied to the vertical and horizontal plates of a polar display, is the radial component and the angular component φ can be scaled to read frequency directly. The FD is often referred to as a polar frequency discriminator because of the sine and cosine terms. Any amplitude or frequency modulation that is present will show up on the display as radial and angular fluctuations respectively.

It should be noted that although the polar display concept sounds relatively easy, it is a very complex task to take the outputs from the FD(s) and produce the correct frequency value. This effort should not be underestimated.

2.5 Digital IFM Receiver

Figure 2.10 is a block diagram of a typical digital IFM (DIFM) receiver. It shows the main functional elements. Most traditional IFM receivers will be of this form.

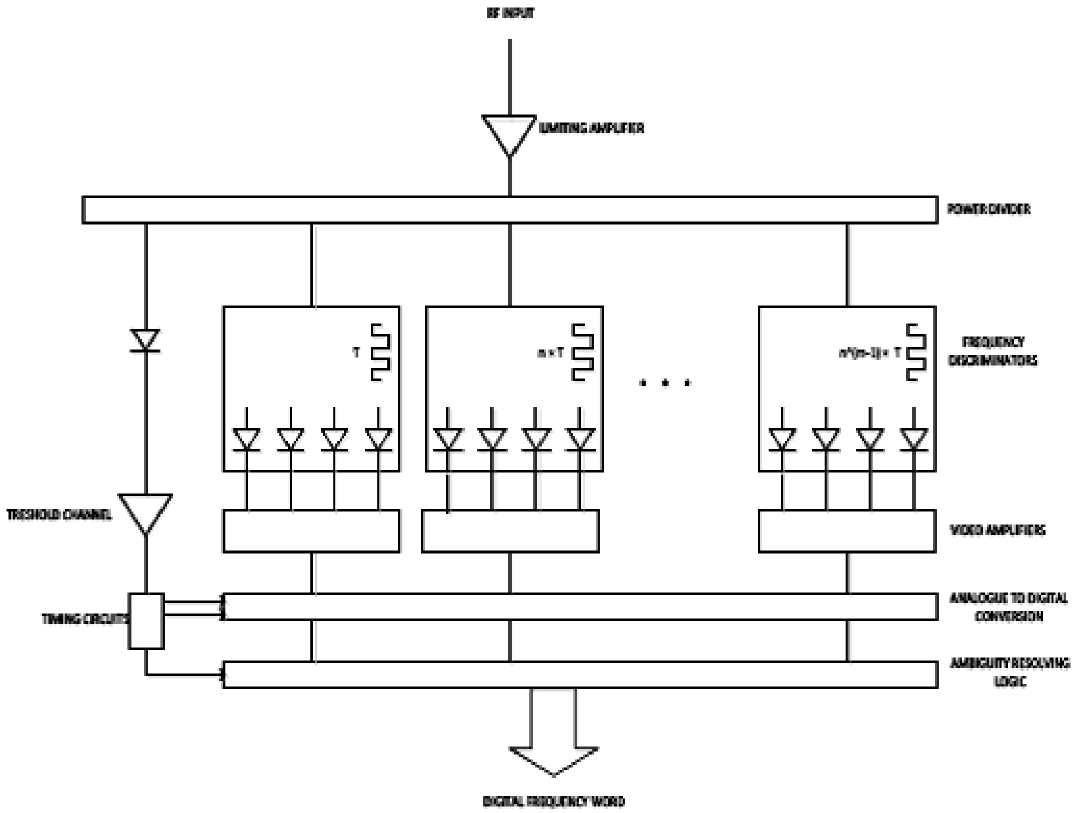

Figure 2.10 block diagram of a typical digital IFM (DIFM) receiver [18]

The filter coarsely defines the operating band of the RF input. The RF amplifier generally has relatively poor gain roll-off characteristics and the main function of the filter is to prevent the generation of spurious signals within the RF amplifier by high-power signals outside the operating band.

Typically 60 dB of RF gain may be required for maximum sensitivity. The amplifier should exhibit good pulse fidelity, low harmonic levels and good limiting at high signal levels. The need for limiting was discussed in 2.4.1. The broadband power divider ensures that the signal is evenly split to the discriminators and the threshold channel. The video amplifiers may be AC-coupled, DC-restored or DC amplifier types. AC-coupled amplifiers are the simplest, but are insensitive to CW and high duty-cycle signals. DC types respond to CW but are affected by RF amplifier noise levels and may be blocked when CW signals are present.

The threshold channel detects the presence or arrival of signals and co-ordinates the analogue-to-digital conversion and logic circuitry sampling. The logic network combines the digital codes from the discriminator quantizes to provide an accurate unambiguous code representing signal frequency. Some form of error correction or temperature compensation may be included here, but processing must be fast to allow a high pulse handling rate. The typical processing time for 10 to 12 bit frequency data is 150 ns [17, 19].

It was found that the term DIFM is ambiguous. Literature was found which also explains a DIFM as an all-digital implementation of the IFM in a FPGA [20], which is a not exactly as explained above.

Chapter Three

Methodology

3.1 Introduction

Previous chapter discus the general terminology of electronic warfare and radar signals parameter that helps to classify RADAR emitters and the classifying process. Extensive backgrounds of IFM receiver are discussed as the hart of EW system. In this chapter we try to investigate the solution to make IFM receiver capable to intercept LPI signals.

3.2 Investigation of a possible solution

However, no evidence of the use of this method could be found in current commercial IFM receivers, indicating that the implementation of Prony’s method is the difficult part.

For this method the FDs are exactly the same as in the traditional IFM receiver. The differences are in the choice of delay lines and the ambiguity-resolving (frequency resolving) part.

3.2.1 Prony’s Method

Prony analysis (Prony's method) was developed by Gaspard Riche de Prony in 1795. However, practical use of the method awaited the digital computer. Similar to the Fourier transform, Prony's method extracts valuable information from a uniformly sampled signal and builds a series of damped complex exponentials or sinusoids. This allows for the estimation of frequency, amplitude, phase and damping components of a signal.

3.2.1.1 Discriminator Outputs and Simultaneous Signal Condition

The FD in an IFM receiver consists of power splitters, delay lines, couplers and video detectors. The output of the discriminator with delay time τ and input signal $A\sin(\omega t)$where A is the amplitude and ω is the angular frequency can be written as

$$R_r(\tau) = \frac{A^2}{2}\cos(\omega\tau) \tag{3.1}$$

$$R_i(\tau) = \frac{A^2}{2}\sin(\omega\tau) \tag{3.2}$$

The output of the discriminator can also be written in complex form as

$R(\tau) = R_r(\tau) + jR_i(\tau) = P\,exp(j\omega\tau)$ where $j$represents the imaginary unit, which satisfies the equation $j^2 = -1$, the input frequency can be found as $\omega = (1/\tau)\tan^{-1}(R_i/R_r)$As explained in the case of the typical IFM receiver.

If one considers the case where two signals are present $V_1\cos(2\pi f_1 t)$ and $V_2\cos(2\pi f_2 t)$, where $f_1$is the signal frequency and $v_1$is the amplitude of the one signal and $f_2$is the signal frequency and $v_2$is the amplitude of the other signal.

Then equation (2.17) can be written as

$$M_1 = \left(\frac{{V_1}^2}{2} + \frac{{V_2}^2}{2}\right)\cos(\varphi) + \frac{{V_1}^2}{2}\cos(4\pi f_1 t - \varphi) + V_1V_2\cos(2\pi(f_1 + f_2)t - \varphi) + V_1V_2\cos(2\pi(f_1 - f_2)t + \varphi) + \frac{{V_2}^2}{2}\cos(4\pi f_2 t - \varphi) \tag{3.3}$$

And equation (2.20) as

$$M_2 = \left(\frac{{V_1}^2}{2} + \frac{{V_2}^2}{2}\right)\cos\left(-\frac{\pi}{2} + \varphi\right) + \frac{{V_1}^2}{2}\cos\left(4\pi f_1 t - \frac{\pi}{2} - \varphi\right) + V_1V_2\cos\left(2\pi(f_1 + f_2)t - \frac{\pi}{2} - \varphi\right) + V_1V_2\cos\left(2\pi(f_1 - f_2)t - \frac{\pi}{2} + \varphi\right) + \frac{{V_2}^2}{2}\cos\left(4\pi f_2 t - \frac{\pi}{2} - \varphi\right) \tag{3.4}$$

If the frequency separation of the two input signals is far apart, such that the difference frequency $(f_1 - f_2)$ will also be filtered out by the low-pass video filter in the discriminator, the outputs from the discriminator are

$$R_r(\tau) = P_1\cos(\omega_1\tau) + P_2 \tag{3.5}$$

$$R_i(\tau) = P_1\sin(\omega_1\tau) + P_2\sin(\omega_2\tau) \tag{3.6}$$

If equation$R(\tau) = R_r(\tau) + jR_i(\tau) = P\,exp(j\omega\tau)$is used to find the frequency, the result will be erroneous.

3.2.2 Two Simultaneous Signals

A solution for the simultaneous signals problem in traditional IFM receivers can be found in Prony's method. In order to solve the simultaneous problem, more discriminators with different delays are needed. For four discriminators with different delays τ,2τ,3τ and 4τ , the outputs from the discriminators can be written as:

$$R(\tau) = P\_1\, exp(j\omega\_1\, \tau) + P\_2\, exp(j\omega\_2\, \tau) \quad (3.7a)$$

$$R(2\tau) = P\_1\, exp(j\omega\_1\, 2\tau) + P\_2\, exp(j\omega\_2\, 2\tau) \quad (3.7b)$$

$$R(3\tau) = P\_1\, exp(j\omega\_1\, 3\tau) + P\_2\, exp(j\omega\_2\, 3\tau) \quad (3.7c)$$

$$R(4\tau) = P\_1\, exp(j\omega\_1\, 4\tau) + P\_2\, exp(j\omega\_2\, 4\tau) \quad (3.7d)$$

Linear prediction can be used to predict future values of time-discrete signals. If linear prediction is used, one can define two constants, a_1 and a_2, such that they satisfy the following equations:

$$-R(4\tau) = a_1 R(3\tau) + \; a_2 R(2\tau) \quad (3.8a)$$

$$-R(3\tau) = a_1 R(2\tau) + \; a_2 R(\tau) \quad (3.8b)$$

From the above equation, Cramer's rule can be used to solve a_1 anda_2. By substituting equation (3.7) into equation (3.8), it can be shown that

$$a_1 = -[exp(j\omega_1 \tau) + \; exp(j\omega_2 \tau)] = -(z_1 + z_2) \quad (3.9a)$$

$$a_2 = exp[j(\omega_1 + \omega_2)\tau] = z_1 z_2 \quad (3.9b)$$

where

$$z_i = \exp(j\omega_i \tau) \quad (3.10)$$

and $= 1,2$. If z_i can be solved, the frequencies of ω_1 and $\omega_2$can be obtained.

From equation (3.9), it can be seen z that will satisfy the following function as

$$\begin{aligned}(z - z_1)(z - z_2) &= z^2 - \; (z_1 + \; z_2)z + \; z_1 z_2 \\ &= z^2 + \; a_1 z + a_2 = 0\end{aligned} \quad (3.11)$$

Therefore one can find z by solving the above equations.

One can summarize the steps of solving two simultaneous signals as follows:

- Four discriminators with proper delays are needed.
- The values of $a_i$can be calculated.
- From equation (3.11), the values of z can be calculated and the individual frequency can be calculated from equation (3.10).

In this case where two simultaneous signals are expected, but only one signal is present, the problem can be solved from:

$$\det(R) = \begin{vmatrix} R(3\tau) & R(2\tau) \\ R(2\tau) & R(\tau) \end{vmatrix} = 0 \tag{3.12}$$

By doing this check, it can be determined if there is one signal or two signals. Noise in the system will cause the check to be very small but not equal to 0. For this condition, the check should be tested against a predetermined value.

3.2.3 Three Simultaneous Signals

In the case of three simultaneous signals, one can choose six discriminators with different delaysτ, 2τ, 3τ, 4τ, 5τ and 6τ . The outputs from the discriminators can be written as

$$R(\tau) = P_1 exp(j\omega_1\tau) + P_2 exp(j\omega_2\tau) + P_3 exp(j\omega_3\tau) \tag{3.13a}$$

$$R(2\tau) = P_1 exp(j2\omega_1\tau) + P_2 exp(j2\omega_2\tau) + P_3 exp(j2\omega_3\tau) \tag{3.13b}$$

$$R(3\tau) = P_1 exp(j3\omega_1\tau) + P_2 exp(j3\omega_2\tau) + P_3 exp(j3\omega_3\tau) \tag{3.13c}$$

$$R(4\tau) = P_1 exp(j4\omega_1\tau) + P_2 exp(j4\omega_2\tau) + P_3 exp(j4\omega_3\tau) \tag{3.13d}$$

$$R(5\tau) = P_1 exp(j5\omega_1\tau) + P_2 exp(j5\omega_2\tau) + P_3 exp(j5\omega_3\tau) \tag{3.13e}$$

$$R(6\tau) = P_1 exp(j6\omega_1\tau) + P_2 exp(j6\omega_2\tau) + P_3 exp(j6\omega_3\tau) \tag{3.13f}$$

If linear prediction is used, one can define three constants (a_1 , $a_2$and a_3) such that they satisfy the following equations

$$-R(6\tau) = a_1R(5\tau) + a_2R(4\tau) + a_3R(3\tau) \tag{3.14a}$$

$$-R(5\tau) = a_1R(4\tau) + a_2R(3\tau) + a_3R(2\tau) \tag{3.14b}$$

$$-R(4\tau) = a_1R(3\tau) + a_2R(2\tau) + a_3R(1\tau) \tag{3.14c}$$

From the above equation, Cramer's rule can be used to solve a_1, $a_2$and a_3.

By substituting equation (3.13) into equation (3.14), it can be shown that

$$a_1 = -(z_1+z_2) \tag{3.15a}$$

$$a_2 = z_1z_2 \tag{3.15b}$$

$$a_3 = -z_1z_2z_3 \tag{3.15c}$$

where

$$z_i = \exp(j\omega_i\tau) \tag{3.16}$$

and i= 1,2,3. If z_i can be solved, the frequencies of ω_1, $\omega_2$and $\omega_3$can be obtained.

From equation (3.15), it can be seen that z will satisfy the following function as

$$(z - z_1)(z - z_2)(z - z_3) = z^3 + a_1z^2 + a_2z + a_3 = 0 \tag{3.17}$$

Therefore one can find z by solving the above equations.

One can summarize the steps of solving two simultaneous signals as follows:

- Six discriminators with proper delays are needed.
- The values of $a_i$can be calculated.
- From equation (3.17), the values of $z$can be calculated and the individual frequency can be calculated from equation (3.16).

In this case where three simultaneous signals are expected, but only two signals are present, the problem can be solved from:

$$\det(R) = \begin{vmatrix} R(5\tau) & R(4\tau) & R(3\tau) \\ R(4\tau) & R(3\tau) & R(2\tau) \\ R(3\tau) & R(2\tau) & R(\tau) \end{vmatrix} = 0 \tag{3.18}$$

By doing this check, it can be determined if there is three signals present. Noise in the system will cause the check to be very small but not equal to 0. For this condition, the check should be tested against a predetermined value.

3.2.4 Four Simultaneous Signals

In the case of four simultaneous signals, one can choose eight discriminators with different delays τ, 2τ , 3τ , 4τ , 5τ ,6τ , 7τ and 8τ .

If linear prediction is used, one can define four constants (a_1,a_2 ,a_3 and a_4) such that they satisfy the following equations:

$$-R(8\tau) = a_1R(7\tau) + a_2R(6\tau) + a_3R(5\tau) + a_4R(4\tau) \tag{3.19a}$$

$$-R(7\tau) = a_1R(6\tau) + a_2R(5\tau) + a_3R(4\tau) + a_4R(3\tau) \tag{3.19b}$$

$$-R(6\tau) = a_1R(5\tau) + a_2R(4\tau) + a_3R(3\tau) + a_4R(2\tau) \tag{3.19c}$$

$$-R(5\tau) = a_1R(4\tau) + a_2R(3\tau) + a_3R(2\tau) + a_4R(\tau) \tag{3.19d}$$

From the above equation, Cramer's rule can be used to solve a_1,a_2 ,a_3 and a_4

It can then be shown that

$$a_1 = -(z_1+z_2) \tag{3.20a}$$

$$a_2 = z_1z_2 \tag{3.20b}$$

$$a_3 = -z_1z_2z_3 \tag{3.20c}$$

$$a_4 = z_1 z_2 z_3 z_4 \tag{3.20d}$$

where

$$z_i = \exp(j\omega_i \tau) \tag{3.21}$$

and i= 1,2,3,4. If z_i can be solved, the frequencies of ω_1, ω_2, ω_3 and ω_4 can be obtained.

From equation (3.20), it can be seen that z will satisfy the following function as

$$\begin{aligned}&(z - z_1)(z - z_2)(z - z_3)(z - z_4)\\ &= z^4 + a_1 z^3 + a_2 z^2 + a_3 z + a_4 = 0\end{aligned} \tag{3.22}$$

Therefore one can find z by solving the above equations.

One can summarize the steps of solving two simultaneous signals as follows:

- Eight discriminators with proper delays are needed.
- The values of a_i can be calculated.
- From equation (3.22), the values of z can be calculated and the individual frequency can be calculated from equation (3.21).

In this case where four simultaneous signals are expected, but only three signals are present, the problem can be solved from:

$$\det(R)\begin{vmatrix} R(7\tau) & R(6\tau) & R(5\tau) & R(4\tau) \\ R(6\tau) & R(5\tau) & R(4\tau) & R(3\tau) \\ R(5\tau) & R(4\tau) & R(3\tau) & R(2\tau) \\ R(4\tau) & R(3\tau) & R(2\tau) & R(\tau) \end{vmatrix} = 0 \tag{3.23}$$

By doing this check, it can be determined if there is four signals present. Noise in the system will cause the check to be very small but not equal to 0.

For this condition, the check should be tested against a predetermined value.

3.3 Simultaneous Signals

Current IFM receivers offer no way of separating signals of different frequencies present in the FDs at the same time. The discussion of simultaneous signals is simplified by considering the case of just two signals present. Consider a receiver of the type described in the previous sections. A single signal produces sine and cosine terms at the inputs of the polar display. A second signal produces its own sine and cosine terms so that the display plots the sum of the two sine terms against the sum of the two cosine terms giving, in general, a result reflective of neither input. In fact, the result will be weighted towards the larger of the two signals and it is this signal that is usually of most interest.

The probability of this occurring when a LPI signal is present may be quite high. If both signals are pulsed, then the probability of them being simultaneous is small. Nevertheless, it may be important to know the integrity of every measurement, as it is made so that doubtful results will be discarded and not acted upon. Many IFM receiver manufacturers include simultaneous signal detection (SSD) circuitry, which sets a flag whenever readings are determined to be in doubt.

Figure 3.1 shows the transition period with its characteristic dip, which some SSD circuitry uses to set a flag when in doubt. The reason for this dip is because of the mixing effect of the limiting amplifier. The non-linear characteristic of the limiting amplifier generates harmonics which reduce the power levels of the original frequencies. Sometime during the rise time of the stronger pulse the most harmonics will be generated by the limiting amplifier, because the most harmonics are generated when the power levelof the signals are approximately the same [17]. Note that other methods is also used for SSD, this is not the only method used.

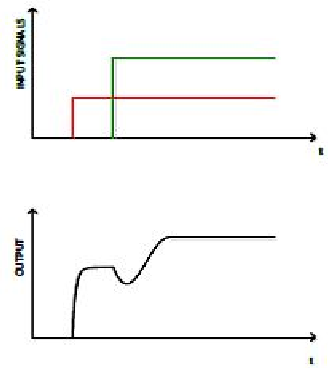

Figure 3.1 : Characteristic dip during simultaneous signals[17]

3.4 Simulation of a Typical Digital IFM (DIFM) Receiver

3.4.1 Single Delay Line

A MATLAB simulation was implemented to simulate a typical (traditional) single-line DIFM receiver. Figure 3.2 shows a block diagram of a typical single-line DIFM receiver.

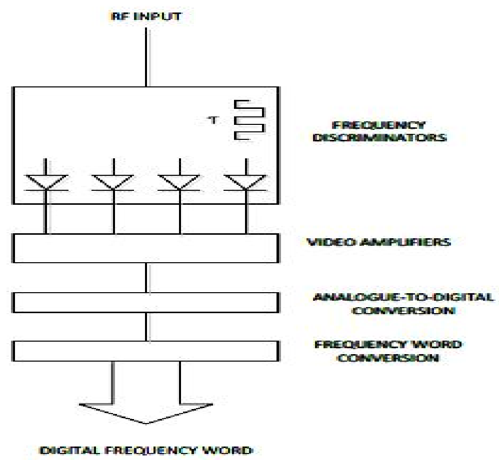

Figure 3.2 : Single-line DIFM receiver [17]

The simulation was implemented as follows:

- Generate time discrete amplitude samples for the input signal.
- Pass the time discrete signal through a Hilbert filter. This produces a complex time discrete signal.
- Choose/Calculate the length τ for an unambiguous phase difference of 2π for the desired input frequencies.
- Generate a delayed complex time discrete signal with delay. τ
- Multiply the non-delayed complex time discrete signal with the conjugate delayed complex time discrete signal to get rid of the sum frequency.
- Determine the angle between the real and imaginary part of the complex number.
- Convert the angle into a frequency value.

As mentioned the Hilbert filter is used to produce a complex time discrete signal from the real time discrete signal. Firstly a complex time discrete signal must be used to be able to calculate the outputs from the FD, $\rho \sin \varphi$ and $\cos \varphi$. Also if a complex time discrete signal is used and the non-delayed complex time discrete signal is multiplied with the conjugate delayed complex time discrete signal the sum frequency shown in equation (17) and (20) is cancelled out. Therefore the need for a LPF is then not necessary.

Chapter Four

The Simulation and Results

4.1 Introduction

In this chapter developed scheme are tested via MATLAB. The results of the scheme are discussed also in this chapter. The parts of FPGA logic as DSP module and hardware implementation are discussed as future work in chapter five.

4.2 Single Delay Lines

4.2.1 Single Input Signal with a Frequency of 200 MHz

Firstly a single signal with a frequency of 200 MHz was simulated. Figure 4.1 shows the phase vs frequency plot. It is clear that the IFM receiver measured the frequency of the input signal at 200 MHz correctly.

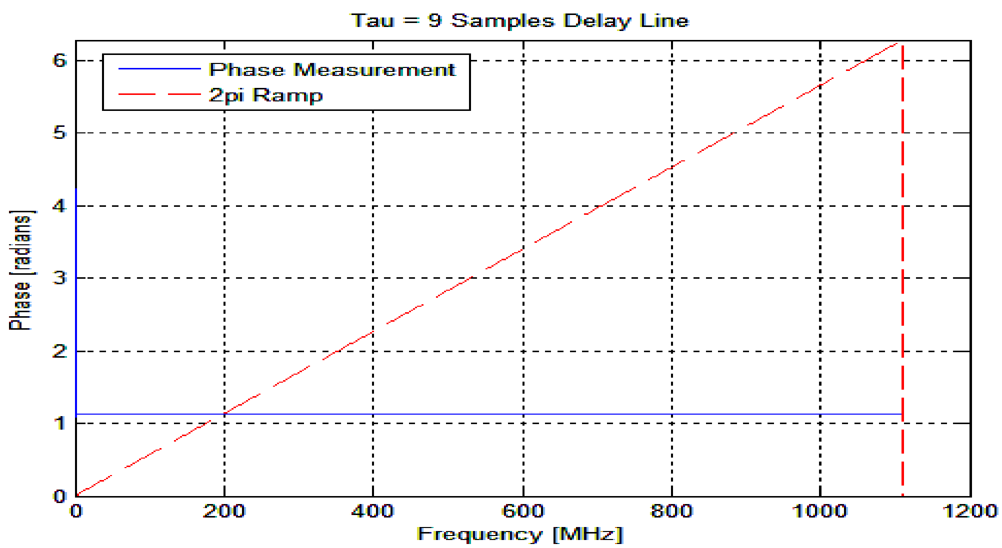

Figure 4.1 : Frequency Output = 200 MHz

4.2.2 Single Input Signal with a Frequency of 1000 MHz

Then a single signal with a frequency of 1000 MHz was simulated. Figure 4.2 shows the phase vs frequency plot.It is clear that the IFM receiver measured the frequency of the input signal at 1000 MHz correctly.

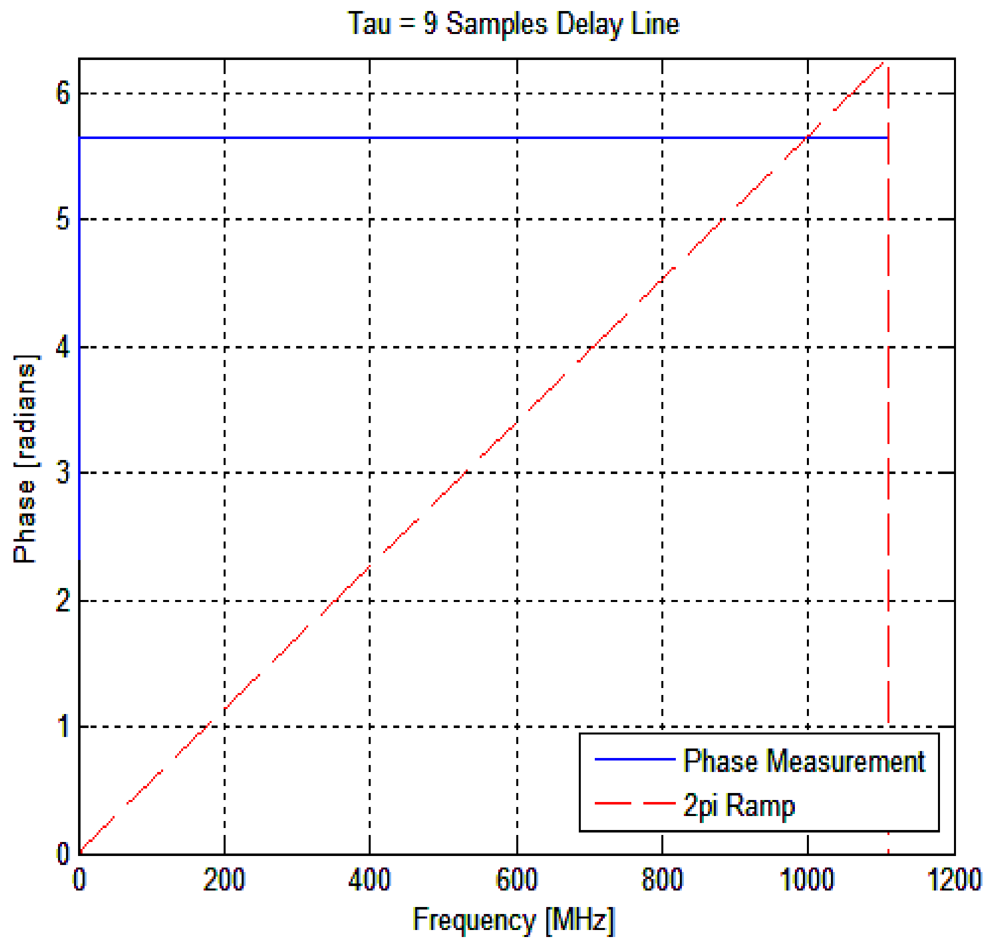

Figure 4.2 : Frequency output = 1000 MHz

4.2.3 Two Input Signals with Frequencies of 200 MHz and 1000 MHz

Then two combined signals with frequencies of 200 MHz and 1000 MHz were simulated. Figure 4.3 shows the phase vs frequency plot. It is clear that the IFM receiver measured the frequency of the input signal at 45 MHz incorrectly.

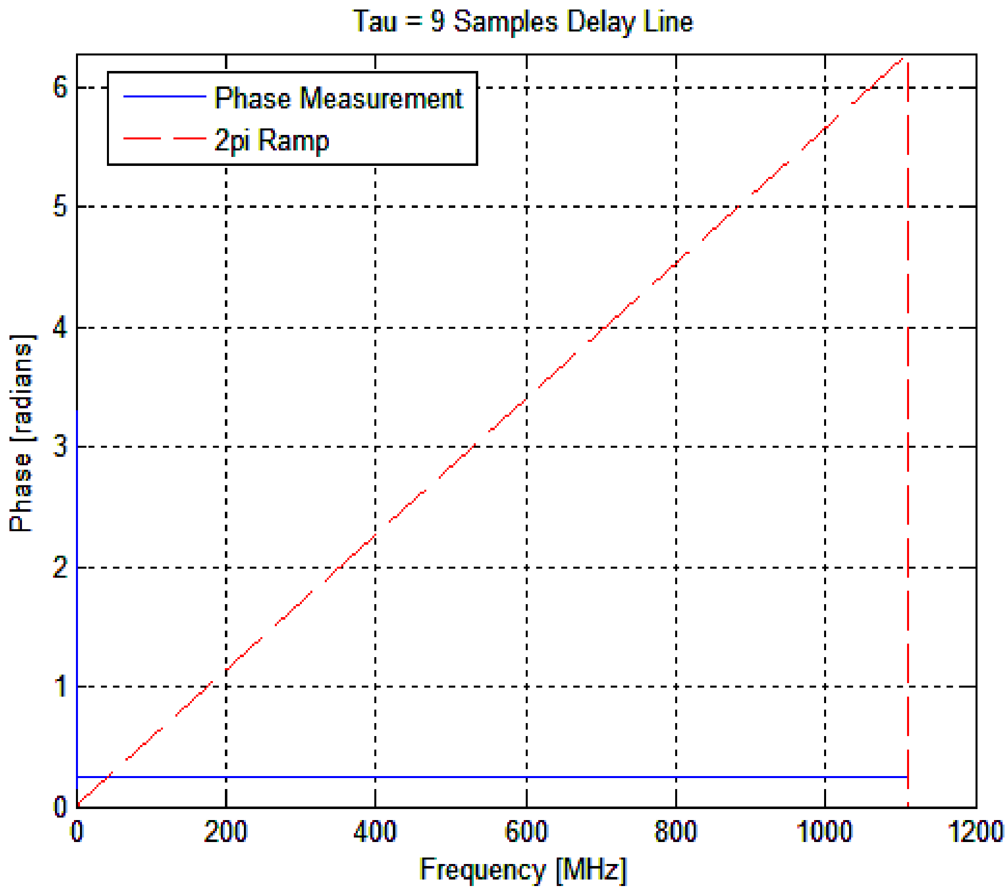

Figure 4.3 : Frequency output = 45 MHz

4.3 Multiple Delay Lines

As explained in section 3.3.2.1, a shorter delay line length will give a larger unambiguous bandwidth and a longer delay line will give a more accurate frequency measurement. By using multiple delay lines, one can therefore have the best of both. One should use a short delay line to get a large unambiguous bandwidth together with a longer line to have more accurate frequency measurement [17].

With the long line one gets multiple answers with very accurate frequency measurement. Then one uses the shorter line to determine which one of the multiple answers is the correct one. This is called ambiguity resolving.

More delay lines (with lengths between longest and shortest) can be used to resolve ambiguities where the longest and the shortest line ratio is too high for the shortest line to resolve the longest line.

A MATLAB simulation was implemented to simulate a multiple delay linedigital IFM receiver. Figure 4.4 shows a block diagram of a typical (traditional) multiple-line DIFM receiver.

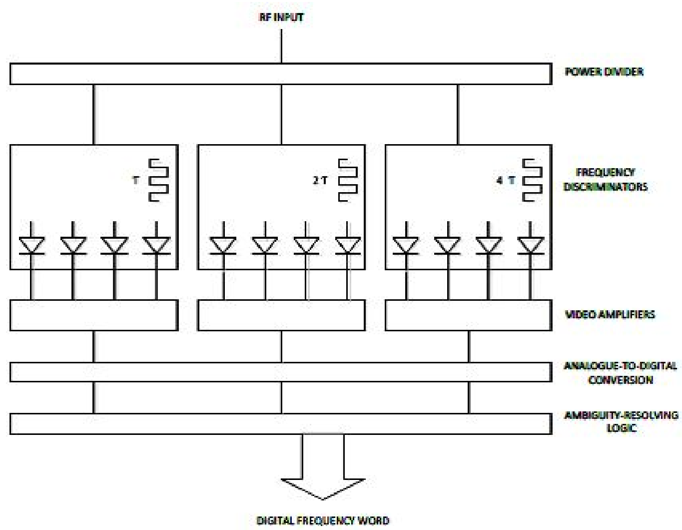

Figure 4.4 : Multiple delay line DIFM receiver

4.3.1 Single Input Signal with a Frequency of 200 MHz

Firstly a single signal with a frequency of 200 MHz was simulated.Figure 4.5 shows the phase vs frequency plots. In the 4τ plot (more accurate frequency measurement) 4 frequencies (200, 480, 755 and 1035 MHz) are possible solutions. In the 2τ plot 2 frequencies (200 and 755 MHz) are possible solutions. Therefore 480 and 1035 MHz can be ignored from the plot. In the 4τ plot τ (large unambiguous bandwidth) it can be seen that the frequency of 200 MHz is the only possible solution. Therefore the 755

MHz can also be ignored and so the IFM receiver measured the frequency of the input signal at 200 MHz correctly.

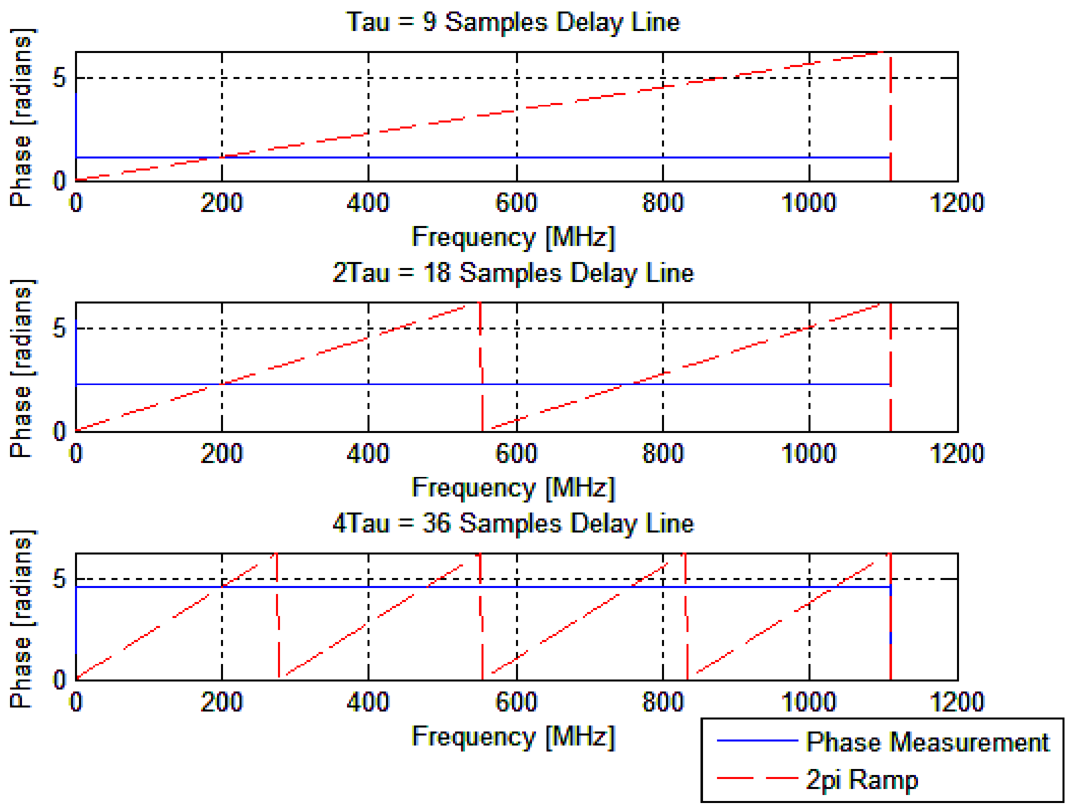

Figure 4.5 : Frequency output = 200 MHz

4.3.2 Single Input Signal with a Frequency of 1000 MHz

Then a single signal with a frequency of 1000 MHz was simulated.Figure 4.6 shows the phase vs frequency plots. In the 4τ plot (more accurate frequency measurement) 4 frequencies (170, 445, 720 and 1000 MHz) are possible solutions. In the 2τ plot 2 frequencies (445 and 1000 MHz) are possible solutions. Therefore 170 and 720 MHz can be ignored from the plot. In the 4τ plot τ (large unambiguous bandwidth) it can be seen that the frequency of 1000 MHz is the only possible solution. Therefore 445

MHz can also be ignored and so the IFM receiver measured the frequency of the input signal at 1000 MHz correctly.

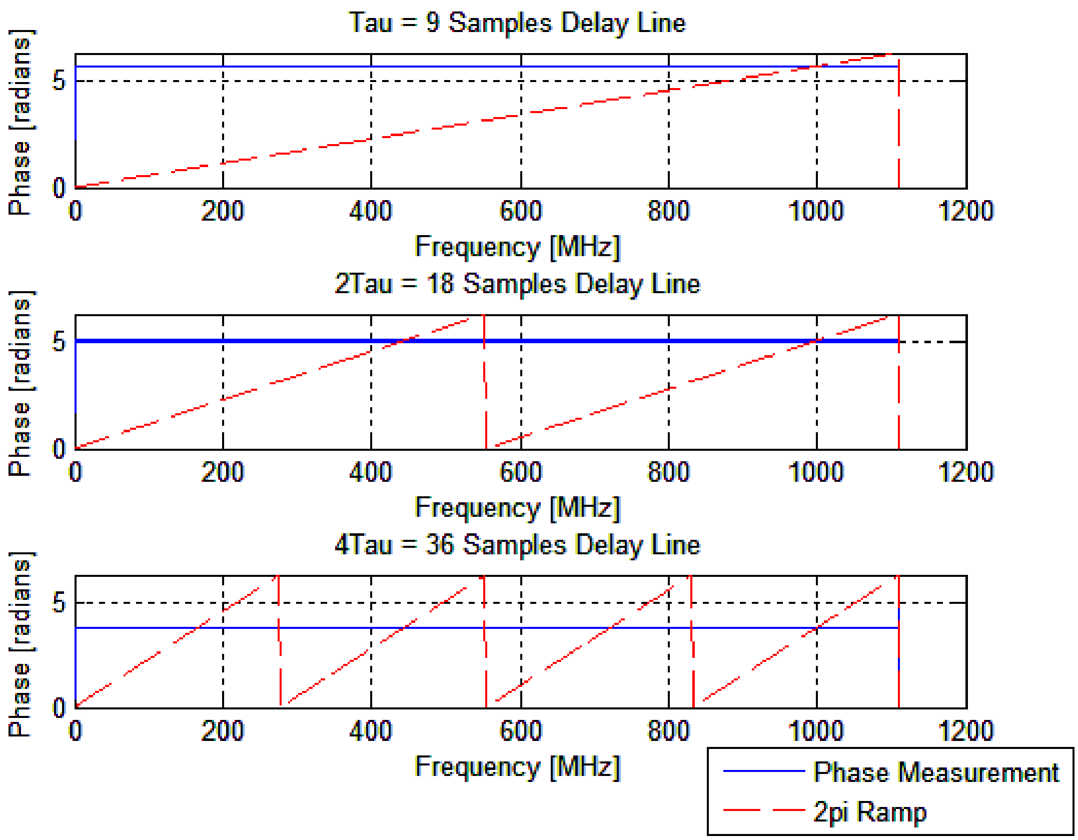

Figure 4.6 : Frequency output = 1000 MHz

4.3.3 Two Input Signals with Frequencies of 200 MHz and 1000 MHz

Then an input signal consisting of two signals with frequencies at 200 MHz and 1000 MHz was simulated.Figure 4.7 shows the phase vs frequency plot. In the 4τ plot 4 frequencies (180, 460, 740 and 1020 MHz) are possible solutions. In the 2τ plot 2 frequencies (320 and 880 MHz) are possible solutions. In the τ plot it can be seen that the frequency of 40 MHz is the only possible solution. It is clear that the IFM receiver cannot resolve the frequency of the input signal correctly, giving non matching solutions for the different delay lines.

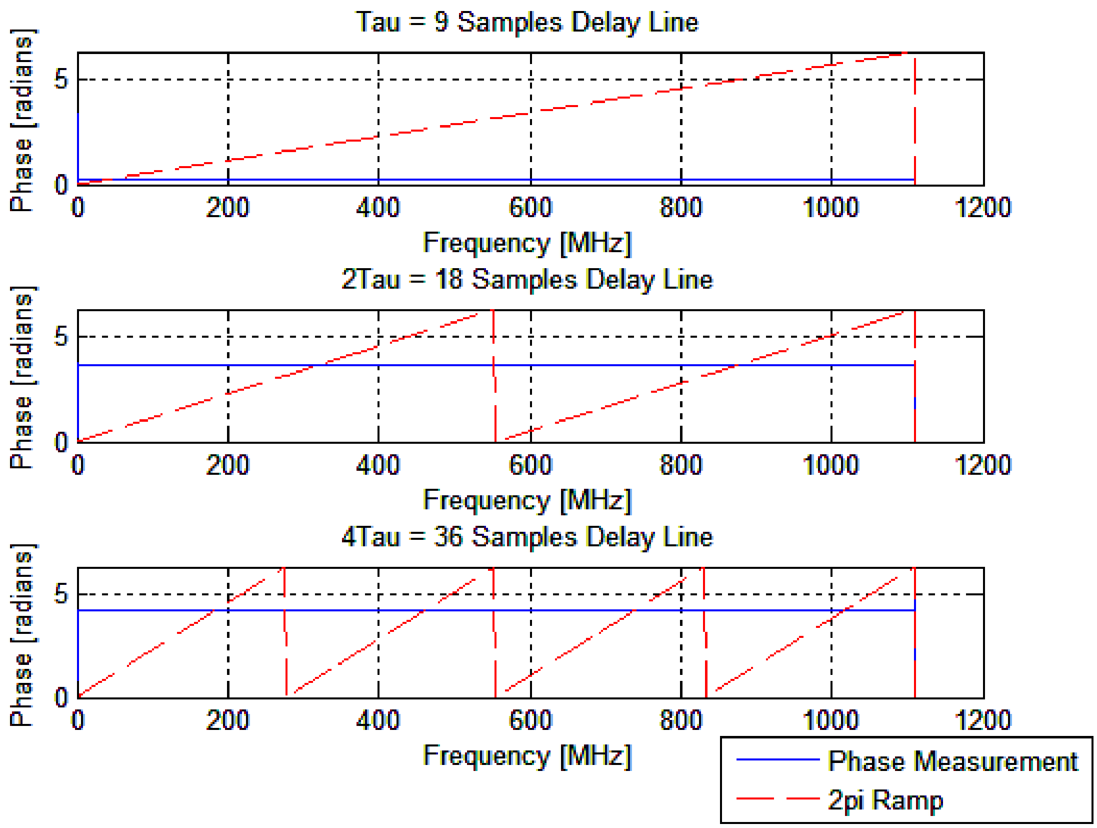

Figure 4.7 : Angle resolving between input signals and delayed versions

4.4 Simulation of a possible solution(Prony's Method)

Figure 4.8 shows a block diagram of a four delay line DIFM receiver using Prony's method.In comparing Figure 3.7 with Figure 4.8, it can be seen that only the ambiguity-resolving logic block is replaced with Prony's resolving method block and that the delay line ratios are different. For the traditional multiple delay line DIFM receivers the delay line ratio is $\tau 2\tau 4\tau$ etc., but for Prony's method the ratios are $\tau 2\tau\tau 3\tau 4\tau 4\tau$ etc

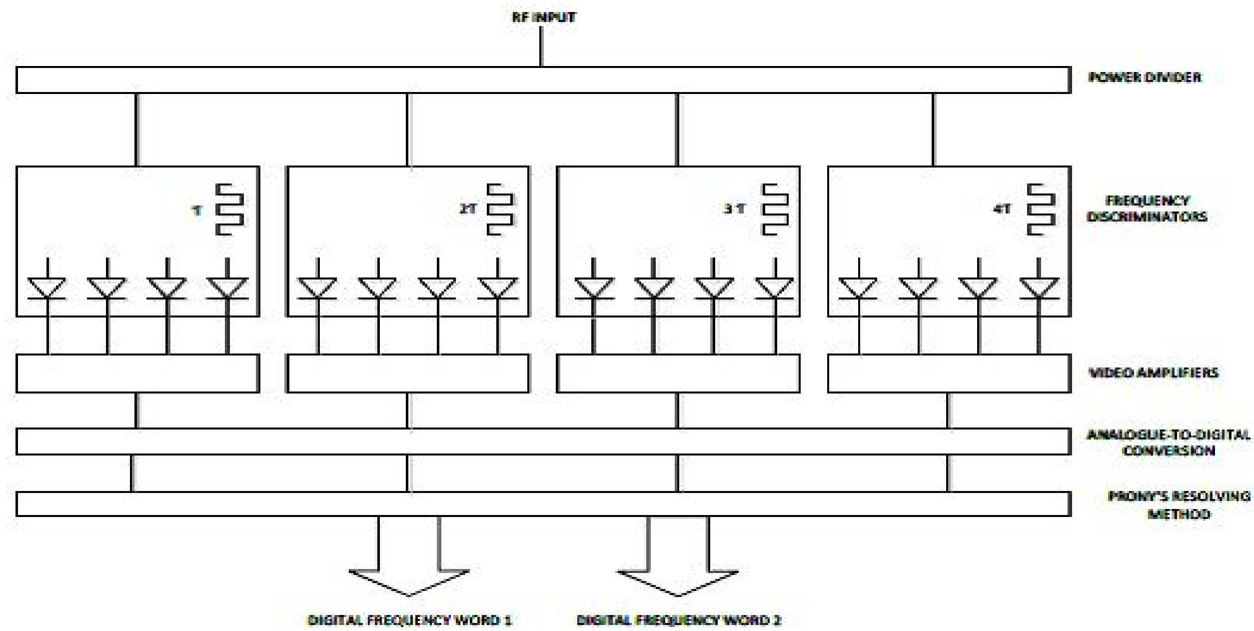

Figure 4.8 : DIFM receiver using Prony's resolving method

4.4.1 Four Delay Lines/Two Simultaneous Signals

In the case of two simultaneous signals, one can choose four discriminators with different with lags (or delay time) , $\tau 2\tau\tau 3\tau 4\tau$ and 4τ .

A MATLAB simulation was implemented to simulate a four delay line DIFM receiver using Prony's method.

Firstly an input signal consisting of two CW signals with frequencies at 200 MHz and 1000 MHz was simulated. Figure 4.9 shows the output of the simulation. It is clear that the simulation measured the frequencies of the input signal at 200 MHz and 1000 MHz correctly.

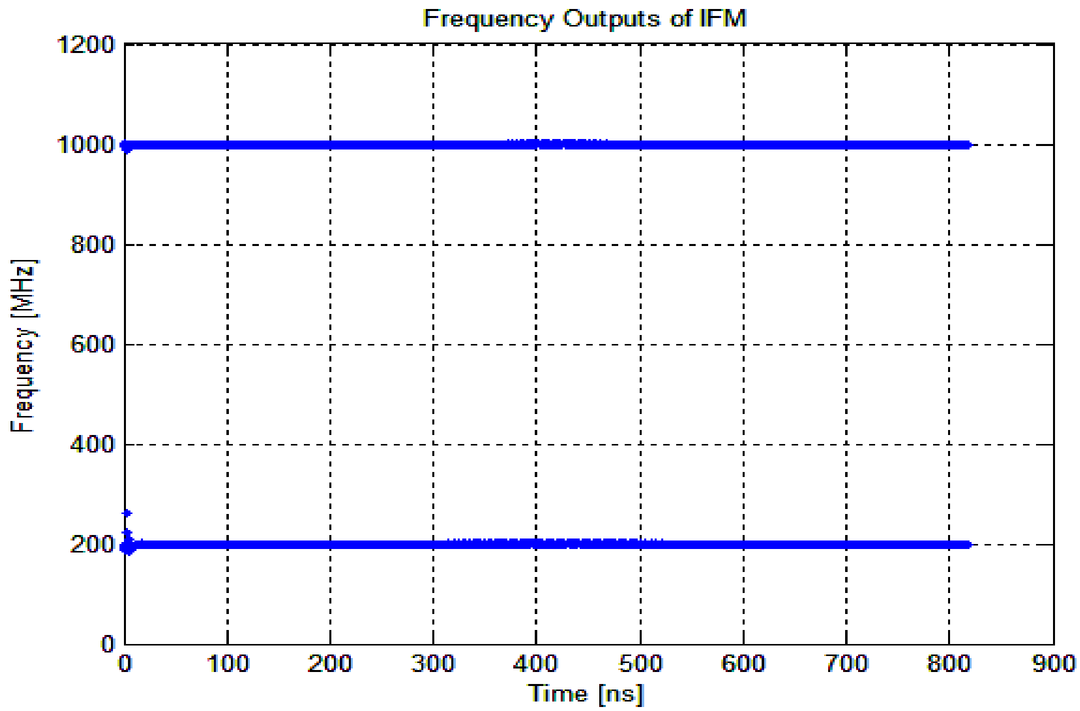

Figure 4.9 : Frequency output = 200 MHz/1000 MHz

Then white noise with a signal-to-noise ratio of 40 dB was added to the input signal. Figure 4.10 shows the output of the simulation.

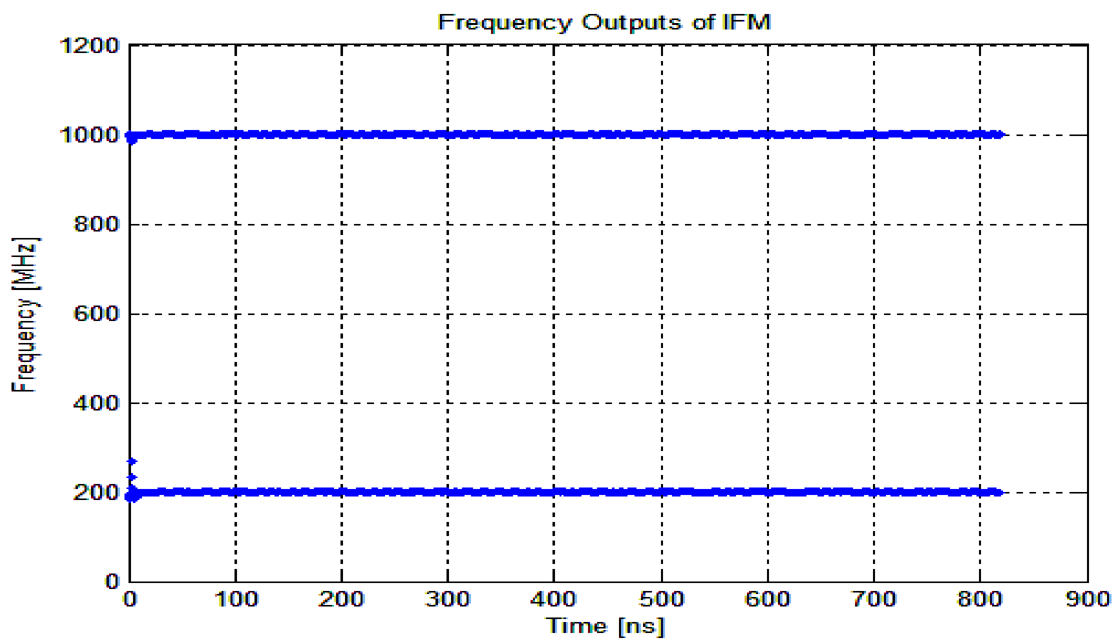

Figure 4.10 : Frequency output = 200 MHz/1000 MHz, SNR = 40 dB

Then an input signal consisting of two signals, one at a fixed frequency of 500 MHz and the other sweeping its frequency from 200 MHz to 1000 MHz, was simulated. Figure 4.11 shows the output of the simulation. It can be seen that the frequencies of the two signals were successfully measured, while sweeping the one signal from 200 MHz to 1000 MHz.Then an input signal consisting of two signals with frequencies at 200 MHz MHz (no phase jumps) and 1000 MHz (which contained two 180º phase jumps) was simulated.

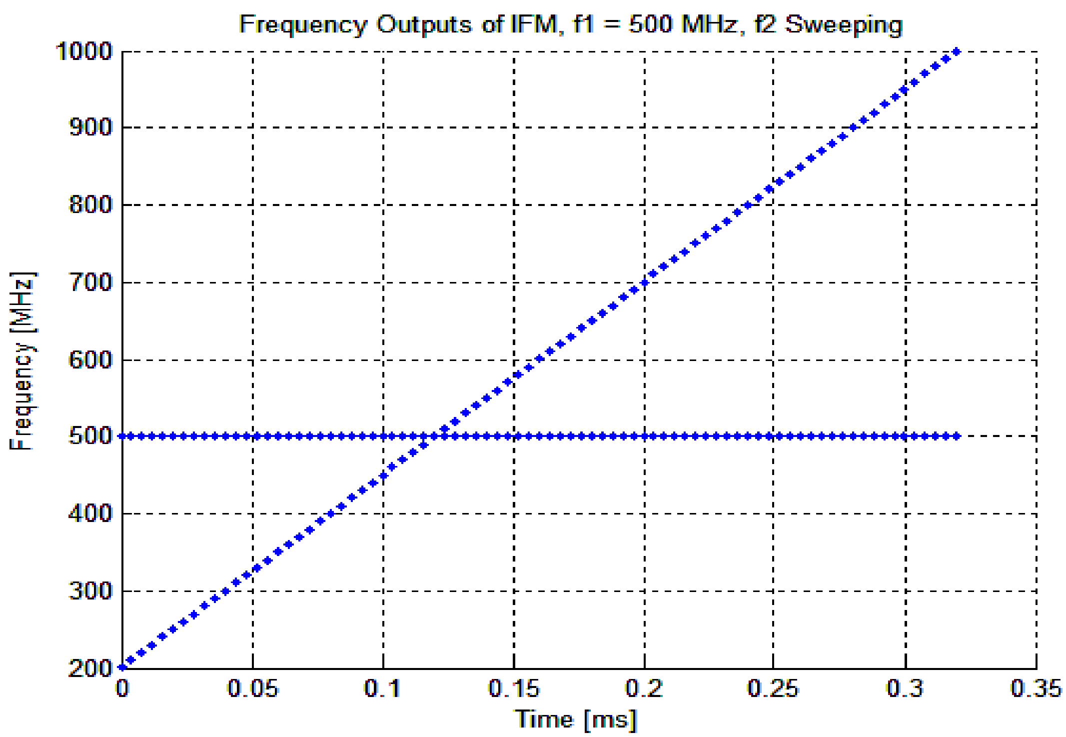

Figure 4.11 : Frequency output = 500 MHz/sweeping

In Figure 4.12 it can be seen that the 180º phase jumps caused jumps in the frequency measurement, but is much better compare to the results achieve with the single line IFM as seen in Figure 3.2. A reason for this is that the delay line lengths used in prony's method is shorter compared to the delay line length used in the single line IFM simulation.

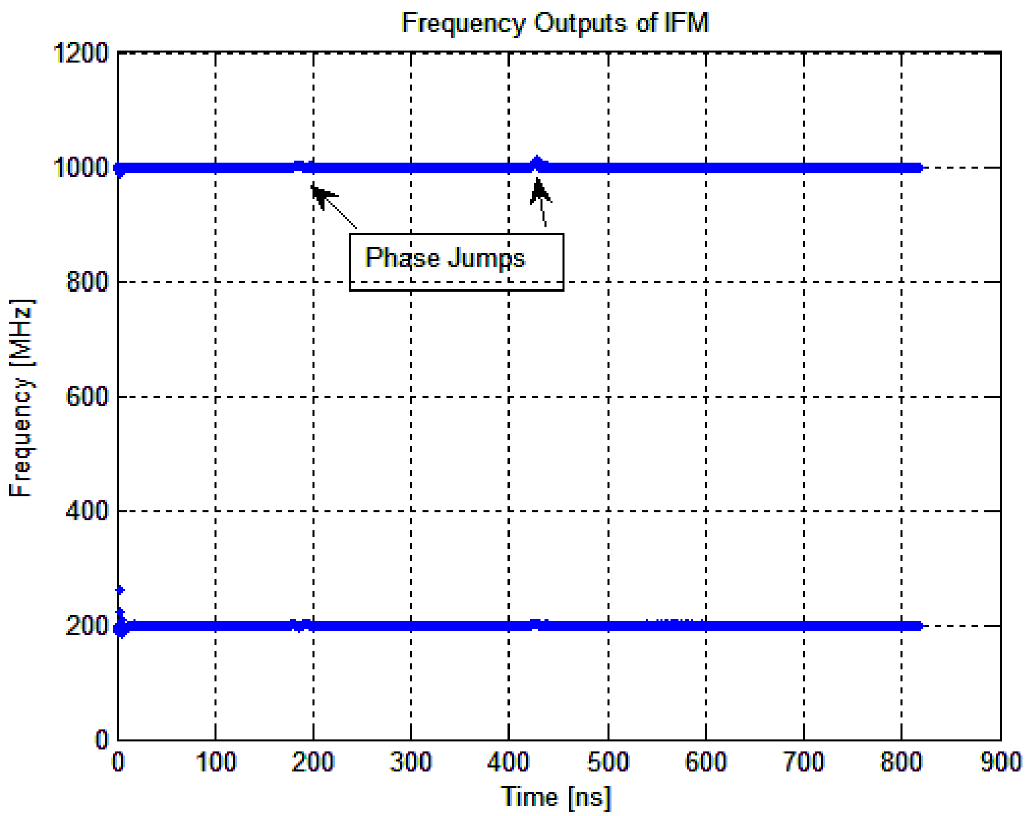

Figure 4.12 : Frequency output = 200 MHz/1000 MHz

4.4.2 Six Delay Lines/Three Simultaneous Signals

Figure 4.13 shows a block diagram of an six delay line DIFM receiver using Prony's method.

In the case of three simultaneous signals, one can choose six discriminators with different lags (or delay time) , $\tau 2\tau\tau 3\tau 4\tau 5\tau$ and 6τ

A MATLAB simulation was implemented to simulate a six delay line DIFM receiver using Prony's method.

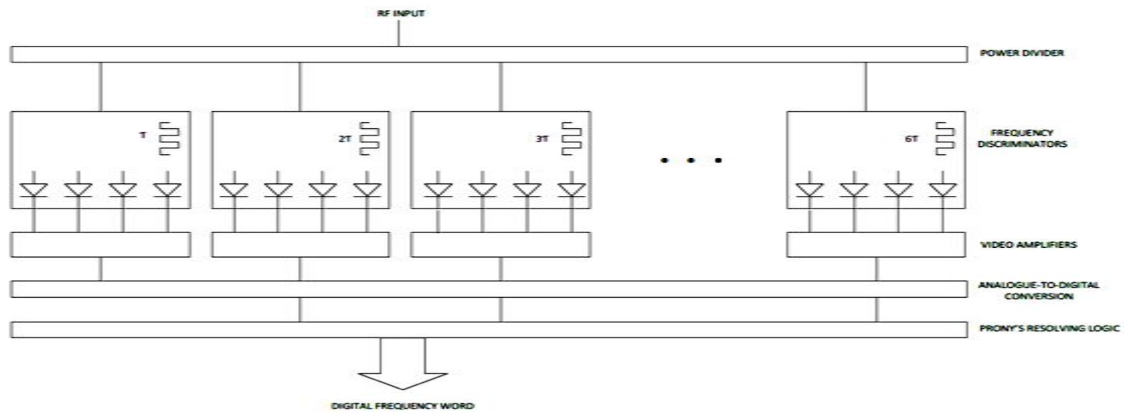

Figure 4.13 : DIFM receiver using Prony's resolving method – six delay lines

Firstly an input signal consisting of three signals with frequencies at 200 MHz, 600 MHz and 1000 MHz was simulated. Figure 4.14 shows the output of the simulation. It is clear that the simulation measured the frequencies of the input signal at 200 MHz, 600 MHz and 1000 MHz correctly.

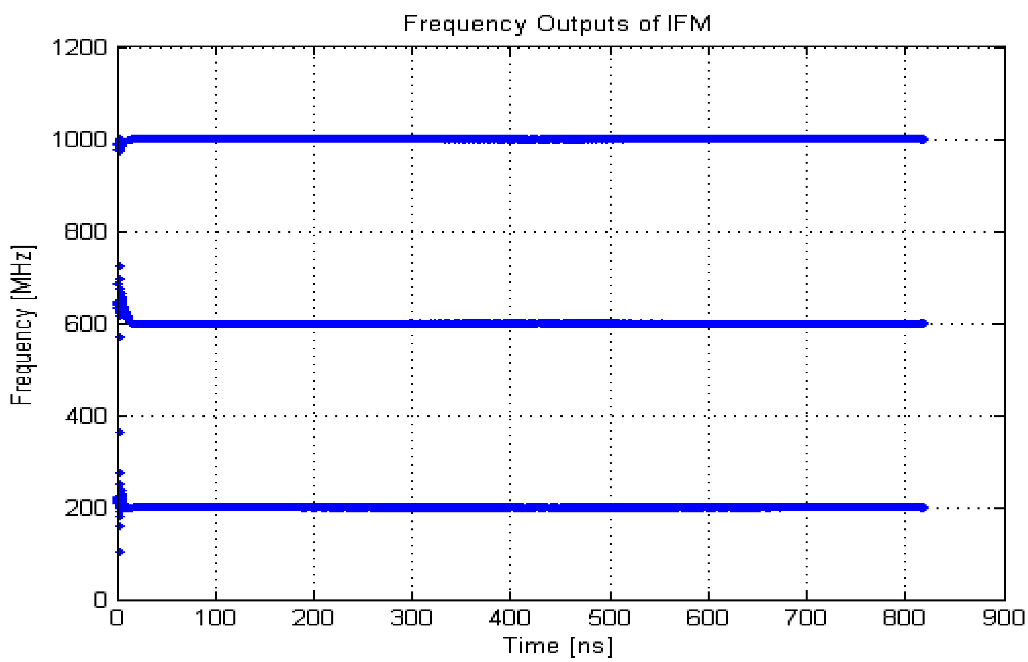

Figure 4.14 : Frequency output = 200 MHz/600 MHz/1000 MHz

Then white noise with a signal-to-noise ratio of 40 dB was added to the input signal. Figure 4.15 shows the output of the simulation.

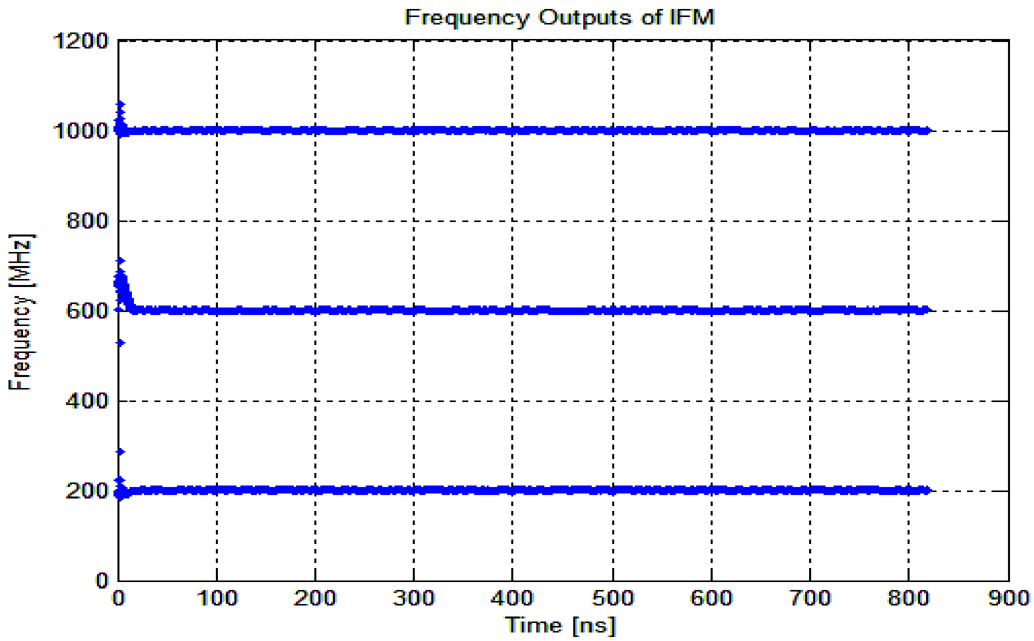

Figure 4.15 : Frequency output = 200 MHz/600 MHz/1000 MHz, SNR = 40 dB

Then an input signal consisting of three signals, two at fixed frequencies of 400 MHz and 700 MHz and the other sweeping its frequency from 200 MHz to 1000 MHz, was simulated.

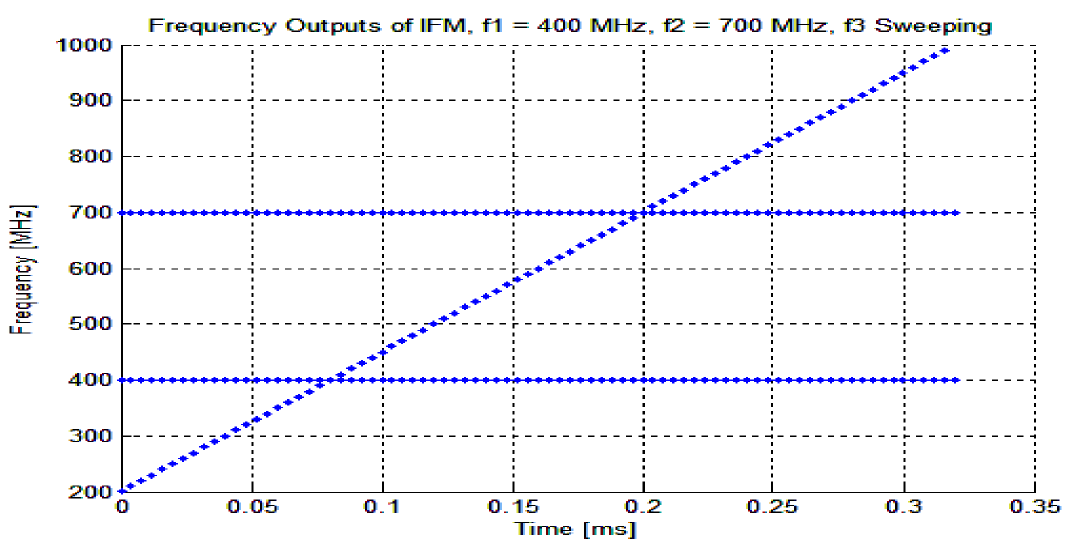

Figure 4.16 : Frequency output = 400 MHz/700 MHz/Sweeping

Figure 4.16 shows the output of the simulation. It can be seen that the frequencies of the three signals were successfully measured, while sweeping the one signal from 200 MHz to 1000 MHz.

4.4.3 Eight Delay Lines/Four Simultaneous Signals

Figure 4.17 shows a block diagram of an eight delay line DIFM receiver using Prony's method. In the case of four simultaneous signals one can choose eight discriminators with different lags (or delay time) , $\tau 2\tau\tau 3\tau 4\tau 5\tau 6\tau 7\tau$ and 8τ.

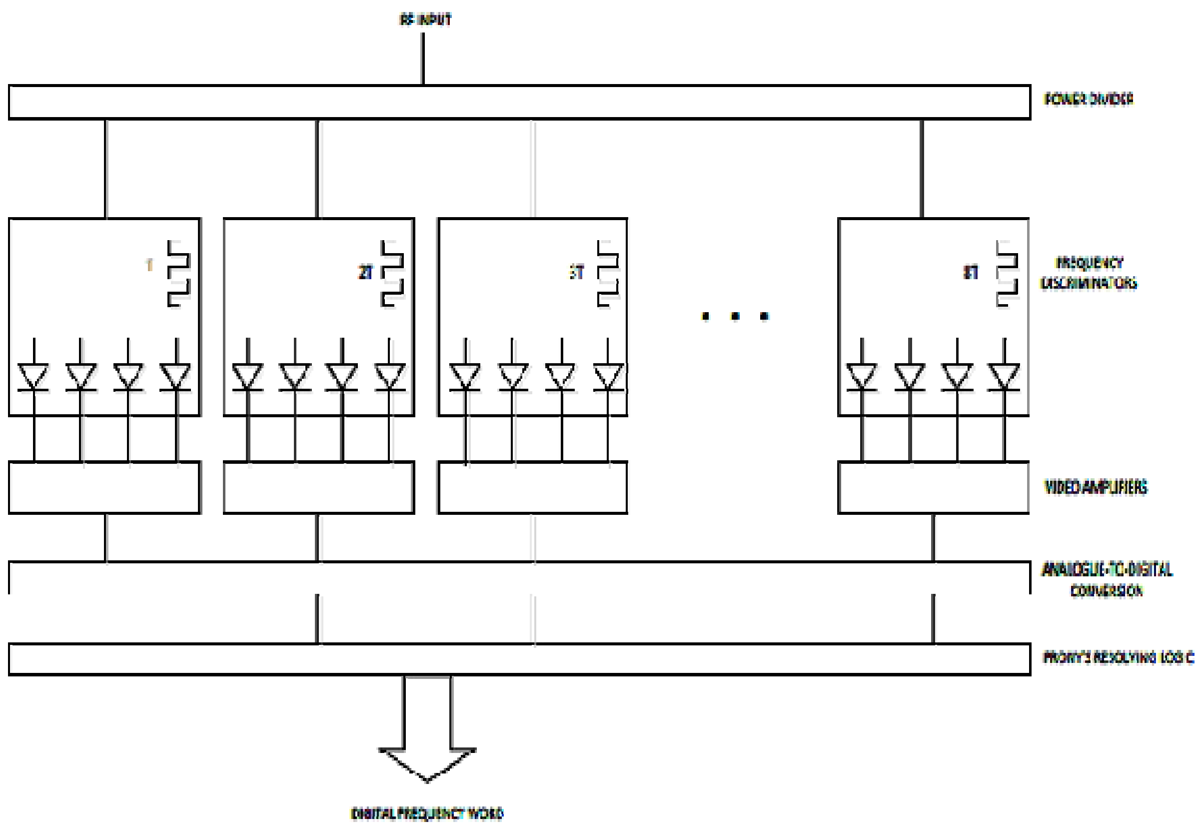

Figure 4.17 : DIFM receiver using Prony's resolving method – eight delay lines

A MATLAB simulation was implemented to simulate an eight delay line DIFM receiver using Prony's method.

Firstly an input signal consisting of four signals with frequencies at 200 MHz, 400 MHz, 700 MHz and 1000 MHz was simulated. Figure 4.18 shows the output of the simulation. It is clear that the simulation measured the frequencies of the input signal at 100 MHz, 400 MHz, 700 MHz and 1000 MHz correctly.

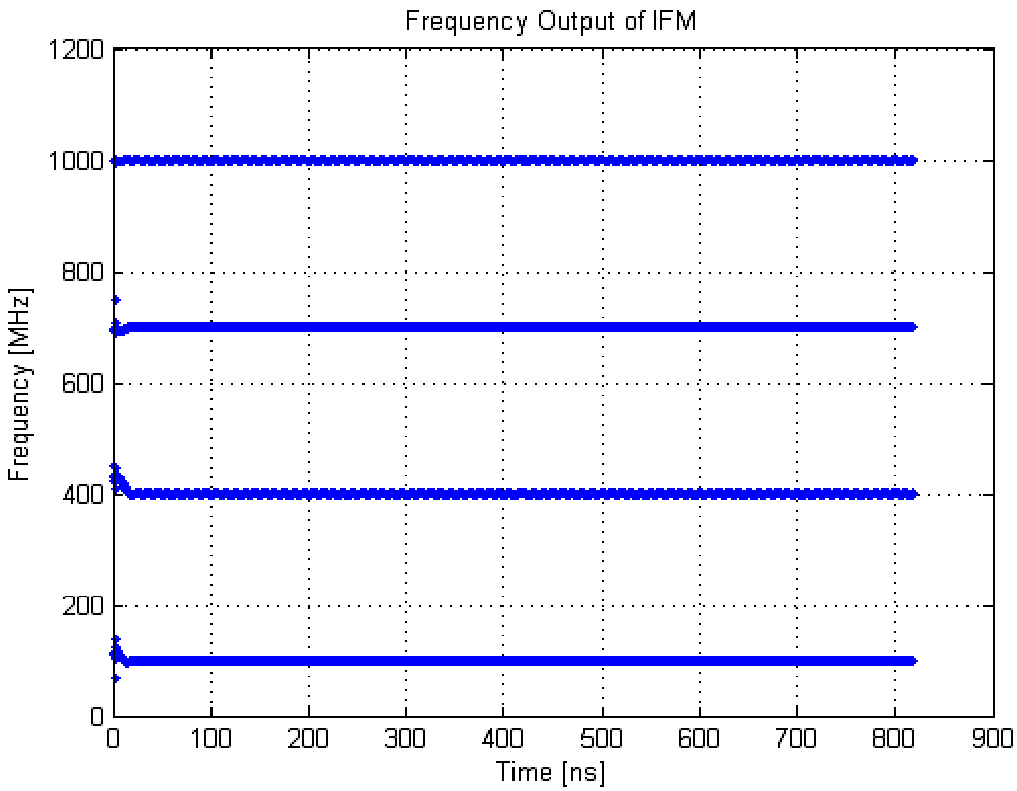

Figure 4.18 : Frequency output = 100 MHz/400 MHz/700MHz/1000 MHz

Then white noise with a signal-to-noise ratio of 40 dB was added to the input signal. Figure 4.19 shows the output of the simulation.

Then an input signal consisting of four signals, three at fixed frequencies of 400 MHz, 600 MHz and 800 MHz and the other sweeping its frequency from 200 MHz to 1000 MHz, was simulated.

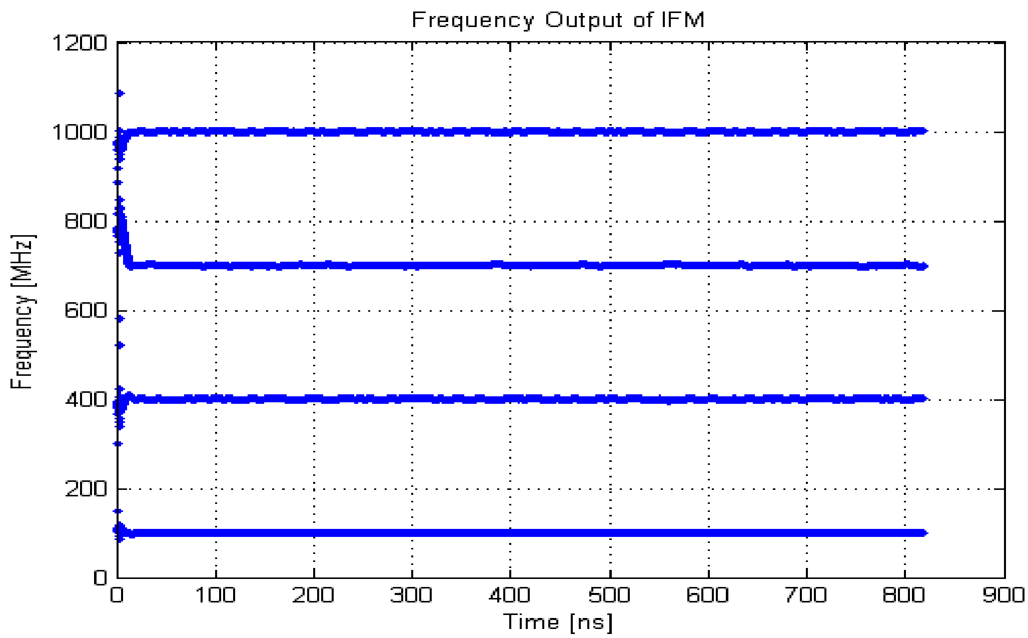

Figure 4.19 : Frequency output = 100 MHz/400 MHz/700MHz/1000 MHz, SNR = 40 dB

. Figure 4.20 shows the output of the simulation. It can be seen that the frequencies of the four signals were successfully measured, while sweeping the one signal from 200 MHz to 1000 MHz.

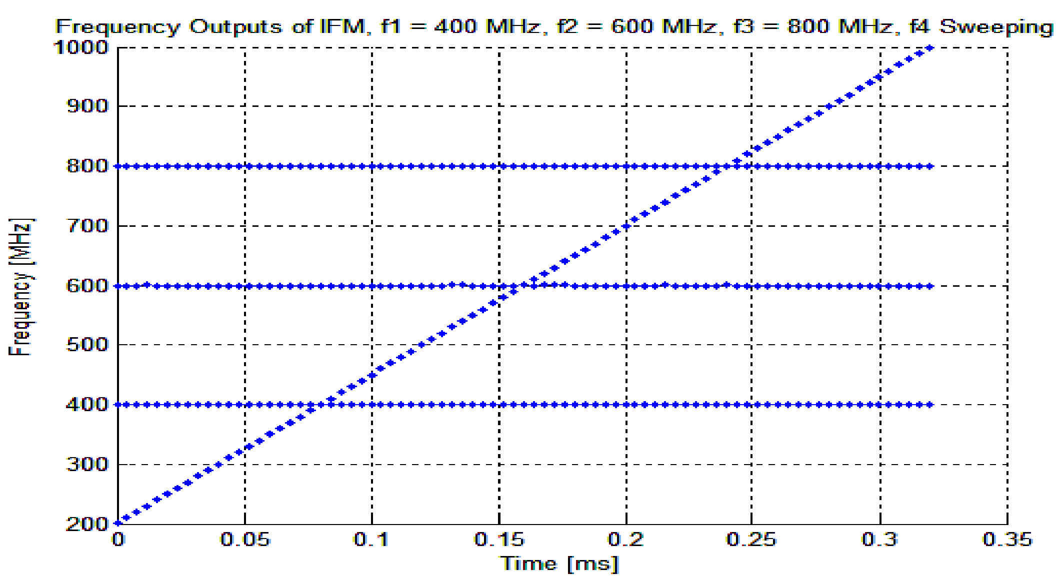

Figure 4.20 : Frequency output = 400 MHz/600 MHz/800MHz/sweeping

Another important point is observed when the results of schemas 2, 3, 4 and 5 are compared to the results of schema 6, 7, 8 and 9. The only difference between these groups of schemas is that the first group does not utilize the clustering technique. The results of the first group of schemas are very poor even though they consume more processing time. The schemas utilizing the clustering technique give better results both in detection capability and in processing time.

When the histogram based schemas are compared to each other, the results are very similar. The wavelet detector based schema seems to give better results even in complicated environment. However this schema consumes more processing power.

These experiments show that the last schema, which is based on the wavelet detector, gives best results. It seems that combining the wavelet detector based method with a clustering technique improves the performance of this method. Other techniques proposed in this schema increase the reliability of the schema. It is observed that applying pre-tracking and post-tracking decreases the processing time of the total schema. Usage of PA for deinterleaving relieves the wavelet pulse detector.

4.4.4 Summary of Simulation Results

The results achieved in MATLAB with Prony’s method to solve the problem with simultaneous signals seem very promising. This strengthens the argument for continuing to implement Prony’s method in hardware and simulate it in hardware.

.

Chapter Five

Conclusion and Recommendation

5.1 Conclusion

In this thesis, several techniques are studied to solve the emitter identification problem in Electronic Warfare systems that using IFM receivers. This done by uses of IFM receiver along with porny's analysis and multiple delay lines to achieve our goals.

We use IFM as a receiver as part of our scheme its function is to measure the frequency of pulse signals over a very wide bandwidth and power dynamic range.These receivers have a very high resolution (typically 1 MHz) over this very wide bandwidth and the error of these receivers is typically a few megahertz.

We use prony's analysis for parameter extraction according to this parameter classification process takes placeshown in chapter two. To solve the IFM problem of receiving simultaneous signals we use multiple delay lines.

We find that our scheme achieved good results in order to intercept LPI signals even when it appear simultaneously at the receiver and number of simultaneously received signals are function of number of delay lines.

5.2 Recommendations

After successful MATLAB simulation of Prony's resolving method and multiple delay lineswe recommend starting an investigation into implementing Prony's resolving logic in an FPGA and multiple delay lines on DIFM.

When pony's method are implemented in FPGA the whale ESM system could be tested to solve the problem of the complex LPI signal and extracting LPI parameters.

Also we recommend computerizing the part of classification process by applying the concept of correlation on the received signals and built database of received signals.

References

[1] A. Denk, "Detection and jamming low probability of intercept (LPI) radars," DTIC Document, 2006.

[2] R. Mohseni, "Interception FMCW radar using Wigner-Ville Distribution (WVD)," *Majlesi Journal of Telecommunication Devices,* vol. 2, 2014.

[3] P. E. Pace, *Detecting and classifying low probability of intercept radar*: Artech House, 2009.

[4] F. L. Taboada, "Detection and classification of low probability of intercept radar signals using parallel filter arrays and higher order statistics," DTIC Document, 2002.

[5] D. Stevens and S. Schuckers, "A Novel Approach for the Characterization of FSK Low Probability of Intercept Radar Signals Via Application of the Reassignment Method," in *2014 IEEE Military Communications Conference*, 2014, pp. 783-787.

[6] M. K. Aslan, "Emitter identification techniques in electronic warfare," middle east technical university, 2006.

[7] A. Güvenlik, "Clustering Techniques for Emitter Identification in Electronic Warfare," M. Sc. Thesis, METU, 1999.

[8] S. Bailie and M. Leeser, "An FPGA implementation of incremental clustering for radar pulse deinterleaving," *Master's thesis, Northeastern University,* 2010.

[9] M. Skolnik, "Introduction to radar systems, McGraw-Hill, Comp," *New York,* 2001.

[10] R. Wiley, "ELINT: The Analysis of Radar Signals," *Artech House,* 1993.

[11] N. Vankayalapati and S. Kay, "Asymptotically optimal detection/localization of LPI signals of emitters using distributed sensors," in *SPIE Defense Security and Sensing*, 2010, pp. 77060U-77060U-9.

[12] L. Anjaneyulu, N. Murthy, and N. Sarma, "A Novel Method for Recognition of Modulation Code of LPI Radar Signals," *signal,* vol. 20, p. 40, 2009.

[13] A. Singh and K. Subba Rao, "Detection, identification and classification of intra pulse modulated LPI radar signal using digital receiver," *International Journal of Emerging Technology and Advanced Engineering,* vol. 2, pp. 433-440, 2012.

[14] J. B. Tsui, *Digital techniques for wideband receivers* vol. 2: SciTech Publishing, 2004.

[15] J. B.-Y. Tsui, "Digital microwave receivers: theory and concepts," *Norwood MA Artech House*, 1989.

[16] J. Tsui, *Microwave receivers with electronic warfare applications*: The Institution of Engineering and Technology, 2005.

[17] D. M. Rachman, "*Polar frequency discriminators*," 1986.

[18] D. Heaton, "The systems engineer's primer on IFM receivers," *Microwave Journal,* vol. 23, pp. 71-73, 1980.

[19] H. Gruchalla-Wesierski, M. Czyzewski, and A. Slowik, "The estimation of simultaneous signals frequencies in the IFM receiver using parametric methods," in *Microwaves, Radar and Wireless Communications, 2008. MIKON 2008. 17th International Conference on*, 2008, pp. 1-4.

[20] P. L. Herselman and J. E. Cilliers, "A digital instantaneous frequency measurement technique using high-speed analogue-to-digital converters and field programmable gate arrays," *South African journal of science*, 2006.

Appendices

Appendix A Single Delay Line Simulation MATLAB Code

```
% This one goes from almost 0 degrees to 360 degrees (200 MHz
- 1000 MHz).
% Close all previous plots
close all;
% Frequency 1 [Hz]
fc1 = 200e6;
% Frequency 2 [Hz]
fc2 = 1000e6;
% Sampling rate [Hz]
fs = 10000e6;
% Time Steps
t = 0:1/fs:50000/fs;
disp('=================== IFM Simulation
=================\n')
disp('== 1. Input signal f = 200 MHz ==\n')
disp('== 2. Input signal f = 1000 MHz ==\n')
disp('== 3. Two input signals f1 = 200 MHz, f2 = 1000 MHz
==\n')
disp('====================================================\n'
)
a = input('Enter a option :');
if (a == 1) s= 50*cos(2*pi*fc1*t);
elseif (a == 2) s= 50*cos(2*pi*fc2*t);
elseif (a == 3) s= 50*cos(2*pi*fc1*t)+50*cos(2*pi*fc2*t);
end;
% Plot the signal
figure(1);
plot(t*1e9,s);
set(gca, 'Xlim', [0 20]);
title('Input signal');
ylabel('Amplitude [mV]');
xlabel('Time [ns]');
grid on;
% Hilbert Transform - Transform Real Signal to Complex Signal
s_imag = hilbert(s);
% Delay Line Length = 9 sample delay
d1 = 9;
% Delay the Complex Signal
s_d1 = [s_imag(d1+1:end) zeros(1,d1)];
% Determine the Angle between the delayed signal and the
original signal
delay1 = angle(s_d1.*conj(s_imag));
% This angle is from -pi(-180 degrees)to +pi(180 degrees)
[radians]
% Convert the angle from 0 to 2*pi (360 degrees) [radians]
```

```
delay1(delay1<=0) = delay1(delay1<=0) + 2*pi;
% Display the angle in radians
delay1(100)
% Display the angle in degrees
delay1(100) * 180 / pi
% Changing angle to frequency (This is normally a lookup
table)
w = delay1 / d1;
f = fs*w/(2*pi);
% Plot 2*pi ramp
x = linspace(0,pi,size(delay1,2));
xf=fs*x/(d1*pi*1000000); % Convert angle to frequency for
plot
y1 = mod(2*x, 2*pi);
figure(2);
subplot(1,1,1);
plot(xf,delay1); % Horizontal Line = Angle
hold on;
plot(xf, y1,'r--'); % 2*pi Ramp
title('T = 9 Samples Delay Line');
ylabel('Phase [radians]');
xlabel('Frequency [MHz]');
grid on;
set(gca, 'Ylim', [0 2*pi]);
figure(3);
plot(f/1e6); ylabel('Frequency [MHz]');
title('Frequency Output of IFM');
xlabel('Time');
grid on;
set(gca, 'Ylim', [0 1200]);
```

Appendix B Multiple Delay Line Simulation MATLABCode

```
% This one goes from almost 0 degrees to 360 degrees (200 MHz
- 1000 MHz).
% Close all previous plots
close all;
% Frequency 1 [Hz]
fc1 = 200e6;
% Frequency 2 [Hz]
fc2 = 1000e6;
% Sampling rate [Hz]
fs = 10000e6;
% Time steps
t = 0:1/fs:50000/fs;
disp('=================== IFM Simulation
=================\n')
disp('== 1. Input signal f = 200 MHz ==\n')
disp('== 2. Input signal f = 1000 MHz ==\n')
disp('== 3. Two input signals f1 = 200 MHz, f2 = 1000 MHz
==\n')
disp('===================================================\n'
)
a = input('Enter a option :');
if (a == 1) s= 50*cos(2*pi*fc1*t);
elseif (a == 2) s= 50*cos(2*pi*fc2*t);
elseif (a == 3) s= 50*cos(2*pi*fc1*t)+50*cos(2*pi*fc2*t);
end;
% Plot the signal
figure(1);
plot(t*1e9,s);
set(gca, 'Xlim', [0 20]);
title('Input signal');
ylabel('Amplitude [mV]');
xlabel('Time [ns]');
grid on;
% Hilbert Transform - Transform Real Signal to Complex Signal
s_imag = hilbert(s);
% Delay Line Length = 9 sample delay
d1 = 9;
% Delay Line Length = 18 (2 * 9) sample delay
d2 = 18;
% Delay Line Length = 36 (2 * 18) sample delay
d3 = 36;
delay1(100) * 180 / pi
delay2(100) * 180 / pi
delay3(100) * 180 / pi
% Ambigiuty Resolving
k1 = zeros(size(delay1));
k2 = zeros(size(delay2));
```

```
k1(delay1 > pi) = 2;
k2(delay2 > pi) = 1;
k = k1+k2;
% Changing anlge to frequency (This is normally a lookup
table)
w = (delay3 + 2*k*pi)/d3;
f = fs*w/(2*pi);
% 2*Pi ramp for Plot
x = linspace(0,pi,size(delay3,2));
xf1=fs*x/(d1*pi*1000000); % Convert angle to frequency for
plot
y1 = mod(2*x, 2*pi);
xf2=2*fs*x/(d2*pi*1000000); % Convert angle to frequency for
plot
y2 = mod(4*x, 2*pi);
xf3=4*fs*x/(d3*pi*1000000); % Convert angle to frequency for
plot
y3 = mod(8*x, 2*pi);
figure(2);
subplot(3,1,1);
plot(xf1,delay1);
hold on;
plot(xf1, y1,'r--');
title('T = 9 Delay Line');
grid on;
set(gca, 'Ylim', [0 2*pi]);
ylabel('Phase [radians]');
xlabel('Frequency [MHz]');
grid on;
subplot(3,1,2);
plot(xf2,delay2);
hold on;
plot(xf2, y2,'r--');
title('T = 18 Delay Line');
grid on;
set(gca, 'Ylim', [0 2*pi]);
ylabel('Phase [radians]');
xlabel('Frequency [MHz]');
grid on;
subplot(3,1,3);
plot(xf3,delay3);
hold on;
plot(xf3, y3,'r--');
title('T = 27 Delay Line');
grid on;
set(gca, 'Ylim', [0 2*pi]);
ylabel('Phase [radians]');
xlabel('Frequency [MHz]');
grid on;
```

```
figure(3);
plot(f/1e6);
ylabel('Frequency [MHz]');
title('Frequency Output of IFM');
xlabel('Time');
grid on;
set(gca, 'Ylim', [0 1200]);
```

Appendix C Prony's Method Simulation MATLAB Code

C.1 Two Simultaneous Signals

```
% Clear all variables
clear all;
% Close all previous plots
close all;
% Sampling rate [Hz]
fs = 5000e6;
% Time steps
t = 0:1/fs:4095/fs;
%t = 0:1/fs:50000/fs;
disp('=================== IFM Simulation
=================\n')
disp('== 1. Input signal f = 200 MHz ==\n')
disp('== 2. Input signal f = 1000 MHz ==\n')
disp('== 3. Two input signals f1 = 200 MHz, f2 = 1000 MHz
==\n')
disp('== 4. Two input signals f1 = 300 MHz, f2 = 900 MHz
==\n')
disp('== 5. Two input signals f1 = 400 MHz, f2 = 800 MHz
==\n')
disp('== 6. Two input signals f1 = 500 MHz, f2 = 700 MHz
==\n')
disp('== 7. Two input signals f1 = 550 MHz, f2 = 650 MHz
==\n')
disp('== 8. Two input signals f1 = 600 MHz, f2 = 600 MHz
==\n')
disp('====================================================\n'
)
a = input('Enter a option :');
if (a == 1) s= 50*cos(2*pi*200e6*t);
elseif (a == 2) s= 50*cos(2*pi*1000e6*t);
elseif (a == 3) s=
50*sin(2*pi*200e6*t)+50*cos(2*pi*1000e6*t);
elseif (a == 4) s= 50*sin(2*pi*300e6*t)+50*cos(2*pi*900e6*t);
elseif (a == 5) s= 50*sin(2*pi*400e6*t)+50*cos(2*pi*800e6*t);
elseif (a == 6) s= 50*sin(2*pi*500e6*t)+50*cos(2*pi*700e6*t);
elseif (a == 7) s= 50*sin(2*pi*550e6*t)+50*cos(2*pi*650e6*t);
elseif (a == 8) s= 50*sin(2*pi*600e6*t)+50*cos(2*pi*600e6*t);
end;
% Plot the signal
figure(1);
plot(t*1e9,s);
set(gca, 'Xlim', [0 20]);
title('Input signal');
ylabel('Amplitude [mV]');
xlabel('Time [ns]');
```

```
grid on;
% Hilbert Transform - Transform Real Signal to Complex Signal
s_imag = hilbert(s);
% Delay Line Length = 2 sample delay
d1 = 2;
% Delay Line Length = 4 (2 * 2) sample delay
d2 = 4;
% Delay Line Length = 6 (3 * 2) sample delay
d3 = 6;
% Delay Line Length = 8 (4 * 2) sample delay
d4 = 8;
% Delay the Complex Signal
s_d1 = [s_imag(d1+1:end) zeros(1,d1)];
s_d2 = [s_imag(d2+1:end) zeros(1,d2)];
s_d3 = [s_imag(d3+1:end) zeros(1,d3)];
s_d4 = [s_imag(d4+1:end) zeros(1,d4)];
% Determine the SIN and COS components
% Multiply the delayed signal with the original signal, apply
lpf to filter
% out high frequency component
%lpf = [-2.6498e-019 7.977e-005 0.00036582 0.00095245
0.0019625 0.0035319 0.0057835 0.0087932 0.012558 0.01697
0.021811 0.026759 0.031423 0.035391 0.038282 0.039806
0.039806 0.038282 0.035391 0.031423 0.026759 0.021811 0.01697
0.012558 0.0087932 0.0057835 0.0035319 0.0019625 0.00095245
0.00036582 7.977e-005 -2.6498e-019];
%OR
% Returns a discrete-time filter object.
%
% MATLAB Code
% Generated by MATLAB(R) 8.0 and the Signal Processing
Toolbox 6.18.
%
% Generated on: 05-Nov-2013 09:29:34
%
% Equiripple Lowpass filter designed using the FIRPM
function.
% All frequency values are in MHz.
Fs = fs/1000000;
N = 100; % Order
Fpass = 1; % Passband Frequency
Fstop = 200; % Stopband Frequency
Wpass = 1; % Passband Weight
wstop = 1; % Stopband Weight
dens = 20; % Density Factor
% Calculate the coefficients using the FIRPM function.
lpf = firpm(N, [0 Fpass Fstop Fs/2]/(Fs/2), [1 1 0 0], [Wpass
Wstop], ...
{dens});
```

```
s_delay1 = filter(lpf,1,s_d1.*conj(s_imag));
s_delay2 = filter(lpf,1,s_d2.*conj(s_imag));
s_delay3 = filter(lpf,1,s_d3.*conj(s_imag));
s_delay4 = filter(lpf,1,s_d4.*conj(s_imag));
% Prony's Method
r_1 = s_delay1;
r_2 = s_delay2;
r_3 = s_delay3;
r_4 = s_delay4;
for counter = 1:1:size(r_1,2)
a1(counter) = det([-1*r_4(counter) r_2(counter);-
1*r_3(counter) r_1(counter)])/det([r_3(counter)
r_2(counter);r_2(counter) r_1(counter)]);
a2(counter) = det([r_3(counter) -1*r_4(counter);r_2(counter)
-1*r_3(counter)])/det([r_3(counter) r_2(counter);r_2(counter)
r_1(counter)]);
r = roots([1 a1(counter) a2(counter)]);
f1(counter) = angle(r(1))/(2*pi*4e-10); % 2*pi*T
f2(counter) = angle(r(2))/(2*pi*4e-10);
end;
figure(2);
plot(t*1e9,f1/1e6,'b.');
hold;
plot(t*1e9,f2/1e6,'b.')
xlabel('Time [ns]');
ylabel('Frequency [MHz]'); title('Frequency Outputs of IFM');
set(gca, 'Ylim', [0 1200]);
grid on;
hold;
```

C.2 Three Simultaneous Signals

```
% Clear all variables
clear all;
% Close all previous plots
close all;
% Sampling rate [Hz]
fs = 5000e6;
% Time steps
t = 0:1/fs:4095/fs;
%t = 0:1/fs:10000/fs;
disp('=================== IFM Simulation
====================================\n')
disp('== 1. Three input signals f1 = 200 MHz, f2 = 600 MHz,
f3 = 1000 MHz ==\n')
disp('== 2. Three input signals f1 = 200 MHz, f2 = 400 MHz,
f3 = 600 MHz ==\n')
disp('=============================================================
===============\n')
```

```
a = input('Enter a option :');
if (a == 1) s=
50*cos(2*pi*200e6*t)+50*cos(2*pi*600e6*t)+50*cos(2*pi*1000e6*
t);
elseif (a == 2) s=
50*cos(2*pi*200e6*t)+50*cos(2*pi*400e6*t)+50*cos(2*pi*600e6*t
);
%elseif (a == 3) s=
50*sin(2*pi*200e6*t)+50*cos(2*pi*1000e6*t);
%elseif (a == 4) s=
50*sin(2*pi*300e6*t)+50*cos(2*pi*900e6*t);
%elseif (a == 5) s=
50*sin(2*pi*400e6*t)+50*cos(2*pi*800e6*t);
%elseif (a == 6) s=
50*sin(2*pi*500e6*t)+50*cos(2*pi*700e6*t);
%elseif (a == 7) s=
50*sin(2*pi*550e6*t)+50*cos(2*pi*650e6*t);
%elseif (a == 8) s=
50*sin(2*pi*600e6*t)+50*cos(2*pi*600e6*t);
end;
% Plot the signal
figure(1);
plot(t*1e9,s);
set(gca, 'Xlim', [0 20]);
title('Input signal');
ylabel('Amplitude [mV]');
xlabel('Time [ns]');
grid on
% Hilbert Transform - Transform Real Signal to Complex Signal
s_imag = hilbert(s);
% Delay Line Length = 2 sample delay
d1 = 2;
% Delay Line Length = 4 (2 * 2) sample delay
d2 = 4;
% Delay Line Length = 6 (3 * 2) sample delay
d3 = 6;
% Delay Line Length = 8 (4 * 2) sample delay
d4 = 8;
% Delay Line Length = 10 (5 * 2) sample delay
d5 = 10;
% Delay Line Length = 12 (6 * 2) sample delay
d6 = 12;
% Delay the Complex Signal
s_d1 = [s_imag(d1+1:end) zeros(1,d1)];
s_d2 = [s_imag(d2+1:end) zeros(1,d2)];
s_d3 = [s_imag(d3+1:end) zeros(1,d3)];
s_d4 = [s_imag(d4+1:end) zeros(1,d4)];
s_d5 = [s_imag(d5+1:end) zeros(1,d5)];
s_d6 = [s_imag(d6+1:end) zeros(1,d6)];
```

```
% Determine the SIN and COS components
% Multiply the delayed signal with the original signal, apply
lpf to filter
% out high frequency component
%lpf = [-2.6498e-019 7.977e-005 0.00036582 0.00095245
0.0019625 0.0035319 0.0057835 0.0087932 0.012558 0.01697
0.021811 0.026759 0.031423 0.035391 0.038282 0.039806
0.039806 0.038282 0.035391 0.031423 0.026759 0.021811 0.01697
0.012558 0.0087932 0.0057835 0.0035319 0.0019625 0.00095245
0.00036582 7.977e-005 -2.6498e-019];
%OR
% Returns a discrete-time filter object.
%
% MATLAB Code
% Generated by MATLAB(R) 8.0 and the Signal Processing
Toolbox 6.18.
%
% Generated on: 05-Nov-2013 09:29:34
%
% Equiripple Lowpass filter designed using the FIRPM
function.
% All frequency values are in MHz.
Fs = fs/1000000;
N = 100; % Order
Fpass = 1; % Passband Frequency
Fstop = 200; % Stopband Frequency
Wpass = 1; % Passband Weight
Wstop = 1; % Stopband Weight
dens = 20; % Density Factor
% Calculate the coefficients using the FIRPM function.
lpf = firpm(N, [0 Fpass Fstop Fs/2]/(Fs/2), [1 1 0 0], [Wpass
Wstop], ...
{dens});
s_delay1 = filter(lpf,1,s_d1.*conj(s_imag));
s_delay2 = filter(lpf,1,s_d2.*conj(s_imag));
s_delay3 = filter(lpf,1,s_d3.*conj(s_imag));
s_delay4 = filter(lpf,1,s_d4.*conj(s_imag));
s_delay5 = filter(lpf,1,s_d5.*conj(s_imag));
s_delay6 = filter(lpf,1,s_d6.*conj(s_imag));
% Prony's Method
r_1 = s_delay1;
r_2 = s_delay2;
r_3 = s_delay3;
r_4 = s_delay4;
r_5 = s_delay5;
r_6 = s_delay6;
for counter = 1:1:size(r_1,2)
%counter = 129;
```

```
deter1(counter) = det([r_5(counter) r_4(counter)
r_3(counter);r_4(counter) r_3(counter)
r_2(counter);r_3(counter) r_2(counter) r_1(counter)]);
a1(counter) = det([-1*r_6(counter) r_4(counter)
r_3(counter);-1*r_5(counter) r_3(counter) r_2(counter);-
1*r_4(counter) r_2(counter) r_1(counter)])/det([r_5(counter)
r_4(counter) r_3(counter);r_4(counter) r_3(counter)
r_2(counter);r_3(counter) r_2(counter) r_1(counter)]);
a2(counter) = det([r_5(counter) -1*r_6(counter)
r_3(counter);r_4(counter) -1*r_5(counter)
r_2(counter);r_3(counter) -1*r_4(counter)
r_1(counter)])/det([r_5(counter) r_4(counter)
r_3(counter);r_4(counter) r_3(counter)
r_2(counter);r_3(counter) r_2(counter) r_1(counter)]);
a3(counter) = det([r_5(counter) r_4(counter) -
1*r_6(counter);r_4(counter) r_3(counter) -
1*r_5(counter);r_3(counter) r_2(counter) -
1*r_4(counter)])/det([r_5(counter) r_4(counter)
r_3(counter);r_4(counter) r_3(counter)
r_2(counter);r_3(counter) r_2(counter) r_1(counter)]);
r = roots([1 a1(counter) a2(counter) a3(counter)]);
f1(counter) = angle(r(1))/(2*pi*4e-10); % 2*pi*T
f2(counter) = angle(r(2))/(2*pi*4e-10);
f3(counter) = angle(r(3))/(2*pi*4e-10);
end;
figure(2);
plot(t*1e9,f1/1e6,'b.');
hold;
plot(t*1e9,f2/1e6,'b.')
plot(t*1e9,f3/1e6,'b.')
xlabel('Time [ns]');
ylabel('Frequency [MHz]'); title('Frequency Outputs of IFM');
set(gca, 'Ylim', [0 1200]);
grid on;
hold;
```

C.3 Four Simultaneous Signals

```
% Clear all variables
clear all;
% Close all previous plots
close all;
% Sampling rate [Hz]
fs = 5000e6;
% Time steps
t = 0:1/fs:4095/fs;
%t = 0:1/fs:10000/fs;
disp('================== IFM Simulation
===============================================\n')
disp('== 1. Four input signals f1 = 100 MHz, f2 = 400 MHz, f3
= 700 MHz f4 = 1000 MHz ==\n')
```

```
disp('=======================================================
===========================\n')
a = input('Enter a option :');
if (a == 1) s=
50*cos(2*pi*100e6*t)+50*cos(2*pi*400e6*t)+50*cos(2*pi*700e6*t
)+50*cos(2*pi*1000e6*t);
%elseif (a == 3) s=
50*sin(2*pi*200e6*t)+50*cos(2*pi*1000e6*t);
%elseif (a == 4) s=
50*sin(2*pi*300e6*t)+50*cos(2*pi*900e6*t);
%elseif (a == 5) s=
50*sin(2*pi*400e6*t)+50*cos(2*pi*800e6*t);
%elseif (a == 6) s=
50*sin(2*pi*500e6*t)+50*cos(2*pi*700e6*t);
%elseif (a == 7) s=
50*sin(2*pi*550e6*t)+50*cos(2*pi*650e6*t);
%elseif (a == 8) s=
50*sin(2*pi*600e6*t)+50*cos(2*pi*600e6*t);
end;
% Plot the signal
figure(1);
plot(t*1e9,s);
set(gca, 'Xlim', [0 20]);
title('Input signal');
ylabel('Amplitude [mV]');
xlabel('Time [ns]');
grid on
% Hilbert Transform - Transform Real Signal to Complex Signal
s_imag = hilbert(s);
% Delay Line Length = 2 sample delay
d1 = 2;
% Delay Line Length = 4 (2 * 2) sample delay
d2 = 4;
% Delay Line Length = 6 (3 * 2) sample delay
d3 = 6;
% Delay Line Length = 8 (4 * 2) sample delay
d4 = 8;
% Delay Line Length = 10 (5 * 2) sample delay
d5 = 10;
% Delay Line Length = 12 (6 * 2) sample delay
d6 = 12;
% Delay Line Length = 14 (7 * 2) sample delay
d7 = 14;
% Delay Line Length = 16 (8 * 2) sample delay
d8 = 16;
% Delay the Complex Signal
s_d1 = [s_imag(d1+1:end) zeros(1,d1)];
s_d2 = [s_imag(d2+1:end) zeros(1,d2)];
s_d3 = [s_imag(d3+1:end) zeros(1,d3)];
```

```
s_d4 = [s_imag(d4+1:end) zeros(1,d4)];
s_d5 = [s_imag(d5+1:end) zeros(1,d5)];
s_d6 = [s_imag(d6+1:end) zeros(1,d6)];
s_d7 = [s_imag(d7+1:end) zeros(1,d7)];
```

```
s_d8 = [s_imag(d8+1:end) zeros(1,d8)];
% Determine the SIN and COS components
% Multiply the delayed signal with the original signal, apply
lpf to filter
% out high frequency component
%lpf = [-2.6498e-019 7.977e-005 0.00036582 0.00095245
0.0019625 0.0035319 0.0057835 0.0087932 0.012558 0.01697
0.021811 0.026759 0.031423 0.035391 0.038282 0.039806
0.039806 0.038282 0.035391 0.031423 0.026759 0.021811 0.01697
0.012558 0.0087932 0.0057835 0.0035319 0.0019625 0.00095245
0.00036582 7.977e-005 -2.6498e-019];
%OR
% Returns a discrete-time filter object.
%
% MATLAB Code
% Generated by MATLAB(R) 8.0 and the Signal Processing
Toolbox 6.18.
%
% Generated on: 05-Nov-2013 09:29:34
%
% Equiripple Lowpass filter designed using the FIRPM
function.
% All frequency values are in MHz.
Fs = fs/1000000;
N = 100; % Order
Fpass = 1; % Passband Frequency
Fstop = 200; % Stopband Frequency
Wpass = 1; % Passband Weight
Wstop = 1; % Stopband Weight
dens = 20; % Density Factor
% Calculate the coefficients using the FIRPM function.
lpf = firpm(N, [0 Fpass Fstop Fs/2]/(Fs/2), [1 1 0 0], [Wpass
Wstop], ...
{dens});
s_delay1 = filter(lpf,1,s_d1.*conj(s_imag));
s_delay2 = filter(lpf,1,s_d2.*conj(s_imag));
s_delay3 = filter(lpf,1,s_d3.*conj(s_imag));
s_delay4 = filter(lpf,1,s_d4.*conj(s_imag));
s_delay5 = filter(lpf,1,s_d5.*conj(s_imag));
s_delay6 = filter(lpf,1,s_d6.*conj(s_imag));
s_delay7 = filter(lpf,1,s_d7.*conj(s_imag));
s_delay8 = filter(lpf,1,s_d8.*conj(s_imag));
% Prony's Method
r_1 = s_delay1;
r_2 = s_delay2;
r_3 = s_delay3;
r_4 = s_delay4;
r_5 = s_delay5;
r_6 = s_delay6;
```

```
r_7 = s_delay7;
r_8 = s_delay8;
for counter = 1:1:size(r_1,2)
%counter = 129;
deter1(counter) = det([r_7(counter) r_6(counter) r_5(counter)
r_4(counter);r_6(counter) r_5(counter) r_4(counter)
r_3(counter);r_5(counter) r_4(counter) r_3(counter)
r_2(counter);r_4(counter) r_3(counter) r_2(counter)
r_1(counter)]);
a1(counter) = det([-1*r_8(counter) r_6(counter) r_5(counter)
r_4(counter);-1*r_7(counter) r_5(counter) r_4(counter)
r_3(counter);-1*r_6(counter) r_4(counter) r_3(counter)
r_2(counter);-1*r_5(counter) r_3(counter) r_2(counter)
r_1(counter)])/deter1(counter);
a2(counter) = det([r_7(counter) -1*r_8(counter) r_5(counter)
r_4(counter);r_6(counter) -1*r_7(counter) r_4(counter)
r_3(counter);r_5(counter) -1*r_6(counter) r_3(counter)
r_2(counter);r_4(counter) -1*r_5(counter) r_2(counter)
r_1(counter)])/deter1(counter);
a3(counter) = det([r_7(counter) r_6(counter) -1*r_8(counter)
r_4(counter);r_6(counter) r_5(counter) -1*r_7(counter)
r_3(counter);r_5(counter) r_4(counter) -1*r_6(counter)
r_2(counter);r_4(counter) r_3(counter) -1*r_5(counter)
r_1(counter)])/deter1(counter);
a4(counter) = det([r_7(counter) r_6(counter) r_5(counter) -
1*r_8(counter);r_6(counter) r_5(counter) r_4(counter) -
1*r_7(counter);r_5(counter) r_4(counter) r_3(counter) -
1*r_6(counter);r_4(counter) r_3(counter) r_2(counter) -
1*r_5(counter)])/deter1(counter);
r = roots([1 a1(counter) a2(counter) a3(counter)
a4(counter)]);
f1(counter) = angle(r(1))/(2*pi*4e-10); % 2*pi*T
f2(counter) = angle(r(2))/(2*pi*4e-10);
f3(counter) = angle(r(3))/(2*pi*4e-10);
f4(counter) = angle(r(4))/(2*pi*4e-10);
end;
figure(2);
plot(t*1e9,f1/1e6,'b.');
hold;
plot(t*1e9,f2/1e6,'b.')
plot(t*1e9,f3/1e6,'b.')
plot(t*1e9,f4/1e6,'b.')
xlabel('Time [ns]');
ylabel('Frequency [MHz]'); title('Frequency Output of IFM');
set(gca, 'Ylim', [0 1200]);
grid on;
hold;
```